

# Automates programmables

Série SYSMAC CS

CS1G/H-CPU□□-EV1

CS1G/H-CPU□□H

CS1D-CPU□□H

CS1D-CPU□□S

Série SYSMAC CJ

CJ1G-CPU□□

CJ1G/H-CPU□□H

CJ1M-CPU□□P

CJ1M-CPU□□

## MANUEL DE PROGRAMMATION



### Présentation

- 1 Fonctionnement de l'UC
- 2 Programmation
- 3 Fonction des instructions
- 4 Tâches

# **Série SYSMAC CS**

**CS1G/H-CPU□□-EV1**

**CS1G/H-CPU□□H**

**CS1D-CPU□□H**

**CS1D-CPU□□S**

# **Série SYSMAC CJ**

**CJ1G-CPU□□**

**CJ1G/H-CPU□□H**

**CJ1G-CPU□□P**

**CJ1M-CPU□□**

# **Automates programmables**

## **Manuel de programmation**

*Révisé en juillet 2004*



## **Avis :**

Les produits OMRON sont conçus pour être utilisés par un opérateur qualifié, en respectant les procédures appropriées et uniquement aux fins précisées dans ce document.

Les conventions suivantes sont utilisées dans ce manuel pour indiquer et catégoriser les consignes de sécurité. Respectez toujours les informations fournies. Le non-respect de ces consignes peut entraîner des blessures ou des dégâts matériels.

 **ATTENTION DANGER** Indique un danger imminent qui, s'il n'est pas évité, peut provoquer des blessures graves ou mortelles.

 **AVERTISSEMENT** Indique un danger potentiel qui, s'il n'est pas évité, peut provoquer des blessures graves ou mortelles.

 **Attention** Indique un danger potentiel qui, s'il n'est pas évité, peut provoquer des blessures moins graves ou endommager des biens.

## **Références des produits OMRON**

Tous les produits OMRON sont écrits en majuscules dans le présent manuel. Le mot « carte » porte également une majuscule lorsqu'il fait référence à un produit OMRON, sous forme de nom propre ou de nom commun.

L'abréviation « Ch » qui apparaît sur certains affichages et sur certains produits OMRON signifie souvent « mot ».

L'abréviation « API » signifie Automate programmable. « PC » est toutefois utilisé dans certains affichages de périphérique de programmation et signifie Programmable Controller.

## **Aide visuelle**

Les intitulés suivants apparaissent dans la colonne gauche du manuel pour vous aider à localiser différents types d'informations.

**Remarque** Désigne des informations particulièrement intéressantes pour une utilisation efficace du produit.

**1,2,3...** 1. Indique la présence d'une liste telle que des procédures ou des listes de contrôles, etc.

### **© OMRON, 2001**

Tous droits réservés. Aucune partie de cette publication ne peut être reproduite, stockée dans un système de mémoire ou transmise, sous quelque forme ou par quelque moyen mécanique, électronique, photocopie, enregistrement que ce soit, sans l'accord écrit préalable d'OMRON.

L'utilisation des informations contenues ci-après n'entraîne aucune responsabilité. De plus, dans un souci d'améliorer sans cesse la qualité de ses produits, OMRON se réserve le droit de modifier toute information contenue dans le présent manuel sans préavis. Malgré tout le soin apporté à l'élaboration de ce manuel, OMRON décline toute responsabilité quant aux erreurs et omissions qui pourraient s'y trouver. Enfin, OMRON décline toute responsabilité concernant tout dommage résultant de l'utilisation des informations contenues dans le présent manuel.

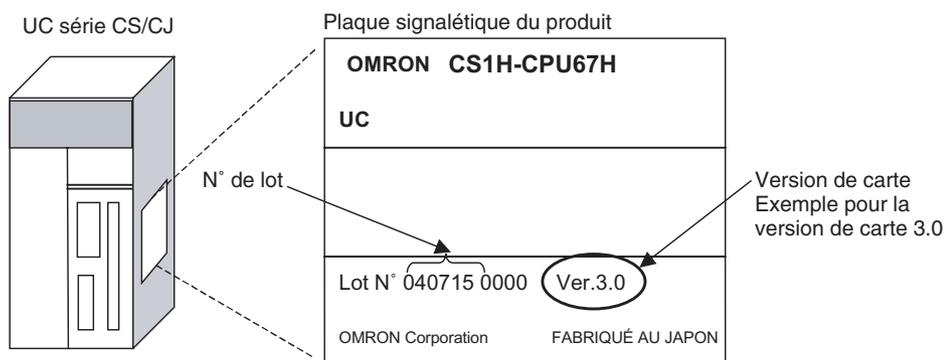
# Version des UCs série CS/CJ

## Version des cartes

### Notation des versions de carte sur les produits

Afin de gérer les UCs de la série CS/CJ en fonction de leurs différences de fonctionnalités dues à leurs mises à niveau, nous avons introduit la notion de « version de carte ». Cela s'applique aux UCs CS1-H, CJ1-H, CJ1M et CS1D.

Le numéro de version apparaît à droite du numéro de lot sur la plaque signalétique des produits concernés par ce mode de gestion des versions, comme illustré ci-dessous.



- La version des UCs CS1-H, CJ1-H et CJ1M (à l'exception des modèles économiques) fabriquées avant le 4 novembre 2003 n'est pas indiquée sur l'UC (l'emplacement réservé à ce numéro, illustré ci-dessus, est vide).
- Les UC CS1-H, CJ1-H et CJ1M, ainsi que les UC CS1D pour systèmes à UC seule, commencent à la version 2.0.
- Les UC CS1D des systèmes d'UC en duplex commencent à la version 1.1.
- Les UC pour lesquelles aucun numéro de version n'est fourni sont appelées UC *pré-ver.* □.□, telles que UC *pré-ver.* 2.0 et UC *pré-ver.* 1.1.

### Vérification des versions de carte avec un logiciel de prise en charge

Vous pouvez utiliser CX-Programmer version 4.0 pour contrôler la version de carte en appliquant l'une des deux méthodes suivantes.

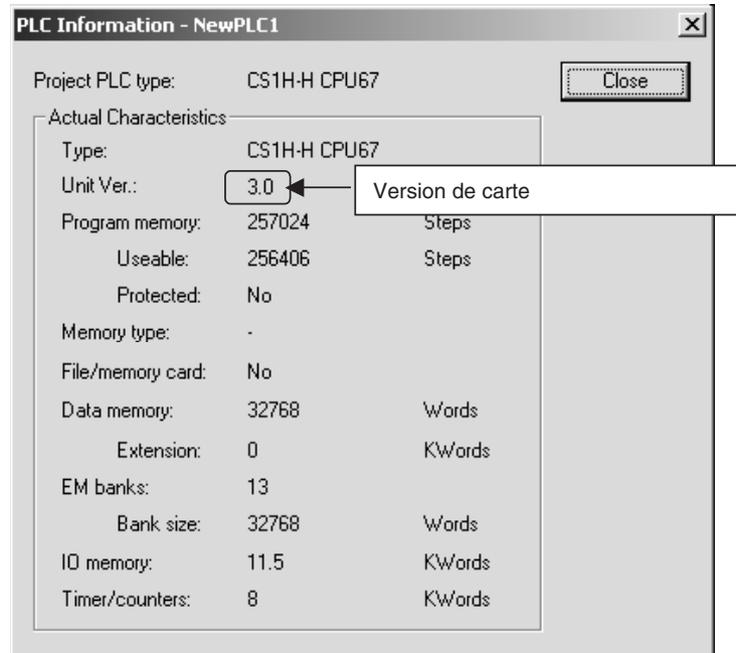
- A l'aide des **informations de l'API**
- A l'aide des **informations sur la fabrication de la carte** (cette méthode peut aussi être utilisée pour les cartes d'E/S spéciales et les cartes réseau.)

**Remarque** CX-Programmer version 3.3 ou antérieure ne permet pas de vérifier les versions de carte.

### Informations de l'API

- Si vous connaissez le type de périphérique et le type d'UC, sélectionnez-les dans la boîte de dialogue *Change PLC*, connectez-vous en ligne, puis sélectionnez **PLC - Edit - Information** dans les menus.
- Si vous ne connaissez pas le type de périphérique ni le type d'UC, mais que vous êtes connecté directement à l'UC sur une ligne série, sélectionnez **PLC - Auto Online** pour passer en ligne, puis sélectionnez **PLC - Edit - Information** dans les menus.

Dans les deux cas, la boîte de dialogue *PLC Information* s'affiche.



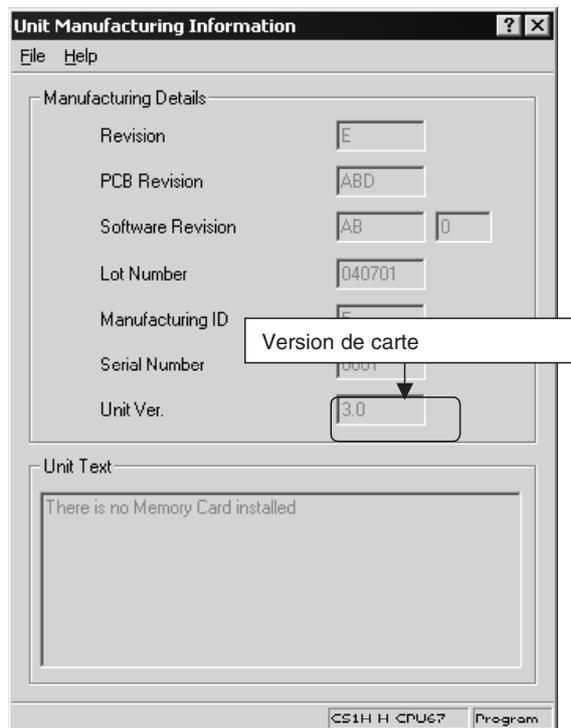
Utilisez l'affichage ci-dessus pour vérifier la version de l'UC.

### **Informations sur la fabrication de la carte**

Dans la fenêtre *IO Table*, cliquez avec le bouton droit et sélectionnez ***Unit Manufacturing information - CPU Unit***.



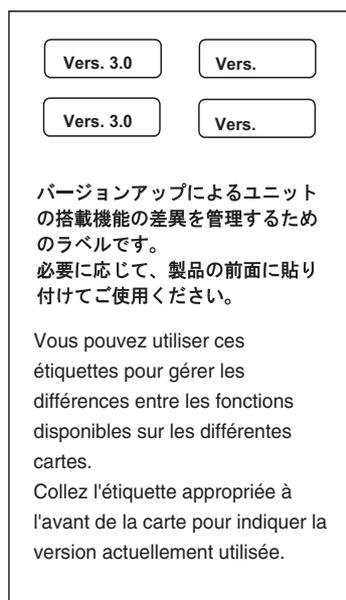
La boîte de dialogue *Unit Manufacturing information* suivante s'affiche.



Utilisez l'affichage ci-dessus pour vérifier la version de l'UC connectée en ligne.

### Utilisation des étiquettes de version de carte

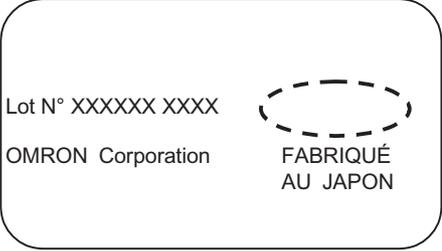
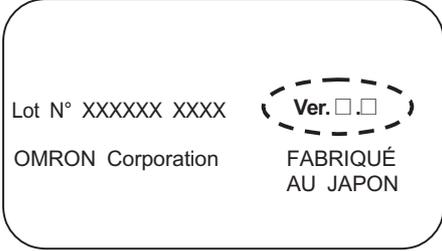
Les étiquettes de version suivantes sont fournies avec l'UC.



Ces étiquettes peuvent être attachées à l'avant des UCs antérieures afin de différencier les versions d'UC.

## Notation des versions de carte

Dans ce manuel, la version d'une UC est indiquée comme dans le tableau ci-dessous.

Plaque signalétique du produit	UC sans numéro de version	Cartes avec numéro de version (Ver. □.□)
<b>Signification</b> Désignation d'UC individuelles (par exemple, CS1H-CPU67H) Désignation de groupes d'UC (par exemple, UC CS1H) Désignation d'une série d'UC (par exemple, les UC série CS)		
	UC pré-ver. 2.0 CS1-H	UC CS1H-CPU67H Ver. □.□
	UC pré-ver. 2.0 CS1-H	UC CS1-H Ver. □.□
	UC pré-ver. 2.0 série CS	UC série CS Ver. □.□

## Numéros de version et de lot

Série	Modèle		Date de fabrication						
			Avant	Sept. 2003	Oct. 2003	Nov. 2003	Déc. 2003	Juin 2004	Après
Série CS	UC CS1	CS1□-CPU□□							
		Pas de numéro de version							
	UC CS1-V1 s	CS1□-CPU□□-V1							
		Pas de numéro de version							
	UC CS1-H	CS1□-CPU□□H							
			UC pré-ver. 2.0	Cartes UC ver. 2.0 (N° lot : 031105 indiqué)			UC ver. 3.0 (N° lot : 040622 indiqué)		
UC CS1D	UC pour système à UC en duplex	CS1D-CPU□□H							
	UC pré-ver. 1.1		UC Ver.1.1 (N° lot : 031120 indiqué)						
	UC pour système à UC unique	CS1D-CPU□□S							
						UC ver. 2.0 (N° lot : 031215 indiqué)			
Série CJ	UC CJ1	CJ1G-CPU□□							
			UC pré-ver. 2.0						
	UC CJ1-H	CJ1□-CPU□□H							
			UC pré-ver. 2.0	UC ver. 2.0 (N° lot : 031105 indiqué)		UC ver. 3.0 (N° lot : 040623 indiqué)			
UC CJ1M excepté modèles économiques	CJ1M-CPU□□								
		UC pré-ver. 2.0	UC ver. 2.0 (N° lot : 031105 indiqué)		UC ver. 3.0 (N° lot : 040624 indiqué)				
UC CJ1M, modèles bas de gamme	CJ1M-CPU11/21								
			Carte ver. 2.0 (N° lot : 031002 indiqué)			UC ver. 3.0 (N° lot : 040629 indiqué)			
Logiciel de prise en charge	CX-Programmer	WS02-CXPC1-EV□							
			Ver. 3.2	Ver. 3.3		Ver. 4.0		Ver. 5.0	

## Fonction prise en charge par la version

### UC CS1-H (CS1□-CPU□□H)

Fonction		Version de carte	
		UC pré-ver. 2.0	UC Ver. 2.0
Téléchargement de tâches individuelles		---	OK
Protection de lecture améliorée grâce aux mots de passe		---	OK
Protection en écriture à partir de commandes FINS envoyées aux UCs via le réseau		---	OK
Connexions au réseau en ligne sans table d'E/S		---	OK
Communications via 8 niveaux de réseau maximum		---	OK
Connexion en ligne aux API via les IHM série NS		OK à partir du numéro de lot 030201	OK
Paramétrage des mots du premier emplacement		OK jusqu'à 8 groupes	OK jusqu'à 64 groupes
Transferts automatiques sous alimentation sans fichier de paramètres		---	OK
Détection automatique de la méthode d'affectation d'E/S lors du transfert automatique à la mise sous tension		---	---
Heures de début/fin de fonctionnement		---	OK
Nouvelles instructions pour l'application	MILH, MILR, MILC	---	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	OK
	BCMP2	---	OK
	GRY	OK à partir du numéro de lot 030201	OK
	TPO	---	OK
	DSW, TKY, HKY, MTR, 7SEG	---	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	OK
	Lecture/écriture de cartes réseau avec IORD/IOWR	OK à partir du numéro de lot 030418	OK
PRV2	---	---	

## UC CS1D

Fonction		UC CS1D pour système à UC en duplex (CS1D-CPU□□H)		UC CS1D pour système d'UC seule (CS1D-CPU□□S)
		UC pré-ver. 1.1	UC Ver. 1.1	UC Ver. 2.0
Fonctions sur les UC CS1D uniquement	UC en duplex	OK	OK	---
	Remplacement de la carte en ligne	OK	OK	OK
	Cartes d'alimentation doubles	OK	OK	OK
	Carte Controller Link double	OK	OK	OK
	Carte Ethernet double	---	OK	OK
Téléchargement de tâches individuelles		---	---	OK
Protection de lecture améliorée grâce aux mots de passe		---	---	OK
Protection en écriture à partir de commandes FINS envoyées aux UCs via le réseau		---	---	OK
Connexions au réseau en ligne sans table d'E/S		---	---	OK
Communications via 8 niveaux de réseau maximum		---	---	OK
Connexion en ligne aux API via les IHM série NS		---	---	OK
Paramétrage des mots du premier emplacement		---	---	OK jusqu'à 64 groupes
Transferts automatiques sous alimentation sans fichier de paramètres		---	---	OK
Détection automatique de la méthode d'affectation d'E/S lors du transfert automatique à la mise sous tension		---	---	---
Heures de début/fin de fonctionnement		---	OK	OK
Nouvelles instructions pour l'application	MILH, MILR, MILC	---	---	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	---	OK
	BCMP2	---	---	OK
	GRY	---	---	OK
	TPO	---	---	OK
	DSW, TKY, HKY, MTR, 7SEG	---	---	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	---	OK
	Lecture/écriture de cartes réseau avec IORD/IOWR	---	---	OK
PRV2	---	---	---	

## UC CJ1-H/CJ1M

Fonction	UC CJ1-H (CJ1□-CPU□□H)		UC CJ1M, excepté modèles économiques (CJ1M-CPU□□)		UC CJ1M, modèles économiques (CJ1M- CPU11/21)
	UC pré-ver. 2.0	UC Ver. 2.0	UC pré-ver. 2.0	UC Ver. 2.0	UC Ver. 2.0
Téléchargement de tâches individuelles	---	OK	---	OK	OK
Protection de lecture améliorée grâce aux mots de passe	---	OK	---	OK	OK
Protection en écriture à partir de commandes FINS envoyées aux UCs via le réseau	---	OK	---	OK	OK
Connexions au réseau en ligne sans table d'E/S	OK, mais uniquement si l'affectation de tables d'E/S est activée sous tension	OK	OK, mais uniquement si l'affectation de tables d'E/S est activée sous tension	OK	OK
Communications via 8 niveaux de réseau maximum	OK jusqu'à 8 groupes	OK jusqu'à 64 groupes	OK jusqu'à 8 groupes	OK jusqu'à 64 groupes	OK jusqu'à 64 groupes
Connexion en ligne aux API via les IHM série NS	OK à partir du numéro de lot 030201	OK	OK à partir du numéro de lot 030201	OK	OK
Paramétrage des mots du premier emplacement	---	OK	---	OK	OK
Transferts automatiques sous alimentation sans fichier de paramètres	---	OK	---	OK	OK
Détection automatique de la méthode d'affectation d'E/S lors du transfert automatique à la mise sous tension	---	OK	---	OK	OK
Heures de début/fin de fonctionnement	---	OK	---	OK	OK
Nouvelles instructions pour l'application	MILH, MILR, MILC	---	OK	OK	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	OK	OK	OK
	BCMP2	---	OK	OK	OK
	GRY	OK à partir du numéro de lot 030201	OK	OK à partir du numéro de lot 030201	OK
	TPO	---	OK	---	OK
	DSW, TKY, HKY, MTR, 7SEG	---	OK	---	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	OK	---	OK
	Lecture/écriture de cartes réseau avec IORD/IOWR	---	OK	---	OK
	PRV2	---	---	---	OK, mais uniquement pour les modèles avec E/S intégrées

## **Fonctions prises en charge par la version de carte 3.0 ou supérieure**

### **UC CS1-H (CS1□-CPU□□H)**

Fonction		Version de carte	
		UC pré-ver. 2.0, Ver. 2.0	Ver. 3.0
Blocs de fonction (pris en charge pour CX-Programmer Ver. 5.0 ou supérieure)		---	OK
Passerelle série (conversion des commandes FINS en commandes CompoWay/F au port série intégré)		---	OK
Mémoire des commentaires (dans la mémoire flash interne)		---	OK
Données de sauvegarde simple étendues		---	OK
Nouvelles instructions pour l'application	TXDU(256), RXDU(255) (prennent en charge les communications sans protocole avec les cartes de communications série, avec la version de carte 1.2 ou supérieure)	---	OK
	Instructions de conversion des modèles : XFERC(565), DISTC(566), COLLC(567), MOVBC(568), BCNTC(621)	---	OK
	Instructions spéciales des blocs de fonction : GETID(286)	---	OK
Fonctions des autres instructions	Instructions TXDU(235) et RXDU(236) (prennent en charge les communications sans protocole avec les cartes de communications série, avec la version de carte 1.2 ou supérieure)	---	OK

### **UC CS1D**

La version de carte 3.0 n'est pas prise en charge.

### **UC CJ1-H/CJ1M (CJ1□-CPU□□H, CJ1G-CPU□□P, CJ1M-CPU□□)**

Fonction		Version de carte	
		UC pré-ver. 2.0, Ver. 2.0	Ver. 3.0
Blocs de fonction (pris en charge pour CX-Programmer Ver. 5.0 ou supérieure)		---	OK
Passerelle série (conversion des commandes FINS en commandes CompoWay/F au port série intégré)		---	OK
Mémoire des commentaires (dans la mémoire flash interne)		---	OK
Données de sauvegarde simple étendues		---	OK
Nouvelles instructions pour l'application	TXDU(256), RXDU(255) (prennent en charge les communications sans protocole avec les cartes de communications série, avec la version de carte 1.2 ou supérieure)	---	OK
	Instructions de conversion des modèles : XFERC(565), DISTC(566), COLLC(567), MOVBC(568), BCNTC(621)	---	OK
	Instructions spéciales des blocs de fonction : GETID(286)	---	OK
Fonctions des autres instructions	Instructions PRV(881) et PRV2(883) : Ajout de méthodes de calcul à haute fréquence pour calculer la fréquence d'impulsion. (UC CJ1M uniquement)	---	OK

## Numéros de version et périphériques de programmation

Vous devez utiliser CX-Programmer version 4.0 ou supérieure pour pouvoir bénéficier des fonctions ajoutées à l'UC ver. 2.0.

Vous devez utiliser CX-Programmer version 5.0 ou supérieure pour pouvoir bénéficier des blocs de fonction ajoutés à l'UC version 3.0.

Les tableaux suivants montrent la relation entre les versions de la carte et les versions de CX-Programmer.

### Numéros de version et périphériques de programmation

UC	Fonctions		CX-Programmer				Console de programmation
			Ver. 3.2 ou antérieure	Ver. 3.3	Ver. 4.0	Ver. 5.0 ou supérieure	
UC CJ1M, modèles économiques, carte ver. 2.0	Fonctions ajoutées pour la version 2.0 de la carte	Nouvelles fonctions utilisées	---	---	OK	OK	Pas de restrictions
		Nouvelles fonctions non utilisées	---	OK	OK	OK	
UC CS1-H, CJ1-H et CJ1M, excepté les modèles économiques, carte ver. 2.0	Fonctions ajoutées pour la version 2.0 de la carte	Nouvelles fonctions utilisées	---	---	OK	OK	
		Nouvelles fonctions non utilisées	OK	OK	OK	OK	
UC CS1D pour système à UC unique, carte ver. 2.0	Fonctions ajoutées pour la version 2.0 de la carte	Nouvelles fonctions utilisées	---	---	OK	OK	
		Nouvelles fonctions non utilisées				OK	
UC CS1D pour systèmes à UC en duplex, carte ver. 1.	Fonctions ajoutées pour la version 1.1 de la carte	Nouvelles fonctions utilisées	---	---	OK	OK	
		Nouvelles fonctions non utilisées	OK	OK	OK	OK	
UC série CS/CJ ver. 3.0	Blocs de fonction ajoutés pour la version de carte 3.0	Utilisation des blocs de fonction	---	---	---	OK	
		Blocs de fonction non utilisés	OK	OK	OK	OK	

**Remarque** Comme indiqué ci-dessus, il n'est pas nécessaire d'effectuer une mise à niveau vers CX-Programmer 4.0 tant que les fonctions ajoutées pour les cartes version 2.0 ou 1.1 ne sont pas utilisées.

### Réglage du type de périphérique

La version de la carte n'affecte pas le réglage effectué pour le type de périphérique avec CX-Programmer. Sélectionnez le type de périphérique comme illustré dans le tableau suivant sans tenir compte de la version de l'UC.

Série	Groupe d'UC	Modèle d'UC	Réglage du type de périphérique dans CX-Programmer Ver. 4.0 ou ultérieure
Série CS	UC CS1-H	CS1G-CPU□□H	CS1G-H
		CS1H-CPU□□H	CS1H-H
	UC CS1D pour systèmes d'UC en duplex	CS1D-CPU□□H	CS1D-H (ou CS1H-H)
	UC CS1D pour systèmes à UC seule	CS1D-CPU□□S	CS1D-S
Série CJ	UC CJ1-H	CJ1G-CPU□□H	CJ1G-H
		CJ1H-CPU□□H	CJ1H-H
	UC CJ1M	CJ1M-CPU□□	CJ1M

## Résolution des problèmes dus aux versions des cartes dans CX-Programmer

Problème	Cause	Solution
 <p>Après l'affichage du message ci-dessus, une erreur de compilation apparaît dans l'onglet <i>Compile</i> de la fenêtre Output.</p>	<p>Vous avez essayé d'utiliser CX-Programmer version 4.0 ou ultérieure pour télécharger vers des UC pré-ver 2.0 un programme contenant des instructions prises en charge uniquement par des UC Version 2.0 ou ultérieure.</p>	<p>Vérifiez le programme ou remplacez l'UC téléchargée par une UC version 2.0 ou ultérieure.</p>
	<p>Vous avez essayé d'utiliser CX-Programmer 4.0 ou supérieur pour télécharger vers des UC pré-ver 2.0 une configuration de l'API contenant des paramètres pris en charge uniquement par des UC Version 2.0 ou ultérieure (c.-à-d. non définies sur leurs valeurs par défaut).</p>	<p>Vérifiez les paramètres dans la configuration de l'API ou remplacez l'UC téléchargée par une UC version 2.0 ou ultérieure.</p>
<p>« ???? » apparaît dans un programme transféré d'un API vers CX-Programmer.</p>	<p>Vous avez utilisé CX-Programmer 3.3 ou inférieur pour télécharger un programme contenant des instructions prises en charge uniquement par les UC version 2.0 ou ultérieure à partir d'une UC version 2.0 ou ultérieure.</p>	<p>Il est impossible de télécharger les nouvelles instructions à l'aide de CX-Programmer 3.3 ou inférieur. Utilisez CX-Programmer version 4.0 ou ultérieure.</p>

# SOMMARE

<b>PRECAUTIONS</b> .....	<b>xxiii</b>
1 Public visé .....	xxiv
2 Précautions générales .....	xxiv
3 Précautions en matière de sécurité .....	xxiv
4 Précautions liées à l'environnement d'utilisation .....	xxvi
5 Précautions en matière d'application .....	xxvi
6 Conformité aux directives CE .....	xxx
<b>CHAPITRE 1</b>	
<b>Fonctionnement de l'UC</b> .....	<b>1</b>
1-1 Configuration initiale (UC CS1 uniquement) .....	2
1-2 Utilisation de l'horloge interne (UC CS1 uniquement) .....	5
1-3 Structure interne de l'UC .....	6
1-4 Modes de fonctionnement .....	8
1-5 Programmes et tâches .....	12
1-6 Description des tâches .....	14
<b>CHAPITRE 2</b>	
<b>Programmation</b> .....	<b>19</b>
2-1 Concepts de base .....	20
2-2 Précautions .....	55
2-3 Vérification des programmes .....	64
<b>CHAPITRE 3</b>	
<b>Fonction des instructions</b> .....	<b>71</b>
3-1 Instructions d'entrée des séquence .....	72
3-2 Instructions de sortie des séquences .....	74
3-3 Instructions de commande des séquences .....	77
3-4 Instructions de temporisation et de compteur .....	81
3-5 Instructions de comparaison .....	85
3-6 Instructions de déplacement de données .....	89
3-7 Instructions de décalage de données .....	92
3-8 Instructions d'incrément/décément .....	96
3-9 Instructions mathématiques à symboles .....	97
3-10 Instructions de conversion .....	102
3-11 Instructions logiques .....	108
3-12 Instructions mathématiques spéciales .....	110
3-13 Instructions mathématiques à virgule flottante .....	111
3-14 Instructions à virgule flottante double précision .....	115
3-15 Instructions de traitement de données de tableaux .....	119
3-16 Instructions de contrôle de données .....	123
3-17 Instructions de sous-programme .....	127
3-18 Instructions de traitement d'interruption .....	128
3-19 Instructions de compteur à grande vitesse et de sortie d'impulsion (CJ1M-CPU21/22/23 uniquement) .....	130
3-20 Instructions de pas .....	132
3-21 Instructions des cartes d'E/S standard .....	132
3-22 Instructions de communications série .....	135
3-23 Instructions réseaux .....	136
3-24 Instructions de mémoire de fichiers .....	139
3-25 Instructions d'affichage .....	140

# SOMMAIRE

3-26	Instructions de temporisation .....	140
3-27	Instructions de débogage.....	141
3-28	Instructions de diagnostic d'erreur .....	142
3-29	Autres instructions .....	143
3-30	Instructions de programmation de bloc.....	144
3-31	Instructions de traitement des chaînes de texte.....	150
3-32	Instructions de contrôle des tâches .....	153
3-33	Instructions de conversion des modèles (UC ver. 3.0 ou supérieure uniquement).....	154
3-34	Instructions spéciales des blocs de fonction .....	155

## CHAPITRE 4

### Tâches..... 157

4-1	Caractéristiques des tâches .....	158
4-2	Utilisation de tâches .....	167
4-3	Tâche d'interruption .....	177
4-4	Fonctionnement du périphérique de programmation pour les tâches .....	189

## CHAPITRE 5

### Fonctions de mémoire de fichiers..... 191

5-1	Mémoire de fichiers .....	192
5-2	Traitement des fichiers .....	211
5-3	Utilisation de la mémoire de fichiers .....	247

## CHAPITRE 6

### Fonctions avancées..... 255

6-1	Temps de cycle et traitement à grande vitesse .....	257
6-2	Registres d'index .....	274
6-3	Communications en série .....	284
6-4	Modification du mode de mise à jour de la PV de temporisation/compteur.....	305
6-5	Utilisation d'une interruption programmée comme temporisation de haute précision (CJIM uniquement) .....	313
6-6	Paramètres du démarrage et maintenance.....	315
6-7	Fonctions de diagnostic.....	326
6-8	Modes de traitement UC.....	331
6-9	Mode prioritaire du traitement des périphériques.....	336
6-10	Fonctionnement sans batterie .....	342
6-11	Autres fonctions .....	344

## CHAPITRE 7

### Transfert du programme, essais de fonctionnement et débogage. 347

7-1	Transfert du programme .....	348
7-2	Essais de fonctionnement et débogage .....	348

## Annexes

A	Tableaux de comparaison des API : API série CJ, série CS, C200HG/HE/HX, CQM1H, CVM1 et série CV .....	357
B	Modifications par rapport aux systèmes Host Link précédents .....	383

## Index..... 387

## Historique des révisions..... 393

## A propos de ce manuel :

Ce manuel décrit la programmation des UC pour les automates programmables (API) série CS/CJ et comprend les chapitres présentés à la page suivante. Les séries CS et CJ sont subdivisées comme illustré dans le tableau suivant.

Carte	Série CS	Série CJ
UC	UC CS1-H : CS1H-CPU□□H CS1G-CPU□□H	UC CJ1-H : CJ1H-CPU□□H CJ1G-CPU□□H
	UC CS1 : CS1H-CPU□□-EV1 CS1G-CPU□□-EV1	UC CJ1 : CJ1G-CPU□□-EV1 UC CJAM : CJ1M-CPU□□
	UC CS1D : UC CS1D pour système à double UC : CS1D-CPU□□H UC CS1D pour système à simple UC : CS1D-CPU□□S UC de process CS1D : CS1D-CPU□□P	
Cartes d'E/S standard	Cartes d'E/S standard série CS	Cartes d'E/S standard série CJ
Cartes d'E/S spéciales	Cartes d'E/S spéciales série CS	Cartes d'E/S spéciales série CJ
Cartes réseau	Cartes réseau série CS	Cartes réseau série CJ
Cartes d'alimentation	Cartes d'alimentation série CS	Cartes d'alimentation série CJ

Veuillez lire ce manuel et tous les manuels répertoriés dans le tableau suivant et vous assurer d'avoir bien compris les informations qu'ils contiennent avant d'essayer d'installer ou d'utiliser des UC série CJ dans un système API.

Ce manuel contient les chapitres suivants :

**Chapitre 1** décrit la structure et le fonctionnement de base de l'UC ;

**Chapitre 2** décrit les informations de base nécessaires pour écrire, vérifier et saisir des programmes ;

**Chapitre 3** décrit les instructions à utiliser pour écrire des programmes utilisateur ;

**Chapitre 4** décrit le fonctionnement des tâches ;

**Chapitre 5** décrit les fonctions utilisées pour gérer la mémoire de fichiers ;

**Chapitre 6** fournit des informations plus détaillées sur les fonctions avancées : traitement à grande vitesse/temps de cycle, registres d'index, communications série, démarrage et maintenance, diagnostic et débogage, périphériques de programmation et temps de réponse d'entrée de la carte d'E/S standard CJ ;

**Chapitre 7** décrit les processus utilisés pour transférer le programme vers l'UC ainsi que les fonctions à utiliser pour tester et déboguer le programme.

Les **Annexes** fournissent une comparaison des séries CS/CJ, et indiquent les restrictions d'utilisation des cartes d'E/S spéciales C200H ainsi que les modifications apportées aux systèmes Host Link.

## A propos de ce manuel, suite

Nom	Cat. N°	Sommaire
Série SYSMAC CS/CJ CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1M-CPU□□, CJ1G-CPU□□P, CJ1G/H-CPU□□H Manuel de programmation des automates programmables	W394	Ce manuel décrit la programmation et les autres modes d'utilisation des fonctions des API série CS/CJ. (Ce manuel)
Série SYSMAC CS CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H Manuel d'utilisation des automates programmables	W339	Présente les grandes lignes et décrit la conception, l'installation, la maintenance et autres opérations de base des API série CS.
Série SYSMAC CJ CJ1G-CPU□□, CJ1M-CPU□□, CJ1G-CPU□□P, CJ1G/H-CPU□□H Manuel d'utilisation des automates programmables	W393	Présente les grandes lignes et décrit la conception, l'installation, la maintenance et autres opérations de base des API série CJ.
Série SYSMAC CJ CJ1M-CPU21/22/23 Manuel d'utilisation des fonctions E/S intégrées	W395	Décrit les fonctions des E/S intégrées des UC CJ1M.
Série SYSMAC CS UC CS1D-CPU□□H UC CS1D-CPU□□S Carte duplex CS1D-DPL01 Carte d'alimentation CS1D-PA207R Manuel d'utilisation du système duplex	W405	Présente les grandes lignes et décrit la conception, l'installation, la maintenance et autres opérations standard d'un système duplex basé sur les UC CS1D.
Série SYSMAC CS/CJ CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1M-CPU□□, CJ1G-CPU□□P, CJ1G/H-CPU□□H Manuel de référence des instructions des automates programmables	W340	Décrit les instructions de programmation du schéma contact pris en charge par les API série CS/CJ.
Série SYSMAC CS/CJ CQM1H-PRO01-E, C200H-PRO27-E, CQM1-PRO01-E Manuel d'utilisation des consoles de programmation	W341	Fournit des informations sur la programmation et l'utilisation des API série CS/CJ à l'aide d'une console de programmation.
Série SYSMAC CS/CJ CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1G/H-CPU□□H, CJ1G-CPU□□P, CJ1M-CPU□□, CS1W-SCB21-V1/41-V1, CS1W-SCU21-V1, CJ1W-SCU21-V1/41-V1 Manuel de référence des commandes de communication	W342	Décrit la série C (Host Link) et les commandes de communication FINS utilisées avec les API série CS/CJ.
SYSMAC WS02-CXP□□-E Manuel d'utilisation du CX-Programmer version 3.□	W414	Fournit des informations sur l'utilisation du CX-Programmer, un périphérique de programmation qui prend en charge les API série CS/CJ et le CX-Net présent dans CX-Programmer.
SYSMAC WS02-CXP□□-E Manuel d'utilisation du CX-Programmer version 4.□	W425	
SYSMAC WS02-CXP□□-E Manuel d'utilisation du CX-Programmer version 5.□	W437	

Nom	Cat. N°	Sommaire
SYSMAC WS02-CXP□□-E Manuel d'utilisation du CX-Programmer – Blocs de fonction	W438	Décrit les spécifications et les méthodes d'utilisation relatives aux blocs de fonction. Ces informations sont requises uniquement lorsque vous utilisez des blocs de fonction avec la combinaison de CX-Programmer ver. 5.0 et UC CS1-H/CJ1-H/CJ1M ver. 3.0. Reportez-vous au <i>Manuel d'utilisation du CX-Programmer version 5.□ (W437)</i> pour des informations détaillées sur les autres opérations du CX-Programmer ver. 5.0.
Série SYSMAC CS/CJ CS1W-SCB21-V1/41-V1, CS1W-SCU21-V1, CJ1W-SCU21-V1/41-V1 Manuel d'utilisation des cartes de communications série	W336	Décrit l'utilisation des cartes de communications série en vue d'effectuer des communications série avec des périphériques externes, y compris l'utilisation des protocoles système standards des produits OMRON.
SYSMAC WS02-PSTC1-E Manuel d'utilisation du CX-Protocol	W344	Décrit l'utilisation du CX-Protocol pour créer des macros protocole en tant que séquences de communication pour communiquer avec des périphériques externes.

**⚠ AVERTISSEMENT** Une lecture partielle ou une mauvaise compréhension des informations contenues dans ce manuel peut provoquer des dysfonctionnements ou endommager les appareils, et présente des risques de blessures corporelles voire un danger de mort. Veuillez lire attentivement chaque chapitre dans son intégralité. Il est essentiel de bien comprendre les informations de chaque chapitre et des chapitres qui lui sont associés avant d'entamer toute procédure ou opération.



# PRECAUTIONS

Ce chapitre présente les précautions générales à prendre lors de l'utilisation des automates programmables (API) séries CS/CJ et des périphériques associés.

**Les informations contenues dans cette section sont importantes pour garantir une utilisation fiable et sans danger des automates programmables. Avant d'essayer de configurer ou de faire fonctionner un système d'API, vous devez lire cette section et comprendre les informations qu'elle renferme.**

1	Public visé . . . . .	xxiv
2	Précautions générales . . . . .	xxiv
3	Précautions en matière de sécurité . . . . .	xxiv
4	Précautions liées à l'environnement d'utilisation. . . . .	xxvi
5	Précautions en matière d'application. . . . .	xxvi
6	Conformité aux directives CE. . . . .	xxx
6-1	Directives applicables . . . . .	xxx
6-2	Concepts . . . . .	xxx
6-3	Conformité aux directives CE. . . . .	xxxi
6-4	Méthodes de réduction des parasites des sorties relais. . . . .	xxxi

## 1 Public visé

Ce manuel est destiné aux personnes suivantes, qui doivent avoir des connaissances en matière de systèmes électriques (ingénieur électricien ou équivalent).

- Personnel chargé d'installer des systèmes d'automatisme.
- Personnel chargé de concevoir des systèmes d'automatisme.
- Personnel chargé de la gestion de sites et de systèmes d'automatisme.

## 2 Précautions générales

L'utilisateur doit se servir du produit conformément aux spécifications de performances énoncées dans les manuels d'utilisation.

Avant d'utiliser ce produit dans des conditions non décrites dans ce manuel ou d'appliquer le produit à des systèmes de contrôle nucléaire, des systèmes ferroviaires, des systèmes aéronautiques, des véhicules, des systèmes de combustion, des équipements médicaux, des machines de jeu, des équipements de sécurité et d'autres systèmes, machines et équipements susceptibles d'avoir des conséquences graves sur la vie et la propriété d'autrui en cas d'utilisation inadéquate, demandez conseil à votre revendeur OMRON.

Vérifiez que les caractéristiques nominales et les performances du produit sont suffisantes pour les systèmes, machines et équipements, et n'oubliez pas de munir les systèmes, machines et équipements de doubles mécanismes de sécurité.

Ce manuel fournit des informations sur la programmation et l'utilisation de la carte. Vous devez absolument lire ce manuel avant d'essayer d'utiliser la carte et le conserver à portée de main pour toute référence ultérieure pendant le fonctionnement du système.

**⚠ AVERTISSEMENT** Il est extrêmement important qu'un API et toutes les cartes API soient utilisés aux fins prévues et dans les conditions spécifiées, en particulier lorsqu'il s'agit d'applications susceptibles d'affecter directement ou indirectement la vie de l'homme. Avant d'utiliser un système d'API dans le cadre des applications mentionnées ci-dessus, vous devez impérativement consulter votre représentant OMRON.

## 3 Précautions en matière de sécurité

**⚠ AVERTISSEMENT** L'UC met à jour les E/S même lorsque le programme est à l'arrêt (c'est-à-dire, même en mode PROGRAM). Contrôlez les conditions de sécurité avant de modifier l'état de toute partie de mémoire réservée aux cartes d'E/S, aux cartes d'E/S spéciales ou aux cartes réseau. Tout changement des données assignées à une carte risque de provoquer un fonctionnement inattendu des charges connectées à la carte. Chacune des opérations suivantes peut provoquer un changement de l'état de la mémoire.

- Transfert de données de la mémoire E/S vers l'UC depuis un périphérique de programmation.
- Modification des valeurs actuelles de la mémoire depuis un périphérique de programmation.
- Configuration/réinitialisation forcée des bits depuis un périphérique de programmation.
- Transfert de fichiers de la mémoire E/S à partir d'une carte mémoire ou de la mémoire de fichiers EM vers l'UC.
- Transfert de la mémoire E/S depuis un ordinateur hôte ou un autre API sur un réseau.

**⚠ AVERTISSEMENT** N'essayez jamais de démonter une carte alors qu'elle est sous tension. Vous pourriez en effet recevoir une décharge électrique.

- ⚠ AVERTISSEMENT** Ne touchez jamais les bornes ou borniers pendant que le système est sous tension. Vous pourriez en effet recevoir une décharge électrique.
- ⚠ AVERTISSEMENT** N'essayez jamais de démonter, de réparer ou de modifier une unité quelconque sous peine de provoquer un dysfonctionnement, un incendie ou une décharge électrique.
- ⚠ AVERTISSEMENT** Prévoyez des mesures de sécurité pour les circuits externes (extérieurs à l'automate programmable), y compris dans les articles suivants, afin de garantir la sécurité du système si une anomalie intervient à la suite d'un dysfonctionnement de l'API ou d'un autre facteur externe affectant le fonctionnement de l'API. Le non-respect de cet avertissement pourrait provoquer des accidents graves.
- Des circuits d'arrêt d'urgence, des circuits à verrouillage, des interrupteurs de fin de course et autres mesures de sécurité similaires doivent être utilisés avec tous les circuits de contrôle externes.
  - L'API désactive toutes les sorties lorsque sa fonction de diagnostic automatique détecte une erreur ou qu'une instruction d'alarme de défaillance grave (FALS) est exécutée. Pour éviter de telles erreurs, des mesures de sécurité externes doivent être prises pour assurer la sécurité du système.
  - Les sorties de l'API peuvent rester sur ON ou sur OFF en raison de l'encrassement ou de la surchauffe des relais de sortie ou de la destruction des transistors de sortie. Pour se prémunir contre ce type de problèmes, des mesures de sécurité externes doivent être prises pour assurer la sécurité du système.
  - Lorsque la sortie de 24 Vc.c. (alimentation électrique de service de l'API) est surchargée ou court-circuitée, il peut y avoir une baisse de tension entraînant une désactivation des sorties. Pour se prémunir contre ce type de problèmes, des mesures de sécurité externes doivent être prises pour assurer la sécurité du système.
- ⚠ Attention** Vérifiez la sécurité avant de transférer des fichiers de données stockés dans la mémoire de fichiers (carte mémoire ou mémoire de fichiers EM) vers la zone E/S (CIO) de l'UC à l'aide d'un outil périphérique. Sinon, les périphériques reliés à la carte de sortie risquent de connaître des dysfonctionnements indépendamment du mode d'opération de l'UC.
- ⚠ Attention** Des mesures de sécurité doivent être prises par le client pour garantir la sécurité dans le cas de signaux incorrects, manquants ou anormaux dus à une rupture des lignes de signaux, à des interruptions momentanées de l'alimentation ou à d'autres causes. Un fonctionnement anormal pourrait provoquer des accidents graves.
- ⚠ Attention** Les UC CS1-H, CJ1-H, CJ1M et CS1D enregistrent automatiquement le programme utilisateur ainsi que les données des paramètres dans la mémoire flash lorsque ces données sont écrites sur l'UC. Cependant, la mémoire E/S (y compris les zones DM, EM et HR) n'est pas sauvegardée dans la mémoire flash. Il est possible de conserver les zones DM, EM et HR en cas d'interruption d'alimentation grâce à une batterie. En cas d'erreur de batterie, il se pourrait que le contenu de ces zones ne soit pas exact après une interruption d'alimentation. Si le contenu des zones DM, EM et HR sert à contrôler les sorties externes, évitez des sorties inadéquates chaque fois que le drapeau d'erreur de batterie (A40204) est activé.
- ⚠ Attention** N'exécutez une édition en ligne qu'après vous être assuré que cette opération n'aura pas d'effets néfastes par suite de l'allongement du temps de cycle. Sinon, les signaux d'entrée risquent d'être illisibles.
- ⚠ Attention** Assurez-vous de la sécurité du nœud de destination avant de transférer un programme vers un autre nœud ou de modifier le contenu de la zone de mémoire E/S. Exécuter l'une de ces opérations sans vérifier la sécurité risque de provoquer des blessures.

-  **Attention** Serrez les vis du bornier de la carte d'alimentation c.a. en respectant le couple spécifié dans le manuel d'utilisation. Des vis mal serrées risquent de provoquer une surchauffe ou un dysfonctionnement.
-  **Attention** Ne touchez pas la carte d'alimentation alors qu'elle est sous tension ou juste après la mise hors tension. Elle risque d'être chaude et de vous brûler.
-  **Attention** Faites attention lorsque vous connectez des ordinateurs personnels ou d'autres périphériques à un API sur lequel est montée une unité non isolée (CS1W-CLK12/52(-V1) ou CS1W-ETN01) connectée à une alimentation externe. Si le côté 24 V de l'alimentation externe est raccordé à la terre et que le côté 0 V de l'appareil périphérique l'est aussi, vous produirez un court-circuit. Lorsque vous connectez un appareil périphérique à ce type d'API, vous pouvez soit raccorder à la terre le côté 0 V de l'alimentation externe, soit ne pas raccorder du tout à la terre l'alimentation externe.

## 4 Précautions liées à l'environnement d'utilisation

-  **Attention** N'utilisez pas le système de contrôle dans les endroits suivants :
- Endroits exposés à la lumière directe du soleil.
  - Endroits soumis à des températures ou des taux d'humidité en dehors des valeurs indiquées dans les spécifications.
  - Endroits soumis à la condensation due à des variations importantes de températures.
  - Endroits en contact avec des gaz corrosifs ou inflammables.
  - Endroits soumis à la poussière (en particulier, la limaille de fer) ou au contact de sels.
  - Endroits où l'on utilise de l'eau, de l'huile ou des produits chimiques.
  - Endroits soumis à des chocs ou des vibrations importantes.
-  **Attention** Prenez les mesures de protection appropriées et suffisantes lorsque vous installez des systèmes dans les endroits suivants :
- Endroits soumis à l'électricité statique ou à d'autres formes de parasites.
  - Endroits soumis à des champs électromagnétiques intenses.
  - Endroits susceptibles d'être exposés à la radioactivité.
  - Endroits proches de sources d'alimentation.
-  **Attention** L'environnement d'utilisation du système d'API peut affecter fortement sa longévité et sa fiabilité. Un environnement d'utilisation hostile peut provoquer des dysfonctionnements, des défaillances et d'autres problèmes imprévisibles au niveau du système API. Vérifiez bien qu'au moment de l'installation, l'environnement d'utilisation est conforme aux conditions spécifiées et qu'il présente toujours les mêmes conditions tout au long de la durée de vie du système.

## 5 Précautions en matière d'application

Observez les précautions suivantes lorsque vous utilisez le système API.

- Vous devez utiliser CX-Programmer (logiciel de programmation qui fonctionne sous Windows) si vous devez programmer plusieurs tâches. La console de programmation peut uniquement être utilisée pour programmer une seule tâche cyclique et des tâches d'interruption. Elle peut toutefois être utilisée pour éditer des programmes multitâches créés initialement avec CX-Programmer.

**⚠ AVERTISSEMENT** Suivez toujours ces précautions. Le non-respect des précautions suivantes pourrait entraîner des blessures graves, voire mortelles.

- Effectuez toujours une mise à la terre de 100  $\Omega$  ou moins lors de l'installation des cartes. L'absence d'une mise à la terre de 100  $\Omega$  ou moins risque de provoquer des décharges électriques.
- Une mise à la terre de 100  $\Omega$  ou moins doit être installée lors du court-circuitage des bornes GR et LG sur la carte d'alimentation.
- Mettez toujours l'API hors tension avant de tenter d'effectuer l'une des opérations suivantes. Si vous ne le faites pas, vous risquez de provoquer un dysfonctionnement ou de recevoir une décharge électrique.
  - Montage ou démontage des cartes d'alimentation, des cartes d'E/S, des UC ou des autres cartes.
  - Assemblage des cartes.
  - Réglage des interrupteurs DIO et autres commutateurs rotatifs.
  - Connexion des câbles ou câblage du système.
  - Connexion ou déconnexion des connecteurs.

**⚠ Attention** Le non-respect des précautions suivantes peut provoquer un dysfonctionnement de l'API ou du système ou endommager l'API ou les cartes API. Suivez toujours ces précautions.

- Les données du programme utilisateur et de la zone de paramètre des UC CS1-H, CS1D, CJ1-H et CJ1M sont sauvegardées dans la mémoire flash intégrée. Le voyant BKUP s'allume sur le devant de l'UC pendant que l'opération de sauvegarde est en cours. Ne pas mettre l'UC hors tension lorsque le voyant BKUP est allumé. Les données ne seront pas sauvegardées si l'alimentation est coupée.
- Lorsque vous utilisez une UC CS1 série CS la première fois, installez la batterie CS1W-BAT1 fournie avec la carte et effacez toutes les zones de mémoire à partir d'un périphérique de programmation avant de commencer à programmer. Lorsque vous utilisez l'horloge interne, mettez le système sous tension après avoir installé la batterie et réglez l'horloge à partir d'un périphérique de programmation ou à l'aide de l'instruction DATE(735). L'horloge ne démarrera pas tant que l'heure n'aura pas été réglée.
- Lorsque l'UC sort de l'usine, l'API est configuré de manière à ce que l'UC démarre dans le mode de fonctionnement réglé sur le commutateur de mode de la console de programmation. Lorsque aucune console de programmation n'est connectée, une UC CS1 série CS démarrera en mode PROGRAM mais une UC CS1-H, CS1D, CJ1, CJ1-H ou CJ1M démarrera en mode RUN et commencera à fonctionner immédiatement. Que ce soit exprès ou pas, évitez de lancer le fonctionnement sans être sûr de sa sécurité.
- Lorsque vous créez un fichier AUTOEXEC.IOM depuis un périphérique de programmation (console de programmation ou CX-Programmer) pour transférer automatiquement des données au démarrage, configurez la première adresse en écriture sur D20000 et assurez-vous que la taille des données écrites n'excède pas la taille de la zone DM. Lorsque le fichier de données est lu depuis la carte mémoire au démarrage, les données sont écrites sur l'UC en commençant par l'adresse D20000 et ce, même si une autre adresse a été configurée lors de la création du fichier AUTOEXEC.IOM. De plus, si la zone DM est saturée (ce qui est possible lorsque CX-Programmer est utilisé), les données restantes sont écrites sur la zone EM.

- Mettez toujours l'API sous tension avant de brancher le système de contrôle. Si l'API est mis sous tension après le système de contrôle, des erreurs temporaires risquent de se produire au niveau des signaux du système de contrôle car les bornes de sortie des cartes de sorties c.c. et des autres cartes sont momentanément activées lors de la mise sous tension de l'API.
- Des mesures de sécurité doivent être prises par le client pour garantir la sécurité dans le cas où les sorties de cartes de sorties restent sur ON à la suite d'un dysfonctionnement interne qui peut survenir au niveau des relais, des transistors et des autres éléments.
- Des mesures de sécurité doivent être prises par le client pour garantir la sécurité dans le cas de signaux incorrects, manquants ou anormaux dus à une rupture des lignes de signaux, à des interruptions momentanées de l'alimentation ou à d'autres causes.
- Des circuits à verrouillage, des interrupteurs de fin de course et des mesures de sécurité similaires dans les circuits externes (c'est-à-dire, extérieurs à l'automate programmable) doivent être fournis par le client.
- Ne débranchez pas l'API pendant le transfert de données. Plus particulièrement, ne débranchez pas l'alimentation lors de la lecture ou de l'écriture d'une carte mémoire. De même, ne retirez pas la carte mémoire lorsque le voyant BUSY est allumé. Pour retirer une carte mémoire, appuyez d'abord sur le commutateur d'alimentation de la carte mémoire et attendez que le voyant BUSY s'éteigne avant de la retirer.
- Si le bit de maintien E/S est sur ON, les sorties de l'API ne sont pas désactivées et conservent leur état précédent lorsque l'API passe du mode RUN ou MONITOR au mode PROGRAM. Assurez-vous que les charges externes ne produisent pas de situations dangereuses lorsque ceci survient. (Lorsque le fonctionnement s'arrête à la suite d'une erreur fatale, y compris celles dues à l'instruction FALS(007), toutes les sorties de la carte de sortie passent à OFF et seul l'état des sorties internes est conservé.)
- Le contenu des zones DM, EM et HR de l'UC est sauvegardé grâce à une batterie. Si la tension de la batterie diminue, ces données risquent d'être perdues. Utilisez des contre-mesures dans le programme en utilisant le drapeau d'erreur de batterie (A40204) pour réinitialiser les données ou pour prendre d'autres dispositions en cas de diminution de la tension de la batterie.
- Si la tension d'alimentation est de 200 à 240 Vc.a. avec un API série CS, enlevez toujours le cavalier métallique des borniers sélecteurs de tension sur la carte d'alimentation (excepté pour les cartes d'alimentation à caractéristiques techniques étendues). Si le cavalier métallique n'est pas enlevé et que l'alimentation est de 200 à 240 Vc.a., le produit sera détruit.
- Utilisez toujours la tension d'alimentation spécifiée dans les manuels d'utilisation. Une tension incorrecte peut provoquer un dysfonctionnement ou une surchauffe.
- Prenez les mesures appropriées pour vous assurer que l'alimentation indiquée est conforme à la tension et à la fréquence nominales. Faire particulièrement attention aux lieux où l'alimentation électrique est instable. Une alimentation incorrecte peut provoquer un dysfonctionnement.
- Installez des disjoncteurs externes et prenez d'autres mesures de sécurité contre d'éventuels courts-circuits au niveau du câblage externe. Des mesures de sécurité insuffisantes contre les courts-circuits peuvent provoquer une surchauffe.
- N'appliquez jamais une tension supérieure à la tension nominale d'entrée aux cartes d'entrées. Une tension excessive peut provoquer une surchauffe.
- N'appliquez pas de tension et ne connectez pas de charges aux cartes de sorties qui dépassent la capacité maximale de commutation. Une tension excessive peut provoquer une surchauffe.
- Séparez la borne de terre des lignes (LG) de la borne de terre fonctionnelle (GR) sur la carte d'alimentation avant de réaliser les essais de tension de maintien ou de résistance d'isolement. Le non-respect de cette consigne peut provoquer une surchauffe..

- Installez la carte correctement, comme indiqué dans les manuels d'utilisation. Une installation incorrecte de la carte peut provoquer un dysfonctionnement.
- Dans le cas des API série CS, assurez-vous que toutes les vis de montage de la carte et du rack arrière sont serrées au couple spécifié dans les manuels correspondants. Un serrage à un couple incorrect peut provoquer un dysfonctionnement.
- Assurez-vous que toutes les vis des bornes et des connecteurs de câble sont serrées au couple spécifié dans les manuels correspondants. Un serrage à un couple incorrect peut provoquer un dysfonctionnement.
- Laissez l'étiquette sur la carte pendant le câblage. Le retrait de l'étiquette peut provoquer un dysfonctionnement si les corps étrangers pénètrent dans la carte.
- Lorsque le câblage est terminé, enlevez l'étiquette pour assurer une bonne dissipation de la chaleur. Laisser l'étiquette peut provoquer un dysfonctionnement.
- Utilisez des bornes serties pour effectuer le câblage. Ne pas raccorder directement des fils toronnés nus à des bornes. Le raccordement de fils multibrins nus peut provoquer une surchauffe.
- Câblez correctement toutes les connexions.
- Vérifiez deux fois le câblage et les paramètres d'interrupteur avant de mettre l'API sous tension. Un câblage incorrect peut provoquer une surchauffe.
- Installez les unités uniquement après avoir complètement vérifié les borniers et les connecteurs.
- Vérifiez que les borniers, les cartes mémoire, les câbles d'extension et autres éléments avec périphériques de verrouillage sont bien connectés. Un verrouillage incorrect peut provoquer un dysfonctionnement.
- Vérifiez les paramètres des interrupteurs, le contenu de la zone DM et d'autres préparatifs avant de commencer à utiliser la carte. Commencer à utiliser la carte sans les réglages ou les données appropriés peut provoquer un fonctionnement inattendu.
- Vérifiez le fonctionnement correct du programme utilisateur avant de l'exécuter sur la carte. Ne pas vérifier le programme peut provoquer un fonctionnement inattendu.
- Assurez-vous que le système ne sera pas perturbé avant de lancer l'une des opérations suivantes. Sans cela, le système risque de fonctionner de façon imprévisible.
  - Modification du mode de fonctionnement de l'API.
  - Configuration/réinitialisation forcée d'un bit en mémoire.
  - Modification de la valeur actuelle d'un mot ou d'une valeur définie dans la mémoire.
- Ne tirez pas sur les câbles et ne les pliez pas au-delà des limites qu'ils peuvent normalement supporter. Sinon, vous risquez de casser les câbles.
- Ne posez aucun objet sur les câbles ou les lignes électriques. Vous pourriez en effet casser les câbles.
- Ne pas utiliser les câbles RS-232C pour ordinateurs disponibles dans le commerce. Toujours utiliser les câbles répertoriés dans ce manuel ou assembler un câble en respectant les caractéristiques techniques du manuel. L'utilisation de câbles disponibles dans le commerce risque d'endommager les périphériques externes ou l'UC.
- Ne connectez jamais la broche 6 (alimentation 5 V) du port RS-232C de l'UC à un autre périphérique qu'un adaptateur NT-AL001 ou CJ1W-CIF11. Cela pourrait endommager le périphérique externe ou l'UC.
- Lorsque vous remplacez des pièces, vérifiez que les caractéristiques nominales des pièces neuves sont correctes. Une différence risque de provoquer un dysfonctionnement ou une surchauffe.

- Avant de toucher une carte, touchez d'abord un objet métallique relié à la terre afin de vous décharger de toute l'électricité statique qui a pu s'accumuler. Le non-respect de cette règle peut provoquer un dysfonctionnement ou des dommages.
- Lors du transport ou du stockage des cartes, couvrez-les d'un matériau antistatique pour les protéger contre l'électricité statique et maintenez la température de stockage appropriée.
- Ne touchez pas les cartes circuit ou les composants montés avec vos mains nues. Des fils pointus et d'autres parties des cartes peuvent causer des blessures en cas de manipulation incorrecte.
- Ne court-circuitez pas les bornes de la batterie et prenez garde de ne pas charger, démonter, chauffer ou incinérer la batterie. Ne soumettez pas la batterie à des chocs importants. Le non-respect de ces consignes peut provoquer une fuite, une rupture, une génération de chaleur ou l'ignition de la batterie. Débarrassez-vous des batteries qui sont tombées sur le sol ou qui ont été soumises à des chocs excessifs. Le liquide des batteries qui ont été soumises à des chocs risque de s'écouler lors de l'utilisation de la batterie.
- Les normes UL exigent que les batteries soient remplacées seulement par des techniciens expérimentés. Ne permettez pas à des personnes non qualifiées de les remplacer.
- Sur une UC série CJ, les cliquets du haut et du bas de la carte d'alimentation, de l'UC, des cartes d'E/S, des cartes d'E/S spéciales et des cartes réseau doivent être complètement verrouillés (jusqu'à ce qu'ils soient bien en place). L'unité ne fonctionnera pas correctement si les cliquets ne sont pas bien en place.
- Dans le cas d'un API série CJ, connectez toujours la plaque d'extrémité à l'unité, à l'extrême droite de l'API. Sans cette plaque, l'API ne fonctionnera pas correctement.
- Un fonctionnement inattendu peut survenir si des tableaux de liaison de données ou des paramètres inappropriés sont configurés. Même si les données de liaison des données et les paramètres corrects ont été définis, assurez-vous que le système de contrôle ne risque pas d'être endommagé avant de lancer ou d'interrompre les liaisons de données.
- Les cartes réseau redémarrent lorsque les tableaux de routage sont transférés depuis un périphérique de programmation vers l'UC. Le redémarrage de ces cartes est nécessaire pour lire et activer les nouveaux tableaux de routage. Assurez-vous que le système ne risque pas d'être endommagé avant de permettre la réinitialisation des cartes réseau.

## 6 Conformité aux directives CE

### 6-1 Directives applicables

- Directives relatives à la compatibilité électromagnétique (CEM)
- Directive relative aux basses tensions

### 6-2 Concepts

#### **Directives sur la CEM**

Les appareils OMRON qui sont en conformité avec les directives CE respectent également les normes de la CEM correspondantes ce qui facilite leur intégration dans d'autres dispositifs ou dans une machine. Les produits commercialisés ont fait l'objet d'un contrôle de conformité avec les normes de la CEM (voir la remarque suivante). C'est au client qu'il appartient de s'assurer que les produits sont en conformité avec les normes du système qu'il utilise.

Les performances liées à la CEM des dispositifs OMRON qui sont en conformité avec les directives CE varient selon la configuration, le câblage et d'autres particularités de l'équipement et du panneau de commande sur lesquels sont installés les dispositifs OMRON. Le client doit dès lors effectuer

un contrôle final pour s'assurer que les dispositifs et l'ensemble de la machine sont en conformité avec les normes de la CEM.

**Remarque** Les normes CEM (Compatibilité électromagnétique) d'application sont les suivantes :

EMS (Electromagnetic Susceptibility, susceptibilité électromagnétique) :

Série CS : EN61131-2 et EN61000-6-2

Série CJ : EN61000-6-2

(Electromagnetic Interference, interférence électromagnétique) :

EN61000-6-4

(Emission de radiations : réglementation 10 m)

#### **Directive sur la basse tension**

Assurez-vous toujours que les dispositifs qui fonctionnent à des tensions comprises entre 50 et 1 000 Vc.a. et 75 à 1 500 Vc.c. respectent les normes de sécurité requises pour l'API (EN61131-2).

### **6-3 Conformité aux directives CE**

Les API série CS/CJ sont conformes aux directives CE. Pour garantir que la machine ou le dispositif sur lequel est utilisé l'API série CS/CJ respecte les directives CE, l'API doit être installé comme suit :

- 1,2,3...**
1. L'API série CS/CJ doit être installé dans un panneau de commande.
  2. Pour les alimentations c.c. connectées à des cartes d'alimentation c.c. et des cartes d'E/S, utilisez une isolation renforcée ou double.
  3. Les API série CJ conformes aux directives CE respectent également la norme d'émission commune (EN61000-6-4). Les caractéristiques des radiations (réglementations de 10 m) peuvent varier en fonction de la configuration du panneau de commande utilisé, des autres périphériques connectés à celui-ci, du câblage et d'autres facteurs. Vous devez dès lors vous assurer que l'ensemble de la machine ou de l'équipement est conforme aux dispositions des directives CE.

### **6-4 Méthodes de réduction des parasites des sorties relais**

Les API série CJ sont conformes aux normes relatives aux émissions communes (EN61000-6-4) des directives sur la CEM. Cependant, il est possible que les parasites générés par la commutation de la sortie relais ne respectent pas ces normes. Dans ce cas, vous devez connecter un filtre antiparasite au côté charge ou prendre d'autres mesures spécifiques à l'extérieur de l'API

Les contre-mesures prises pour être en conformité avec les normes varient en fonction des périphériques qui sont du côté charge, du câblage, de la configuration des machines, etc. Les exemples suivants décrivent des contre-mesures permettant de réduire les parasites générés.

#### **Contre-mesures**

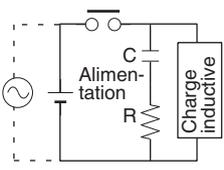
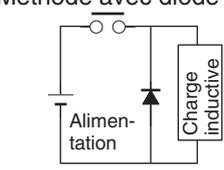
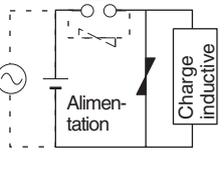
(Pour plus de détails, reportez-vous à la norme EN61000-6-4.)

Les contre-mesures sont inutiles si la fréquence de commutation de la charge de l'ensemble du système - API inclus - est inférieure à 5 fois par minute.

Des contre-mesures sont obligatoires si la fréquence de commutation de la charge pour l'ensemble du système - API inclus - est supérieure à 5 fois par minute.

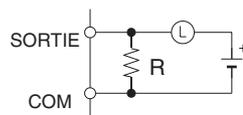
**Exemples de contre-mesures**

En cas de commutation d'une charge inductive, connectez un limiteur de tension, des diodes, etc. en parallèle avec la charge ou le contact, comme illustré ci-dessous.

Circuit	Courant		Caractéristique	Élément requis
	c.a.	c.c.		
<p>Méthode CR</p> 	Oui	Oui	<p>Si la charge est un relais ou un solénoïde, il y a un décalage entre l'ouverture du circuit et la réinitialisation de la charge.</p> <p>Si la tension d'alimentation est de 24 ou 48 V, placez le limiteur de tension en parallèle avec la charge. Si la tension d'alimentation est comprise entre 100 et 200 V, placez le limiteur entre les contacts.</p>	<p>La capacité du condensateur doit être de 1 à 0,5 <math>\mu\text{F}</math> pour un courant de contact de 1 A et la résistance du condensateur doit être de 0,5 à 1 <math>\Omega</math> pour une tension de contact de 1 V. Toutefois, ces valeurs peuvent varier en fonction de la charge et des caractéristiques du relais. Ces valeurs doivent être choisies à partir d'expérimentations en tenant compte du fait que la capacité supprime la décharge à étincelles lorsque les contacts sont séparés et que la résistance limite le courant qui circule dans la charge lorsque le circuit est à nouveau fermé.</p> <p>La rigidité diélectrique du condensateur doit être comprise entre 200 et 300 V. S'il s'agit d'un circuit c.a., utilisez un condensateur sans polarité.</p>
<p>Méthode avec diode</p> 	Non	Oui	<p>La diode connectée en parallèle avec la charge transforme l'énergie accumulée par la bobine en courant, qui circule alors dans la bobine afin d'être converti en chaleur par la résistance de la charge inductive.</p> <p>Le décalage entre l'ouverture du circuit et la réinitialisation de la charge, qui est provoqué par cette méthode, est plus long que celui obtenu par la méthode CR.</p>	<p>La valeur de la rigidité diélectrique inverse de la diode doit être au moins 10 fois plus grande que la valeur de la tension du circuit. Le courant direct de la diode doit être supérieur ou égal au courant de la charge.</p> <p>La valeur de la rigidité diélectrique inverse de la diode peut être deux ou trois fois plus grande que la tension d'alimentation si le limiteur de tension est appliqué à des circuits électroniques présentant de faibles tensions de circuit.</p>
<p>Méthode à varistor</p> 	Oui	Oui	<p>La méthode du varistor empêche d'appliquer une tension élevée entre les contacts grâce à la caractéristique de tension constante du varistor. Il y a un décalage entre l'ouverture du circuit et la réinitialisation de la charge.</p> <p>Si la tension d'alimentation est de 24 ou 48 V, placez le varistor en parallèle avec la charge. Si la tension d'alimentation est comprise entre 100 et 200 V, placez-le entre les contacts.</p>	---

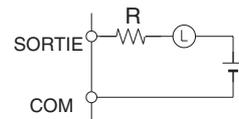
Lorsque vous commutez une charge avec un courant élevé induit tel qu'une lampe à incandescence, supprimez le courant induit comme illustré ci-dessous.

**Contre-mesure 1**



Fourniture d'un courant d'obscurité d'environ un tiers de la valeur nominale via une lampe à incandescence.

**Contre-mesure 2**



Fourniture d'une résistance de limitation

# CHAPITRE 1

## Fonctionnement de l'UC

Ce chapitre décrit la structure et le fonctionnement de base de l'UC.

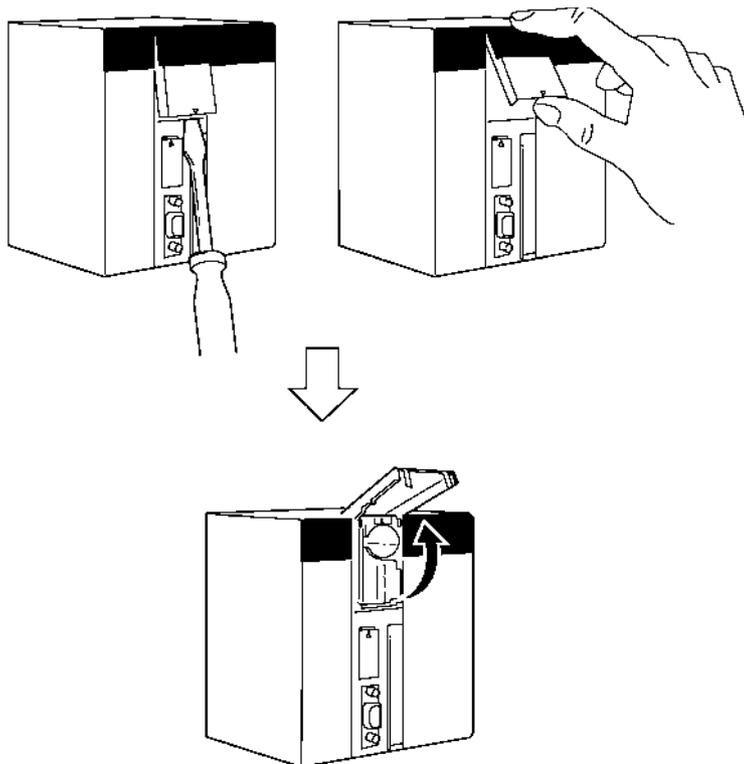
1-1	Configuration initiale (UC CS1 uniquement) . . . . .	2
1-2	Utilisation de l'horloge interne (UC CS1 uniquement). . . . .	5
1-3	Structure interne de l'UC. . . . .	6
1-3-1	Présentation. . . . .	6
1-3-2	Diagramme des blocs de la mémoire de l'UC. . . . .	7
1-4	Modes de fonctionnement. . . . .	8
1-4-1	Description des modes de fonctionnement . . . . .	8
1-4-2	Initialisation de la mémoire E/S . . . . .	10
1-4-3	Mode de démarrage . . . . .	11
1-5	Programmes et tâches . . . . .	12
1-6	Description des tâches. . . . .	14

## 1-1 Configuration initiale (UC CS1 uniquement)

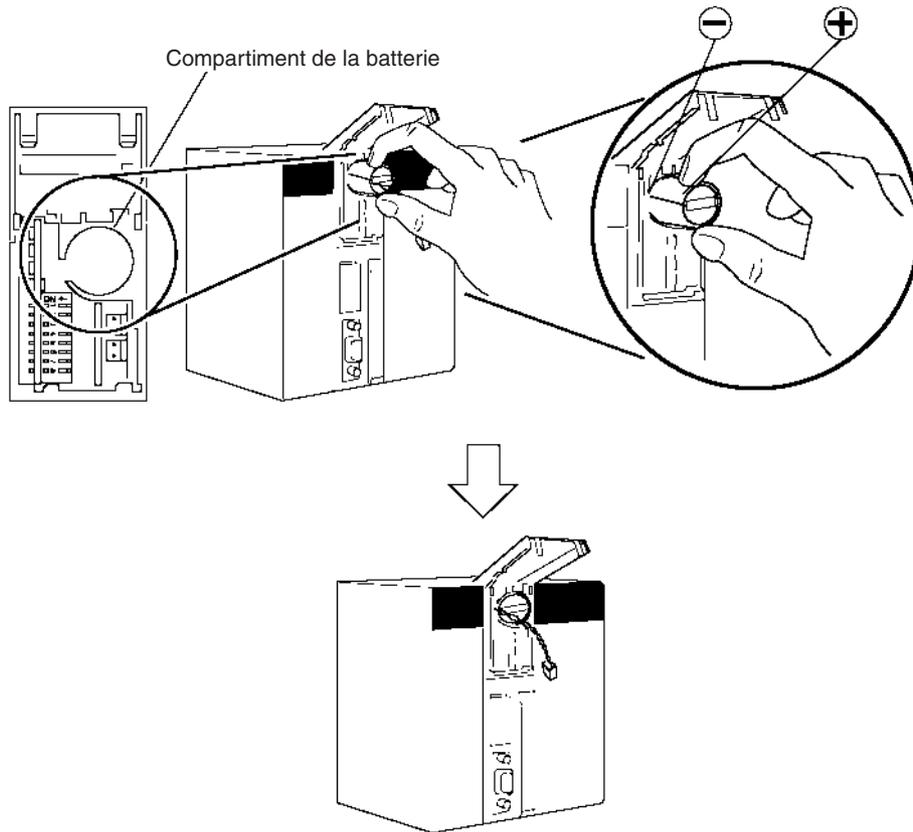
### Installation de la batterie

Avant d'utiliser l'UC CS1, vous devez installer la batterie dans l'UC en respectant la procédure suivante.

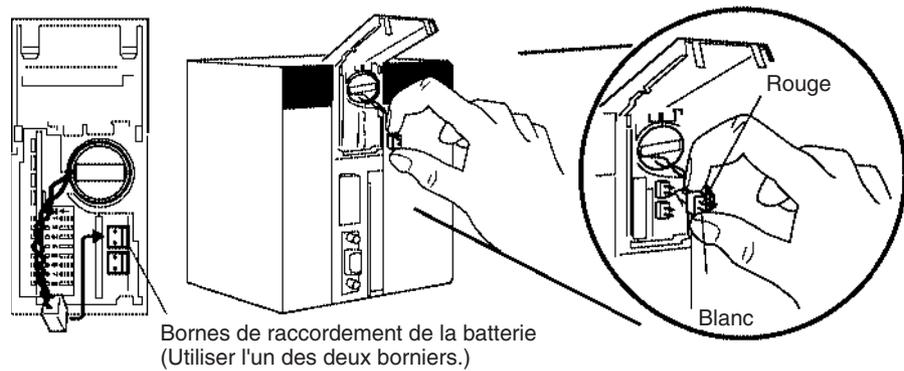
- 1,2,3... 1. Insérez un tournevis plat dans le petit trou du bas du compartiment de la batterie et tirez le couvercle vers le haut pour l'ouvrir.



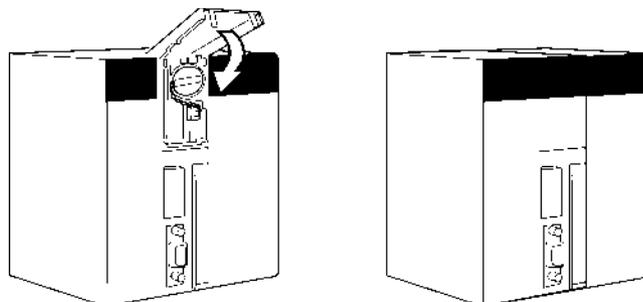
2. Tenez la batterie, le câble vers l'extérieur, et insérez-la dans le compartiment.



3. Branchez le connecteur de la batterie dans la borne du connecteur de batterie. Connectez le fil rouge dans le bornier du haut et le fil blanc dans le bornier du bas. Il y a deux borniers de connecteur de batterie ; connectez la batterie dans l'un des deux. Peu importe celui que vous utilisez.



4. Pliez le câble et fermez le couvercle.



**Effacement de la mémoire** Après avoir installé la batterie, effacez la mémoire grâce à la fonction d'effacement de la mémoire afin d'initialiser la mémoire RAM de l'UC.

**Console de programmation**

A partir d'une console de programmation, procédez comme suit :



**Remarque** Lorsque vous effacez la mémoire à partir d'une console de programmation, vous ne pouvez pas spécifier plus d'une tâche cyclique. Vous pouvez spécifier une tâche cyclique et une tâche d'interruption, ou une tâche cyclique et pas de tâche d'interruption. Consultez le *Manuel de fonctionnement* pour plus d'informations sur l'effacement de la mémoire. Consultez *CHAPITRE 1 Fonctionnement de l'UC* et *CHAPITRE 4 Tâches* pour plus d'informations sur les tâches.

**CX-Programmer**

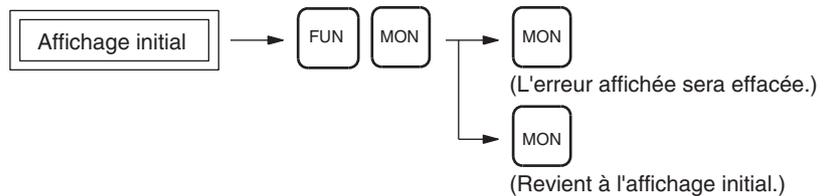
Il est également possible d'effacer la mémoire à partir de CX-Programmer. Veuillez vous reporter au *Manuel d'utilisation de CX-Programmer* pour de plus amples informations sur les procédures actuelles.

**Erreurs d'effacement**

Après avoir effacé la mémoire, effacez toutes les erreurs de l'UC, y compris les erreurs de tension de batterie faible.

**Console de programmation**

A partir d'une console de programmation, procédez comme suit :



**CX-Programmer**

Il est également possible d'effacer les erreurs à partir de CX-Programmer. Veuillez vous reporter au *Manuel d'utilisation de CX-Programmer* pour de plus amples informations sur les procédures actuelles.

**Remarque** Lorsqu'une carte interne est montée, il se peut qu'une erreur de table de routage de la carte interne apparaisse toujours, même après avoir annulé l'erreur à l'aide de CX-Programmer. (A42407 sera sur ON pour une carte de communications série.) Si cela se produit, remettez l'appareil sous tension ou redémarrez la carte interne, puis annulez une nouvelle fois l'erreur.

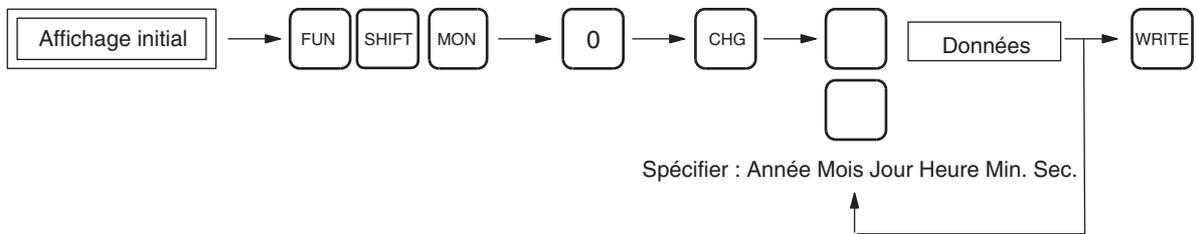
## 1-2 Utilisation de l'horloge interne (UC CS1 uniquement)

Lorsque la batterie est installée dans l'UC série CS, les paramètres de l'horloge interne de l'UC sont les suivants : année 00, mois 01, jour 01 (00-01-01), 00 heure, 00 minute, 00 seconde (00:00:00) et Dimanche (SUN).

Pour utiliser l'horloge interne, mettez le système sous tension après avoir installé la batterie et 1) utilisez un périphérique de programmation (console de programmation ou CX-Programmer) pour définir l'heure, 2) exécutez l'instruction CLOCK ADJUSTMENT (DATE) ou 3) envoyez une commande FINS pour lancer l'horloge interne à partir de l'heure et la date actuelle correcte.

Vous trouverez ci-dessous la procédure d'utilisation de la console de programmation pour régler l'horloge interne.

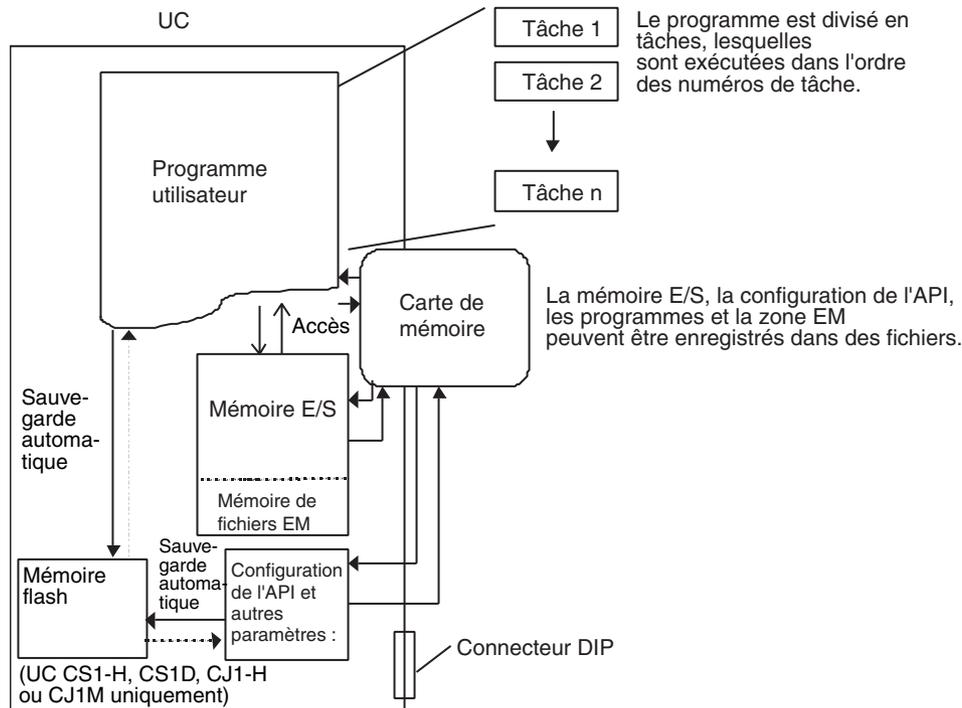
### Séquence des opérations



## 1-3 Structure interne de l'UC

### 1-3-1 Présentation

Le diagramme suivant décrit la structure interne de l'UC.



#### Le programme utilisateur

Le programme utilisateur est créé à partir de 288 tâches de programme maximum, y compris les tâches d'interruption. Les tâches sont transférées vers l'UC à partir du logiciel de programmation CX-Programmer.

Il existe deux types de tâche : une tâche cyclique, exécutée une fois par cycle (maximum 32), et une tâche d'interruption, exécutée uniquement lorsque se produisent les conditions d'interruption (maximum 256). Les tâches cycliques sont exécutées dans l'ordre numérique.

#### Remarque

1. Dans le cas d'une UC CS1-H, CJ1-H, CJ1M ou CS1D, les tâches d'interruption peuvent être exécutées de manière cyclique, comme les tâches cycliques. On les appelle alors des « tâches cycliques extra ». Le nombre total de tâches pouvant être exécutées de manière cyclique doit être de maximum 288.
2. Avec CX-Programmer, utilisez la version 2.1 ou supérieure pour une UC CS1-H ou CJ1-H et la version 3.0 ou supérieure pour une UC CJ1M (excepté pour les modèles économiques) ou l'UC CS1D pour les systèmes d'UC en duplex. Lorsque vous utilisez une UC CJ1M bas de gamme (CJ1M-CPU11/CPU21), utilisez la version 3.3 ou supérieure de CX-Programmer. Si vous utilisez une UC CS1-H, CJ1-H CJ1M ou CS1D version 2.0 ou supérieure, utilisez CX-Programmer version 4.0 ou supérieure.

Les instructions du programme sont lues et écrites dans la mémoire E/S et sont exécutées en commençant par le haut du programme. Lorsque toutes les tâches cycliques sont exécutées, les E/S de toutes les unités sont mises à jour et le cycle recommence en débutant au numéro de tâche cyclique le plus bas.

Consultez le chapitre sur le fonctionnement de l'UC dans le *Manuel d'utilisation de la série CS/CJ* pour plus de détails sur la mise à jour d'E/S.

<b>Mémoire E/S</b>	<p>La mémoire E/S est la zone RAM utilisée pour lire et écrire à partir du programme utilisateur. Elle comprend une zone effacée lors de la mise sous et hors tension et une autre zone qui conserve les données.</p> <p>La mémoire E/S est également découpée en deux zones : une qui échange les données avec toutes les unités et l'autre réservée à l'usage interne. Les données sont échangées avec toutes les unités une fois par cycle d'exécution du programme et lorsque des instructions spécifiques sont envoyées.</p>
<b>Configuration de l'API</b>	<p>La configuration de l'API permet de définir plusieurs paramètres initiaux ou autres grâce aux interrupteurs du logiciel.</p>
<b>Interrupteurs DIP</b>	<p>Les interrupteurs DIP permettent de définir des paramètres initiaux ou autres via les interrupteurs matériels.</p>
<b>Cartes mémoire</b>	<p>Les cartes mémoire sont utilisées, en cas de nécessité, pour stocker des données telles que les programmes, les données de mémoire E/S, la configuration de l'API et les commentaires d'E/S créés par les périphériques de programmation. Il est possible d'écrire les programmes et les différents paramètres du système automatiquement à partir de la carte mémoire lors de la mise sous tension (transfert automatique au démarrage).</p>
<b>Mémoire flash (UC CS1-H, CS1D, CJ1-H ou CJ1M uniquement)</b>	<p>Avec une CU CS1-H, CJ1-H, CJ1M ou CS1D, les données du programme utilisateur et de la zone des paramètres, telles que la configuration de l'API, sont automatiquement sauvegardées dans la mémoire flash intégrée chaque fois que l'utilisateur écrit des données vers l'UC. Ceci permet un fonctionnement sans batterie sans utiliser la carte mémoire. La mémoire E/S, ainsi que la majorité de la zone DM, ne sont pas sauvegardées sans batterie.</p>

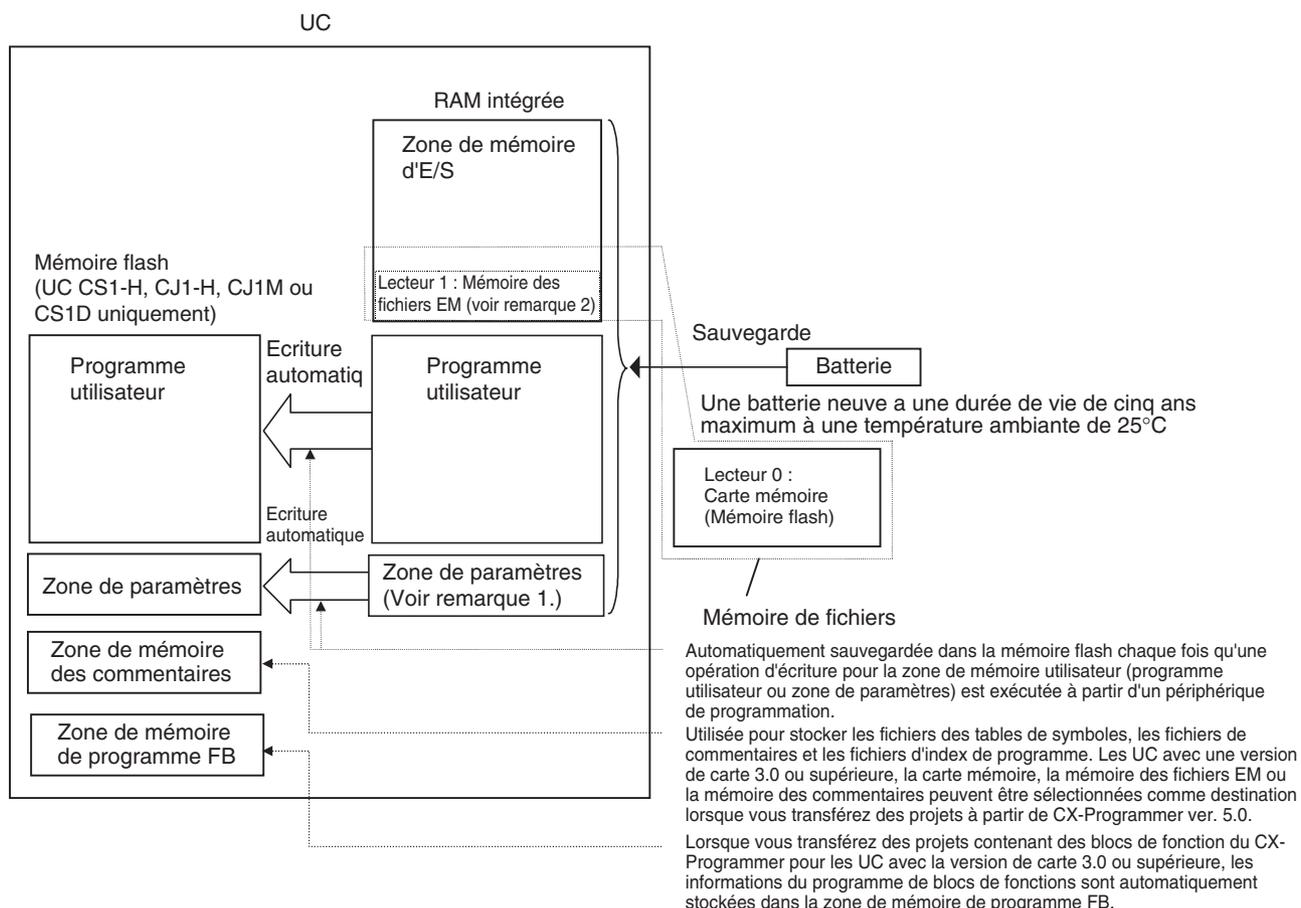
### 1-3-2 Diagramme des blocs de la mémoire de l'UC

La mémoire de l'UC (RAM) est formée des blocs suivants, dans le cas de la série CS/CJ :

- La zone des paramètres (configuration de l'API, table d'E/S enregistrées, tableau de routage et paramètres des cartes réseau)
- Zones de mémoire E/S
- Le programme utilisateur

Les données de la zone des paramètres et des zones de mémoire E/S sont sauvegardées grâce à une batterie (série CS : CS1W-BAT01, CJ1-H : CPM2A-BAT01) et seront perdues si la tension de la batterie est trop faible.

Cependant, les UC CS1-H, CJ1-H, CJ1M ou CS1D disposent d'une mémoire flash intégrée pour la sauvegarde des données. Les données du programme utilisateur et de la zone de paramètres sont automatiquement sauvegardées dans la mémoire flash intégrée chaque fois que l'utilisateur écrit des données vers l'UC à partir d'un périphérique de programmation (CX-Programmer ou la console de programmation) : transfert de données, édition en ligne, transfert à partir de la carte mémoire, etc. Ainsi, les données du programme utilisateur et de la zone de paramètres ne seront jamais perdues, même si la batterie tombait en panne.



- Remarque**
1. La zone de paramètres et le programme utilisateur (c-à-d la mémoire utilisateur) peuvent être protégés en écriture en définissant à ON la broche 1 de l'interrupteur DIP sur la face avant de l'UC.
  2. La mémoire de fichiers EM fait partie de la zone EM convertie en mémoire de fichiers dans la configuration de l'API. Toutes les banques EM, de la banque spécifiée jusqu'à la fin de la zone EM, ne peuvent être utilisées que comme mémoire de fichiers pour le stockage de données et de fichiers programmes.
  3. Veillez à installer la batterie fournie (CS1W-BAT01) avant d'utiliser l'UC CS1 pour la première fois. Lorsque la batterie est installée, utilisez un périphérique de programmation pour effacer la mémoire RAM de l'API (zone de paramètre, zone de mémoire E/S et programme utilisateur).
  4. Une batterie est installée sur les UC CS1-H, CJ1, CJ1-H, CJ1M ou CS1D en usine. Dans ce cas, il n'est pas nécessaire d'effacer la mémoire ou régler l'heure.
  5. Le voyant BKUP à l'avant de l'UC s'allume lorsque des données sont écrites dans la mémoire flash. Ne mettez pas l'UC hors tension tant que la sauvegarde n'est pas terminée (c-à-d tant que le voyant BKUP n'est pas éteint). Consultez la section 6-6-11 *Mémoire flash* pour plus d'informations.

## 1-4 Modes de fonctionnement

### 1-4-1 Description des modes de fonctionnement

L'UC dispose des modes de fonctionnement suivants. Ces modes contrôlent tout le programme utilisateur et sont communs à toutes les tâches.

**Mode PROGRAM**

L'exécution du programme s'arrête en mode PROGRAM et le voyant RUN n'est pas allumé. Ce mode est utilisé lors de l'édition du programme ou d'autres opérations de préparation telles que :

- l'enregistrement de la table d'E/S
- la modification de la configuration de l'API et d'autres paramètres
- le transfert et le contrôle des programmes
- la configuration et la réinitialisation forcée des bits pour vérifier le câblage et l'affectation des bits.

Dans ce mode, aucune tâche cyclique et d'interruption ne s'exécute (INI), elles s'arrêtent. Voir la section 1-6 *Description des tâches* pour plus de détails sur les tâches.

La mise à jour d'E/S s'effectue en mode PROGRAM. Consultez le *Manuel de fonctionnement* pour plus d'informations sur la mise à jour d'E/S.

**⚠ AVERTISSEMENT**

L'UC met à jour les E/S même lorsque le programme est à l'arrêt (c'est-à-dire, même en mode PROGRAM). Contrôlez les conditions de sécurité avant de modifier l'état de toute partie de mémoire réservée aux cartes d'E/S, aux cartes d'E/S spéciales ou cartes réseau. Tout changement des données assignées à une carte risque de provoquer un fonctionnement inattendu des charges connectées à la carte. Chacune des opérations suivantes peut provoquer un changement de l'état de la mémoire.

- Transfert de données de la mémoire E/S vers l'UC depuis un périphérique de programmation.
- Modification des valeurs actuelles de la mémoire depuis un périphérique de programmation.
- Configuration/réinitialisation forcée des bits depuis un périphérique de programmation.
- Transfert de fichiers de la mémoire E/S à partir d'une carte mémoire ou de la mémoire de fichiers EM vers l'UC.
- Transfert de la mémoire E/S depuis un ordinateur hôte ou un autre API sur un réseau.

**Mode MONITOR**

Lorsque le programme s'exécute en mode MONITOR, il est possible d'effectuer les opérations suivantes via les périphériques de programmation. Le voyant RUN est allumé. Ce mode est utilisé pour effectuer des tests ou d'autres réglages.

- Edition en ligne
- Reconfiguration et réinitialisation des bits
- Changement des valeurs de la mémoire E/S

Dans ce mode, les tâches cycliques qui doivent être exécutées au démarrage (voir remarque) et celles exécutables par TKON(820) seront exécutées lorsque l'exécution du programme atteindra leur numéro de tâche. Les tâches d'interruption seront exécutées si les conditions d'interruption se présentent.

**Remarque**

Les tâches exécutées au démarrage sont spécifiées dans les propriétés du programme à partir de CX-Programmer.

**Mode RUN**

Ce mode est utilisé pour l'exécution normale du programme. Le voyant RUN est allumé. Certaines opérations du périphérique de programmation telles que l'édition en ligne, la configuration et la réinitialisation forcées ainsi que le changement des valeurs de la mémoire E/S sont désactivées dans ce mode. Par contre, d'autres opérations du périphérique de programmation telles que la surveillance de l'état de l'exécution du programme (surveillance des programmes et de la mémoire E/S) sont activées.

Utilisez ce mode pour le fonctionnement normal du système. L'exécution des tâches est la même qu'en mode MONITOR.

Consultez *Modes d'utilisation de l'UC 10-2* dans le *Manuel d'utilisation* pour plus de détails sur les opérations disponibles avec chaque mode de fonctionnement.

## 1-4-2 Initialisation de la mémoire E/S

Le tableau suivant présente les zones de données qui seront effacées lorsque le mode de fonctionnement est commuté du mode PROGRAM au mode RUN/MONITOR ou vice-versa.

Modification du mode	Zones non-conservées (Remarque 1)	Zones conservées (Remarque 2)
RUN/MONITOR → PROGRAM	Effacé (Remarque 3.)	Conservé
PROGRAM → RUN/MONITOR	Effacé (Remarque 3.)	Conservé
RUN ↔ MONITOR	Conservé	Conservé

### Remarque

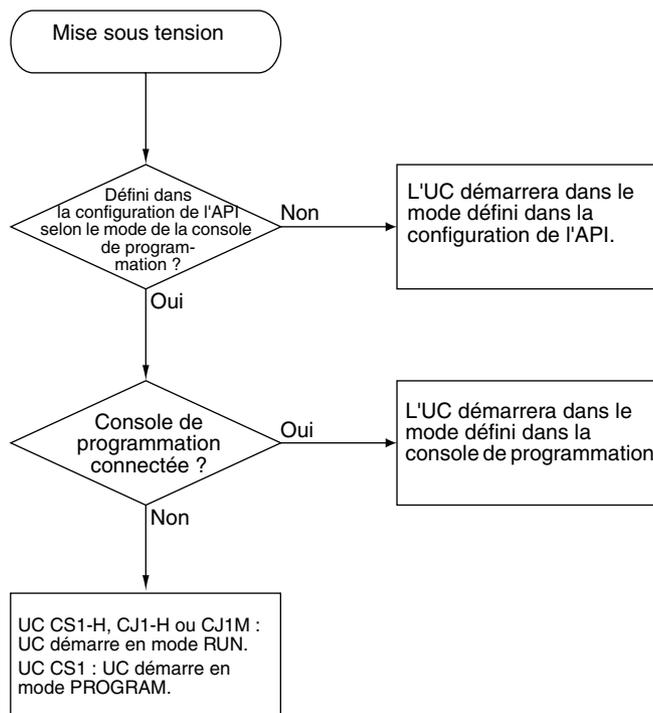
1. Zones non-conservées : zone CIO, zone de travail, valeurs actuelles de temporisation, drapeaux de fin de temporisation, registres d'index, registres de données, drapeaux des tâches et drapeaux de conditions.  
(L'état de certaines adresses de la zone auxiliaire sont conservées et d'autres sont effacées.)
2. Zones conservées : zone de maintien, zone DM, zone EM, valeurs actuelles du compteur et drapeaux de fin de compteur.
3. Les données dans la mémoire E/S seront conservées tant que le bit de maintien IOM (050012) est ON. Lorsque le bit de maintien (A50012) est ON et que le fonctionnement s'arrête à cause d'une erreur fatale (y compris FALS(007)), le contenu de la mémoire E/S est conservé mais les sorties des cartes de sorties passeront toutes à OFF.

### 1-4-3 Mode de démarrage

Consultez le *Manuel d'utilisation* pour plus de détails sur le réglage du mode de démarrage de l'UC.

**Remarque** Dans le cas des UC CJ1, CS1-H, CJ1-H, CJ1M ou CS1D, l'UC démarrera en mode RUN si aucune console de programmation n'est connectée. Ce phénomène est contraire au fonctionnement par défaut d'une UC CS1 puisque l'UC démarre en mode PROGRAM par défaut si aucune console de programmation n'est connectée.

Conditions	UC CS1	UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement
La configuration de l'API est telle que le démarrage doit se faire en fonction du mode défini sur la console de programmation, mais aucune console de programmation n'est connectée.	Mode PROGRAM	Mode RUN



## 1-5 Programmes et tâches

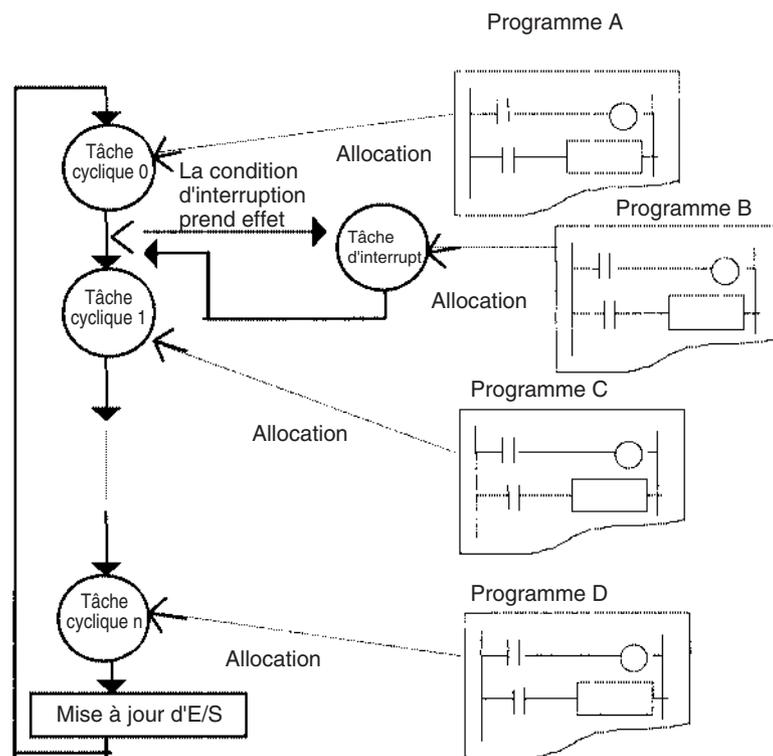
Les tâches précisent l'ordre et les conditions d'interruption dans lesquelles les programmes individuels seront exécutés. Elles sont regroupées en deux types :

- 1,2,3...**
1. les tâches exécutées de manière séquentielle, appelées tâches cycliques,
  2. les tâches exécutées par des conditions d'interruption, appelées tâches d'interruption.

**Remarque** Dans le cas des UC CS1-H, CJ1-H, CJ1M ou CS1D, les tâches d'interruption peuvent être exécutées de manière cyclique, comme les tâches cycliques. On les appelle alors des « tâches cycliques extra ».

Les programmes affectés aux tâches cycliques sont exécutés par ordre de numéro de tâche et les E/S sont mises à jour une fois par cycle, lorsque toutes les tâches ont été exécutées (ou plutôt, uniquement les tâches dont l'état est « exécutable »). Si une condition d'interruption se présente au cours du traitement des tâches cycliques, la tâche cyclique sera interrompue et le programme affecté à cette tâche d'interruption sera exécuté.

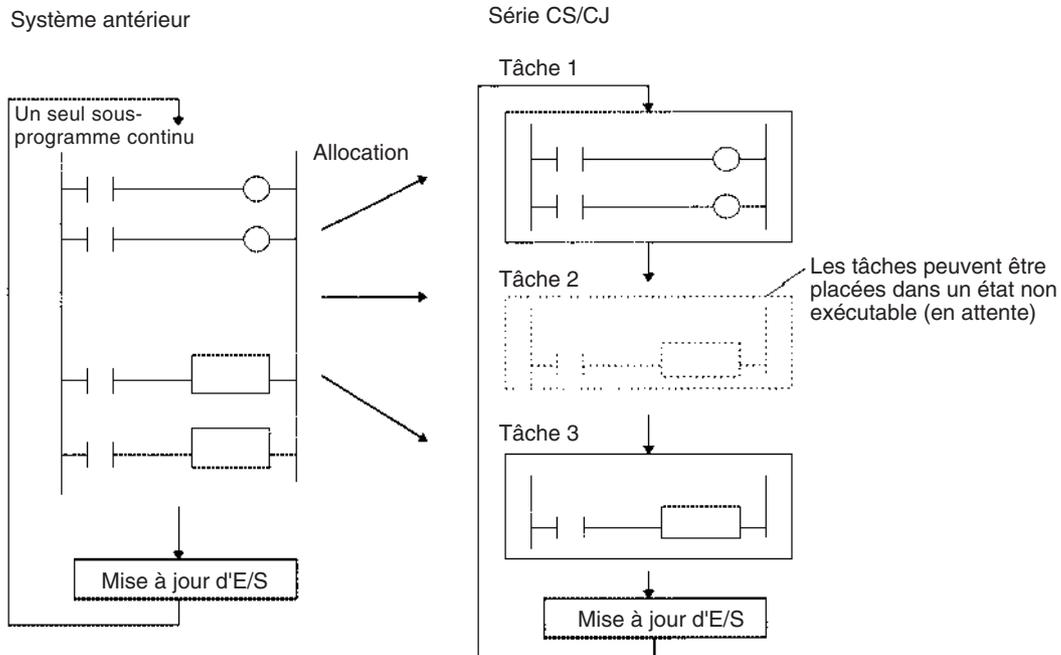
Consultez le chapitre sur le fonctionnement de l'UC dans le *Manuel d'utilisation de la série CS/CJ* pour plus d'informations sur la mise à jour d'E/S.



Dans l'exemple ci-dessus, la programmation s'effectuera dans l'ordre suivant : démarrage de A, B, reste de A, C, puis D. Cela suppose que la condition d'interruption de la tâche d'interruption 100 a été établie lors de l'exécution du programme A. Lorsque le programme B est complètement exécuté, le système reprend l'exécution du programme A là où il s'était arrêté.

Avec les anciens API d'OMRON, un programme continu est formé de plusieurs parties continues. Les programmes affectés à chaque tâche sont des programmes uniques qui se terminent par une instruction END, comme le programme unique des anciens API.

Les tâches cycliques présentent notamment l'avantage suivant : elles peuvent être activées (état exécutable) et désactivées (état en attente) par les instructions de contrôle des tâches. Cela signifie que plusieurs composants du programme peuvent être rassemblés pour former une tâche et que seuls des programmes spécifiques (tâches) peuvent ensuite être exécutés selon les besoins du modèle ou du traitement du produit réalisé (commutation des étapes du programme). Ce système permet d'améliorer nettement les performances (temps de cycle) car seuls les programmes nécessaires sont exécutés.



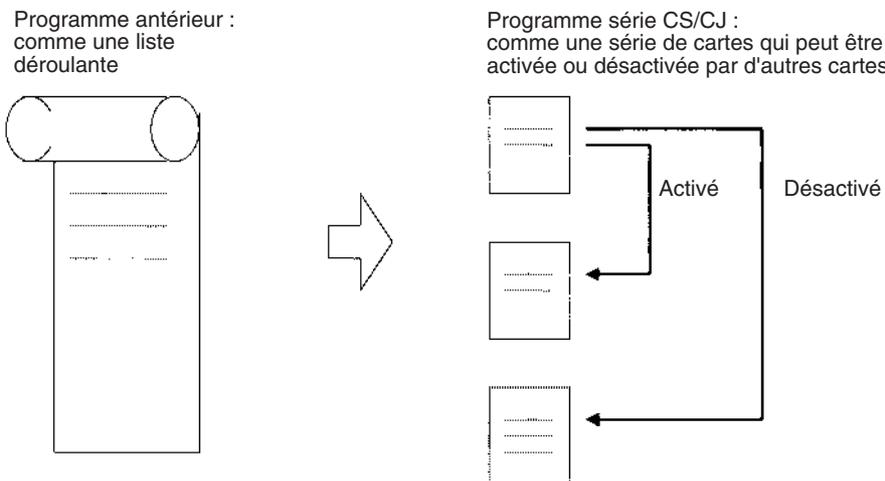
Une tâche qui a été exécutée sera exécutée lors des cycles suivants et une tâche mise en attente le restera lors des cycles suivants, sauf si elle est à nouveau exécutée à partir d'une autre tâche.

**Remarque**

Contrairement aux anciens programmes, qui pouvaient être comparés à la lecture d'une liste déroulante, les tâches peuvent être comparées à la lecture d'une série de cartes individuelles.

- Chaque carte est lue dans un ordre prédéfini, en commençant par le plus petit chiffre.
- Chaque carte reçoit l'état actif ou inactif et les cartes inactives sont ignorées. (Les cartes sont activées ou désactivées par les instructions de contrôle des tâches.)

- Une carte définie comme activée restera activée et sera lue lors des séquences suivantes. Une carte définie comme désactivée restera désactivée et sera passée jusqu'à ce qu'elle soit réactivée par une autre carte.



## 1-6 Description des tâches

Les tâches sont regroupées en deux types :

**1,2,3...**

### 1. Tâches cycliques (maximum 32)

Ces tâches sont exécutées une fois par cycle si elles sont exécutables. Si nécessaire, il est possible de désactiver l'exécution des tâches cycliques.

### 2. Tâches d'interruption

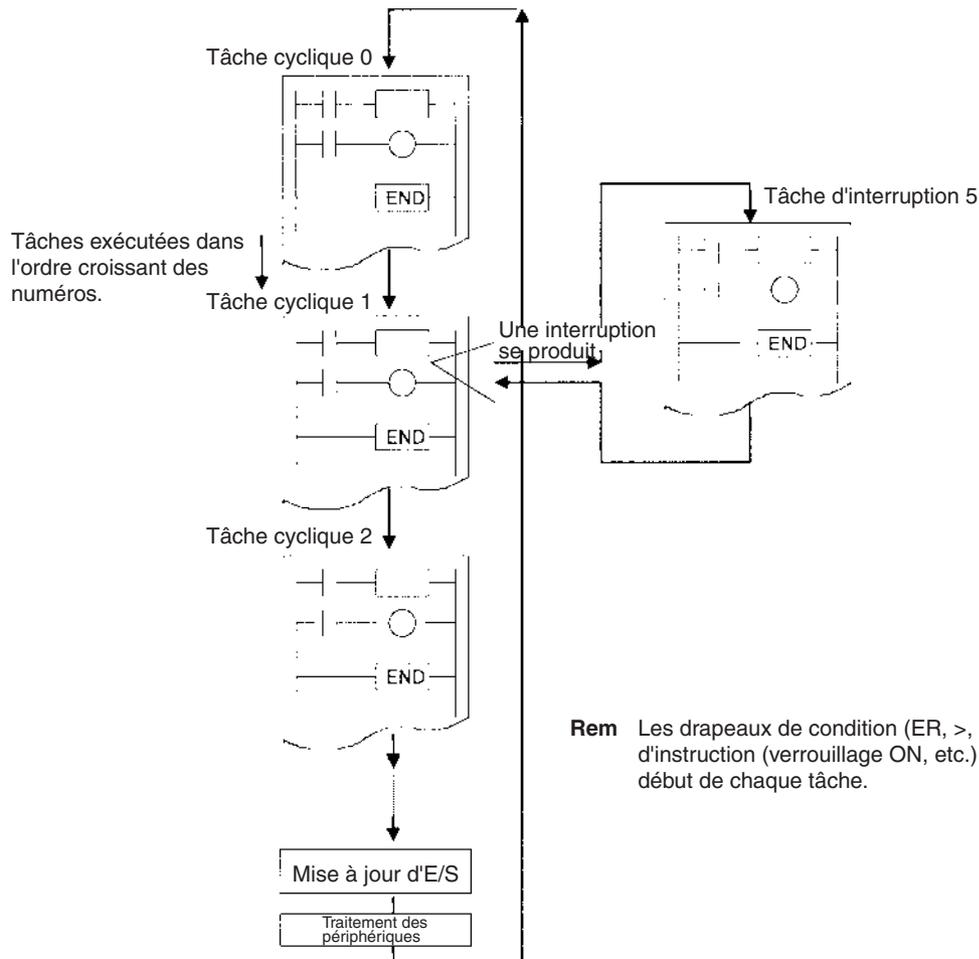
Ces tâches sont exécutées lorsqu'une interruption se produit, qu'une tâche cyclique soit ou non exécutée. Les tâches d'interruption (voir remarques 1 et 2) sont regroupées sous les quatre types suivants (cinq types, si l'on compte les tâches cycliques extra pour les UC CS1-H, CJ1-H, CJ1M ou CS1D) :

- Tâche d'interruption de mise hors tension (pas prise en charge par les UC CS1D pour les systèmes d'UC en duplex) : exécutée lorsque l'alimentation est interrompue. (1 max.)
- Tâche d'interruption programmée (pas prise en charge par les UC CS1D pour les systèmes d'UC en duplex) : exécutée à intervalles précis. (2 max.)
- Tâche d'interruption d'E/S (pas prise en charge par les UC CJ1 ou CS1D pour les systèmes d'UC en duplex) : exécutée lorsque l'entrée de la carte d'entrée d'interruption passe à ON (32 max.)
- Tâche d'interruption externe (non prise en charge par les UC CJ1 ou CS1D pour systèmes d'UC en duplex) : exécutée (256 max.) suite à la demande d'une carte d'E/S spéciales, une carte réseau ou une carte interne (Série CS uniquement).
- Tâche cyclique supplémentaire (prise en charge uniquement par les UC CS1-H, CJ1-H, CJ1M et CS1D) : Tâches d'interruption traitées comme des tâches cycliques. Les tâches cycliques extra sont exécutées une fois dans chaque cycle tant qu'elles ont l'état exécutable.

Avec CX-Programmer, il est possible de créer et de contrôler jusqu'à 288 tâches avec 288 programmes (jusqu'à 32 tâches cycliques et 256 tâches d'interruption).

- Remarque**
1. Les UC CJ1 ne prennent actuellement pas en charge les tâches d'interruption d'E/S et externes. Le nombre maximal de tâches pour une UC CJ1 est donc de 35, c-à-d 32 tâches cycliques et 3 tâches d'interruption. Le nombre total de programmes pouvant être créés et gérés est également de 35.
  2. Les UC CS1D ne prennent en charge aucune tâche d'interruption. Cependant, avec les UC CS1D, vous pouvez utiliser les tâches d'interruption en tant que tâches cycliques extra.

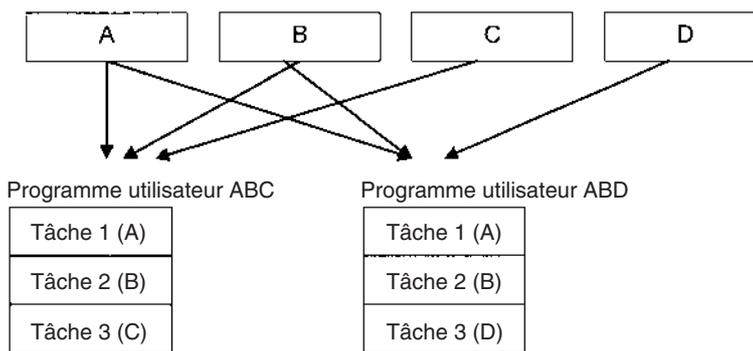
Le rapport entre le nombre de programmes alloués à une tâche est de 1:1. Ce rapport est déterminé dans CX-Programmer, dans les différents paramètres de propriétés du programme.



### Structure du programme

Il est possible de créer et d'affecter des sous-programmes standard aux tâches si vous en avez besoin pour créer des programmes. Cela signifie qu'il est possible de créer des programmes dans des modules (composants standard) et de déboguer des tâches de manière individuelle.

Sous-programmes standard



Lors de la création de programmes modulaires, vous pouvez préciser des adresses à l'aide de symboles afin de faciliter la standardisation.

**Etat exécutable et en attente**

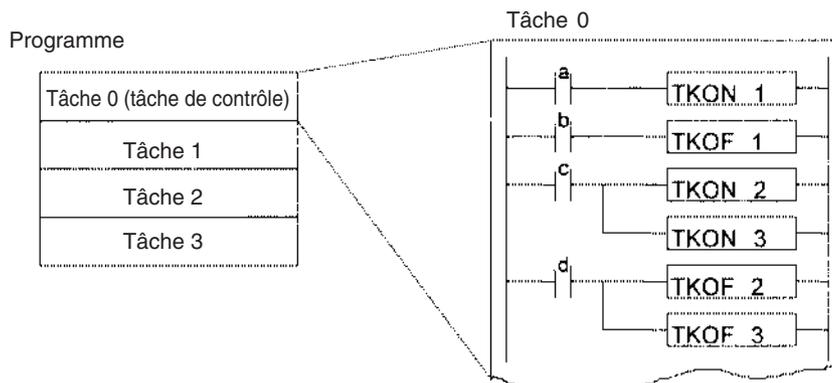
Les instructions TASK ON et TASK OFF (TKON(820) et TKOF(821)) peuvent être exécutées en une seule tâche afin de pouvoir attribuer à une autre tâche l'état exécutable ou en attente.

Les instructions qui se trouvent dans des tâches mise en attente ne seront pas exécutées mais l'état d'E/S sera conservé. Lorsqu'une tâche récupère son état exécutable, les instructions seront exécutées avec l'état d'E/S qui avait été conservé.

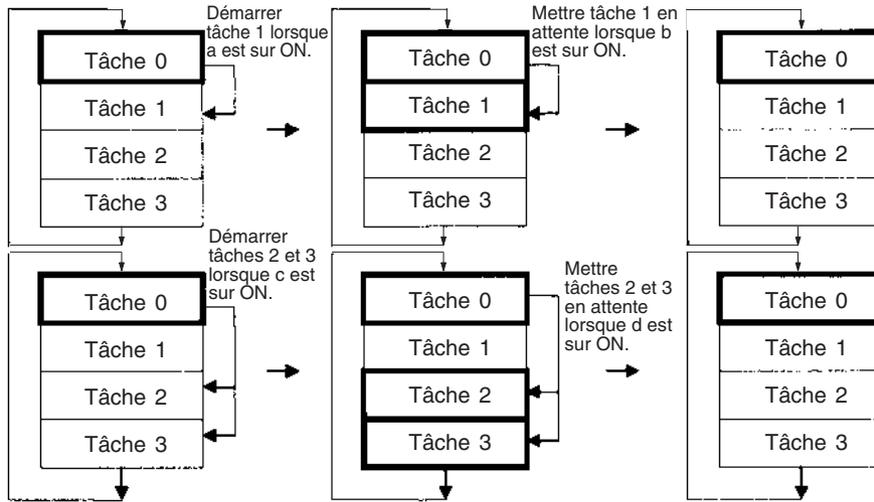
**Exemple : Programmation à l'aide d'une tâche de contrôle**

Dans cet exemple, la tâche 0 est une tâche de contrôle exécutée d'abord au début de l'opération. Les autres tâches peuvent être définies à partir de CX-Programmer (mais pas d'une console de programmation) de manière à ce qu'elles démarrent, ou non, au début de l'opération.

Lorsque l'exécution du programme a commencé, il est possible de contrôler les tâches grâce aux instructions TKON(820) et TKOF(821).

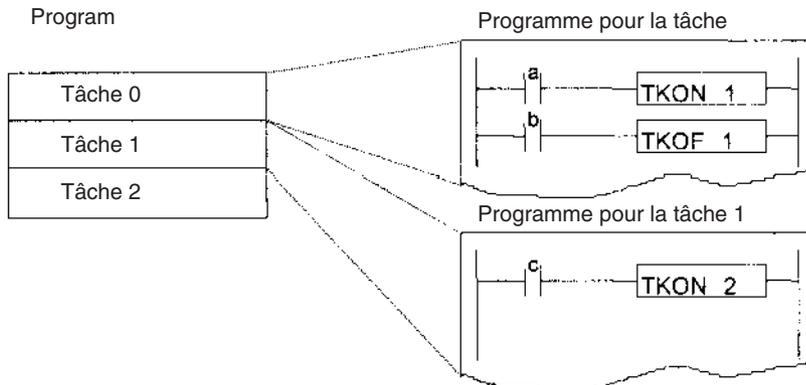


Exemple : La tâche 0 est configurée de manière à être exécutée au début de l'opération (défini dans les propriétés du programme à partir du CX-Programmer).  
 La tâche 1 est exécutable lorsque a est sur ON.  
 La tâche 1 est mise en attente lorsque b est sur ON.  
 Les tâches 2 et 3 sont exécutables lorsque c est sur ON.  
 Les tâches 2 et 3 sont mises en attente lorsque d est sur ON.

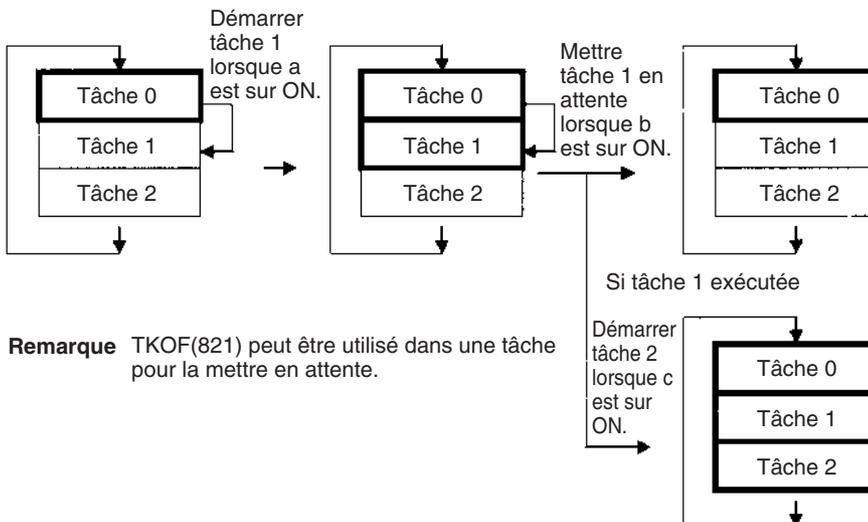


**Exemple : chaque tâche est contrôlée par une autre tâche**

Dans cet exemple, chaque tâche est contrôlée par une autre tâche.



Exemple : La tâche 1 est configurée pour être exécutée sans condition au début de l'opération.  
 Tâche 1 exécutable lorsque a est sur ON.  
 Tâche 1 mise en attente lorsque b est sur ON.  
 Tâche 2 exécutable lorsque c est sur ON et que tâche 1 a été exécutée.



**Remarque** TKOF(821) peut être utilisé dans une tâche pour la mettre en attente.

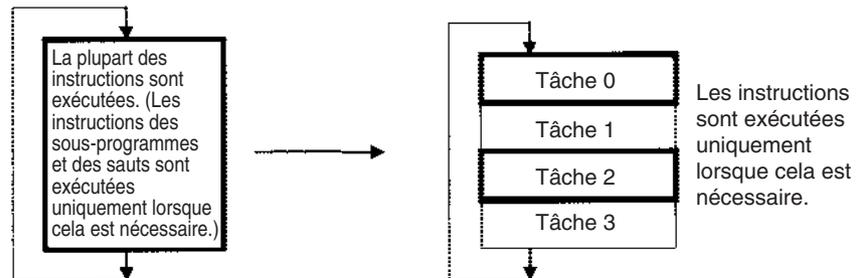
**Temps d'exécution des tâches**

Lorsqu'une tâche est en attente, les instructions de cette tâche ne sont pas exécutées ; le temps d'exécution de l'instruction OFF n'est donc pas ajouté au temps de cycle.

**Remarque**

A partir de là, les instructions qui se trouvent dans une tâche en attente peuvent être comparées aux instructions qui se trouvent dans une partie de programme sautée (JMP-JME).

Etant donné que les instructions se trouvant dans une tâche non exécutée n'augmentent pas le temps de cycle, il est possible d'améliorer de manière significative les performances globales du système en scindant le système en une tâche de contrôle globale et des tâches individuelles exécutées uniquement lorsque cela s'avérera nécessaire.



# CHAPITRE 2

## Programmation

Ce chapitre décrit les informations de base nécessaires pour écrire, vérifier et entrer des programmes.

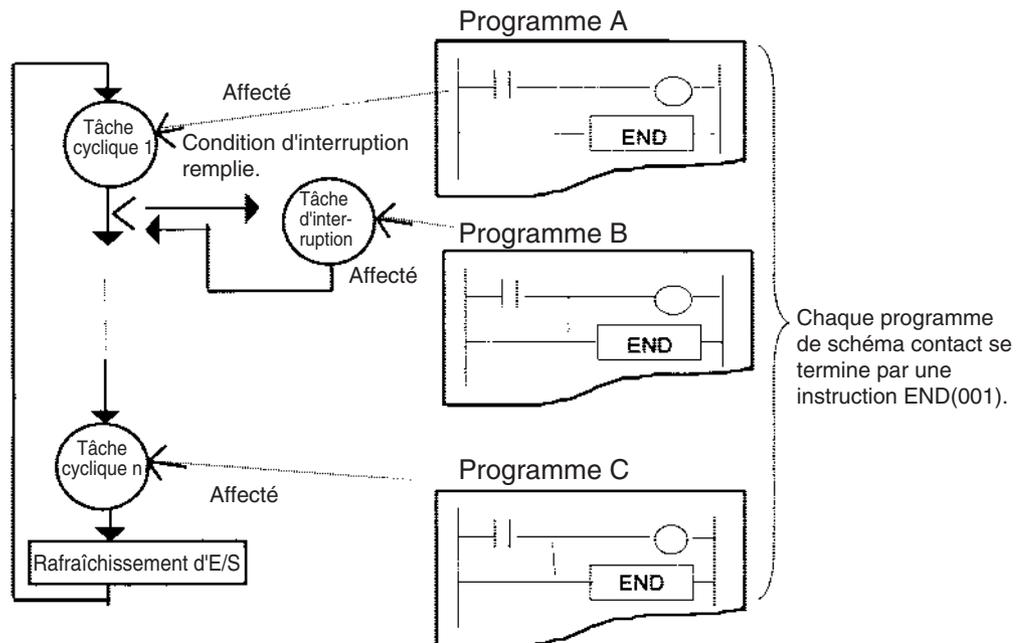
2-1	Concepts de base . . . . .	20
2-1-1	Programmes et tâches . . . . .	20
2-1-2	Informations de base concernant les instructions . . . . .	21
2-1-3	Emplacement de l'instruction et conditions d'exécution . . . . .	23
2-1-4	Adressage des zones de mémoire E/S . . . . .	24
2-1-5	Spécification des opérandes . . . . .	25
2-1-6	Formats de données . . . . .	30
2-1-7	Variations des instructions . . . . .	34
2-1-8	Conditions d'exécution . . . . .	34
2-1-9	Temporisation des instructions d'E/S . . . . .	37
2-1-10	Programmation de la mise à jour . . . . .	39
2-1-11	Capacité de programme . . . . .	42
2-1-12	Concepts de base de programmation du schéma contact . . . . .	42
2-1-13	Insertion de mnémoniques . . . . .	47
2-1-14	Exemples de programmes . . . . .	50
2-2	Précautions . . . . .	55
2-2-1	Drapeaux de condition . . . . .	55
2-2-2	Sections de programmes spéciales . . . . .	60
2-3	Vérification des programmes . . . . .	64
2-3-1	Erreurs lors de l'entrée de périphériques de programmation . . . . .	64
2-3-2	Vérification du programme par CX-Programmer . . . . .	64
2-3-3	Vérification de l'exécution du programme . . . . .	66
2-3-4	Vérification des erreurs fatales : . . . . .	68

## 2-1 Concepts de base

### 2-1-1 Programmes et tâches

Les API série CS/CJ exécutent les programmes du schéma contact inclus dans les tâches. Le programme de schéma contact de chaque tâche se termine par une instruction END(001), comme dans le cas des API traditionnels.

Les tâches servent à déterminer l'ordre d'exécution des programmes de schéma contact ainsi que les conditions d'interruption d'exécution.



Ce chapitre décrit les concepts de base nécessaires à l'écriture de programmes dans la série CS/CJ. Consultez le *CHAPITRE 4 Tâches* pour plus d'informations sur les tâches et leur relation avec les programmes de schéma contact.

#### Remarque **Tâches et périphériques de programmation**

Les tâches sont gérées de la manière décrite ci-après sur les périphériques de programmation. Consultez *4-4 Fonctionnement du périphérique de programmation pour les tâches* ainsi que le *Manuel d'utilisation des consoles de programmation série CS/CJ (W341)* et le *Manuel d'utilisation de CX-Programmer* pour plus de détails.

#### **CX-Programmer**

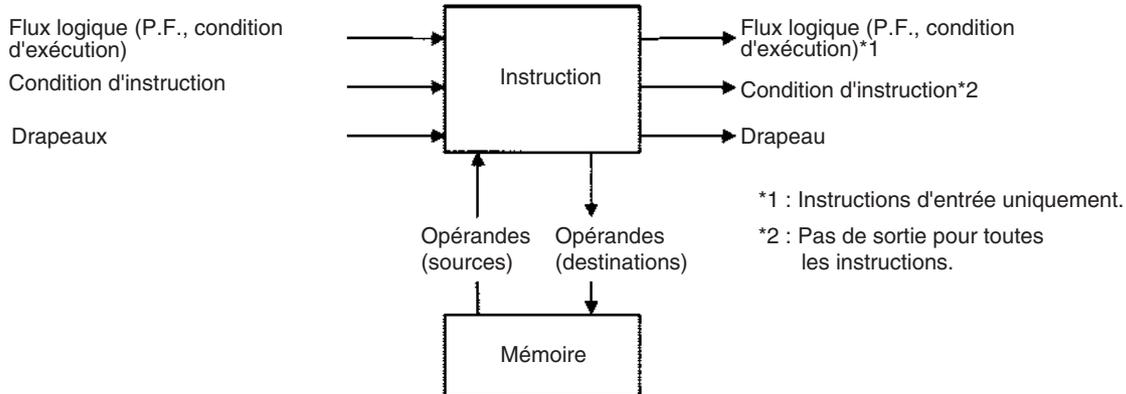
CX-Programmer sert à attribuer des types de tâches et des numéros de tâches aux programmes individuels.

#### **Console de programmation**

L'accès et l'édition des programmes se font sur une console de programmation en spécifiant CT00 à CT31 pour les tâches cycliques et IT00 à IT25 pour les tâches d'interruption. Lors de l'effacement de la mémoire avec une console de programmation, seule la tâche cyclique 0 (CT00) peut être réécrite dans un nouveau programme. Utilisez CX-Programmer pour créer des tâches cycliques de 1 à 31 (CT01 à CT31).

### 2-1-2 Informations de base concernant les instructions

Un programme, c'est une suite d'instructions. Le diagramme suivant présente la structure conceptuelle des entrées et sorties d'une instruction.

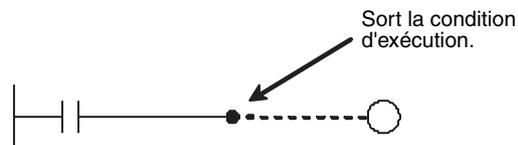


#### Flux d'alimentation

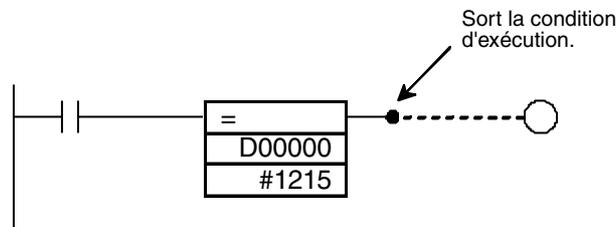
Le flux d'alimentation est la condition d'exécution utilisée pour contrôler l'exécution et les instructions lorsque les programmes s'exécutent normalement. Dans un schéma contact, le flux d'alimentation représente l'état de la condition d'exécution.

#### Instructions d'entrée

- Les instructions de chargement indiquent un départ logique et sortent la condition d'exécution.

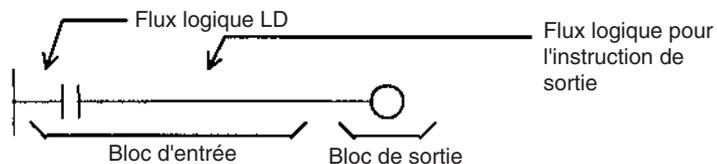


- Les instructions intermédiaires entrent le flux d'alimentation en tant que condition d'exécution et le sortent vers une instruction intermédiaire ou de sortie.



#### Instructions de sortie

Les instructions de sortie exécutent toutes les fonctions et utilisent le flux d'alimentation comme condition d'exécution.



#### Conditions d'instruction

Les conditions d'instruction sont des conditions spéciales liées à l'exécution générale des instructions fournies par les instructions suivantes. Le niveau de priorité des conditions d'instruction est supérieur à celui du flux d'alimentation lorsqu'il s'agit de décider si une instruction doit être exécutée ou non. Il se peut qu'une instruction ne doive plus être exécutée ou qu'elle agisse différemment en fonction des conditions d'instruction. Les conditions

d'instruction sont redéfinies (annulées) au début de chaque tâche, c-à-d qu'elles sont redéfinies lorsque l'on passe à une autre tâche.

Les instructions suivantes sont utilisées par deux pour définir et annuler certaines conditions d'instruction. Ces instructions groupées doivent se trouver dans la même tâche.

Condition d'instruction	Description	Définition de l'instruction	Annulation de l'instruction
Verrouillé	Un verrouillage désactive une partie du programme. Les conditions spéciales, telles la désactivation (OFF) des bits de sortie, la réinitialisation des temporisations et le maintien des compteurs fonctionnent toujours.	IL(002)	ILC(003)
Exécution de BREAK(514)	Met fin à une boucle FOR(512) – NEXT(513) pendant l'exécution. (Empêche l'exécution de toutes les instructions tant qu'il n'y a pas d'instruction NEXT(513)).	BREAK(514)	NEXT(513)
	Exécute un saut de JMP0(515) à JME0(516)	JMP0(515)	JME0(516)
Exécution d'un programme de bloc	Exécute un bloc de programmes de BPRG(096) à BEND(801).	BPRG(096)	BEND(801)

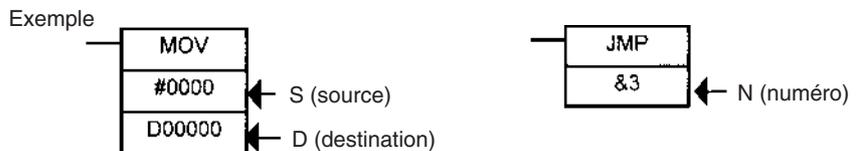
**Drapeaux**

Dans ce contexte, un drapeau est un bit qui sert d'interface entre des instructions.

Drapeaux d'entrée	Drapeaux de sortie
<ul style="list-style-type: none"> <li>• Drapeaux de différenciation Drapeaux de résultat de différenciation. Les états de ces drapeaux sont envoyés automatiquement à l'instruction pour toutes les instructions de sortie haut/bas différenciées et pour les instructions DIFU(013)/DIFD(014).</li> <li>• Drapeau de retenue (CY) Le drapeau de retenue sert d'opérande non spécifiée pour les instructions de décalage de données et les instructions d'addition/soustraction.</li> <li>• Drapeaux pour instructions spéciales Il s'agit des drapeaux d'apprentissage pour les instructions FPD(269) et les communications réseau activées par drapeaux.</li> </ul>	<ul style="list-style-type: none"> <li>• Drapeaux de différenciation Drapeaux de résultat de différenciation. Les états de ces drapeaux sont reçus automatiquement de l'instruction pour toutes les instructions d'entrée haut/bas différenciées et pour l'instruction UP(521)/DOWN(522).</li> <li>• Drapeaux de condition Il s'agit des drapeaux Toujours ON/OFF ainsi que des drapeaux mis à jour par le résultat de l'exécution d'une instruction. Dans les programmes utilisateur, ces drapeaux peuvent être spécifiés par des étiquettes telles que ER, CY, &gt;, =, A1, A0 plutôt que par des adresses.</li> <li>• Drapeaux pour instructions spéciales Il s'agit des drapeaux d'instruction de carte mémoire et des drapeaux d'exécution terminée MSG(046).</li> </ul>

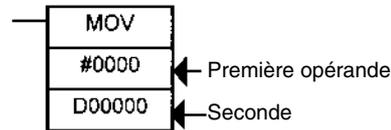
**Opérandes**

Les opérandes spécifient les paramètres des instructions prédéfinies (cases des schémas de contact) utilisés pour préciser le contenu ou les constantes de la zone de mémoire E/S. Il est possible d'exécuter une instruction en entrant une adresse ou une constante comme opérande. Les opérandes sont classées dans trois groupes : source, destination ou numéro.



Types d'opérandes		Symbole de l'opérande	Description	
Source	Précise l'adresse des données à lire ou une constante.	S	Opérande source	Opérande source autre que les données de contrôle (C)
		C	Données de contrôle	Données composées dans une opérande source dont la signification diffère en fonction de l'état du bit.
Destination (résultats)	Précise l'adresse à laquelle seront écrites les données.	D (R)	---	---
Numéro	Précise un numéro particulier utilisé dans l'instruction, comme par exemple, un numéro de saut ou de sous-programme.	N	---	---

**Remarque** On peut également distinguer les opérandes en les appelant première opérande, seconde opérande, etc. en commençant par le haut de l'instruction.



### 2-1-3 Emplacement de l'instruction et conditions d'exécution

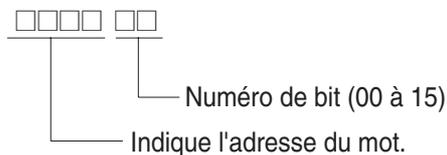
Le tableau suivant montre les différents emplacements possibles pour les instructions. Les instructions sont réparties en deux groupes : celles pour lesquelles les conditions d'exécution sont obligatoires, et les autres. Voir *CHAPITRE 3 Fonction des instructions* pour plus de détails sur les différentes instructions.

Type d'instruction		Emplacement possible	Condition d'exécution	Diagramme	Exemples
Instructions d'entrée	Départ logique (instructions de chargement)	Connectée directement à la barre de bus gauche ou se trouve au début d'un bloc d'instructions.	Non obligatoire.		LD, LD TST(350), LD > (et autres instructions de comparaison de symboles)
	Instructions intermédiaires	Entre un départ logique et une instruction de sortie.	Obligatoire.		AND, OR, AND TEST(350), AND > (et autres instructions de comparaison de symboles ADD), UP(521), DOWN(522), NOT(520), etc.
Instructions de sortie		Connectées directement à la barre de bus droite.	Obligatoire.		La plupart des instructions comprenant OUT et MOV(021).
			Non obligatoire.		END(001), JME(005), FOR(512), ILC(003), etc.

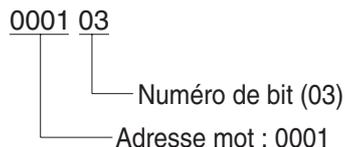
- Note**
1. Il existe un autre groupe d'instructions qui exécutent une série d'instructions mnémotechniques basées sur une seule entrée. On les appelle les instructions de programmes de bloc. Consultez le *Manuel de référence des instructions pour les UC série CS/CJ* pour plus de détails sur ces programmes de bloc.
  2. Si une instruction qui a besoin d'une condition d'exécution est directement connectée à la barre de bus gauche sans instruction de départ logique, le système affichera une erreur de programme lorsqu'il contrôlera le programme sur un périphérique de programmation (CX-Programmer ou une console de programmation).

## 2-1-4 Adressage des zones de mémoire E/S

### Adresses bits



**Exemple :** L'adresse du bit 03 du mot 0001 de la zone CIO est illustrée ci-dessous. Dans ce manuel, l'adresse est fournie sous la forme « CIO 000103 ».



Mot ↓	Bit : CIO 000103															
	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
0000																
0001																
0002																

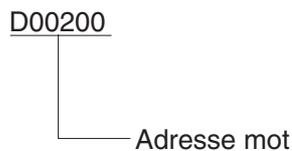
### Adresses mots



**Exemple :** L'adresse des bit 00 à 15 du mot 0010 de la zone CIO est illustrée ci-dessous. Dans ce manuel, l'adresse est fournie sous la forme « CIO 00010 ».



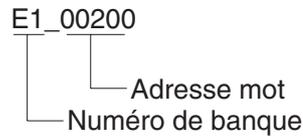
Les adresses des zones DM et EM commencent par les préfixes « D » ou « E », comme illustré pour l'adresse D00200.



**Exemple :** L'adresse du mot 2000 dans la banque actuelle de Mémoire de données étendue (EM) est la suivante :



L'adresse du mot 2000 dans la banque 1 de Mémoire de données étendue (EM) est la suivante :



### 2-1-5 Spécification des opérandes

Opérande	Description	Notation	Exemples d'applications
Spécification des adresses bits	<p>Les numéros de mot et de bit sont spécifiés directement pour spécifier un bit (bits d'entrée).</p> <p>□□□□ □□</p> <p>Indique l'adresse du mot.</p> <p>Numéro de bit (00 à 15)</p> <p><b>Remarque</b> Les mêmes adresses sont utilisées pour accéder aux drapeaux de fin de temporisation/compteur et aux valeurs actuelles. Il n'existe également qu'une seule adresse pour un drapeau de tâche.</p>	<p>0001 02</p> <p>Numéro de bit (02)</p> <p>Numéro de mot : 0001</p>	<p>0001 02</p>
Spécification des adresses mots	<p>Le numéro de mot est spécifié directement pour spécifier le mot 16 bits.</p> <p>□□□□</p> <p>Indique l'adresse du mot.</p>	<p>0003</p> <p>Numéro de mot : 0003</p> <p>D00200</p> <p>Numéro de mot : 00200</p>	<p>MOV 0003 D00200</p>

Opérande	Description	Notation	Exemples d'applications
<p>Spécification des adresses DM/EM indirectes en mode binaire</p>	<p>Le décalage par rapport au début de la zone est spécifié. Le contenu de l'adresse sera traité comme une donnée BCD (0000 à 32767) pour spécifier l'adresse du mot dans la mémoire de données (DM) ou la mémoire de données étendue (EM). Ajoutez le symbole @ au début pour spécifier une adresse indirecte en mode binaire.</p> <p style="text-align: center;">@D□□□□□</p> <p style="text-align: center;">↓</p> <p>Contenu <span style="border: 1px solid black; display: inline-block; width: 60px; height: 15px; vertical-align: middle;"></span> 00000 à 32767 (0000 Hex à 7FFF Hex dans BIN)</p> <p style="text-align: center;">↓</p> <p>D <span style="border: 1px solid black; display: inline-block; width: 60px; height: 15px; vertical-align: middle;"></span></p>		
	<p>1) D00000 à D32767 sont spécifiés si @D(□□□□□) contient 0000 hex. à 7FFF hex. (00000 à 32 767).</p>	<p>@D00300</p> <p style="text-align: center;"><span style="border: 1px solid black; display: inline-block; width: 40px; height: 15px; vertical-align: middle;">0 1 0 0</span> Contenu Binaire : 256</p> <p style="text-align: center;">↓</p> <p style="text-align: center;">Spécifie D00256.</p> <p style="text-align: center;">— Ajoute le symbole @.</p>	<p>MOV #0001 @00300</p>
	<p>2) E0_00000 à E0_32767 de la banque 0 dans la mémoire de données étendue (EM) sont spécifiés si @D(□□□□□) contient 8000 hex. à FFFF hex. (32768 à 65 535).</p>	<p>@D00300</p> <p style="text-align: center;"><span style="border: 1px solid black; display: inline-block; width: 40px; height: 15px; vertical-align: middle;">8 0 0 1</span> Contenu Binaire : 32769</p> <p style="text-align: center;">↓</p> <p style="text-align: center;">Spécifie E0 00001.</p>	
	<p>3) E□_00000 à E□_32767 dans la banque spécifiée sont spécifiés si @E□_□□□□□ contient 0000 hex. à 7FFF hex. (00000 à 32 767).</p>	<p>@E1_00200</p> <p style="text-align: center;"><span style="border: 1px solid black; display: inline-block; width: 40px; height: 15px; vertical-align: middle;">0 1 0 1</span> Contenu Binaire : 257</p> <p style="text-align: center;">↓</p> <p style="text-align: center;">Spécifie E1_00257.</p>	<p>MOV #0001 @E1_00200</p>
	<p>4) E(□+1)_00000 à E(□+1)_32767 dans la banque suivant la banque spécifiée □ sont spécifiés si @E□_□□□□□ contient 8000 hex. à FFFF hex. (32768 à 65 535).</p>	<p>@E1_00200</p> <p style="text-align: center;"><span style="border: 1px solid black; display: inline-block; width: 40px; height: 15px; vertical-align: middle;">8 0 0 2</span> Contenu Binaire : 32770</p> <p style="text-align: center;">↓</p> <p style="text-align: center;">Spécifie E2_00002.</p>	
<p><b>Remarque</b> Lorsque vous spécifiez une adresse indirecte en mode binaire, vous devez traiter les zones de mémoire de données (DM) et de mémoire de données étendue (EM) (banques 0 à C) comme une série d'adresses. Si le contenu d'une adresse comprenant le symbole @ dépasse 32 767, le système considérera que l'adresse de la mémoire de données étendue (EM) et continuera à partir de 00000 dans la banque 0.</p> <p><b>Exemple :</b> Si le mot de la mémoire de données (DM) contient 32 768, le système spécifiera E_100000 dans la banque 0 de la mémoire de données étendue (EM).</p> <p><b>Remarque</b> Si le numéro de banque de la mémoire de données étendue (EM) spécifié est « n » et que le contenu du mot dépasse 32 767, le système supposera que l'adresse est une adresse de la mémoire de données étendue et continuera à partir de 00000 dans la banque N+1.</p> <p><b>Exemple :</b> Si la banque 2 de la mémoire de données étendue (EM) contient 32 768, le système spécifiera E3_00000 dans la banque 3 de la mémoire de données étendue (EM).</p>			

Opérande	Description	Notation	Exemples d'applications
Spécification des adresses DM/EM indirectes en mode BCD	<p>Le décalage par rapport au début de la zone est spécifié. Le contenu de l'adresse sera traité comme une donnée BCD (0000 à 9999) pour spécifier l'adresse du mot dans la mémoire de données (DM) ou la mémoire de données étendue (EM). Ajoutez un astérisque (*) au début pour spécifier une adresse indirecte en mode BCD.</p> <p style="text-align: center;">*D□□□□□</p> <p>↓</p> <p>Table des matières □□□□□ 00000 à 9999 (BCD)</p> <p>↓</p> <p>D □□□□□</p>	<p>*D00200</p> <p style="text-align: center;">0 1 0 0 Table des matières</p> <p style="text-align: center;">↓</p> <p style="text-align: center;">Spécifie D0100</p> <p style="text-align: center;">— Ajoutez un astérisque (*).</p>	<p>MOV #0001 *D00200</p>

Opérande	Description	Notation	Exemples d'applications
Spécification directe d'un registre	Un registre d'index (IR) ou un registre de données (DR) peut être directement spécifié en spécifiant IR□ (□ : 0 à 15) ou DR□ (□ : 0 à 15).	<p>IR0</p> <p>IR1</p>	<p>MOVR 000102 IR0 Sauvegarde l'adresse mémoire de l'API pour CIO 0010 dans IR0.</p> <p>MOVR 00010 IR1 Sauvegarde l'adresse mémoire de l'API pour CIO 0010 dans IR1.</p>
Spécification d'une adresse indirecte à l'aide d'un registre	Adresse indirecte (sans décalage)	<p>,IR0</p> <p>,IR1</p>	<p>LD ,IR0 Charge le bit avec l'adresse mémoire de l'API dans IR0.</p> <p>MOV #0001 ,IR1 Sauvegarde #0001 dans le mot avec la mémoire de l'API dans IR1.</p>
	Décalage constant	<p>+5,IR0</p> <p>+31,IR1</p>	<p>LD + 5 ,IR0 Charge le bit avec l'adresse mémoire de l'API dans IR0 + 5.</p> <p>MOV #0001 +31 ,IR1 Sauvegarde #0001 dans le mot avec la mémoire de l'API dans IR1 + 31</p>
	Décalage DR	<p>DR0 ,IR0</p> <p>DR0 ,IR1</p>	<p>LD DR0 ,IR0 Charge le bit avec l'adresse mémoire de l'API dans IR0 + la valeur dans DR0.</p> <p>MOV #0001 DR0 ,IR1 Sauvegarde #0001 dans le mot avec la mémoire de l'API dans IR1 + la valeur dans DR0.</p>
	Auto-incrémentation	<p>,IR0 ++</p> <p>,IR1 +</p>	<p>LD ,IR0 ++ Incrémente le contenu d'IR0 de 2 après le chargement du bit avec l'adresse mémoire de l'API dans IR0.</p> <p>MOV #0001 ,IR1 + Incrémente le contenu d'IR0 d'1 après le stockage de #0001 dans le mot avec l'adresse mémoire de l'API dans IR1.</p>
	Auto-décrémentation	<p>,- -IR0</p> <p>,-IR1</p>	<p>LD ,- -IR0 Lorsque le contenu d'IR0 a été décrémenté de 2, le bit avec l'adresse mémoire de l'API dans IR0 est chargé.</p> <p>MOV #0001 ,-IR1 Lorsque le contenu d'IR1 est décrémenté d'1, #0001 est stocké dans le mot avec l'adresse mémoire de l'API dans IR1.</p>

Données	Opérande	Formulaire de données	Symbole	Intervalle	Exemple d'application																						
Constante 16 bits	Toutes les données binaires ou un intervalle limité de données binaires	Binaires non signées	#	De #0000 à #FFFF	---																						
		Décimales signées	±	De -32 768 à +32 767	---																						
		Décimales non signées	& (Voir remarque.)	De &0 à &65535	---																						
	Toutes les données BCD ou un intervalle limité de données BCD	BCD	#	De #0000 à #9999	---																						
Constante 32 bits	Toutes les données binaires ou un intervalle limité de données binaires	Binaires non signées	#	De #00000000 à #FFFFFFFF	---																						
		Binaires signées	+	De -2 147 483 648 à +2 147 483 647	---																						
		Décimales non signées	& (Voir remarque.)	De &0 à &429467295	---																						
	Toutes les données BCD ou un intervalle limité de données BCD	BCD	#	De #00000000 à #99999999	---																						
Chaîne de texte	<b>Description</b>		<b>Symbole</b>	<b>Exemples</b>	---																						
	<p>Les données chaînes de texte sont stockées en ASCII (un octet, sauf pour les caractères spéciaux), dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite et du mot le plus à droite (le plus petit) au mot le plus à gauche.</p> <p>00 hex. (code NUL) est stocké dans l'octet le plus à droite du dernier mot si le nombre de caractères est impair.</p> <p>0000 hex. (2 codes NUL) est stocké dans l'octet vacant le plus à gauche et le plus à droite du dernier mot + 1 si le nombre de caractères est pair.</p>		---	<p>'ABCDE'</p> <table border="1"> <tr><td>'A'</td><td>'B'</td></tr> <tr><td>'C'</td><td>'D'</td></tr> <tr><td>'E'</td><td>NUL</td></tr> </table> <p>  </p> <table border="1"> <tr><td>41</td><td>42</td></tr> <tr><td>43</td><td>44</td></tr> <tr><td>45</td><td>00</td></tr> </table> <p>'ABCD'</p> <table border="1"> <tr><td>'A'</td><td>'B'</td></tr> <tr><td>'C'</td><td>'D'</td></tr> <tr><td>NUL</td><td>NUL</td></tr> </table> <p>  </p> <table border="1"> <tr><td>41</td><td>42</td></tr> <tr><td>43</td><td>44</td></tr> <tr><td>00</td><td>00</td></tr> </table>	'A'	'B'	'C'	'D'	'E'	NUL	41	42	43	44	45	00	'A'	'B'	'C'	'D'	NUL	NUL	41	42	43	44	00
'A'	'B'																										
'C'	'D'																										
'E'	NUL																										
41	42																										
43	44																										
45	00																										
'A'	'B'																										
'C'	'D'																										
NUL	NUL																										
41	42																										
43	44																										
00	00																										
<p>Les caractères ASCII pouvant être utilisés dans une chaîne de texte sont les caractères alphanumériques, Katakana et les symboles (sauf pour les caractères spéciaux). Le tableau suivant présente les caractères acceptés.</p>																											

**Remarque** Notation en décimales non signées uniquement si elles sont utilisées pour CX-Programmer.

Caractères ASCII

Bits 0 à 3		Bits 4 à 7															
Binaire	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
	hex.	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0			Espace	0	@	P	~	F				ー	ウ	ミ		
0001	1			!	1	A	Q	a	4			■	ア	チ	ム		
0010	2			"	2	B	R	b	r			「	イ	ツ	ヌ		
0011	3			#	3	C	S	c	s			」	ウ	テ	モ		
0100	4			\$	4	D	T	d	t			、	エ	ト	フ		
0101	5			%	5	E	U	e	u			・	オ	ナ	ユ		
0110	6			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
0111	7			'	7	G	W	g	w			フ	キ	ヌ	ラ		
1000	8			<	8	H	X	h	x			ィ	ク	ネ	リ		
1001	9			>	9	I	Y	i	y			ウ	ケ	ル	ル		
1010	A			*	:	J	Z	j	z			エ	コ	ハ	レ		
1011	B			+	;	K	[	k	[			オ	サ	ヒ	ロ		
1100	C			,	<	L	¥	l	l			ヤ	シ	フ	ワ		
1101	D			-	=	M	]	m	]			ユ	ス	ヘ	ン		
1110	E			.	>	N	^	n	~			ヨ	セ	ホ	ク		
1111	F			/	?	O	_	o	_			ッ	リ	マ	■		

### 2-1-6 Formats de données

Le tableau suivant présente les formats de données pris en charge par la série CS CJ.

Type de données	Format de données	Décimal	4 chiffres hexadécimaux																																																																																			
Binaires non signés	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">15</td><td style="text-align: center;">14</td><td style="text-align: center;">13</td><td style="text-align: center;">12</td><td style="text-align: center;">11</td><td style="text-align: center;">10</td><td style="text-align: center;">9</td><td style="text-align: center;">8</td><td style="text-align: center;">7</td><td style="text-align: center;">6</td><td style="text-align: center;">5</td><td style="text-align: center;">4</td><td style="text-align: center;">3</td><td style="text-align: center;">2</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td> </tr> <tr> <td style="border: 1px solid black; width: 20px; height: 20px;"></td><td style="border: 1px solid black; width: 20px; height: 20px;"></td> </tr> <tr> <td>Binaire</td><td><math>2^{15}</math></td><td><math>2^{14}</math></td><td><math>2^{13}</math></td><td><math>2^{12}</math></td><td><math>2^{11}</math></td><td><math>2^{10}</math></td><td><math>2^9</math></td><td><math>2^8</math></td><td><math>2^7</math></td><td><math>2^6</math></td><td><math>2^5</math></td><td><math>2^4</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td> </tr> <tr> <td>Décimal</td><td>32768</td><td>16384</td><td>8192</td><td>4092</td><td>2048</td><td>1024</td><td>512</td><td>256</td><td>128</td><td>64</td><td>32</td><td>16</td><td>8</td><td>4</td><td>2</td><td>1</td> </tr> <tr> <td>Hex.</td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td> </tr> </table>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																	Binaire	$2^{15}$	$2^{14}$	$2^{13}$	$2^{12}$	$2^{11}$	$2^{10}$	$2^9$	$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	Décimal	32768	16384	8192	4092	2048	1024	512	256	128	64	32	16	8	4	2	1	Hex.	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	De 0 à 65 535	0000 à FFFF
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																							
Binaire	$2^{15}$	$2^{14}$	$2^{13}$	$2^{12}$	$2^{11}$	$2^{10}$	$2^9$	$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$																																																																						
Décimal	32768	16384	8192	4092	2048	1024	512	256	128	64	32	16	8	4	2	1																																																																						
Hex.	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$																																																																						
Binaires signés	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">15</td><td style="text-align: center;">14</td><td style="text-align: center;">13</td><td style="text-align: center;">12</td><td style="text-align: center;">11</td><td style="text-align: center;">10</td><td style="text-align: center;">9</td><td style="text-align: center;">8</td><td style="text-align: center;">7</td><td style="text-align: center;">6</td><td style="text-align: center;">5</td><td style="text-align: center;">4</td><td style="text-align: center;">3</td><td style="text-align: center;">2</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td> </tr> <tr> <td style="border: 1px solid black; width: 20px; height: 20px;"></td><td style="border: 1px solid black; width: 20px; height: 20px;"></td> </tr> <tr> <td>Binaire</td><td><math>2^{15}</math></td><td><math>2^{14}</math></td><td><math>2^{13}</math></td><td><math>2^{12}</math></td><td><math>2^{11}</math></td><td><math>2^{10}</math></td><td><math>2^9</math></td><td><math>2^8</math></td><td><math>2^7</math></td><td><math>2^6</math></td><td><math>2^5</math></td><td><math>2^4</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td> </tr> <tr> <td>Décimal</td><td>32768</td><td>16384</td><td>8192</td><td>4092</td><td>2048</td><td>1024</td><td>512</td><td>256</td><td>128</td><td>64</td><td>32</td><td>16</td><td>8</td><td>4</td><td>2</td><td>1</td> </tr> <tr> <td>Hex.</td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td> </tr> </table> <p style="margin-left: 40px;">↑ Bit de signe : 0 : positif, 1 :</p>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																	Binaire	$2^{15}$	$2^{14}$	$2^{13}$	$2^{12}$	$2^{11}$	$2^{10}$	$2^9$	$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	Décimal	32768	16384	8192	4092	2048	1024	512	256	128	64	32	16	8	4	2	1	Hex.	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	De -32 768 à +32 767	De 8000 à 7FFF
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																							
Binaire	$2^{15}$	$2^{14}$	$2^{13}$	$2^{12}$	$2^{11}$	$2^{10}$	$2^9$	$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$																																																																						
Décimal	32768	16384	8192	4092	2048	1024	512	256	128	64	32	16	8	4	2	1																																																																						
Hex.	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$																																																																						
BCD (Binary Coded Decimal - décimales codées binaires)	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">15</td><td style="text-align: center;">14</td><td style="text-align: center;">13</td><td style="text-align: center;">12</td><td style="text-align: center;">11</td><td style="text-align: center;">10</td><td style="text-align: center;">9</td><td style="text-align: center;">8</td><td style="text-align: center;">7</td><td style="text-align: center;">6</td><td style="text-align: center;">5</td><td style="text-align: center;">4</td><td style="text-align: center;">3</td><td style="text-align: center;">2</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td> </tr> <tr> <td style="border: 1px solid black; width: 20px; height: 20px;"></td><td style="border: 1px solid black; width: 20px; height: 20px;"></td> </tr> <tr> <td>Binaire</td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td><td><math>2^3</math></td><td><math>2^2</math></td><td><math>2^1</math></td><td><math>2^0</math></td> </tr> <tr> <td>Décimal</td><td colspan="3" style="text-align: center;">0 à 9</td><td colspan="3" style="text-align: center;">0 à 9</td><td colspan="3" style="text-align: center;">0 à 9</td><td colspan="3" style="text-align: center;">0 à 9</td> </tr> </table>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																	Binaire	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	Décimal	0 à 9			0 à 9			0 à 9			0 à 9			De 0 à 9999	De 0000 à 9999																					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																							
Binaire	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$	$2^3$	$2^2$	$2^1$	$2^0$																																																																						
Décimal	0 à 9			0 à 9			0 à 9			0 à 9																																																																												

Type de données	Format de données	Décimal	4 chiffres hexadécimaux
Virgule décimale flottante à simple précision	<p>                     Valeur = <math>(-1)^{\text{Signe}} \times 1.[\text{Mantisse}] \times 2^{\text{Exposant}}</math>                      Signe (bit 31)      1 : négatif ou 0 : positif                      Mantisse            Les 23 bits entre le 00 et le bit 22 contiennent la mantisse, c'est-à-dire la portion au-dessous du point décimal dans 1.□□□....., en binaire.                      Exposant            Les 8 bits compris entre les bits 23 et 30 contiennent l'exposant. L'exposant est exprimé en binaire sous la forme 127 plus n dans <math>2^n</math>.                 </p> <p><b>Remarque</b> Ce format, conforme aux normes IEEE754 concernant les données à virgule flottante double précision, est utilisé uniquement avec les instructions qui convertissent ou calculent des données à virgule flottante. Il peut être utilisé pour la configuration ou le contrôle à partir de l'écran d'édition et de surveillance de la mémoire E/S sur le CX-Programmer (non pris en charge par les consoles de programmation). Les utilisateurs n'ont pas besoin de connaître ce format. Il leur suffit de savoir que le formatage occupe quatre mots.</p>	---	---
Virgule décimale flottante à double précision	<p>                     Valeur = <math>(-1)^{\text{Signe}} \times 1.[\text{Mantisse}] \times 2^{\text{Exposant}}</math>                      Signe (bit 63)      1 : négatif ou 0 : positif                      Mantisse            Les 52 bits entre le 00 et le bit 51 contiennent la mantisse, c'est-à-dire la portion au-dessous du point décimal dans 1.□□□....., en binaire.                      Exposant            Les 11 bits compris entre les bits 52 et 62 contiennent l'exposant. L'exposant est exprimé en binaire sous la forme 1023 plus n dans <math>2^n</math>.                 </p> <p><b>Remarque</b> Ce format, conforme aux normes IEEE754 concernant les données à virgule flottante double précision, sert uniquement avec les instructions qui convertissent ou calculent des données à virgule flottante. Il peut être utilisé pour la configuration ou le contrôle à partir de l'écran d'édition et de surveillance de la mémoire E/S sur le CX-Programmer (non pris en charge par les consoles de programmation). Les utilisateurs n'ont pas besoin de connaître ce format. Il leur suffit de savoir que le formatage occupe quatre mots.</p>	---	---

**Données binaires signées**

Dans le cas des données binaires signées, le bit le plus à gauche indique le signe des données binaires 16 bits. La valeur est exprimée par 4 chiffres hexadécimaux.

**Nombres positifs :** Une valeur est positive ou nulle si le bit le plus à gauche est 0 (OFF). En 4 chiffres hexadécimaux, cela correspond à 0000 à 7FFF hex..

**Nombres négatifs :** Une valeur est négative si le bit le plus à gauche est 1 (ON). En 4 chiffres hexadécimaux, cela correspond à 8000 à FFFF hex. L'absolu de la valeur négative (décimale) est exprimé sous la forme d'un complément 2.

**Exemple :** Pour traiter -19 en décimales comme valeur binaire signée, on soustrait 0013 hex. (la valeur absolue de 19) de FFFF hex., puis on ajoute 0001 hex. pour atteindre FFED hex.

	F	F	F	F
	1111	1111	1111	1111
Vrai nombre	0	0	1	3
-)	0000	0000	0001	0011

---

	F	F	E	C
	1111	1111	1110	1100
	0	0	0	1
+)	0000	0000	0000	0001

---

Complément deux	F	F	E	D
	1111	1111	1110	1101

### Compléments

Généralement, le complément de base  $x$  fait référence à un nombre obtenu lorsque tous les chiffres d'un nombre donné sont soustraits de  $x - 1$ , puis que l'on ajoute 1 au chiffre le plus à droite. (Exemple : le complément dix de 7556 est  $9999 - 7556 + 1 = 2444$ .) Un complément permet d'exprimer une soustraction et d'autres fonctions telles que l'addition.

**Exemple :**  $8954 - 7556 = 1398$ ,  $8954 + (\text{le complément dix de } 7556) = 8954 + 2444 = 11\ 398$ . Si vous ne connaissez pas le bit le plus à gauche, le résultat de la soustraction est 1398.

### Compléments deux

Un complément deux est un complément de base deux. Dans ce cas-ci, nous soustrayons tous les chiffres de 1 ( $2 - 1 = 1$ ) et ajoutons un.

**Exemple :** le complément deux du nombre binaire 1101 est 1111 (F hex.) - 1101 (D hex.) + 1 (1 hex.) = 0011 (3 hex.). La ligne suivante présente cette valeur exprimée par 4 chiffres hexadécimaux.

Le complément deux b hex. de a hex. est  $\text{FFFF hex.} - a \text{ hex.} + 0001 \text{ hex.} = b \text{ hex.}$ . Pour déterminer le complément deux b hex. de « a hex. », utilisez  $b \text{ hex.} = 10\ 000 \text{ hex.} - a \text{ hex.}$ .

**Exemple :** pour déterminer le complément deux de 3039 hex., utilisez  $10\ 000 \text{ hex.} - 3030 \text{ hex.} = \text{CFC7 hex.}$ .

De même, utilisez  $a \text{ hex.} = 10000 \text{ hex.} - b \text{ hex.}$  pour déterminer la valeur a hex. à partir du complément deux b hex..

**Exemple :** pour déterminer le complément deux de 3039 hex., utilisez  $10\ 000 \text{ hex.} - 3030 \text{ hex.} = \text{CFC7 hex.}$ .

La série CS/CJ possède deux instructions : NEG(160)(2'S COMPLEMENT) et NEGL(161) (DOUBLE 2'S COMPLEMENT). Elles peuvent servir à déterminer le complément deux à partir du vrai nombre ou à déterminer le vrai nombre à partir du complément deux.

### Données BCD signées

Les données BCD signées sont un format de données spécial permettant d'exprimer des nombres négatifs en BCD. Même si ce format se trouve dans certaines applications, il n'est pas rigoureusement défini et dépend de l'application spécifique. La série CS/CJ prend en charge les instructions suivantes pour convertir les formats de données : SIGNED BCD-TO-BINARY : BINS(470), DOUBLE SIGNED BCD-TO-BINARY : BISL(472), SIGNED BINARY-TO-BCD : BCDS(471) et DOUBLE SIGNED BINARY-TO-

BCD : BDSL(473). Consultez le *Manuel de référence des instructions pour les automates programmables série CS/CJ (W340)* pour plus d'informations.

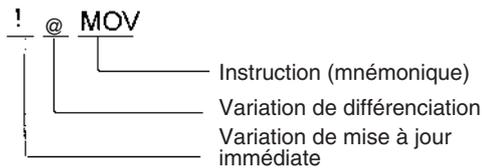
Décimal	hex.adécimal	Binaire	BCD	
0	0	0000		0000
1	1	0001		0001
2	2	0010		0010
3	3	0011		0011
4	4	0100		0100
5	5	0101		0101
6	6	0110		0110
7	7	0111		0111
8	8	1000		1000
9	9	1001		1001
10	A	1010	0001	0000
11	B	1011	0001	0001
12	C	1100	0001	0010
13	D	1101	0001	0011
14	E	1110	0001	0100
15	F	1111	0001	0101
16	10	10000	0001	0110

Décimal	Binaire non signé (4 chiffres hexadécimaux)	Binaire signé (4 chiffres hexadécimaux)
+65 535	FFFF	Impossible à exprimer.
+65 534	FFFE	
.	.	
.	.	
.	.	
+32 769	8001	
+32 768	8000	
+32 767	7FFF	7FFF
+32 766	7FFE	7FFE
.	.	
.	.	
.	.	
+2	0002	0002
+1	0001	0001
0	0000	0000
-1	Impossible à exprimer.	FFFF
-2		FFFE
.		
.		
.		
-32 767		8001
-32 768	8000	

### 2-1-7 Variations des instructions

Les variations suivantes sont disponibles pour les instructions, afin de différencier les conditions d'exécution et de mettre à jour les données lors de l'exécution de l'instruction (mise à jour immédiate).

Variation		Symbole	Description
Différenciation	ON	@	L'instruction fait la différenciation lorsque la condition d'exécution passe à ON.
	OFF	%	L'instruction fait la différenciation lorsque la condition d'exécution passe à OFF.
Mise à jour immédiate		!	Met à jour les données dans la zone E/S spécifiée par les opérandes ou les mots de la carte d'E/S spéciales lors de l'exécution de l'instruction.  (La mise à jour immédiate n'est pas prise en charge par les UC CS1D pour les systèmes d'UC en duplex.)



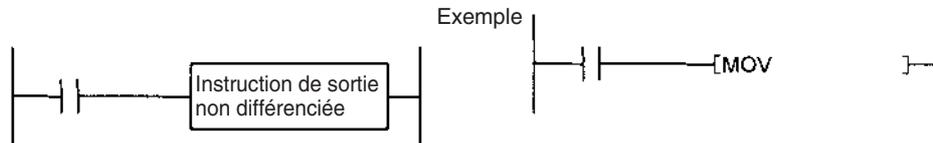
### 2-1-8 Conditions d'exécution

La série CS/CJ contient les types d'instructions standard et spéciales suivantes :

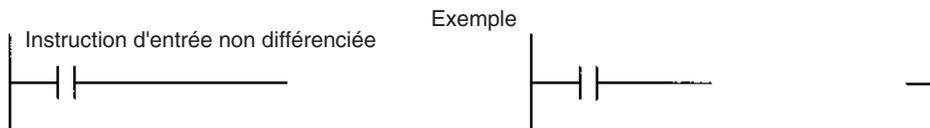
- instructions non différenciées exécutées lors de chaque cycle
- instructions différenciées exécutées une seule fois

#### Instructions non différenciées

Instructions de sortie pour lesquelles les conditions d'exécution doivent être exécutées une fois par cycle pendant que la condition d'exécution est valide (ON ou OFF).



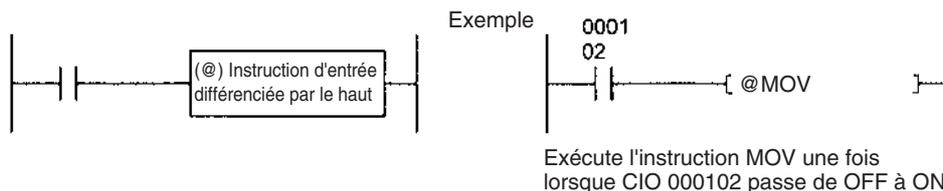
Instructions d'entrée qui créent des départs logiques et instructions intermédiaires qui lisent l'état des bits, effectuent des comparaisons, testent les bits ou effectuent d'autres types de traitement à chaque cycle. Si les résultats sont ON, le flux d'alimentation est envoyé (c-à-d, la condition d'exécution passe à ON).



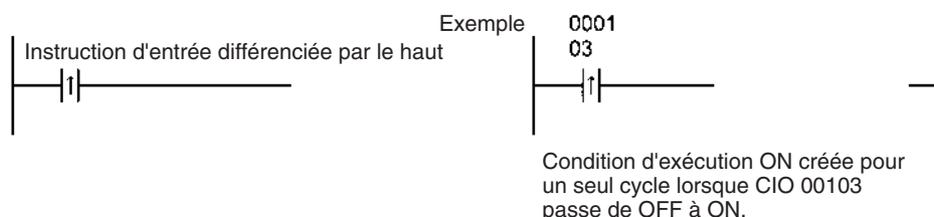
### Instructions différenciées à l'entrée

#### Instructions différenciées par le haut (instruction précédée de @)

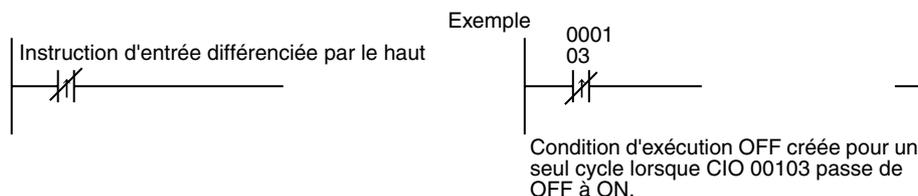
- **Instructions de sortie** : L'instruction n'est exécutée que pendant le cycle au cours duquel la condition d'exécution passe à ON (OFF → ON) et n'est pas exécutée lors des cycles suivants.



- **Instructions d'entrée (départs logiques et instructions intermédiaires)** : L'instruction lit l'état des bits, effectue des comparaisons, teste des bits ou effectue d'autres types de traitement lors de chaque cycle ; elle enverra une condition d'exécution ON (flux d'alimentation) lorsque les résultats passeront de OFF à ON. La condition d'exécution passera à OFF lors du cycle suivant.

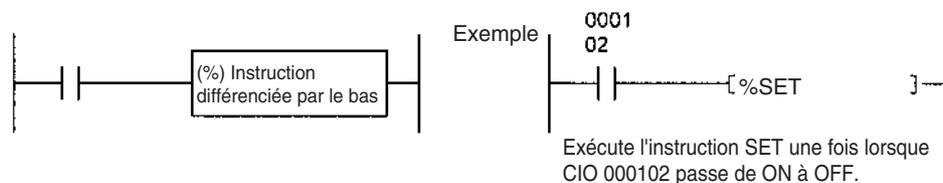


- **Instructions d'entrée (départs logiques et instructions intermédiaires)** : L'instruction lit l'état des bits, effectue des comparaisons, teste des bits ou effectue d'autres types de traitement lors de chaque cycle ; elle enverra une condition d'exécution OFF (arrêt du flux d'alimentation) lorsque les résultats passeront de OFF à ON. La condition d'exécution passera à ON lors du cycle suivant.

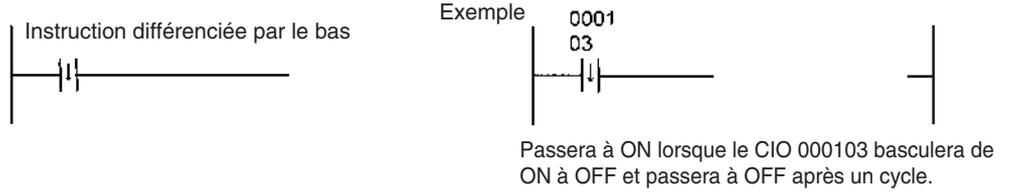


#### Instructions différenciées par le bas (instruction précédée de %)

- **Instructions de sortie** : L'instruction n'est exécutée que pendant le cycle au cours duquel la condition d'exécution passe à OFF (ON → OFF) et n'est pas exécutée lors des cycles suivants.

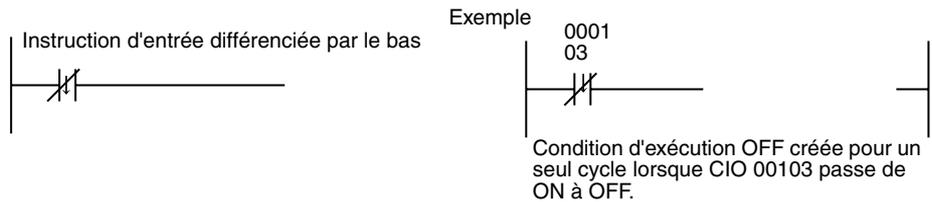


- Instructions d'entrée (départs logiques et instructions intermédiaires) :** L'instruction lit l'état des bits, effectue des comparaisons, teste des bits ou effectue d'autres types de traitement lors de chaque cycle ; elle enverra la condition d'exécution (flux d'alimentation) lorsque les résultats passeront de ON à OFF. La condition d'exécution passera à OFF lors du cycle suivant.



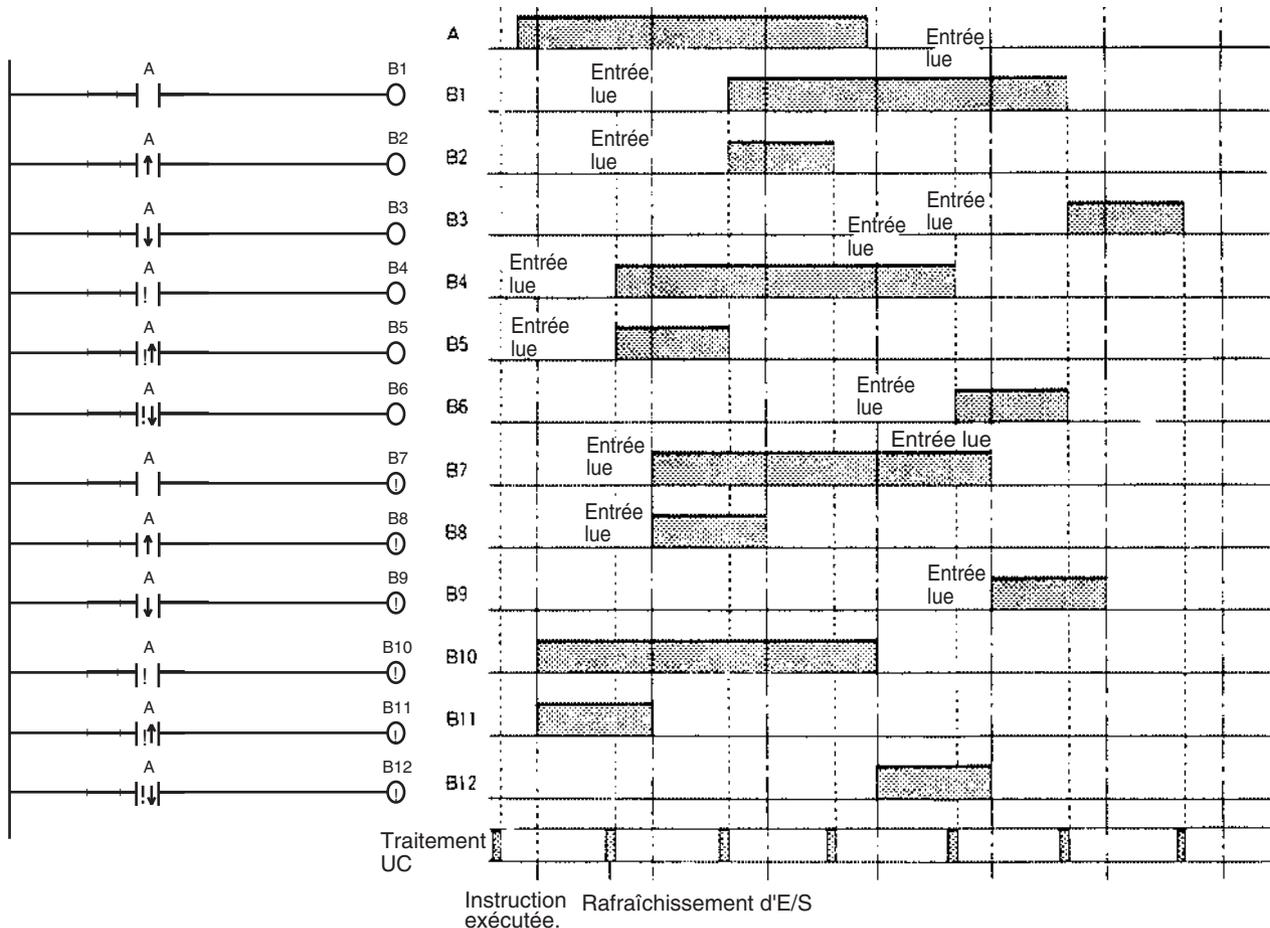
**Remarque** Contrairement aux instructions différenciées par le haut, la variation (%) de différenciation par le bas ne peut être ajoutée qu'aux instructions LD, AND, OR, SET et RSET. Pour exécuter la différenciation par le bas avec d'autres instructions, combinez les instructions à une instruction DIFD ou DOWN. NOT ne peut être ajouté aux instructions que si vous utilisez une UC CS1-H, CJ1-H, CJ1M ou CS1D.

- Instructions d'entrée (départs logiques et instructions intermédiaires) :** L'instruction lit l'état des bits, effectue des comparaisons, teste des bits ou effectue d'autres types de traitement lors de chaque cycle ; elle enverra une condition d'exécution OFF (arrêt du flux d'alimentation) lorsque les résultats passeront de ON à OFF. La condition d'exécution passera à ON lors du cycle suivant.



### 2-1-9 Temporisation des instructions d'E/S

Le graphique suivant présente les différentes phases de fonctionnement des instructions individuelles qui utilisent un programme ne contenant que des instructions LD et OUT.

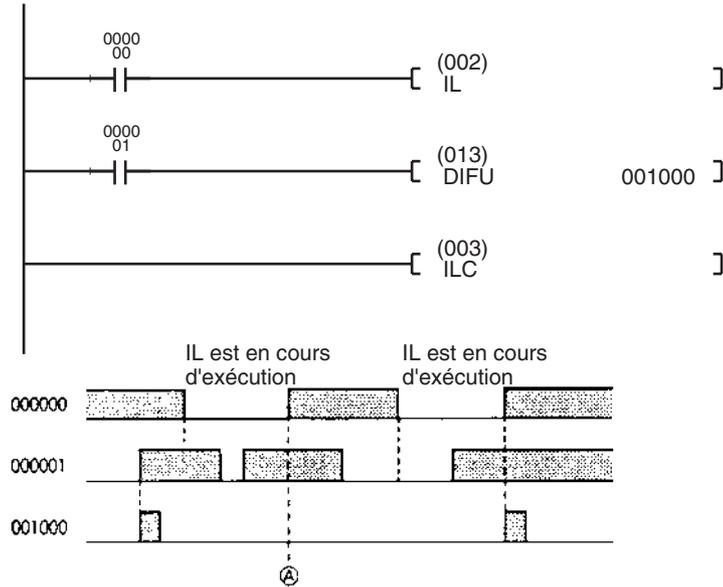


### Instructions différenciées

- Une instruction différenciée contient un drapeau interne signalant l'état de la valeur précédente (ON ou OFF). Au début du fonctionnement, les drapeaux de la valeur précédente des instructions différenciées par le haut (DIFU et @) sont définis sur ON et ceux des instructions différenciées par le bas (DIFD et %) sont définis sur OFF. Cela permet d'éviter que les résultats de la différenciation ne soient envoyés accidentellement au début de l'opération.
- Une instruction différenciée par le haut (DIFU ou @) n'enverra ON que lorsque la condition d'exécution sera sur ON et que le drapeau de la valeur précédente sera sur OFF.

• **Utilisation dans Interlocks (instructions IL – ILC)**

Dans l'exemple suivant, le drapeau de la valeur précédente de l'instruction différenciée garde la valeur verrouillée précédente et n'enverra pas de résultat différencié au point A car la valeur ne sera pas mise à jour tant que le verrouillage sera activé.



• **Utilisation dans les sauts (instructions JMP – JME) :** Comme dans le cas des verrouillages, le drapeau de la valeur précédente d'une instruction différenciée ne change pas si une instruction est sautée. La valeur précédente est donc maintenue. Les instructions différenciées par le haut et par le bas n'enverront la condition d'exécution que si l'état de l'entrée a changé par rapport à l'état indiqué dans le drapeau de la valeur précédente.

- Remarque a)** N'utilisez pas le drapeau Toujours ON ou A20011 (drapeau du premier cycle) comme bit d'entrée pour une instruction différenciée par le haut. L'instruction ne sera jamais exécutée.
- b)** N'utilisez pas le drapeau Toujours OFF comme bit d'entrée pour une instruction de différenciation par le bas. L'instruction ne sera jamais exécutée.

### 2-1-10 Programmation de la mise à jour

Pour mettre à jour les E/S externes, vous pouvez utiliser l'une des méthodes suivantes :

- Mise à jour cyclique
- Mise à jour immédiate (instruction spécifiée !, instruction IORF)

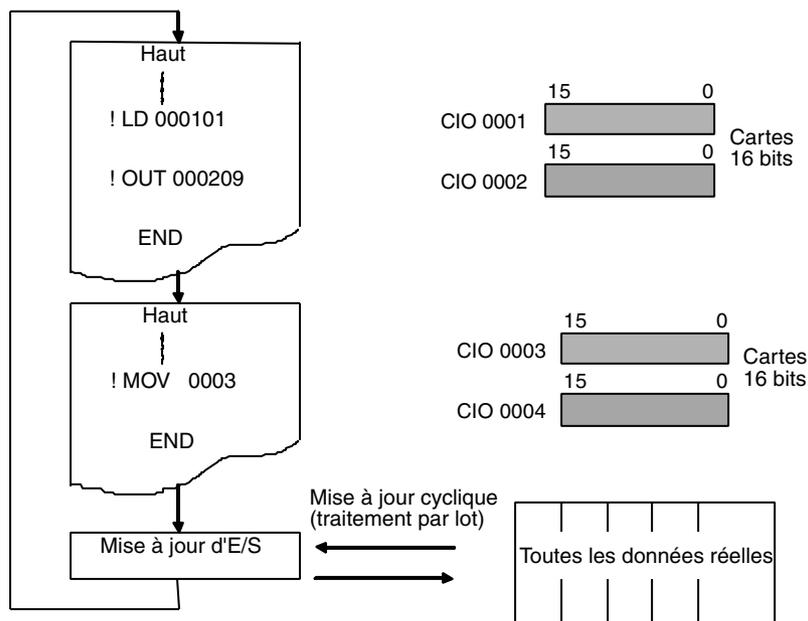
Consultez le chapitre sur le fonctionnement de l'UC dans le *Manuel d'utilisation de la série CS/CJ* pour plus de détails sur la mise à jour d'E/S.

#### Mise à jour cyclique

Chaque programme affecté à une tâche cyclique prête ou à une tâche répondant à la condition d'interruption sera exécuté depuis l'adresse de début du programme jusqu'à l'instruction END(001). Lorsque toutes ces tâches auront été exécutées (tâches cycliques prêtes ou tâches répondant à la condition d'interruption), la mise à jour cyclique commence la mise à jour de tous les points E/S en même temps.

**Remarque**

Les programmes peuvent être exécutés dans plusieurs tâches. Les E/S seront mises à jour après la dernière instruction END(001) dans les programmes affectés au nombre le plus élevé (parmi toutes les tâches cycliques prêtes) et elles ne seront pas mises à jour après l'instruction END(001) dans les programmes affectés à d'autres tâches cycliques.



Exécutez une instruction IORF pour tous les mots nécessaires avant l'instruction END(001) si la mise à jour d'E/S est nécessaire dans d'autres tâches.

#### Mise à jour immédiate

##### Instructions avec variation de mise à jour (!)

Les E/S sont mises à jour comme expliqué dans le tableau ci-dessous lorsqu'une instruction est exécutée si un bit d'E/S réel est spécifié comme opérande.

Cartes	Données mises à jour
Cartes d'E/S standard C200H (série CS uniquement)	Les E/S des 16 bits contenant le bit seront mises à jour.
Cartes d'E/S standard CJ	

- Si une opérande mot est spécifiée pour une instruction, les 16 bits spécifiés des E/S seront mis à jour.
- Les entrées de l'opérande d'entrée ou de l'opérande source seront mises à jour juste avant l'exécution d'une instruction.
- Les sorties de l'opérande de sortie ou de l'opérande de destination (D) seront mise à jour juste après l'exécution d'une instruction.

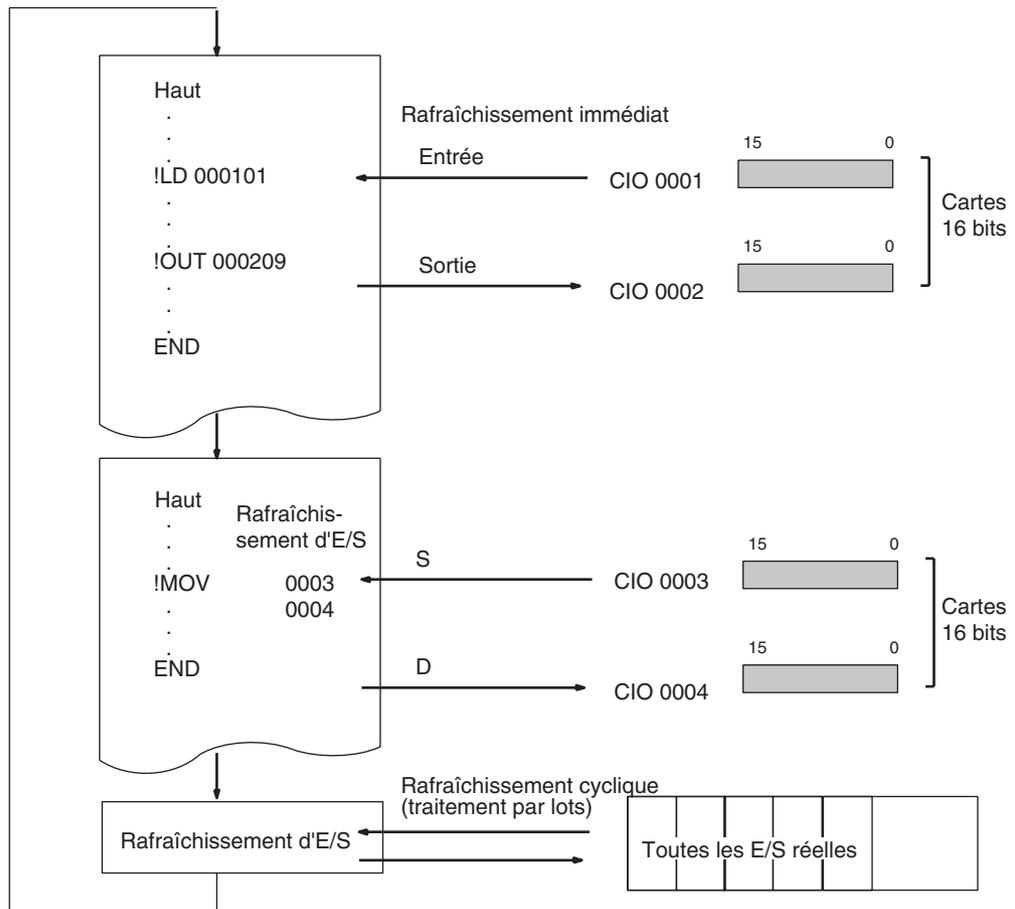
Ajoutez un point d'exclamation (!) (option de mise à jour immédiate) devant l'instruction.

**Remarque** La mise à jour immédiate n'est pas prise en charge par les UC CS1D pour systèmes d'UC en duplex, mais par contre elles prennent en charge la mise à jour pour les instructions IORF(097) et DLNK(226).

**Cartes mises à jour pour l'instruction I/O REFRESH**

<b>Emplacement</b>	Rack UC ou rack d'extension E/S (mais pas les racks esclaves SYSMAC BUS)		
<b>Cartes</b>	Cartes d'E/S standard	Cartes d'E/S standard série CS/CJ	Mises à jour
		Cartes d'E/S standard C200H (voir remarque.)	Mises à jour
		Cartes d'E/S haute densité groupe 2 C200H (voir remarque.)	Pas de mise à jour
	Cartes d'E/S spéciales	Pas de mise à jour	

**Remarque** Les cartes d'E/S C200H ne peuvent pas être montées sur les API série CJ.



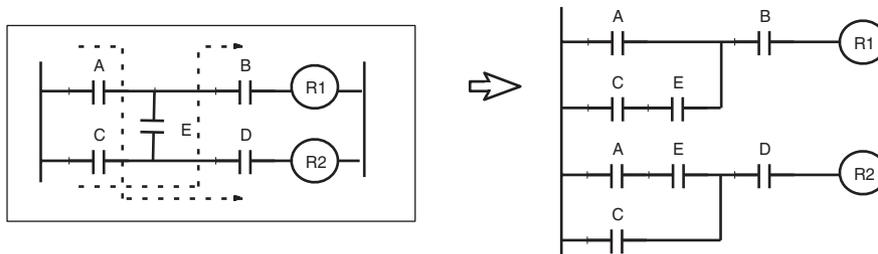
**Cartes mises à jour pour les instructions IORF(097) ou DLNK(226)**

Il existe une instruction spéciale, I/O REFRESH (IORF(097)) qui met à jour toutes les données d'E/S réelles d'une plage de mots spécifiée. Grâce à cette instruction, toutes les données réelles, ou uniquement celles de la plage spécifiée, peuvent être mises à jour pendant un cycle. IORF peut également servir à mettre à jour les mots affectés aux cartes d'E/S spéciales.

Une autre instruction, CPU BUS UNIT REFRESH (DLNK(226)) permet de mettre à jour tous les mots affectés aux cartes réseau des zones CIO et DM ainsi que d'effectuer une mise à jour spéciale de la carte (mise à jour des liaisons de données). DLNK(226) n'est prise en charge que par les UC CS1-H, CJ1-H, CJ1M ou CS1D.

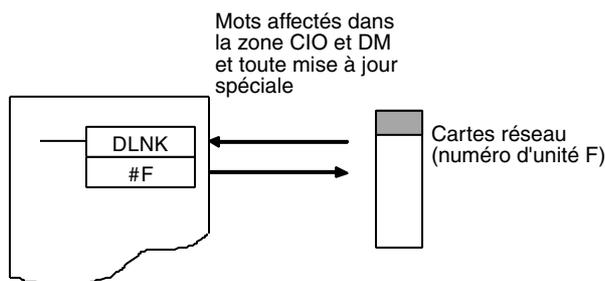
**Cartes mises à jour pour les instructions IORF(097)**

<b>Emplacement</b>	Rack UC ou rack d'extension E/S (mais pas les racks esclaves SYSMAC BUS)		
<b>Cartes</b>	Cartes d'E/S standard	Cartes d'E/S standard série CS/CJ	Mises à jour
		Cartes d'E/S standard C200H	Mises à jour
		Cartes d'E/S haute densité groupe 2 C200H	Mises à jour
	Cartes d'E/S spéciales		Mises à jour
	Cartes réseau		Pas mises à jour



**Cartes mises à jour pour DLNK(226)**

<b>Emplacement</b>	Rack UC ou rack d'extension E/S (mais pas les racks esclaves SYSMAC BUS)	
<b>Cartes</b>	Cartes d'E/S standard	Pas mises à jour
	Cartes d'E/S spéciales	Pas mises à jour
	Cartes réseau Mots affectés à la carte dans la zone CIO Mots affectés à la carte dans la zone DM Mise à jour spéciale pour la carte (liaisons de données pour les cartes Controller Link ou SYSMAC Link ou E/S déportées pour les cartes DeviceNet)	Mises à jour



## 2-1-11 Capacité de programme

Le tableau suivant présente les capacités maximales du programme des UC série CS/CJ de tous les programmes utilisateurs (c-à-d, la capacité totale de toutes les tâches). La capacité est exprimée en nombre maximal de pas. Il ne faut absolument pas dépasser la capacité du programme. En cas de tentative de dépassement, l'écriture du programme sera désactivée.

Chaque instruction prend de 1 à 7 pas. Consultez *10-5 Temps d'exécution des instructions et nombre de pas* dans le *Manuel d'utilisation* afin de connaître le nombre de pas spécifiques de chaque instruction. (La longueur de chaque instruction augmentera de 1 si vous utilisez une opérande deux fois plus longue.)

Série	UC	Capacité max. du programme	Points E/S
Série CS	CS1H-CPU67H/CPU67-E	250 Kpas	5 120
	CS1D-CPU67H	250 Kpas	
	CS1D-CPU67S	250 Kpas	
	CS1H-CPU66H/CPU66-E	120 Kpas	
	CS1H-CPU65H/CPU65-E	60 Kpas	
	CS1D-CPU65H	60 Kpas	
	CS1D-CPU65S	60 Kpas	
	CS1H-CPU64H/CPU64-E	30 Kpas	
	CS1H-CPU63H/CPU63-E	20 Kpas	
	CS1G-CPU45H/CPU45-E	60 Kpas	
	CS1G-CPU44H/CPU44-E	30 Kpas	1 280
	CS1D-CPU44S	30 Kpas	
	CS1G-CPU43H/CPU43-E	20 Kpas	960
	CS1G-CPU42H/CPU42-E	10 Kpas	
	CS1D-CPU42S	10 Kpas	
Série CJ	CJ1H-CPU67H	250 Kpas	2 560
	CJ1H-CPU66H	120 Kpas	
	CJ1H-CPU65H	60 Kpas	
	CJ1G-CPU45H/CPU45	60 Kpas	1 280
	CJ1G-CPU44H/CPU44	30 Kpas	
	CJ1G-CPU43H	20 Kpas	960
	CJ1G-CPU42H	10 Kpas	
	CJ1M-CPU23/CPU13	20 Kpas	640
	CJ1M-CPU22/CPU12	10 Kpas	320
CJ1M-CPU11/CPU21	5 K pas	160	

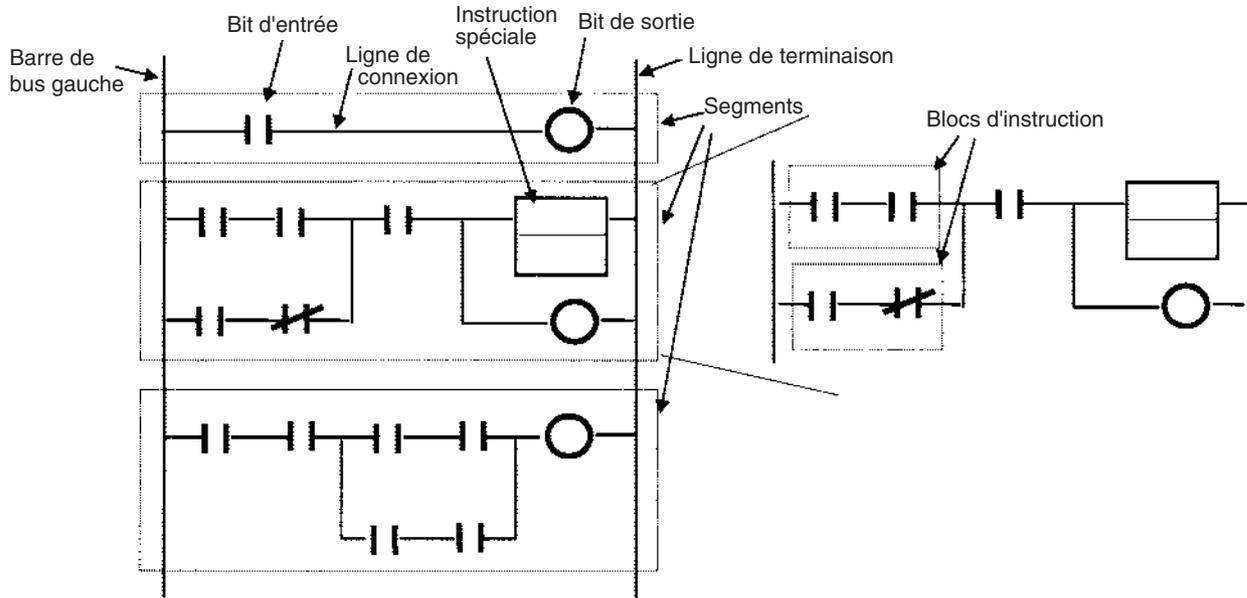
**Remarque** La capacité du programme des API série CJ est mesurée en pas tandis que la capacité de mémoire des anciens API OMRON, tels que ceux des séries C200HX/HG/HE et CV, était mesurée en mots. Consultez les informations à la fin du chapitre *10-5 Temps d'exécution des instructions et nombre de pas* dans le *Manuel d'utilisation* de votre API pour connaître les lignes directrices de la conversion des capacités des programmes par rapport aux anciens API d'OMRON.

## 2-1-12 Concepts de base de programmation du schéma contact

Les instructions sont exécutées dans l'ordre établi dans la mémoire (ordre mnémonique). Les concepts de base de programmation ainsi que l'ordre d'exécution doivent être respectés.

**Structure générale du schéma contact**

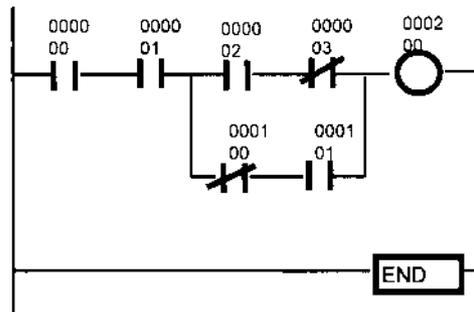
Un schéma contact comprend les éléments suivants : des barres de bus gauche et droite, des lignes de connexions, des bits d'entrée, des bits de sortie et des instructions spéciales. Un programme consiste en une ou plusieurs équations logiques. Une équation logique est une unité qui peut être partagée lorsque le bus est séparé horizontalement. En forme mnémotechnique, une équation logique est l'ensemble des instructions allant de l'instruction LD/LD NOT à l'instruction de sortie située juste avant les instructions LD/LD NOT suivantes. Une équation logique d'un programme consiste en un bloc d'instructions qui commence par une instruction LD/LD NOT indiquant un départ logique.



**Mnémotechnique**

Un programme mnémotechnique est une série d'instructions de schéma contact exprimées sous la forme mnémotechnique. Il possède des adresses de programmes et chaque adresse de programme correspond à une instruction. Les adresses de programmes sont formées de six chiffres, la première étant 000000.

**Exemple**

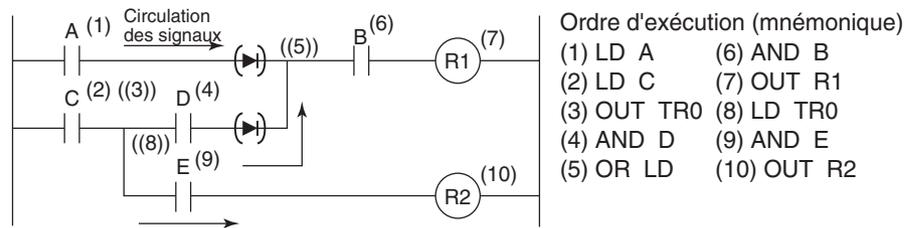


Adresse de programme	Instruction (mnémotechnique)	Opérande
000000	LD	000000
000001	AND	000001
000002	LD	000002
000003	AND NOT	000003
000004	LD NOT	000100
000005	AND	000101
000006	OR LD	
000007	AND LD	
000008	OUT	000200
000009	END	

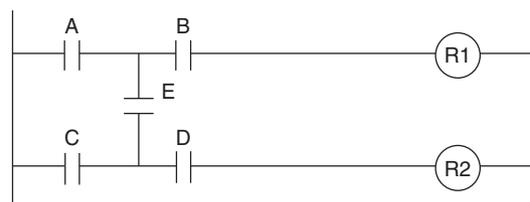
Concepts de base du schéma contact

- 1,2,3... 1. Lorsque les API exécutent les schémas contacts, la circulation des signaux (flux d'alimentation) se fait toujours de gauche à droite. Vous ne pouvez pas utiliser de programmation pour laquelle il faut un flux d'alimentation de droite à gauche. Par conséquent, le débit est différent de celui des circuits faits de relais de commandes câblées. Par exemple, lorsque le circuit « a » est implémenté dans un programme API, l'alimentation passe comme si les diodes entre parenthèses étaient insérées et que la bobine R2 ne pouvait pas être activée avec le contact D inclus. L'ordre réel d'exécution est indiqué sur la droite sous forme mnémotechnique. Pour obtenir cette opération sans les diodes imaginaires, il faut réécrire le circuit. Le flux d'alimentation du circuit « b » ne peut pas être non plus programmé directement et doit être réécrit.

Circuit "a"



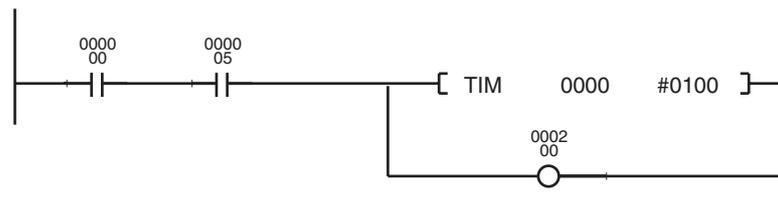
Circuit "b"



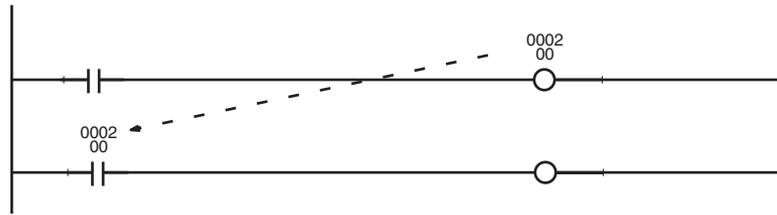
Dans le circuit « a », la bobine R2 ne peut être activée si le contact D est inclus.

Dans le circuit « b », le contact E inclus ne peut être réécrit dans un schéma contact. Le programme doit être réécrit.

2. Le nombre de bits d'E/S, de bits de travail, de temporisations et d'autres bits d'entrées est illimité. Cependant, les équations logiques doivent rester aussi claires et simples que possible, même si cela implique l'utilisation de plus de bits d'entrée afin qu'elles soient plus faciles à comprendre et à gérer.
3. Le nombre de bits d'entrée pouvant être connectés aux équations logiques série ou parallèles est illimité.
4. Vous pouvez connecter deux ou plusieurs bits de sortie en parallèle.



5. Les bits de sortie peuvent également servir de bits d'entrée.



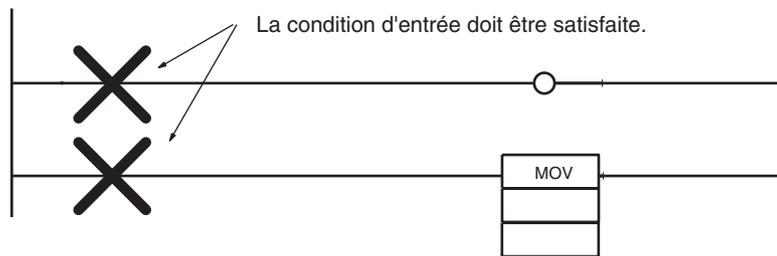
**Restrictions**

1,2,3...

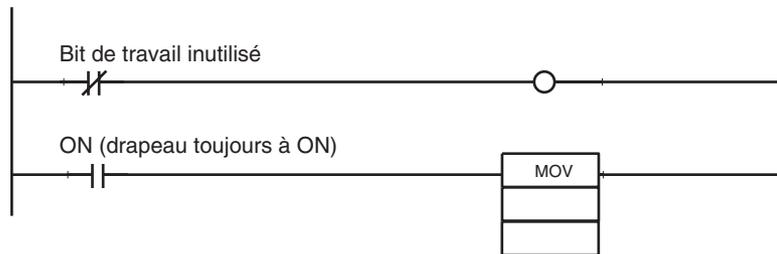
1. Un schéma contact doit être fermé de manière à ce que les signaux (flux d'alimentation) passent de la barre de bus gauche à la barre de bus droite. Une erreur d'équation logique se produira si le programme n'est pas fermé (mais le programme peut être exécuté).



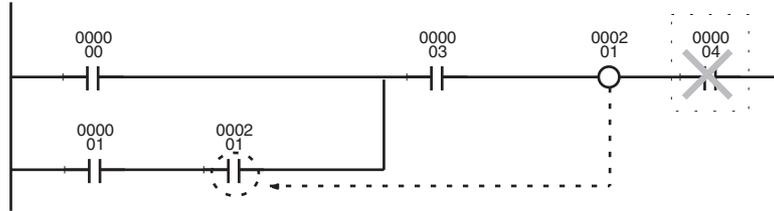
2. Les bits de sortie, les temporisations, les compteurs et les autres instructions de sortie ne peuvent être connectés directement à la barre de bus gauche. Si l'un de ces éléments est connecté directement à la barre de bus gauche, la vérification du programme par un périphérique de programmation mentionnera une erreur d'équation logique. (Le programme peut s'exécuter mais les instructions Out et MOV(021) ne seront pas exécutées.)



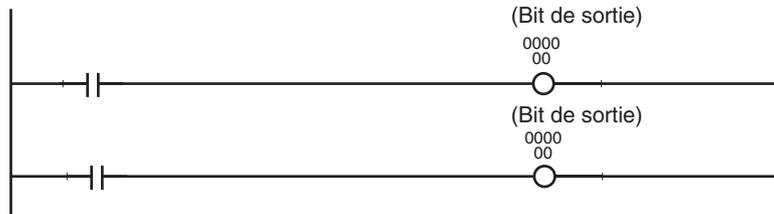
Insérez un bit de travail NF ou un drapeau de condition ON (drapeau Toujours ON) si l'entrée doit rester sur ON en permanence.



3. Un bit d'entrée doit toujours être inséré avant, et jamais après, une instruction de sortie telle qu'un bit de sortie. Si vous l'insérez avant une instruction de sortie, la vérification du programme par un périphérique de programmation affichera une erreur d'emplacement.



4. Un même bit de sortie ne peut être programmé qu'une seule fois dans une instruction de sortie. Les instructions d'un schéma contact sont exécutées dans l'ordre depuis l'équation logique supérieure, en un seul cycle. Par conséquent, le résultat de l'instruction de sortie des équations logiques inférieures se reflètera en fin de compte dans le bit de sortie et les résultats de toutes les instructions précédentes contrôlant le même bit seront réécrits.

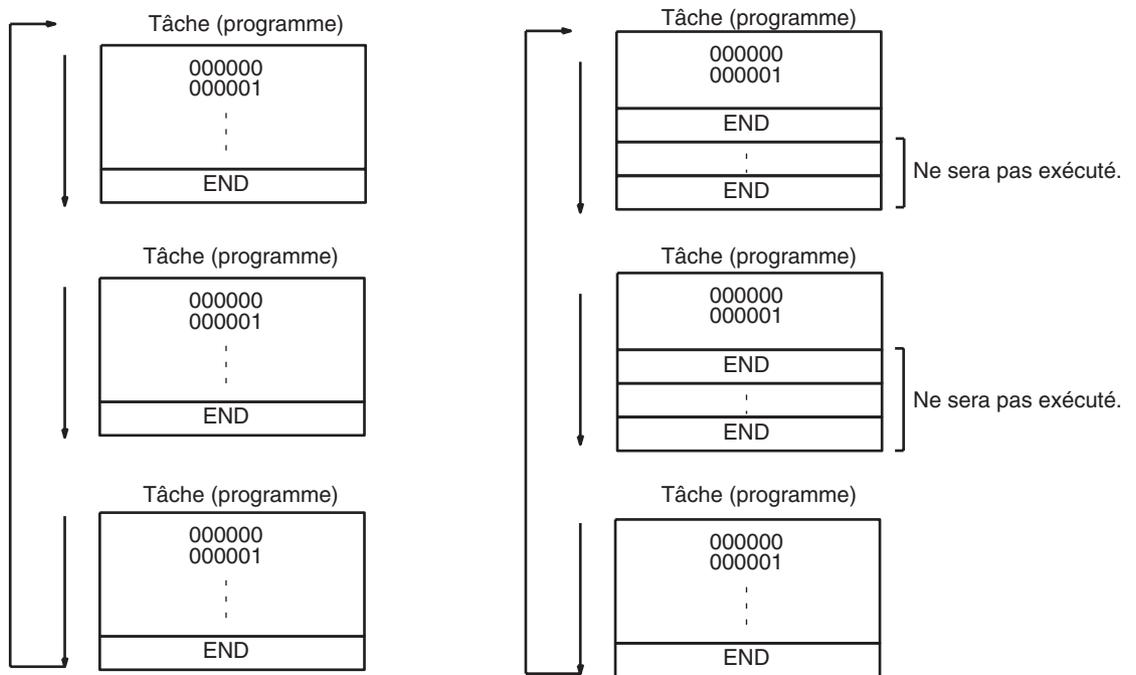


5. Un bit d'entrée ne peut pas être utilisé dans une instruction de SORTIE (OUT).



6. Une instruction END(001) doit être insérée à la fin du programme dans chaque tâche.
  - Si un programme ne contenant pas d'instruction END(001) démarre, une erreur de programme « No End Instruction » s'affiche, le voyant ERR/ALM sur la face avant de l'UC s'allume et le programme n'est pas exécuté.
  - Si un programme contient plus d'une instruction END(001), il sera exécuté uniquement jusqu'à la première instruction END(001).

- Les programmes de débogage fonctionneront mieux si une instruction END(001) est insérée à plusieurs points de rupture entre les équations logiques séquentielles et que l'instruction END(001) du milieu est supprimée après la vérification du programme.

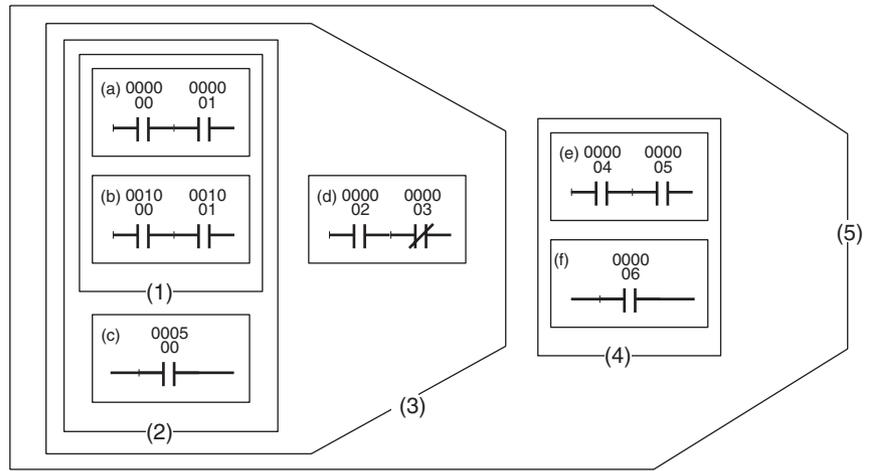
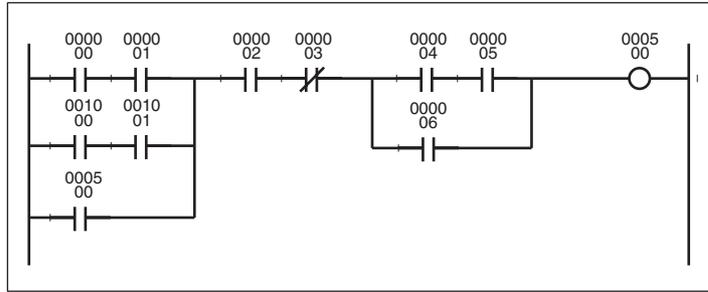


### 2-1-13 Insertion de mnémoniques

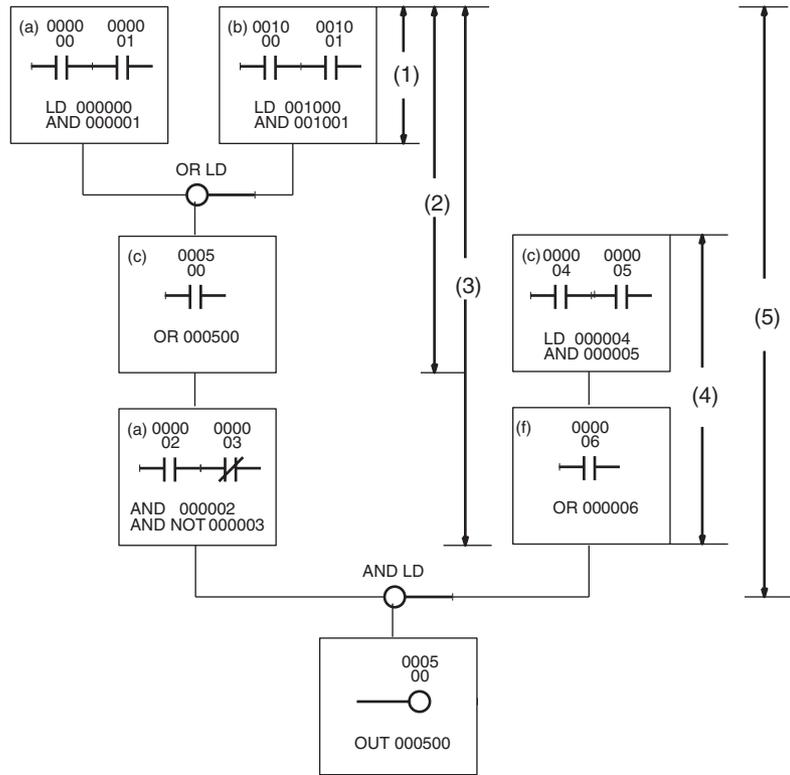
Un départ logique est effectué grâce à une instruction LD/LD NOT. La zone comprise entre le départ logique et l'instruction située juste avant l'instruction LD/LD NOT suivante est considérée comme un seul bloc d'instructions.

Créez une seule équation logique constituée de deux blocs d'instructions à l'aide d'une instruction AND LD sur AND les blocs ou à l'aide d'une instruction OR LD sur OR les blocs. L'exemple suivant illustre une équation logique complexe qui permet d'expliquer la procédure d'insertion de mnémoniques (résumé et ordre des équations logiques).

1,2,3... 1. Commencez par séparer l'équation logique en petits blocs de (a) à (f).



- Programmer les blocs du haut vers le bas, puis de gauche à droite.

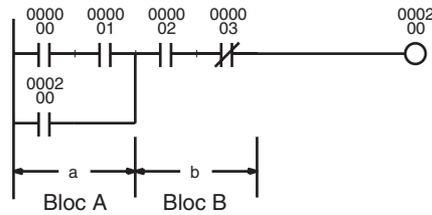


	Adresse	Instruction	Opérande
(a)	000200	LD	000000
	000201	AND	000001
(b)	000202	LD	001000
	000203	AND	001001
	000204	OR LD	---
(c)	000205	OR	000500
(d)	000206	AND	000002
	000207	AND NOT	000003
	000208	LD	000004
(e)	000209	AND	000005
	000210	OR	000006
	000211	AND LD	---
	000212	SORTIE	000500

Vertical dimension lines (1) through (5) are shown to the right of the table, corresponding to the vertical extent of the logic blocks in the diagram above.

### 2-1-14 Exemples de programmes

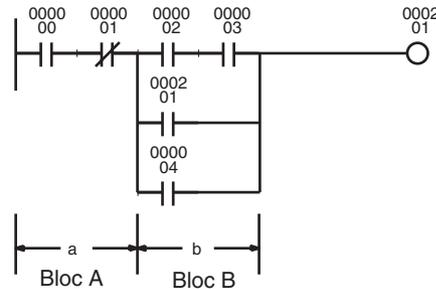
1,2,3... 1. Equations logiques parallèles/série



Instruction	Opérandes
LD	000000
AND	000001
OR	000200
AND	000002
AND NOT	000003
SORTIE	000200

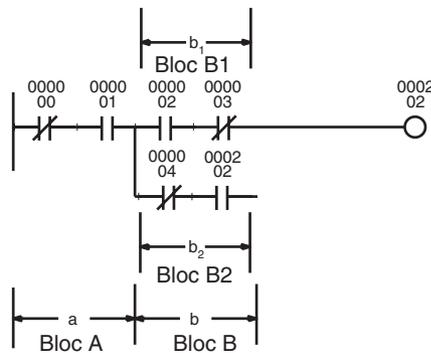
Programmez l'instruction parallèle dans le bloc A, puis dans le bloc B.

2. Equations logiques série/parallèles



Instruction	Opérandes
LD	000000
AND NOT	000001
LD	000002
AND	000003
OR	000201
OR	000004
AND LD	---
SORTIE	000201

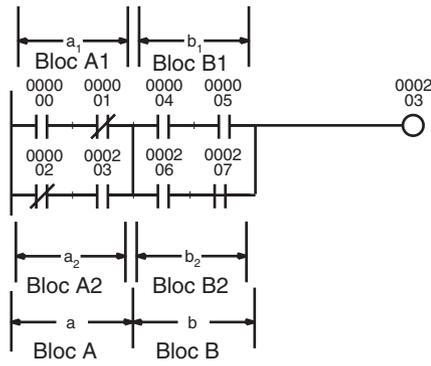
- Séparez le segment en deux blocs, A et B, et programmez-les chacun individuellement.
- Reliez les blocs A et B avec un AND LD.
- Programmez le bloc A.



Instruction	Opérandes
LD NOT	000000
AND	000001
LD	000002
AND NOT	000003
LD NOT	000004
AND	000202
OR LD	---
AND LD	---
SORTIE	000202

- Programmez le bloc B<sub>1</sub>, puis programmez le bloc B<sub>2</sub>.
- Reliez les blocs B<sub>1</sub> et B<sub>2</sub> avec un OR LD, puis les blocs A et B avec un AND LD.

3. Exemples de connexion série dans une équation logique série



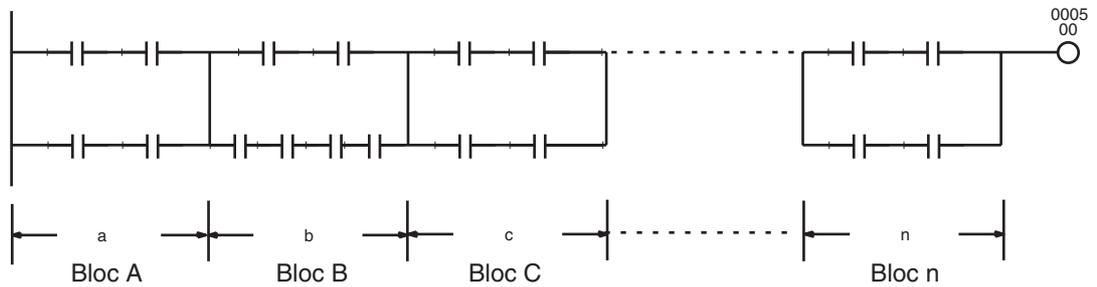
Instruction	Opérandes	
LD	000000	} a <sub>1</sub>
AND NOT	000001	
LD NOT	000002	} a <sub>2</sub>
AND	000003	
OR LD	---	} a <sub>1</sub> + a <sub>2</sub>
LD	000004	} b <sub>1</sub>
AND	000005	
LD	000006	} b <sub>2</sub>
AND	000007	
OR LD	---	} b <sub>1</sub> + b <sub>2</sub>
AND LD	---	} a · b
SORTIE	000203	

Programmez le bloc A<sub>1</sub>, programmez le bloc A<sub>2</sub>, puis connectez les blocs A<sub>1</sub> et A<sub>2</sub> avec un OR LD.

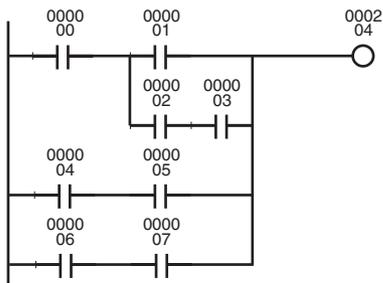
Programmez B<sub>1</sub> et B<sub>2</sub> de la même façon.

Connectez les blocs A et B avec un AND LD.

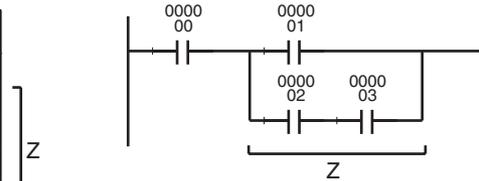
Répétez l'opération pour tous les blocs A à n présents.



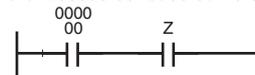
4. Equations logiques complexes



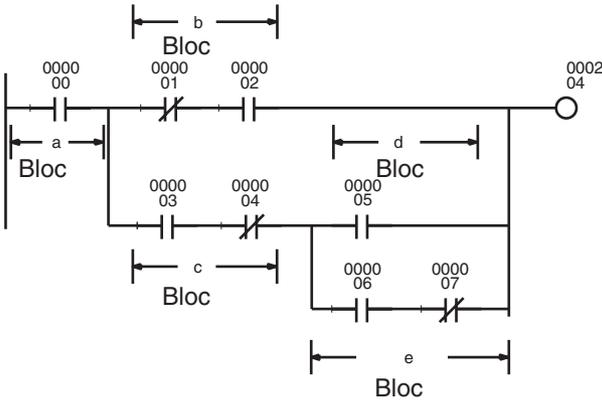
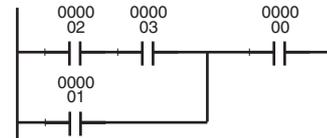
Instruction	Opérande
LD	000000
LD	000001
LD	000002
AND	000003
OR LD	---
AND LD	---
LD	000004
AND	000005
OR LD	---
LD	000006
AND	000007
OR LD	---
SORTIE	000204



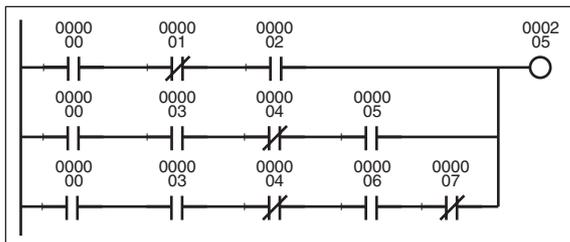
Le schéma ci-dessus est basé sur le schéma suivant.



Vous pouvez écrire un programme plus simple en le réécrivant comme indiqué ci-après.

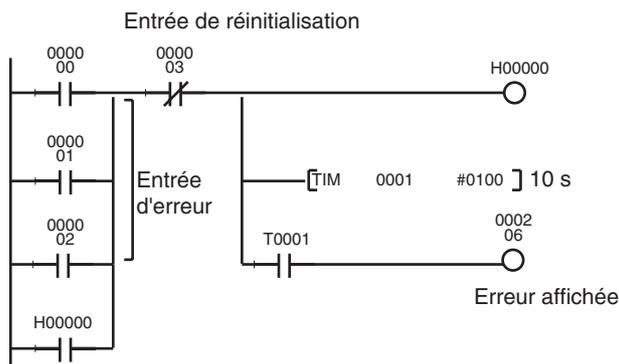


Vous pouvez réécrire le segment ci-dessus comme indiqué ci-après :



Instruction	Opérande
LD	000000
LD NOT	000001
AND	000002
LD	000003
AND NOT	000004
LD	000005
LD	000006
AND NOT	000007
OR LD	---
AND LD	---
OR LD	---
AND LD	---
SORTIE	000205

a  
 b  
 c  
 d  
 e  
 $d + e$   
 $(d + e) \cdot c$   
 $(d + e) \cdot c + b$   
 $((d + e) \cdot c + b) \cdot a$



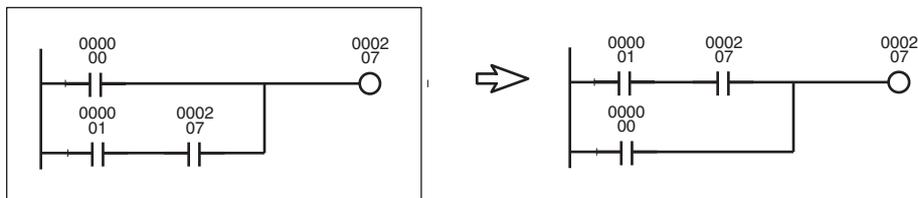
Instruction	Opérande
LD	000000
OR	000001
OR	000002
OR	H00000
AND NOT	000003
SORTIE	H00000
TIM	0001
	0100
AND	T0001
SORTIE	000206

Si un bit de maintien est utilisé, l'état ON/OFF sera conservé en mémoire, même en cas de mise hors tension, et le signal d'erreur restera actif à la prochaine mise sous tension.

5. Equations logiques auxquelles il faut faire attention ou devant être réécrites

**Instructions OR et OL LD**

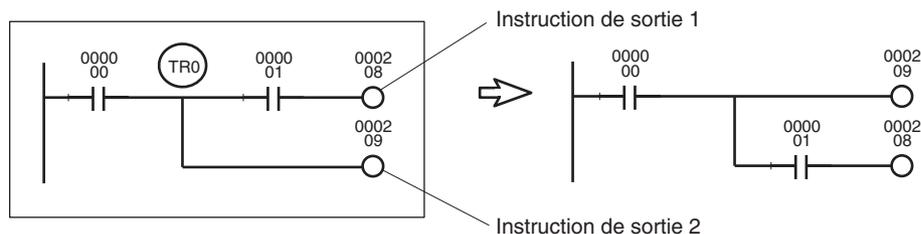
Dans le cas d'une instruction OR ou OR NOT, un OR est pris en compte avec les résultats de la logique du schéma contact depuis l'instruction LD ou LD NOT jusqu'à l'instruction OR ou OR NOT. Ainsi, il est possible de réécrire les équations logiques afin que l'instruction OR LD ne soit pas obligatoire.



Exemple : Une instruction OR LD sera nécessaire si les équations logiques sont programmées comme illustré, sans modification. Il est possible de supprimer quelques étapes en réécrivant les équations logiques comme illustré.

**Dérivations de l'instruction de sortie**

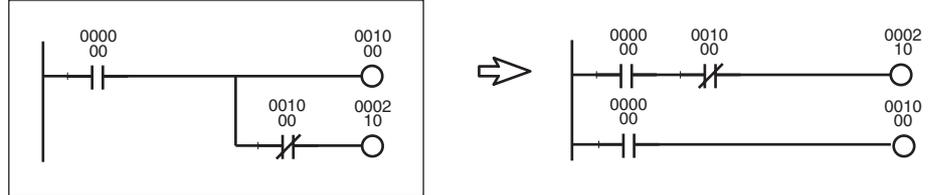
Un bit TR sera nécessaire si une dérivation se trouve devant une instruction AND ou AND NOT. Le bit TR ne sera pas nécessaire si la dérivation se trouve à un point connecté directement aux instructions de sortie ; l'instruction AND ou AND NOT ou les instructions de sortie peuvent se poursuivre comme si de rien n'était.



Exemple : Une instruction de sortie TR0 de bit de stockage temporaire et une instruction de chargement (LD) sont nécessaires au point de dérivation si les équations logiques sont programmées sans modification. Il est possible de supprimer quelques étapes en réécrivant les équations logiques.

**Ordre d'exécution mnémorique**

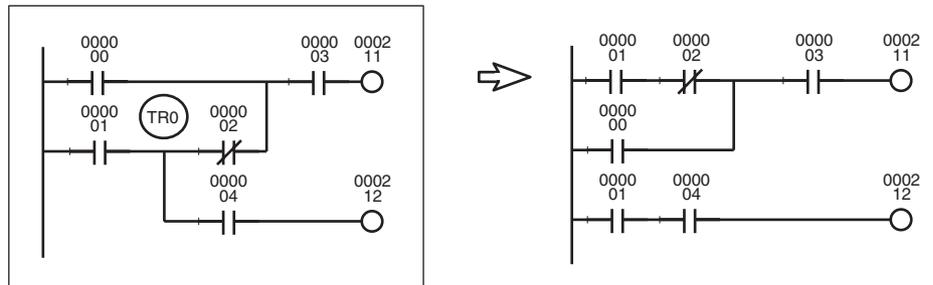
Les API exécutent les schémas contacts dans l'ordre de saisie des mnémoniques ; il se peut donc que les instructions ne fonctionnent pas comme prévu, en fonction de la manière dont sont écrites les équations logiques. Pensez toujours à l'ordre d'exécution mnémorique lorsque vous écrivez les schémas contacts.



Exemple : Dans le schéma ci-dessus, CIO 000210 ne peut être envoyé. En réécrivant l'équation logique comme illustré ci-dessus, il est possible de faire passer CIO 000210 à ON pour un cycle.

**Equations logiques devant être réécrites**

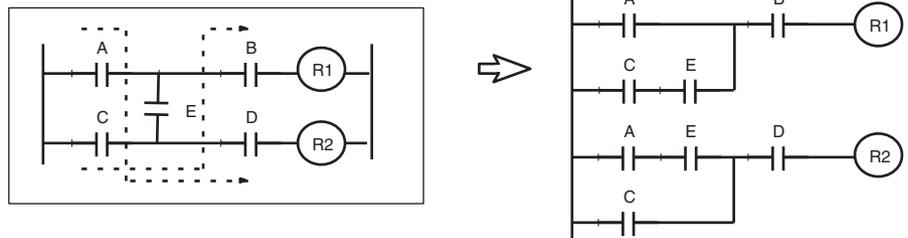
Les API exécutent les instructions dans l'ordre de saisie des mnémoniques de manière à ce que la circulation des signaux (flux d'alimentation) se fasse de gauche à droite dans le schéma contact. Il n'est pas possible de programmer des débits de puissance de droite à gauche.



Exemple : Le programme peut être écrit comme dans le schéma de gauche où TR0 reçoit la dérivation. Dans le schéma de droite, les équations logiques reçoivent la même valeur et le schéma est plus facile à comprendre. Par conséquent, nous conseillons de réécrire les équations logiques de gauche dans les équations logiques de droite.

Réécrivez les équations logiques de gauche ci-dessous. Elles ne peuvent pas être exécutées.

Les flèches indiquent la circulation des signaux (flux d'alimentation) lorsque les équations logiques sont des relais de contrôle.



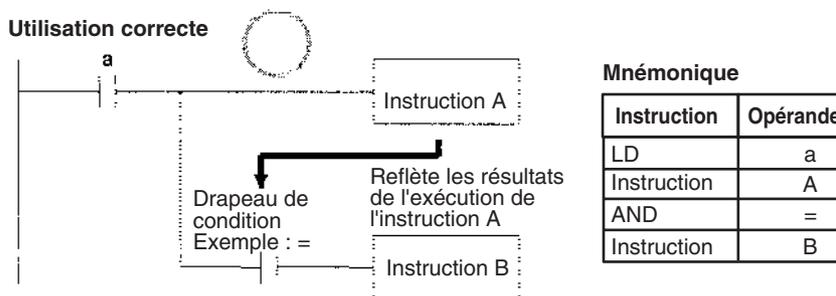
## 2-2 Précautions

### 2-2-1 Drapeaux de condition

#### Utilisation des drapeaux de condition

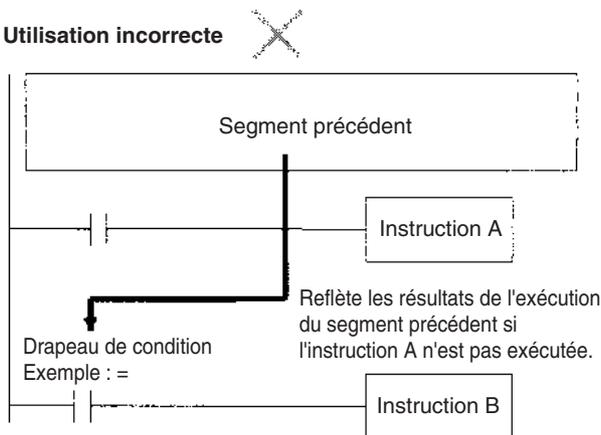
Les drapeaux de conditions sont partagés par toutes les instructions et changeront au cours d'un cycle en fonction des résultats de l'exécution de chaque instruction. Par conséquent, veillez à utiliser les drapeaux de condition sur une sortie dérivée avec les mêmes conditions d'exécution directement après une instruction afin de refléter les résultats de l'exécution de l'instruction. Ne connectez jamais un drapeau de condition directement à la barre de bus car il pourrait refléter les résultats d'exécution d'autres instructions.

**Exemple :** Utilisation des résultats d'exécution de l'instruction A



La même condition d'exécution (a) sert aux instructions A et B pour exécuter l'instruction B en fonction des résultats d'exécution de l'instruction A. Dans ce cas, l'instruction B ne sera exécutée en fonction du drapeau de condition que si l'instruction A est exécutée.

#### Utilisation incorrecte



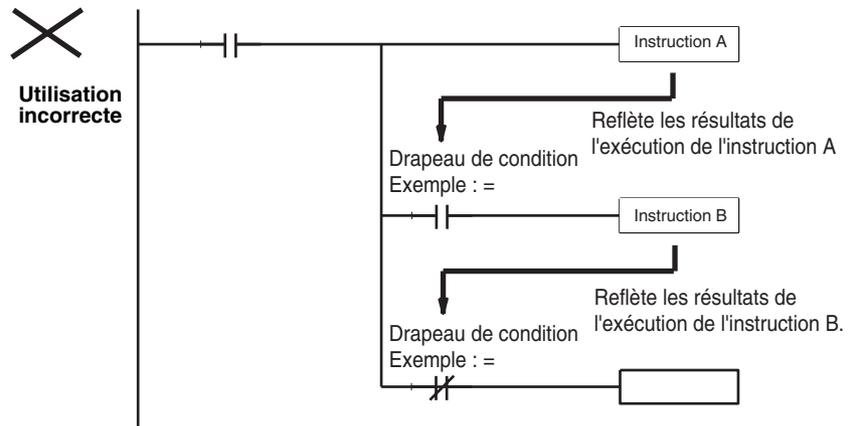
Si le drapeau de condition est connecté directement à la barre de bus gauche, l'instruction B sera exécutée en fonction des résultats d'exécution d'une équation logique précédente si l'instruction A n'est pas exécutée.

#### Remarque

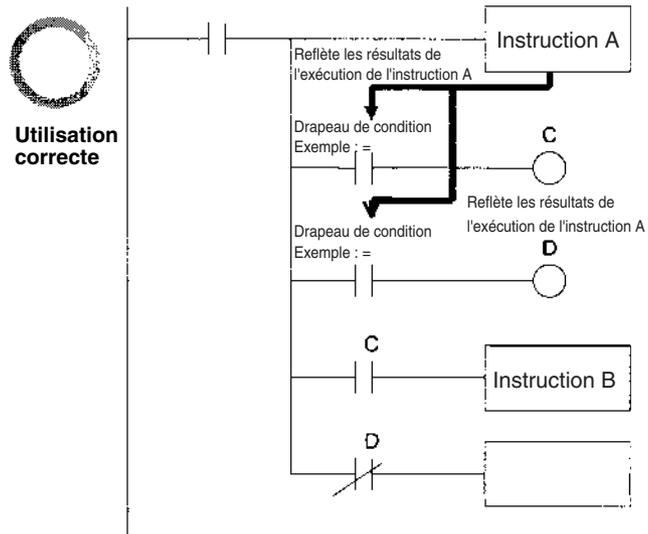
Les drapeaux de condition sont utilisés par toutes les instructions au sein d'un seul programme (tâche) mais ils sont effacés lorsque la tâche change. Par conséquent, les résultats d'exécution de la tâche précédente ne seront pas reflétés dans les tâches ultérieures. Etant donné que les drapeaux de condition sont partagés par toutes les instructions, veillez absolument à ce qu'ils n'interfèrent pas les uns avec les autres dans un même schéma contact. Les schémas ci-après sont des exemples.

**Utilisation des résultats d'exécution dans les entrées NF et NO**

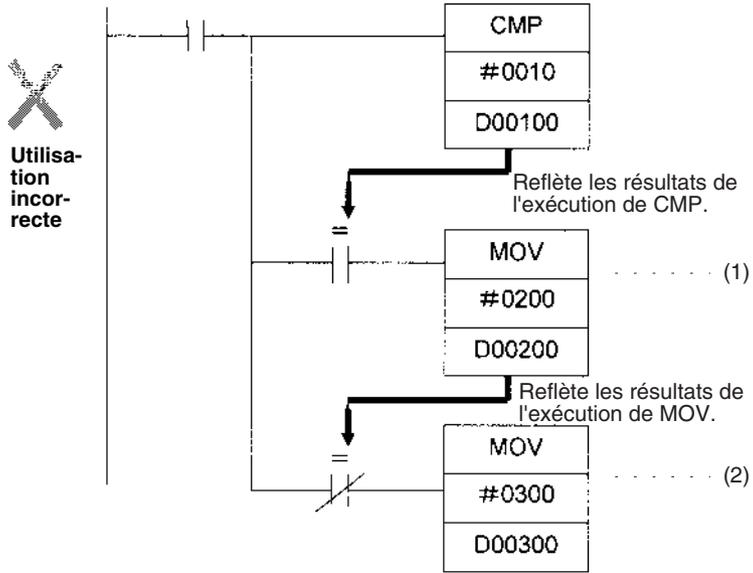
Les drapeaux de condition prennent les résultats d'exécution de l'instruction B, comme illustré dans l'exemple ci-dessous, même si les bits d'entrée NF et NO sont exécutés à partir de la même dérivation de sortie.



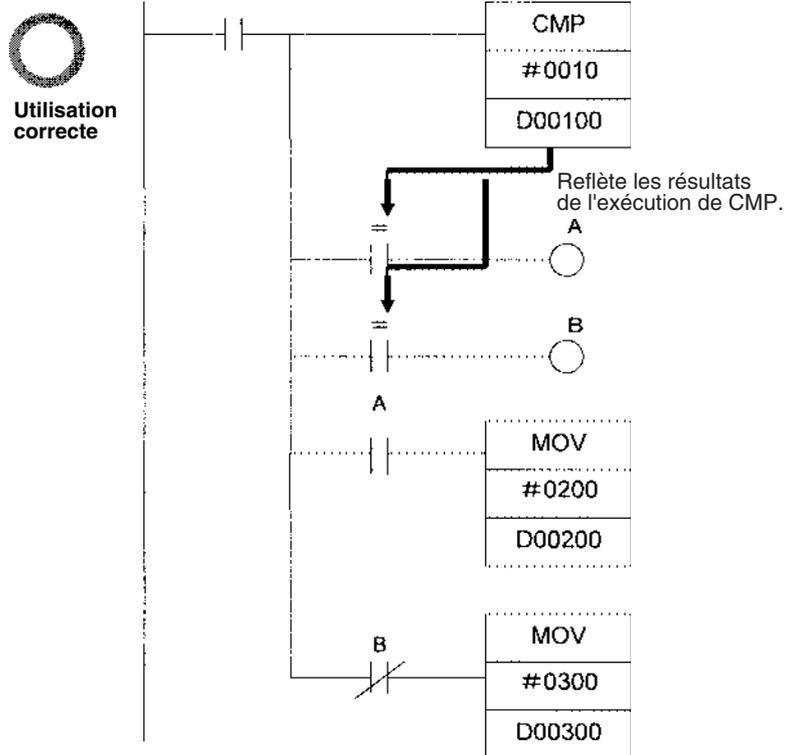
Veillez à ce que chaque résultat ne soit pris qu'une seule fois par une instruction de sortie afin de garantir que les résultats d'exécution de l'instruction B ne seront pas repris.



**Exemple :** L'exemple suivant déplacera #0200 vers D00200 si D00100 contient #0010 et déplacera #0300 vers D00300 si D00100 ne contient pas #0010.



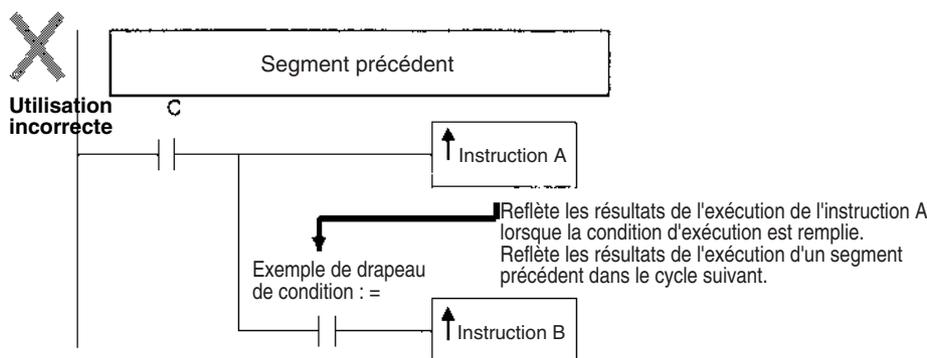
Le drapeau d'égalité passera à ON si D00100 de l'équation logique ci-dessus contient #0010. #0200 sera déplacé vers D00200 pour l'instruction (1), mais ensuite le drapeau d'égalité passera à OFF car les données source #0200 ne sont pas 0000 hex. L'instruction MOV (2) sera ensuite exécutée et #0300 sera déplacé vers D00300. Il faudra donc insérer une équation logique, comme illustré ci-dessous, afin d'éviter que les résultats de la première instruction MOVE ne soient repris.



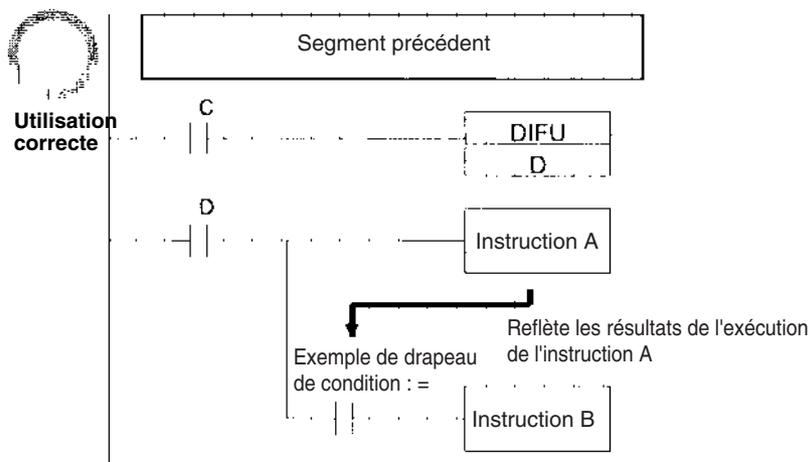
**Utilisation des résultats d'exécution d'instructions différenciées**

Dans le cas des instructions différenciées, les résultats d'exécution sont reflétés dans les drapeaux de condition uniquement si la condition d'exécution est rencontrée. Les résultats d'une équation logique précédente (plutôt que les résultats d'exécution de l'instruction différenciée) seront reflétés dans les drapeaux de condition du cycle suivant. Par conséquent, vous devez connaître l'action réalisée par les drapeaux de conditions dans le cycle suivant s'il faut utiliser les résultats d'exécution des instructions différenciées.

Dans l'exemple suivant, les instructions A et B ne seront exécutées que si la condition d'exécution C se présente, mais le problème suivant se posera si l'instruction B reprend les résultats d'exécution de l'instruction A. Si la condition d'exécution C reste à ON dans le cycle suivant l'exécution de l'instruction A, l'instruction B s'exécutera de manière imprévue (par la condition d'exécution) lorsque le drapeau de condition passera de OFF à ON, suite aux résultats en provenance d'une équation logique précédent.



Dans ce cas, les instructions A et B ne sont pas des instructions différenciées, l'instruction DIFU (de DIFD) est utilisée à leur place, comme illustré ci-dessous, et les instructions A et B sont toutes deux différenciées par le haut (ou par le bas) et exécutées pour un seul cycle.



**Remarque** Les UC CS1-H, CJ1-H, CJ1M ou CS1D prennent en charge les instructions de sauvegarde et de chargement de l'état des drapeaux de condition (CCS(282) et CCL(283)). Elles peuvent être utilisées pour accéder aux drapeaux de condition à d'autres emplacements dans une même tâche ou dans une tâche différente.

**Conditions principales pour faire passer les drapeaux de condition à ON**

**Drapeau d'erreur**

Le drapeau ER passera à ON dans certaines conditions, comme par exemple lorsque les données opérandes d'une instruction ne sont pas correctes. L'instruction ne sera pas exécutée lorsque le drapeau ER passera à ON.

Lorsque le drapeau ER est à ON, l'état des autres drapeaux de condition tels que <, >, OF et UF ne changera pas et l'état des drapeaux = et N changera d'une instruction à l'autre.

Consultez les descriptions de chaque instruction dans le *Manuel de référence des instructions pour les automates programmables série CS/CJ* pour connaître les conditions qui font passer le drapeau ER à ON. Soyez prudent car certaines instructions feront passer le drapeau ER à OFF quelles que soient les conditions.

**Remarque**

Les paramètres de la configuration de l'API spécifiant quand une erreur d'instruction se produit déterminent si le fonctionnement doit s'arrêter lorsque le drapeau ER passe à ON. Selon la valeur par défaut, le fonctionnement se poursuivra lorsque le drapeau ER passera à ON. Si le système spécifie « Arrêter le fonctionnement » lorsque le drapeau ER passe à ON et que le fonctionnement s'arrête (comme s'il s'agissait d'une erreur de programme), l'adresse du programme au point auquel s'est arrêté le fonctionnement sera sauvegardée en A298 et A299. En même temps, A29508 passera à ON.

**Drapeau d'égalité**

Le drapeau d'égalité est un drapeau temporaire pour toutes les instructions, sauf quand les résultats des comparaisons sont égaux (=). Il est défini automatiquement par le système et sera modifié. Une instruction peut faire passer le drapeau d'égalité à OFF (ON) après qu'une instruction précédente l'a fait passer à ON (OFF). Par exemple, le drapeau d'égalité passera à ON lorsque l'instruction MOV, ou toute autre instruction de déplacement, fera passer 0000 hex. comme données source, mais il sera à OFF tout le reste du temps. Même si une instruction fait passer le drapeau d'égalité à ON, l'instruction de déplacement s'exécutera immédiatement et le drapeau d'égalité passera à ON ou OFF en fonction de la valeur des données source de l'instruction (0000 hex. ou non).

**Drapeau de retenue (CY)**

Le drapeau de retenue sert pour les instructions de décalage, les instructions d'addition et de soustraction avec entrée de retenue, les emprunts et les retenues d'instruction d'addition et de soustraction ainsi que pour les instructions de cartes d'E/S spéciales, les instructions PID et les instructions FPD. Veuillez prendre note des précautions suivantes.

**Note**

1. Le drapeau de retenue peut rester sur ON (OFF) étant donné les résultats d'exécution d'une certaine instruction, puis servir dans une autre instruction (une instruction d'addition et de soustraction avec instruction de retenue ou de décalage). N'oubliez pas d'effacer le drapeau de retenue si nécessaire.
2. Le drapeau de retenue peut passer à ON (OFF) à cause des résultats d'exécution d'une certaine instruction et repasser à OFF (ON) dans une autre instruction. Veillez à ce que les résultats corrects soient reflétés dans le drapeau de retenue lorsque vous l'utilisez.

**Drapeaux « inférieur à » et « supérieur à »**

Les drapeaux < et > sont utilisés dans les instructions de comparaison ainsi que dans les instructions LMT, BAND, ZONE, PID et autres.

Le drapeau < ou > peut être mis sur OFF (ON) par une autre instruction même s'il est mis sur ON (OFF) suite aux résultats d'exécution d'une certaine instruction.

**Drapeau négatif**

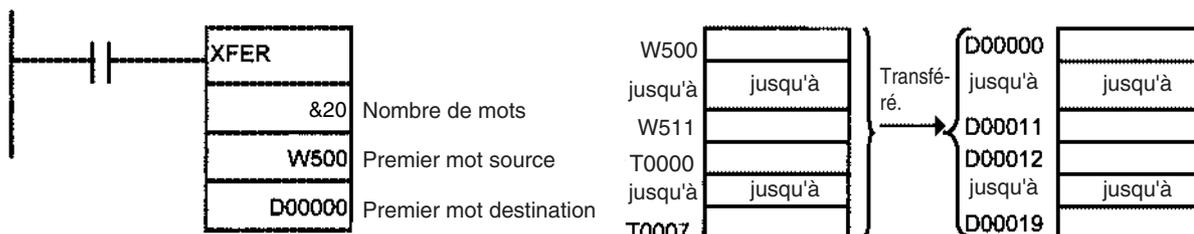
Le drapeau N (négatif) est à ON lorsque le bit le plus à gauche du mot des résultats d'exécution de l'instruction est « 1 » pour certaines instructions et il est à OFF sans condition pour les autres instructions.

**Spécification d'opérandes pour plusieurs mots**

Dans le cas des API série CS/CJ, une instruction sera exécutée telle qu'elle est écrite même si une opérande exigeant plusieurs mots a été spécifiée et que donc tous les mots de cette opérande ne se trouvent pas dans la même zone. Dans ce cas, les mots seront pris dans l'ordre des adresses mémoire de l'API. Le drapeau d'erreur ne passera **pas** à ON.

Prenons comme exemple les résultats de l'exécution d'un transfert par bloc avec XFER(070) si 20 mots sont spécifiés pour le transfert, en commençant à W500. Dans ce cas, la zone de travail, qui se termine à W511, sera dépassée, mais l'instruction sera exécutée sans faire passer le drapeau d'erreur à ON. Dans les adresses mémoire de l'API, les valeurs actuelles des temporisations sont maintenues en mémoire après la zone de travail. Par conséquent, lors de l'instruction suivante, W500 à W511 seront transférés vers D00000 à D00011 et les valeurs actuelles de T0000 à T0007 seront transférées vers D00012 à D00019.

**Remarque** Consultez l'annexe *Plan de la mémoire des adresses mémoire de l'API* pour les adresses mémoire de l'API spécifiques.



### 2-2-2 Sections de programmes spéciales

Les programmes de la série CS/CJ contiennent des sections de programmes spéciales qui contrôlent les conditions des instructions. Vous avez le choix entre les sections de programmes spéciales suivantes.

Section de programme	Instructions	Condition d'instruction	Etat
Sous-programme	Instructions SBS, SBN et RET	Le sous-programme est exécuté.	La section du sous-programme comprise entre les instructions SBN et RET est exécutée.
Section IL - ILC	Instructions IL et ILC	La section est verrouillée	Les bits de sortie passent sur ON et les temporisations sont réinitialisées. Les autres instructions ne seront pas exécutées et l'état précédent sera maintenu.
Section de séquence échelonnée	Instructions STEP S et instructions STEP		
Boucle FOR-NEXT	Instructions FOR et NEXT	Rupture en cours.	En boucles
Section JMP0 – JME0	Instructions JMP0 et JME0		Saut
Section de programmes de bloc	Instructions BPRG et BEND	Le programme de bloc s'exécute.	Le programme de bloc repris dans les mnémoniques entre les instructions BPRG et BEND est exécuté.

#### Combinaisons d'instructions

Le tableau suivant illustre le type d'instruction spéciale qui peut être utilisé dans d'autres sections de programmes.

	Sous-programme	Section IL - ILC	Section de séquence échelonnée	Boucle FOR-NEXT	Section JMP0 – JME0	Section de programmes de bloc
<b>Sous-programme</b>	Impossible	Impossible	Impossible	Impossible	Impossible	Impossible
<b>IL - ILC</b>	OK	Impossible	Impossible	OK	OK	Impossible
<b>Section de séquence échelonnée</b>	Impossible	OK	Impossible	Impossible	OK	Impossible
<b>Boucle FOR-NEXT</b>	OK	OK	Impossible	OK	OK	Impossible
<b>JMP0 – JME0</b>	OK	OK	Impossible	Impossible	Impossible	Impossible
<b>Section de programmes de bloc</b>	OK	OK	OK	Impossible	OK	Impossible

**Remarque** Les instructions qui précisent des zones de programmes ne peuvent être utilisées pour des programmes dans d'autres tâches. Consultez la section 4-2-2 *Limites des instructions relatives aux tâches* pour plus de détails.

**Sous-programmes** Rassemblez tous les sous-programmes juste avant l'instruction END(001) dans tous les programmes mais après la programmation des éléments autres que les sous-programmes. (Un sous-programme ne peut donc pas être placé dans une séquence échelonnée, un programme de bloc, une section FOR – NEXT ou une section JMP0 – JME0.) Si un programme autre qu'un sous-programme est placé après un sous-programme (SBN à RET), ce programme ne sera pas exécuté.



**Instructions non disponibles dans les sous-programmes**

Les instructions suivantes ne peuvent pas être placées dans un sous-programme.

Fonction	Mnémonique	Instruction
Contrôle de processus pas à pas	STEP(008)	Définit la section de séquence échelonnée
	SNXT(009)	Passe à travers les étapes de la séquence échelonnée

**Remarque Sections du programme de bloc**  
 Un sous-programme peut inclure une section de programme de bloc. Cependant, si l'état du programme de bloc est WAIT lorsque l'exécution revient du sous-programme vers le programme principal, la section de programme de bloc gardera l'état WAIT la prochaine fois qu'elle sera appelée.

**Instructions non disponibles dans les sections de séquence échelonnée**

Fonction	Mnémonique	Instruction
Commandes des séquences	FOR(512), NEXT(513) et BREAK(514)	FOR, NEXT et BREAK LOOP
	END(001)	END
	IL(002) et ILC(003)	INTERLOCK et INTERLOCK CLEAR
	JMP(004) et JME(005)	JUMP et JUMP END
	CJP(510) et CJPN(511)	CONDITIONAL JUMP et CONDITIONAL JUMP NOT
	JMP0(515) et JME0(516)	MULTIPLE JUMP et MULTIPLE JUMP END
Sous-programmes	SBN(092) et RET(093)	SUBROUTINE ENTRY et SUBROUTINE RETURN
Programmes de bloc	IF(802) (NOT), ELSE(803) et IEND(804)	Instructions de dérivation
	BPRG(096) et BEND(801)	BLOCK PROGRAM BEGIN/END
	EXIT(806) (NOT)	CONDITIONAL BLOCK EXIT (NOT)
	LOOP(809) et LEND(810) (NOT)	Loop Control
	WAIT(805) (NOT)	ONE CYCLE WAIT (NOT)
	TIMW(813)	TIMER WAIT
	TMHW(815)	HIGH-SPEED TIMER WAIT
	CNTW(814)	COUNTER WAIT
	BPPS(811) et BPRS(812)	BLOCK PROGRAM PAUSE et RESTART

- Note**
1. Une section de schéma contact échelonné peut être utilisée dans une section verrouillée (entre IL et ILC). La section de séquence échelonnée sera complètement réinitialisée lorsque le verrouillage passera à ON.
  2. Une section de schéma contact échelonnée peut être utilisée entre les instructions MULTIPLE JUMP (JMP0) et MULTIPLE JUMP END (JME0).

**Instructions non disponibles dans les sections de programmes de bloc**

Les instructions suivantes ne peuvent pas être placées dans des sections de programmes de bloc.

Classement par fonction	Mnémonique	Instruction
Commandes des séquences	FOR(512), NEXT(513) et BREAK(514)	FOR, NEXT et BREAK LOOP
	END(001)	END
	IL(002) et ILC(003)	INTERLOCK et INTERLOCK CLEAR
	JMP0(515) et JME0(516)	MULTIPLE JUMP et MULTIPLE JUMP END
Entrée de séquence	UP(521)	CONDITION ON
	DOWN(522)	CONDITION OFF
Sortie de séquence	DIFU	DIFFERENTIATE UP
	DIFD	DIFFERENTIATE DOWN
	KEEP	KEEP
	OUT	OUTPUT
	OUT NOT	OUTPUT NOT
Temporisation/compteur	TIM	TIMER
	TIMH	HIGH-SPEED TIMER
	TMHH(540)	ONE-MS TIMER
	TTIM(087)	ACCUMULATIVE TIMER
	TIML(542)	LONG TIMER
	MTIM(543)	MULTI-OUTPUT TIMER
	CNT	COUNTER
	CNTR	REVERSIBLE COUNTER
Sous-programmes	SBN(092) et RET(093)	SUBROUTINE ENTRY et SUBROUTINE RETURN
Décalage de données	SFT	SHIFT
Contrôle de schéma contact pas à pas	STEP(008) et SNXT(009)	STEP DEFINE et STEP START
Contrôles des données	PID	PID CONTROL
Programme de bloc	BPRG(096)	BLOCK PROGRAM BEGIN
Diagnostic des problèmes	FPD(269)	FAILURE POINT DETECTION

- Note**
1. Les programmes de bloc peuvent être utilisés dans une section de schéma contact échelonnée.
  2. Un schéma contact pas à pas peut être utilisé dans une section verrouillée (entre IL et ILC). La section du programme de bloc ne sera pas exécutée si le verrouillage est à ON.
  3. Une section de programme de bloc peut être utilisée entre les instructions MULTIPLE JUMP (JMP0) et MULTIPLE JUMP END (JME0).
  4. Une instruction JUMP (JMP) et une instructions CONDITIONAL JUMP (CJP/ CJPN) peuvent être utilisées dans une section de programmes de bloc. Les instructions JUMP (JMP) et JUMP END (JME) ainsi que les instructions CONDITIONAL JUMP (CJP/CJPN) et JUMP END (JME) ne peuvent pas être utilisées dans la section de programmes de bloc sauf si elles sont groupées par paire. Le programme ne s'exécutera pas correctement sauf si ces instructions sont groupées par paire.

## 2-3 Vérification des programmes

Les programmes de la série CS/CJ peuvent être vérifiés aux niveaux suivants :

- vérification à l'entrée lors du fonctionnement à l'entrée de la console de programmation ;
- vérification du programme par CX-Programmer ;
- vérification des instructions pendant l'exécution ;
- vérification des erreurs fatales (erreurs de programme) pendant l'exécution.

### 2-3-1 Erreurs lors de l'entrée de périphériques de programmation

#### Console de programmation

La console de programmation affichera les erreurs aux points suivants lors de l'entrée.

Erreur affichée	Cause
CHK MEM	La broche 1 de l'interrupteur DIP de l'UC est à ON (protection en écriture).
IO No. ERR	Une entrée d'E/S illégale a été tentée.

#### CX-Programmer

Le programme sera automatiquement vérifié par CX-Programmer aux moments suivants.

Moment	Contenu vérifié
A l'insertion de schémas contacts	Saisies des instructions, saisies des opérandes et modèles de programmation
Lors du chargement de fichiers	Toutes les opérandes de toutes les instructions et tous les modèles de programmation
Lors du téléchargement de fichiers	Modèles pris en charge par la série CS/CJ et toutes les opérandes de toutes les instructions
Pendant l'édition en ligne	Capacité, etc.

Les résultats des vérifications sont envoyés dans l'onglet texte de la fenêtre Output. De même, la barre de bus gauche des sections de programmes illégales s'affichera en rouge dans la présentation du schéma.

### 2-3-2 Vérification du programme par CX-Programmer

Les erreurs détectées lors de la vérification du programme par CX-Programmer sont reprises dans le tableau suivant.

CX-Programmer ne vérifie pas les erreurs de plage des opérandes dont l'adresse est indirecte dans les instructions. Les erreurs d'adressage indirect seront détectées lors de la vérification de l'exécution du programme et le drapeau ER passera à ON, comme décrit dans le paragraphe suivant. Consultez le *Manuel de référence des instructions pour les automates programmables série CS/CJ (W340)* pour plus de détails.

Lors de la vérification du programme par CX-Programmer, l'opérateur peut préciser le niveau A, B ou C de vérification (en fonction de la gravité de l'erreur), ou même un niveau de vérification personnalisé.

Zone	Vérification
Données illégales : schéma contact	Emplacements des instructions
	Lignes d'E/S
	Connexions
	Achèvement des instructions et du fonctionnement
Prise en charge d'instructions par l'API	Instructions et opérandes prises en charge par l'API
	Variations d'instructions (NOT, !, @ et %)
	Intégrité du code de l'objet

Zone	Vérification
Plages des opérandes	Plages des zones des opérandes
	Types des données d'opérandes
	Vérification de l'accès aux mots en lecture seule
	Vérification de la plage des opérandes, y compris : <ul style="list-style-type: none"> <li>• constantes (#, &amp;, +, -)</li> <li>• codes de commandes</li> <li>• vérification des limites des zones pour les opérandes à plusieurs mots</li> <li>• vérification des relations de taille pour les opérandes à plusieurs mots</li> <li>• chevauchement des plages d'opérandes</li> <li>• affectations de plusieurs mots</li> <li>• opérandes double longueur</li> <li>• vérification des limites des zones pour les décalages</li> </ul>
Capacité du programme pour les API	Nombre de pas
	Capacité générale
	Nombre de tâches
Syntaxe	Vérification des appels pour les instructions par paire <ul style="list-style-type: none"> <li>• IL-ILC</li> <li>• JMP-JME, CJP/CJPN-JME</li> <li>• SBS-SBN-RET, MCRO-SBN-RET</li> <li>• STEP-SNXT</li> <li>• BPRG-BEND</li> <li>• IF-IEND</li> <li>• LOOP-LEND</li> </ul>
	Emplacement de programmation limité pour BPRG-BEND
	Emplacement de programmation limité pour SBN-RET
	Emplacement de programmation limité pour STEP-SNXT
	Emplacement de programmation limité pour FOR-NEXT
	Emplacement de programmation limité pour les tâches d'interruption
	Emplacement de programmation obligatoire pour BPRG-BEND
	Emplacement de programmation obligatoire pour FOR-NEXT
	Imbrication illégale
	Instruction END(001)
	Cohérence des numéros
Structure du schéma contact	Dépassements de piles
Duplication des résultats	Duplique la vérification des résultats <ul style="list-style-type: none"> <li>• Par bit</li> <li>• Par mot</li> <li>• Instructions de temporisation/compteur</li> <li>• Longs mots (de 2 ou 4 mots)</li> <li>• Mots affectés plusieurs fois</li> <li>• Plages de début/fin</li> <li>• Numéros de l'instruction FAL</li> <li>• Instructions comprenant plusieurs opérandes de sortie</li> </ul>
Tâches	Vérifie les tâches devant commencer au début du fonctionnement
	Affectation du programme des tâches

**Remarque**

La duplication des résultats n'est pas vérifiée entre les tâches ; elle ne l'est qu'au sein des tâches individuelles.

**Opérandes à plusieurs mots**

Les limites des zones de mémoire des opérandes à plusieurs mots sont vérifiées pour la vérification du programme, comme illustré dans le tableau suivant.

CX-Programmer	Consoles de programmation
CX-Programmer offre les fonctionnalités suivantes pour les opérandes à plusieurs mots qui dépassent une limite de zone de mémoire. <ul style="list-style-type: none"> <li>• Impossible de transférer le programme vers l'UC.</li> <li>• Impossible de lire le programme à partir de l'UC.</li> <li>• Des erreurs de compilation sont générées pour la vérification du programme.</li> <li>• Lors de la programmation hors ligne, des messages d'avertissement apparaîtront à l'écran.</li> <li>• Des messages d'avertissement apparaîtront à l'écran lors de l'édition en ligne en mode PROGRAM ou MONITOR.</li> </ul>	Vérifiées lorsque les programmes sont installés, c-à-d que les opérandes qui dépassent une limite de zone de mémoire ne peuvent pas être écrites.

**2-3-3 Vérification de l'exécution du programme**

Les vérifications concernant l'emplacement des instructions et des opérandes sont effectuées sur les instructions lors de la saisie à partir des périphériques de programmation (y compris les consoles de programmation) ainsi que pendant les vérifications de programmes à partir des périphériques de programmation (excepté les consoles de programmation). Cependant, il ne s'agit pas de vérifications finales.

Les vérifications suivantes sont effectuées lors de l'exécution de l'instruction.

Type d'erreur	Drapeau passant à ON en cas d'erreur	Arrêt/poursuite du fonctionnement
1. Erreur de traitement de l'instruction	Drapeau ER Le drapeau d'erreur de traitement d'instruction (A29508) passera aussi à ON s'il est spécifié qu'il faut arrêter le fonctionnement lorsqu'une erreur se produit.	Dans la configuration de l'API, vous pouvez définir un paramètre de façon à préciser si le fonctionnement doit s'arrêter ou continuer en cas d'erreur de traitement d'instruction. Par défaut, le fonctionnement continuera. Une erreur de programme sera générée et le fonctionnement s'arrêtera uniquement si Arrêter le fonctionnement a été spécifié.
2. Erreur d'accès	Drapeau AER Le drapeau d'erreur d'accès (A29510) passera aussi à ON s'il est spécifié qu'il faut arrêter le fonctionnement lorsqu'une erreur se produit.	Dans la configuration de l'API, vous pouvez définir un paramètre de façon à préciser si le fonctionnement doit s'arrêter ou continuer en cas d'erreur d'accès. Par défaut, le fonctionnement continuera. Une erreur de programme sera générée et le fonctionnement s'arrêtera uniquement si Arrêter le fonctionnement a été spécifié.
3. Erreur d'instruction illégale	Drapeau d'erreur d'instruction illégale (A29514)	(Erreur de programme) fatale
4. Erreur de dépassement de la MU (mémoire utilisateur)	Drapeau d'erreur de dépassement MU	(Erreur de programme) fatale

**Erreurs de traitement d'instruction**

Une erreur de traitement d'instruction se produira si des données incorrectes ont été fournies lors de l'exécution d'une instruction ou si quelqu'un a essayé d'exécuter une instruction en dehors d'une tâche. Dans ce cas-ci, les données obligatoires au début du traitement de l'instruction ont été vérifiées. Le résultat est le suivants : l'instruction n'avait pas été exécutée, le drapeau ER (d'erreur) est passé à ON et les drapeaux EQ et N sont conservés ou mis sur OFF en fonction de l'instruction.

Le drapeau ER (d'erreur) passera à OFF si l'instruction (excepté les instructions d'entrée) se termine normalement. Les conditions faisant passer le drapeau ER à ON varient en fonction de chaque instruction. Consultez les descriptions de chaque instruction dans le *Manuel de référence des instructions pour les automates programmables série CS/CJ (W340)* pour plus de détails.

Si dans la configuration de l'API, Erreurs d'instruction est programmé sur Arrêter le fonctionnement, le fonctionnement s'arrêtera (erreur fatale) et le drapeau d'erreur de traitement d'instruction (A29508) passera à ON si une erreur de traitement d'instruction se produit et que le drapeau ER passe à ON.

#### Erreurs d'accès illégal

Les erreurs d'accès illégal indiquent qu'il y a eu un accès à une mauvaise zone d'une des manières suivantes lors de l'accès à l'adresse spécifiant l'opérande de l'instruction.

- a. Lecture ou écriture d'une zone de paramètres
- b. Lecture dans une zone de mémoire non montée (voir remarque)
- c. Ecriture dans une zone EM spécifiée comme mémoire de fichiers EM
- d. Ecriture dans une zone de lecture seule
- e. La valeur spécifiée dans une adresse DM/EM indirecte en mode BCD n'était pas une donnée BCD (par exemple, \*D00001 contient #A000).

Le traitement d'instruction se poursuivra et le drapeau d'erreur (drapeau ER) ne passera pas à ON si une erreur d'accès se produit, mais c'est le drapeau d'erreur d'accès (AER) qui passera à ON.

#### Remarque

Une erreur d'accès se produira pour les raisons suivantes :

- lorsqu'une adresse EM spécifiée dépasse 32767 (exemple : E32768) pour la banque actuelle ;
- la banque finale (exemple : C) est spécifiée pour une adresse EM indirecte en mode BIN et le mot spécifié contient 8000 à FFFF hex. (exemple : @EC\_00001 contient #8000) ;
- la banque actuelle (exemple : C) est spécifiée pour une adresse EM indirecte en mode BIN et le mot spécifié contient 8000 à FFFF hex. (exemple : @EC\_00001 contient #8000) ;
- un registre d'index (IR) contenant l'adresse de mémoire interne d'un bit est utilisé en tant qu'adresse de mot ou un IR contenant l'adresse de mémoire interne d'un mot est utilisé en tant qu'adresse de bit.

Si dans la configuration de l'API, Erreurs d'instruction est programmé sur Arrêter le fonctionnement, le fonctionnement s'arrêtera (erreur fatale) et le « drapeau d'erreur d'accès illégal » (A29510) passera à ON si une erreur d'accès illégal se produit et que le drapeau AER passe à ON.

#### Remarque

Le drapeau d'erreur d'accès (AER) ne sera pas effacé après l'exécution d'une tâche. Si Poursuivre le fonctionnement a été choisi pour Erreurs d'instruction, ce drapeau peut être surveillé jusqu'à l'instruction END(001) afin de vérifier si une erreur d'accès illégal s'est produite dans le programme de tâches. (L'état du drapeau AER final, après l'exécution de tout le programme utilisateur, sera surveillé si ce drapeau est surveillé par une console de programmation.)

### Autres erreurs

#### Erreurs d'instruction illégale

Les erreurs d'instruction illégale indiquent qu'a eu lieu une tentative d'exécution des données d'instruction autres que celles définies dans le système. Cette erreur ne doit normalement pas se produire tant que le programme est créé sur un périphérique de programmation série CS/CJ (y compris les consoles de programmation).

Dans le cas fort peu probable où cette erreur se produirait, elle serait traitée comme une erreur de programme : le fonctionnement s'arrêtera (erreur fatale) et le drapeau d'instruction illégale (A29514) passera à ON.

#### Erreur de dépassement de la MU (mémoire utilisateur)

Les erreurs de dépassement de MU indiquent qu'a eu lieu une tentative d'exécution de données d'instruction stockées au-delà de la dernière adresse dans la mémoire utilisateur (MU) définie comme zone de stockage du programme. Cette erreur ne doit normalement pas se produire tant que le programme est créé sur un périphérique de programmation série CS/CJ (y compris les consoles de programmation).

Dans le cas fort peu probable où cette erreur se produirait, elle serait traitée comme une erreur de programme : le fonctionnement s'arrêtera (erreur fatale) et le drapeau de dépassement de la MU (A29515) passera à ON.

### 2-3-4 Vérification des erreurs fatales :

Les erreurs suivantes sont des erreurs de programme fatales ; si l'une d'entre elles se produit, l'UC s'arrêtera. Lorsque le fonctionnement s'arrête suite à une erreur de programme, le numéro de tâche auquel s'est arrêté le fonctionnement est sauvegardé en A294 et l'adresse du programme est sauvegardée en A298 et A299. Cette information permet de déterminer la cause de l'erreur de programme.

Adresse	Description	Données stockées
A294	Le type de tâche ainsi que le numéro de tâche auquel le fonctionnement s'est arrêté sera stocké ici si le fonctionnement s'arrête suite à une erreur de programme.  FFFF hex. sera stocké si aucune tâche cyclique active ne fait partie d'un cycle, c-à-d, aucune tâche cyclique ne doit être exécutée.	Tâche cyclique : 0000 à 001F hex. (tâches cycliques 0 à 31) Tâche d'interruption : 8000 à 80FF hex. (tâches d'interruption 0 à 255)
A298/A299	L'adresse de programme au point auquel le fonctionnement s'est arrêté sera stockée ici en binaire si le fonctionnement s'arrête à cause d'une erreur de programme.  Si l'instruction END(001) manque (A29511 est à ON), c'est l'adresse où devait se trouver END(001) qui sera stockée.  En cas d'erreur d'exécution de tâche (A29512 est à ON), FFFFFFFF hex. sera stocké en A298/A299.	A298 : Partie la plus à droite de l'adresse de programme A299 : Partie la plus à gauche de l'adresse de programme

**Remarque** Si le drapeau d'erreur ou le drapeau d'erreur d'accès passe à ON, l'erreur sera traitée comme une erreur de programme et pourra servir à arrêter le fonctionnement de l'UC. Définissez le fonctionnement des erreurs de programme dans la configuration de l'API.

Erreur de programme	Description	Drapeaux associés :
Pas d'instruction END	Le programme ne contient pas d'instruction END.	Le drapeau No END (A29511) passe à ON.
Erreur lors de l'exécution de la tâche	Aucune tâche n'est prête dans le cycle. Aucun programme n'est affecté à une tâche. Le numéro de tâche d'interruption correspondant n'est pas présent même si la condition d'exécution de la tâche d'interruption a été rencontrée.	Le drapeau d'erreur de tâches (29512) passe à ON.
Erreur de traitement d'instruction (Drapeau ER sur ON) et dans la configuration de l'API, Arrêter le fonctionnement a été défini en cas d'erreurs d'instruction.	Lors de la tentative d'exécution d'une instruction, de mauvaises valeurs de données ont été fournies dans l'opérande.	Le drapeau ER et le drapeau d'erreur de traitement d'instruction (A29508) passent à ON si, dans la configuration de l'API, Arrêter le fonctionnement a été défini en cas d'erreurs d'instruction.
Erreur d'accès illégal (Drapeau AER sur ON) et dans la configuration de l'API, Arrêter le fonctionnement a été défini en cas d'erreurs d'instruction.	Lecture ou écriture d'une zone de paramètres Lecture dans une zone de mémoire non montée (voir remarque) Ecriture dans une zone EM spécifiée comme mémoire de fichiers EM Ecriture dans une zone de lecture seule La valeur spécifiée dans une adresse DM/EM indirecte en mode BCD n'était pas une donnée BCD.	Le drapeau AER et le drapeau d'erreur d'accès illégal (A29510) passent à ON si, dans la configuration de l'API, Arrêter le fonctionnement a été défini en cas d'erreurs d'instruction.
Erreur BCD de DM/EM indirecte et dans la configuration de l'API, Arrêter le fonctionnement a été défini en cas d'erreurs d'instruction.	La valeur spécifiée dans une adresse DM/EM indirecte en mode BCD n'était pas une donnée BCD.	Le drapeau AER et le drapeau d'erreur BCD de DM/EM indirecte (A29509) passent à ON si, dans la configuration de l'API, Arrêter le fonctionnement a été défini en cas d'erreurs d'instruction.
Erreur de dépassement d'adresse de différenciation	Pendant l'édition en ligne, plus de 131 071 instructions différenciées ont été insérées ou supprimées.	Le drapeau d'erreur de dépassement de différenciation (A29513) passe à ON.
Erreur de dépassement de MU (mémoire utilisateur)	Une tentative a eu lieu pour exécuter des données d'instruction stockées au-delà de la dernière adresse dans la mémoire utilisateur (MU) définie comme zone de stockage du programme.	Le drapeau de dépassement de MU (mémoire utilisateur) (A29516) passe à ON.
Erreur d'instruction illégale	Une tentative d'exécution d'une instruction qui ne peut être exécutée a eu lieu.	Le drapeau d'instruction illégale (A29514) passe à ON.



# CHAPITRE 3

## Fonction des instructions

Cette section décrit les instructions que vous pouvez utiliser pour écrire des programmes utilisateur.

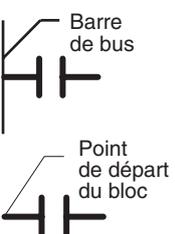
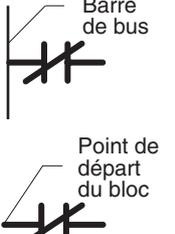
3-1	Instructions d'entrée des séquence . . . . .	72
3-2	Instructions de sortie des séquences . . . . .	74
3-3	Instructions de commande des séquences . . . . .	77
3-4	Instructions de temporisation et de compteur . . . . .	81
3-5	Instructions de comparaison . . . . .	85
3-6	Instructions de déplacement de données . . . . .	89
3-7	Instructions de décalage de données . . . . .	92
3-8	Instructions d'incrément/décément . . . . .	96
3-9	Instructions mathématiques à symboles . . . . .	97
3-10	Instructions de conversion . . . . .	102
3-11	Instructions logiques . . . . .	108
3-12	Instructions mathématiques spéciales . . . . .	110
3-13	Instructions mathématiques à virgule flottante . . . . .	111
3-14	Instructions à virgule flottante double précision . . . . .	115
3-15	Instructions de traitement de données de tableaux . . . . .	119
3-16	Instructions de contrôle de données . . . . .	123
3-17	Instructions de sous-programme . . . . .	127
3-18	Instructions de traitement d'interruption . . . . .	128
3-19	Instructions de compteur à grande vitesse et de sortie d'impulsion (CJ1M-CPU21/22/23 uniquement) . . . . .	130
3-20	Instructions de pas . . . . .	132
3-21	Instructions des cartes d'E/S standard . . . . .	132
3-22	Instructions de communications série . . . . .	135
3-23	Instructions réseaux . . . . .	136
3-24	Instructions de mémoire de fichiers . . . . .	139
3-25	Instructions d'affichage . . . . .	140
3-26	Instructions de temporisation . . . . .	140
3-27	Instructions de débogage . . . . .	141
3-28	Instructions de diagnostic d'erreur . . . . .	142
3-29	Autres instructions . . . . .	143
3-30	Instructions de programmation de bloc . . . . .	144
3-31	Instructions de traitement des chaînes de texte . . . . .	150
3-32	Instructions de contrôle des tâches . . . . .	153
3-33	Instructions de conversion des modèles (UC ver. 3.0 ou supérieure uniquement) . . . . .	154
3-34	Instructions spéciales des blocs de fonction . . . . .	155

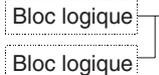
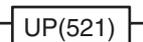
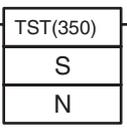
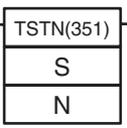
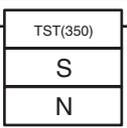
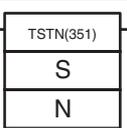
### 3-1 Instructions d'entrée des séquence

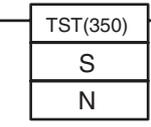
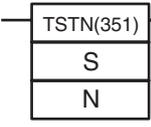
\*1 : non prise en charge par les UC CS1D pour les systèmes d'UC en duplex.

\*2 : prise en charge par les UC CS1-H, CJ1-H, CJ1M et CS1D seulement.

\*3 : prise en charge par les UC CS1-H, CJ1-H et CJ1M uniquement.

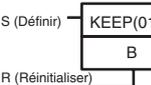
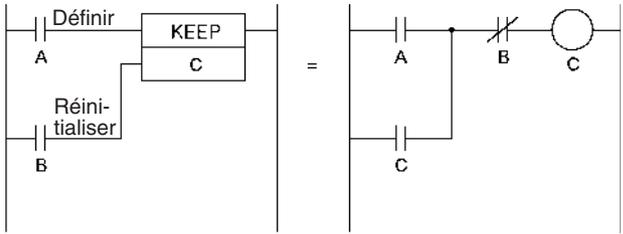
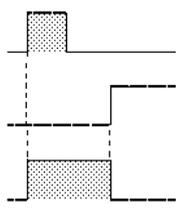
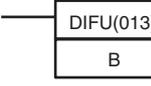
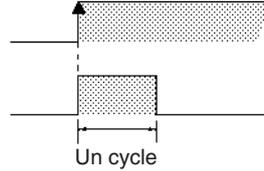
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>LOAD</b>  LD @LD %LD !LD*1 !@LD*1 !%LD*1		Indique un départ logique et crée une condition d'exécution ON/OFF fondée sur l'état ON/OFF du bit d'opérande spécifié.	Démarrage de la logique Non obligatoire
<b>LOAD NOT</b>  LD NOT @LD NOT*2 %LD NOT*2 !LD NOT*1 !@LD NOT*3 !%LD NOT*3		Indique un départ logique et crée une condition d'exécution ON/OFF fondée sur l'état ON/OFF inverse du bit d'opérande spécifié.	Démarrage de la logique Non obligatoire
<b>AND</b>  AND @AND %AND !AND*1 !@AND*1 !%AND*1		Effectue un AND logique de l'état du bit d'opérande spécifié et de la condition d'exécution courante.	En continu sur l'équation logique Obligatoire
<b>AND NOT</b>  AND NOT @AND NOT*2 %AND NOT*2 !AND NOT*1 !@AND NOT*3 !%AND NOT*3		Inverse l'état du bit d'opérande spécifié et contient un AND logique avec la condition d'exécution courante.	En continu sur l'équation logique Obligatoire
<b>OR</b>  OR @OR %OR !OR*1 !@OR*1 !%OR*1		Effectue un OR logique de l'état ON/OFF du bit d'opérande spécifié et de la condition d'exécution courante.	En continu sur l'équation logique Obligatoire
<b>OR NOT</b>  OR NOT @OR NOT*2 %OR NOT*2 !OR NOT*1 !@OR NOT*3 !%OR NOT*3		Inverse l'état du bit d'opérande spécifié et contient un OR logique avec la condition d'exécution courante.	En continu sur l'équation logique Obligatoire

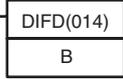
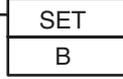
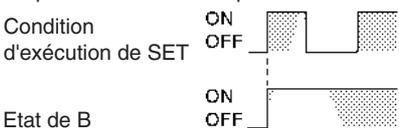
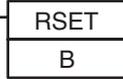
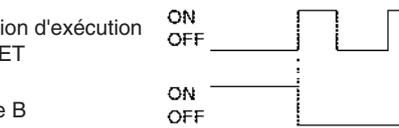
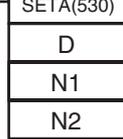
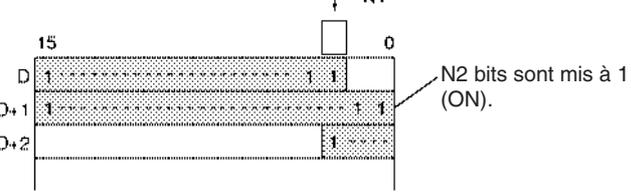
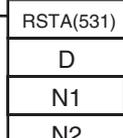
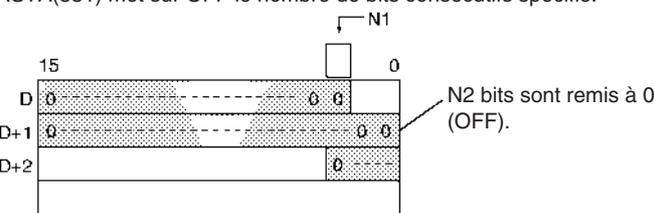
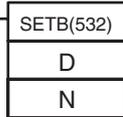
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>AND LOAD</b> AND LD		Effectue un AND logique entre les blocs logiques.  LD } jusqu'à } Bloc logique A  LD } jusqu'à } Bloc logique B  AND LD ..... Connexion série entre le bloc logique A et le bloc logique B.	En continu sur l'équation logique Obligatoire
<b>OR LOAD</b> OR LD		Effectue un OR logique entre les blocs logiques.  LD } jusqu'à } Bloc logique A  LD } jusqu'à } Bloc logique B  OR LD ..... Connexion parallèle entre le bloc logique A et le bloc logique B.	En continu sur l'équation logique Obligatoire
<b>NOT</b> NOT 520	---	Inverse la condition d'exécution.	En continu sur l'équation logique Obligatoire
<b>CONDITION ON</b> UP 521		UP(521) met à ON la condition d'exécution pendant un cycle lorsque la condition d'exécution passe de OFF à ON.	En continu sur l'équation logique Obligatoire
<b>CONDITION OFF</b> DOWN 522		DOWN(522) met à ON la condition d'exécution pendant un cycle lorsque la condition d'exécution passe de ON à OFF.	En continu sur l'équation logique Obligatoire
<b>BIT TEST</b> LD TST 350	 S : Mot source N : Numéro de bit	LD TST(350), AND TST(350) et OR TST(350) sont utilisées dans le programme comme LD, AND et OR ; la condition d'exécution est ON lorsque le bit spécifié dans le mot spécifié est ON et OFF lorsque le bit est OFF.	En continu sur l'équation logique Non obligatoire
<b>BIT TEST</b> LD TSTN 351	 S : Mot source N : Numéro de bit	LD TSTN(351), AND TSTN(351) et OR TSTN(351) sont utilisées dans le programme comme LD NOT, AND NOT et OR NOT ; la condition d'exécution est OFF lorsque le bit spécifié dans le mot spécifié est ON et ON lorsque le bit est à OFF.	En continu sur l'équation logique Non obligatoire
<b>BIT TEST</b> AND TST 350	 S : Mot source N : Numéro de bit	LD TST(350), AND TST(350) et OR TST(350) sont utilisées dans le programme comme LD, AND et OR ; la condition d'exécution est ON lorsque le bit spécifié dans le mot spécifié est ON et OFF lorsque le bit est OFF.	En continu sur l'équation logique Obligatoire
<b>BIT TEST</b> AND TSTN 351	 S : Mot source N : Numéro de bit	LD TSTN(351), AND TSTN(351) et OR TSTN(351) sont utilisées dans le programme comme LD NOT, AND NOT et OR NOT ; la condition d'exécution est OFF lorsque le bit spécifié dans le mot spécifié est ON et ON lorsque le bit est OFF.	En continu sur l'équation logique Obligatoire

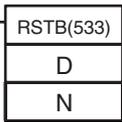
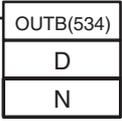
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>BIT TEST</b> OR TST 350	 <p>S : Mot source N : Numéro de bit</p>	LD TST(350), AND TST(350) et OR TST(350) sont utilisées dans le programme comme LD, AND et OR ; la condition d'exécution est ON lorsque le bit spécifié dans le mot spécifié est ON et OFF lorsque le bit est OFF.	En continu sur l'équation logique Obligatoire
<b>BIT TEST</b> OR TSTN 351	 <p>S : Mot source N : Numéro de bit</p>	LD TSTN(351), AND TSTN(351) et OR TSTN(351) sont utilisées dans le programme comme LD NOT, AND NOT et OR NOT ; la condition d'exécution est OFF lorsque le bit spécifié dans le mot spécifié est ON et ON lorsque le bit est à OFF.	En continu sur l'équation logique Obligatoire

### 3-2 Instructions de sortie des séquences

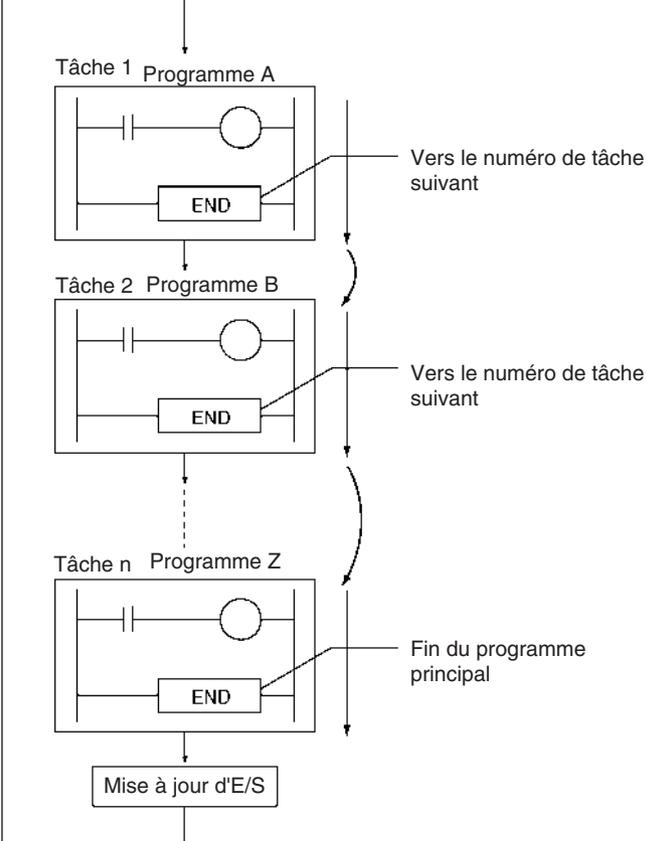
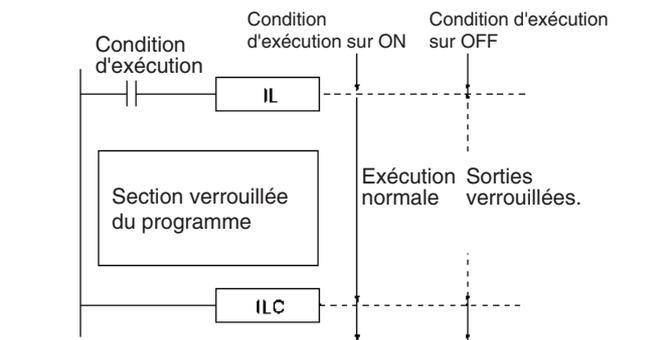
\*1 : non prise en charge par les UC CS1D pour les systèmes d'UC en duplex.

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>OUTPUT</b> OUT !OUT*1		Sort le résultat (condition d'exécution) du traitement logique vers le bit spécifié.	Sortie obligatoire
<b>OUTPUT NOT</b> OUT NOT !OUT NOT*1		Inverse le résultat (condition d'exécution) du traitement logique et le sort vers le bit spécifié.	Sortie obligatoire
<b>KEEP</b> KEEP !KEEP*1 011	 <p>S (Définir) R (Réinitialiser) B : Bit</p>	<p>Fonctionne comme un relais verrouillé.</p>  <p>Condition d'exécution S</p>  <p>Condition d'exécution R</p> <p>Etat de B</p>	Sortie Obligatoire
<b>DIFFERENTIATE UP</b> DIFU !DIFU*1 013	 <p>B : Bit</p>	<p>DIFU(013) met le bit désigné sur ON pour un cycle lorsque la condition d'exécution passe de OFF sur ON (front montant).</p> <p>Condition d'exécution</p>  <p>Etat de B</p> <p>Un cycle</p>	Sortie Obligatoire

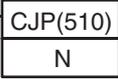
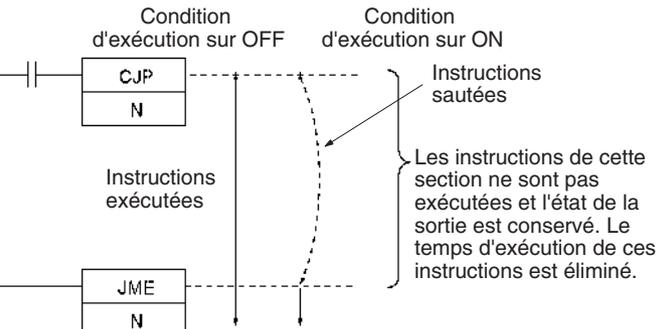
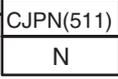
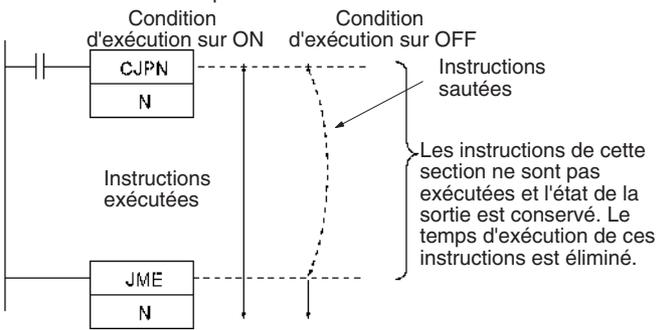
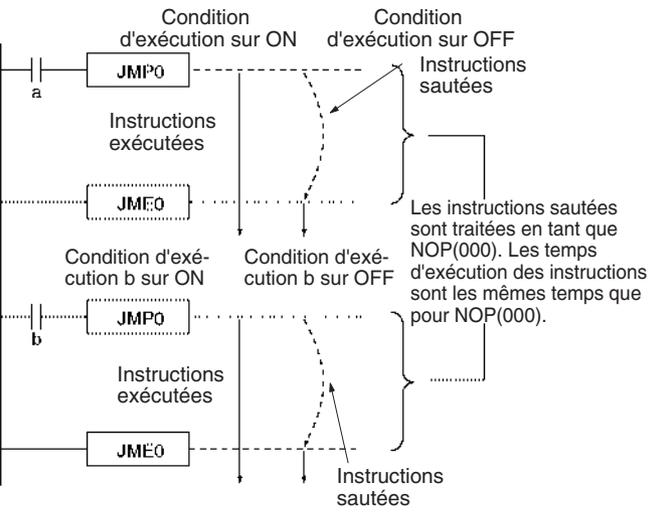
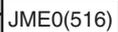
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DIFFERENTIATE DOWN</b>  DIFD !DIFD <sup>*1</sup>  014	 B : Bit	DIFU(014) met le bit désigné sur ON pour un cycle lorsque la condition d'exécution passe de ON sur OFF (front descendant).  	Sortie Obligatoire
<b>SET</b>  SET @SET %SET !SET <sup>*1</sup> !@SET <sup>*1</sup> !%SET <sup>*1</sup>	 B : Bit	SET met le bit d'opérande sur ON lorsque la condition d'exécution est sur ON.  	Sortie Obligatoire
<b>RRESET</b>  RSET @RSET %RSET !RSET <sup>*1</sup> !@RSET <sup>*1</sup> !%RSET <sup>*1</sup>	 B : Bit	RSET met le bit d'opérande sur OFF lorsque la condition d'exécution est sur ON.  	Sortie Obligatoire
<b>MULTIPLE BIT SET</b>  SETA @SETA 530	 D : Mot de début N1 : Bit de début N2 : Nombre de bits	SETA(530) met sur ON le nombre de bits consécutifs spécifié.  	Sortie Obligatoire
<b>MULTIPLE BIT RESET</b>  RSTA @RSTA 531	 D : Mot de début N1 : Bit de début N2 : Nombre de bits	RSTA(531) met sur OFF le nombre de bits consécutifs spécifié.  	Sortie Obligatoire
<b>SINGLE BIT SET (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b>  SETB @SETB !SETB <sup>*1</sup> !@SETB <sup>*1</sup>	 D : Adresse mot N : Numéro de bit	SETB(532) met à ON le bit spécifié dans le mot spécifié lorsque la condition d'exécution est ON. A la différence de l'instruction SET, SETB(532) peut être utilisée pour initialiser un bit dans un mot DM ou EM.	Sortie Obligatoire

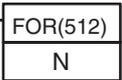
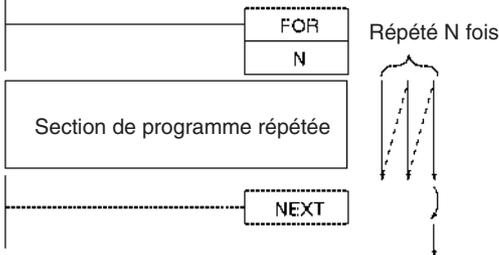
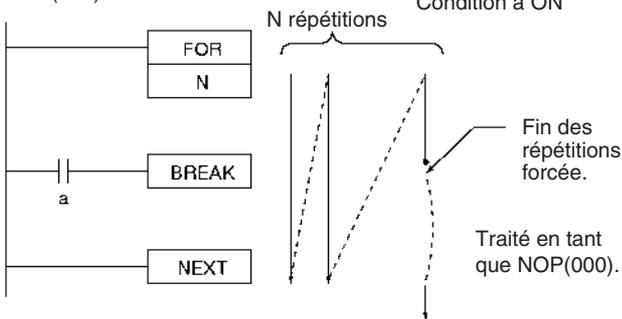
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>SINGLE BIT RESET (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b></p> <p>RSTB @RSTB !RSTB<sup>*1</sup> !@RSTB<sup>*1</sup></p>	 <p>D : Adresse mot N : Numéro de bit</p>	<p>RSTB(533) met à OFF le bit spécifié dans le mot spécifié lorsque la condition d'exécution est ON. A la différence de l'instruction RSET, RSTB(533) peut être utilisée pour réinitialiser un bit dans un mot DM ou EM.</p>	<p>Sortie Obligatoire</p>
<p><b>SINGLE BIT OUTPUT (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b></p> <p>OUTB @OUTB !OUTB<sup>*1</sup></p>	 <p>D : Adresse mot N : Numéro de bit</p>	<p>OUTB(534) sort le résultat (condition d'exécution) du traitement logique vers le bit spécifié. A la différence de l'instruction OUT, OUTB(534) peut être utilisée pour contrôler un bit dans un mot DM ou EM.</p>	<p>Sortie Obligatoire</p>

### 3-3 Instructions de commande des séquences

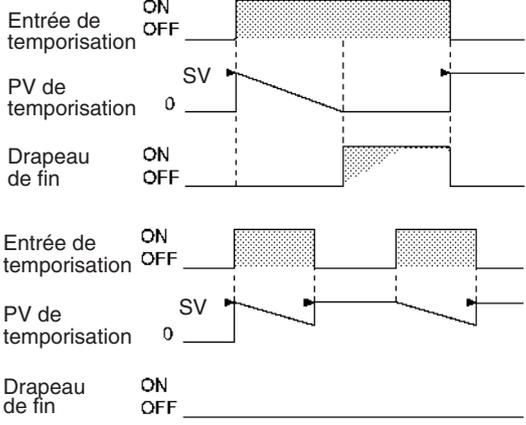
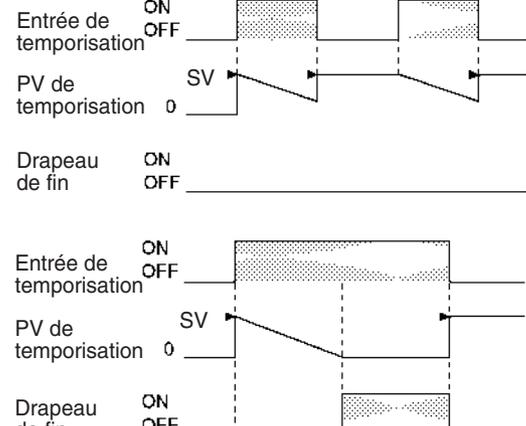
Instruction Mnémomique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>END</b></p> <p>END 001</p>	<p>END(001)</p>	<p>Indique la fin d'un programme. END(001) termine l'exécution d'un programme pour ce cycle. Aucune instruction n'est écrite après l'exécution de END(001). L'exécution passe au programme correspondant au numéro de tâche suivant. Lorsque le programme en cours d'exécution correspond au numéro de tâche le plus élevé dans le programme, END(001) marque la fin de l'ensemble du programme principal.</p> 	<p>Sortie Non obligatoire</p>
<p><b>NO OPERATION</b></p> <p>NOP 000</p>		<p>Cette instruction n'a aucune fonction (aucun traitement n'est effectué pour NOP(000).)</p>	<p>Sortie Non obligatoire</p>
<p><b>INTERLOCK</b></p> <p>IL 002</p>	<p>IL(002)</p>	<p>Verrouille toutes les sorties situées entre IL(002) et ILC(003) lorsque la condition d'exécution de IL(002) est sur OFF. IL(002) et ILC(003) sont normalement utilisées sous forme de paire.</p> 	<p>Sortie Obligatoire</p>

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>INTERLOCK CLEAR</b>  ILC 003		Toutes les sorties situées entre IL(002) et ILC(003) sont verrouillées lorsque la condition d'exécution de IL(002) est OFF. IL(002) et ILC(003) sont normalement utilisées sous forme de paire.	Sortie Non obligatoire
<b>MULTI-INTERLOCK DIFFERENTIATION HOLD</b>  MILH 517  UC série CS/CJ Ver. 2.0 ou supérieure uniquement	  N : Numéro de verrouillage D : Bit d'état du verrouillage	Lorsque la condition d'exécution de MILH(517) est OFF, les sorties de toutes les instructions situées entre cette instruction MILH(517) et la prochaine instruction MILC(519) sont verrouillées. MILH(517) et MILC(519) sont utilisées sous forme de paire.  Les verrouillages MILH(517)/MILC(519) peuvent être imbriqués (exemple : MILH(517)—MILH(517)—MILC(519)—MILC(519)).  S'il existe une instruction différenciée (DIFU, DIFD ou une instruction avec un préfixe @ ou %) entre MILH(517) et la MILC(519) correspondante, cette instruction sera exécutée après la libération du verrouillage si la condition de différenciation de l'instruction a été établie.	Sortie Obligatoire
<b>MULTI-INTERLOCK DIFFERENTIATION RELEASE</b>  MILR 518  UC série CS/CJ Ver. 2.0 ou supérieure uniquement	  N : Numéro de verrouillage D : Bit d'état du verrouillage	Lorsque la condition d'exécution de MILR(518) est OFF, les sorties de toutes les instructions situées entre cette instruction MILR(518) et la prochaine instruction MILC(519) sont verrouillées. MILR(518) et MILC(519) sont utilisées en tant que paire.  Les verrouillages MILR(518)/MILC(519) peuvent être imbriqués (exemple : MILR(518)—MILR(518)—MILC(519)—MILC(519)).  S'il existe une instruction différenciée (DIFU, DIFD ou une instruction avec un préfixe @ ou %) entre MILR(518) et la MILC(519) correspondante, cette instruction ne sera pas exécutée après la suppression du verrouillage, même si la condition de différenciation de l'instruction a été établie.	Sortie Obligatoire
<b>MULTI-INTERLOCK CLEAR</b>  MILC 519  UC série CS/CJ Ver. 2.0 ou supérieure uniquement	  N : Numéro de verrouillage	Efface un verrouillage déclenché par une MILH(517) ou une MILR(518) portant le même numéro de verrouillage.  Toutes les sorties situées entre MILH(517)/MILR(518) et la MILC(519) correspondante, et qui portent le même numéro de verrouillage, sont verrouillées lorsque la condition d'exécution de MILH(517)/MILR(518) est OFF.	Sortie Non obligatoire
<b>JUMP</b>  JMP 004	  N : Numéro de saut	Lorsque la condition d'exécution de JMP(004) est sur OFF, l'exécution du programme passe directement au premier JME(005) du programme avec le même numéro de saut. JMP(004) et JME(005) sont utilisées dans des paires.  	Sortie Obligatoire
<b>JUMP END</b>  JME 005	  N : Numéro de saut	Indique la fin d'un saut déclenché par JMP(004) ou CJP(510).	Sortie Non obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>CONDITIONAL JUMP</b>  CJP 510	 N : Numéro de saut	<p>Le fonctionnement de CJP(510) est fondamentalement à l'opposé de celui de JMP(004). Lorsque la condition d'exécution de CJP(510) est sur ON, l'exécution du programme passe directement au premier JME(005) du programme avec le même numéro de saut. CJP(510) et JME(005) sont utilisées dans des paires.</p> 	Sortie Obligatoire
<b>CONDITIONAL JUMP</b>  CJPN 511	 N : Numéro de saut	<p>Le fonctionnement de CJPN(511) est presque identique à celui de JMP(004). Lorsque la condition d'exécution de CJP(004) est sur OFF, l'exécution du programme passe directement au premier JME(005) du programme avec le même numéro de saut. CJPN(511) et JME(005) sont utilisées dans des paires.</p> 	Sortie Non obligatoire
<b>MULTIPLE JUMP</b> JMP0 515		<p>Lorsque la condition d'exécution de JMP0(515) est OFF, toutes les instructions situées entre JMP0(515) et la prochaine JME0(516) dans le programme sont traitées en tant que NOP(000). Utilisez JMP0(515) et JME0(516) sous forme de paire. Le nombre de paires pouvant être utilisées dans le programme est illimité.</p> 	Sortie Obligatoire
<b>MULTIPLE JUMP END</b> JME0 516		<p>Lorsque la condition d'exécution de JMP0(515) est OFF, toutes les instructions situées entre JMP0(515) et la prochaine JME0(516) dans le programme sont traitées en tant que NOP(000). Utilisez JMP0(515) et JME0(516) sous forme de paire. Le nombre de paires pouvant être utilisées dans le programme est illimité.</p>	Sortie Non obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>FOR-NEXT LOOPS</b>  FOR 512	 <p>N : Nombre de boucles</p>	<p>Les instructions situées entre FOR(512) et NEXT(513) sont répétées un nombre de fois spécifié. FOR(512) et NEXT(513) sont utilisées sous forme de paire.</p>  <p>Répété N fois</p>	Sortie Non obligatoire
<b>BREAK LOOP</b> BREAK 514		<p>Programmé dans une boucle FOR-NEXT de façon à annuler l'exécution de la boucle pour une condition d'exécution donnée. Les autres instructions de la boucle sont traitées comme des instructions NOP(000).</p>  <p>N répétitions</p> <p>Condition a ON</p> <p>Fin des répétitions forcée.</p> <p>Traité en tant que NOP(000).</p>	Sortie Obligatoire
<b>FOR-NEXT LOOPS</b>  NEXT 513		<p>Les instructions situées entre FOR(512) et NEXT(513) sont répétées un nombre de fois spécifié. FOR(512) et NEXT(513) sont utilisées sous forme de paire.</p>	Sortie Non obligatoire

### 3-4 Instructions de temporisation et de compteur

Instruction Mnémorique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>TIMER</b></p> <p>TIM (BCD)</p> <p>TIMX (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>TIM</p> <p>N</p> <p>S</p> <p>N : Numéro de temporisation S : Valeur de consigne</p> <p>TIMX(550)</p> <p>N</p> <p>S</p> <p>N : Numéro de temporisation S : Valeur de consigne</p>	<p>TIM/TIMX(550) effectue une temporisation décroissante par pas de 0,1 s. La plage de configuration pour la valeur de consigne (SV) va de 0 à 999,9 s pour les données BCD et de 0 à 65 553,5 s pour les données binaires (décimales ou hexadécimales).</p> 	<p>Sortie Obligatoire</p>
<p><b>HIGH-SPEED TIMER</b></p> <p>TIMH 015 (BCD)</p> <p>TIMHX 551 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>TIMH(015)</p> <p>N</p> <p>S</p> <p>N : Numéro de temporisation S : Valeur de consigne</p> <p>TIMHX(551)</p> <p>N</p> <p>S</p> <p>N : Numéro de temporisation S : Valeur de consigne</p>	<p>TIMH(015)/TIMHX(551) effectue une temporisation décroissante par pas de 10 ms. La plage de configuration pour la valeur de consigne (SV) va de 0 à 99,9 s pour les données BCD et de 0 à 655,35 s pour les données binaires (décimales ou hexadécimales).</p> 	<p>Sortie Obligatoire</p>
<p><b>ONE-MS TIMER</b></p> <p>TMHH 540 (BCD)</p> <p>TMHXX 552 (BCD) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>TMHH(540)</p> <p>N</p> <p>S</p> <p>N : Numéro de temporisation S : Valeur de consigne</p> <p>TMHXX(552)</p> <p>N</p> <p>S</p> <p>N : Numéro de temporisation S : Valeur de consigne</p>	<p>TMHH(540)/TMHXX(552) effectue une temporisation décroissante par pas de 1 ms. La plage de configuration pour la valeur de consigne (SV) va de 0 à 9,999 s pour les données BCD et de 0 à 65,535 s pour les données binaires (décimales ou hexadécimales).</p> <p>Les tableaux de temporisation pour TMHH(540) et TMHXX(552) sont les mêmes que ceux donnés ci-dessus pour TIMH(015).</p>	<p>Sortie Obligatoire</p>

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution								
<p><b>ACCUMULATIVE TIMER</b></p> <p>TTIM 087 (BCD)</p> <p>TTIMX 555 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>Entrée de tempo- risation</p> <table border="1" data-bbox="400 266 512 363"> <tr><td>TTIM(087)</td></tr> <tr><td>N</td></tr> <tr><td>S</td></tr> </table> <p>Entrée de réinitialisation</p> <p>N : Numéro de temporisation S : Valeur de consigne</p> <p>Entrée de tempori- sation</p> <table border="1" data-bbox="400 527 512 625"> <tr><td>TTIMX(555)</td></tr> <tr><td>N</td></tr> <tr><td>S</td></tr> </table> <p>Entrée de réinitialisation</p> <p>N : Numéro de temporisation S : Valeur de consigne</p>	TTIM(087)	N	S	TTIMX(555)	N	S	<p>TTIM(087)/TTIMX(555) effectue une temporisation croissante par pas de 0,1 s. La plage de configuration pour la valeur de consigne (SV) va de 0 à 999,9 s pour les données BCD et de 0 à 65 553,5 s pour les données binaires (décimales ou hexadécimales).</p> <p>Entrée de ON OFF</p> <p>Entrée de ON OFF</p> <p>PV de temporisation</p> <p>SV</p> <p>0</p> <p>Drapeau de fin ON OFF</p> <p>Entrée de ON OFF</p> <p>La temporisation reprend</p> <p>PV conservée.</p>	<p>Sortie Obligatoire</p>		
TTIM(087)											
N											
S											
TTIMX(555)											
N											
S											
<p><b>LONG TIMER</b></p> <p>TIML 542 (BCD)</p> <p>TIMLX 553 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>Entrée de tempo- risation</p> <table border="1" data-bbox="400 776 512 938"> <tr><td>TIML(542)</td></tr> <tr><td>D1</td></tr> <tr><td>D2</td></tr> <tr><td>S</td></tr> </table> <p>Entrée de réinitialisation</p> <p>D1 : Drapeau de fin D2 : Mot PV S : Mot SV</p> <p>Entrée de tempori- sation</p> <table border="1" data-bbox="400 1081 512 1221"> <tr><td>TIMLX(553)</td></tr> <tr><td>D1</td></tr> <tr><td>D2</td></tr> <tr><td>S</td></tr> </table> <p>D1 : Drapeau de fin D2 : Mot PV S : Mot SV</p>	TIML(542)	D1	D2	S	TIMLX(553)	D1	D2	S	<p>TIML(542)/TIMLX(553) effectue une temporisation décroissante par pas de 0,1 s qui peut aller jusqu'à 115 jours pour les données BCD et 49 710 jours pour les données binaires (décimales ou hexadécimales).</p> <p>Entrée de ON OFF</p> <p>Entrée de ON OFF</p> <p>PV de temporisation</p> <p>SV</p> <p>0</p> <p>Drapeau de fin ON OFF</p> <p>(Bit 00 de D1)</p>	<p>Sortie Obligatoire</p>
TIML(542)											
D1											
D2											
S											
TIMLX(553)											
D1											
D2											
S											

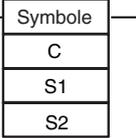
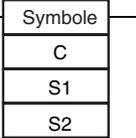
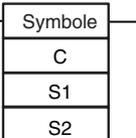
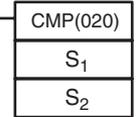
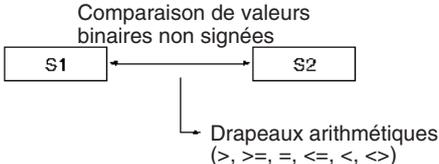
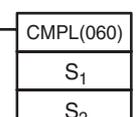
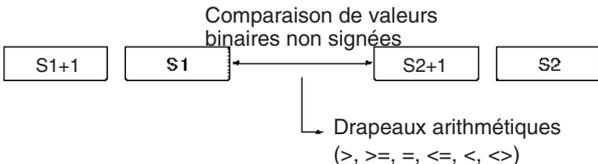
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>MULTI-OUTPUT TIMER</b></p> <p>MTIM 543 (BCD)</p> <p>MTIMX 554 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p>MTIM(543)</p> <p>D1</p> <p>D2</p> <p>S</p> </div> <p><b>D1</b> : Drapeaux de fin <b>D2</b> : Mot PV <b>S</b> : 1er mot SV</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p>MTIMX(554)</p> <p>D1</p> <p>D2</p> <p>S</p> </div> <p><b>D1</b> : Drapeaux de fin <b>D2</b> : Mot PV <b>S</b> : 1er mot SV</p>	<p>MTIM(543)/MTIMX(554) prend en charge une temporisation basée sur des incréments de 0,1 s avec 8 valeurs de consigne (SV) et drapeaux de fin indépendants. La plage de configuration pour la valeur de consigne (SV) est comprise entre 0 et 999,9 s pour les données BCD, entre 0 et 6 553,5 s pour les données binaires (décimales ou hexadécimales).</p> <p>The diagram illustrates the timer's internal structure and timing. At the top, 'PV de temporisation' (D2) is shown as a single input. Below it, 'SV de temporisation' (S) is shown as a stack of 8 values: S, S+1, S+2, ..., jusqu'à, S+7. Each SV value is associated with a bit of the 'Drapeaux de fin' (D1), labeled 'Bit 7 jusqu'à Bit 2', 'Bit 1', and 'Bit 0'. A timing diagram shows the 'Entrée de temporisation' (ON/OFF) signal, the 'PV de temporisation (D2)' value, and the 'Exécution Drapeaux (D1)' bits. The PV value increases linearly over time, and the execution flags are set when the PV reaches a specific SV value.</p>	<p>Sortie Obligatoire</p>
<p><b>COUNTER</b></p> <p>CNT (BCD)</p> <p>CNTX 546 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p>CNT</p> <p>N</p> <p>S</p> </div> <p>Entrée de comptage</p> <p>Entrée de réinitialisation</p> <p><b>N</b> : Numéro de compteur <b>S</b> : Valeur de consigne</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p>CNTX(546)</p> <p>N</p> <p>S</p> </div> <p>Entrée de comptage</p> <p>Entrée de réinitialisation</p> <p><b>N</b> : Numéro de compteur <b>S</b> : Valeur de consigne</p>	<p>CNT/CNTX(546) prend en charge un compteur décrémental. La plage de configuration pour la valeur de consigne (SV) va de 0 à 9 999 pour les données BCD et de 0 à 65 535 pour les données binaires (décimales ou hexadécimales).</p> <p>The diagram shows the counter's operation. It features a 'Drapeau de fin' (ON/OFF) signal, an 'Entrée de comptage' (ON/OFF) signal, and an 'Entrée de réinitialisation' (ON/OFF) signal. The 'PV de compteur' (SV) is shown as a value that decreases from its initial value towards 0 as the counter counts down. The counter is reset to its initial value when the 'Entrée de réinitialisation' is active.</p>	<p>Sortie Obligatoire</p>

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>REVERSIBLE COUNTER</b></p> <p>CNTR 012 (BCD)</p> <p>CNTRX 548 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>Entrée incrémentielle</p> <p>Entrée décré- mentielle</p> <p>Entrée de réinitialisation</p> <p><b>N</b> : Numéro de compteur <b>S</b> : Valeur de consigne</p> <p>Entrée incrémentielle</p> <p>Entrée décré- mentielle</p> <p>Entrée de réinitialisation</p> <p><b>N</b> : Numéro de compteur <b>S</b> : Valeur de consigne</p>	<p>CNTR(012)/CNTRX(548) prend en charge un compteur réversible.</p> <p>Entrée incrémentielle</p> <p>Entrée décré- mentielle</p> <p>PV de compteur</p> <p>0</p> <p>PV de compteur</p> <p>SV</p> <p>0</p> <p>+1</p> <p>Drapeau de fin</p> <p>ON</p> <p>OFF</p> <p>PV de compteur</p> <p>SV</p> <p>0</p> <p>1</p> <p>Drapeau de fin</p> <p>ON</p> <p>OFF</p>	<p>Sortie Obligatoire</p>
<p><b>RESET TIMER/ COUNTER</b></p> <p>CNR @CNR 545 (BCD)</p> <p>CNRX @CNRX 547 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>CNR(545)</p> <p>N1</p> <p>N2</p> <p><b>N1</b> : Premier numéro de la plage <b>N2</b> : Dernier numéro de la plage</p> <p>CNRX(547)</p> <p>N1</p> <p>N2</p> <p><b>N1</b> : Premier numéro de la plage <b>N2</b> : Dernier numéro de la plage</p>	<p>CNR(545)/CNRX(547) réinitialise les temporisations ou les compteurs dans l'intervalle des numéros de temporisation et de compteur spécifié. Attribue à la valeur de consigne (SV) la valeur maximale de 9999.</p>	<p>Sortie Obligatoire</p>

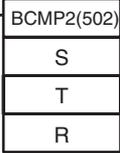
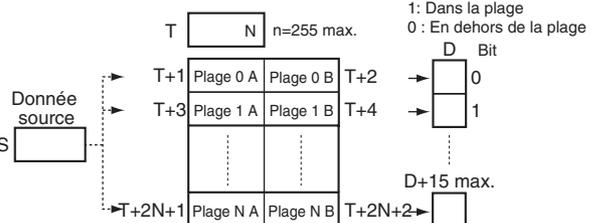
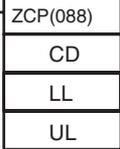
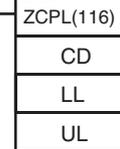
### 3-5 Instructions de comparaison

\*1 : non prise en charge par les UC CS1D pour les systèmes d'UC en duplex.

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution			
<p><b>Comparaison de symboles (non signé)</b> LD, AND, OR + =, &lt;&gt;, &lt;, &lt;=, &gt;, &gt;= 300 (=) 305 (&lt;&gt;) 310 (&lt;) 315 (&lt;=) 320 (&gt;) 325 (&gt;=)</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">Symbole et options</td> </tr> <tr> <td style="text-align: center;">S1</td> </tr> <tr> <td style="text-align: center;">S2</td> </tr> </table> <p>S1 : Donnée de comparaison 1 S2 : Donnée de comparaison 2</p>	Symbole et options	S1	S2	<p>Les instructions de comparaison de symboles (non signées) comparent deux valeurs (constantes et/ou le contenu des mots spécifiés) dans les données binaires 16 bits et créent une condition d'exécution ON lorsque la condition de comparaison est vraie. Il existe trois types d'instructions de comparaison à symbole : LD (LOAD), AND et OR.</p> <p>LD Condition d'exécution sur ON lorsque le résultat de la comparaison est vrai.</p> <p>AND Condition d'exécution sur ON lorsque le résultat de la comparaison est vrai.</p> <p>OR Condition d'exécution sur ON lorsque le résultat de la comparaison est vrai.</p>	<p>LD : non obligatoire AND, OR : obligatoire</p>
Symbole et options						
S1						
S2						
<p><b>Comparaison de symboles (mot double, non signé)</b> LD, AND, OR + =, &lt;&gt;, &lt;, &lt;=, &gt;, &gt;= + L 301 (=) 306 (&lt;&gt;) 311 (&lt;) 316 (&lt;=) 321 (&gt;) 326 (&gt;=)</p>	<p>S1 : données de comparaison 1 S2 : données de comparaison 2</p>	<p>Les instructions de comparaison de symboles (mot double, non signé) comparent deux valeurs (constantes et/ou le contenu des données de mots doubles spécifiés) dans les données binaires 32 bits non signées et créent une condition d'exécution ON lorsque la condition de comparaison est vraie. Il existe trois types d'instructions de comparaison à symbole : LD (LOAD), AND et OR.</p>	<p>LD : non obligatoire AND, OR : obligatoire</p>			
<p><b>Comparaison de symboles (signé)</b> LD, AND, OR + =, &lt;&gt;, &lt;, &lt;=, &gt;, &gt;= + S 302 (=) 307 (&lt;&gt;) 312 (&lt;) 317 (&lt;=) 322 (&gt;) 327 (&gt;=)</p>	<p>S1 : données de comparaison 1 S2 : données de comparaison 2</p>	<p>Les instructions de comparaison de symboles (signé) comparent deux valeurs (constantes et/ou le contenu des mots spécifiés) dans les données binaires 16 bits signées (4 chiffres hexadécimaux) et créent une condition d'exécution ON lorsque la condition de comparaison est vraie. Il existe trois types d'instructions de comparaison à symbole : LD (LOAD), AND et OR.</p>	<p>LD : non obligatoire AND, OR : obligatoire</p>			

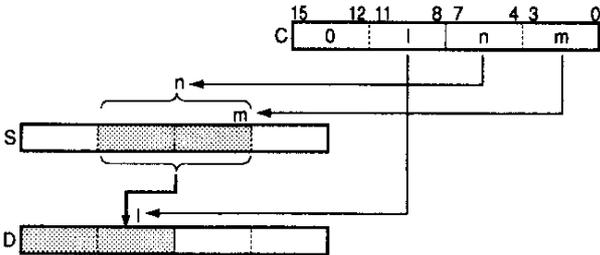
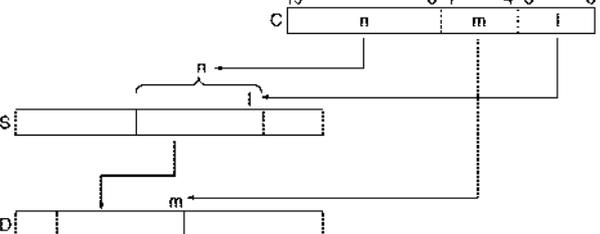
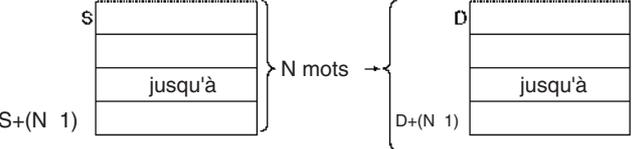
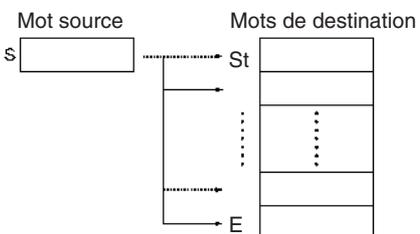
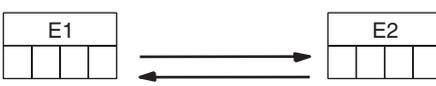
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>Comparaison de symboles (mot double, signé)</b> LD, AND, OR +=, &lt;&gt;, &lt;, &lt;=, &gt;, &gt;= +SL 303 (=) 308 (&lt;&gt;) 313 (&lt;) 318 (&lt;=) 323 (&gt;) 328 (&gt;=)</p>	<p>S<sub>1</sub> : données de comparaison 1 S<sub>2</sub> : données de comparaison 2</p>	<p>Les instructions de comparaison de symboles (mot double, signé) comparent deux valeurs (constantes et/ou le contenu des données de mots doubles spécifiés) dans les données binaires 32 bits signées (8 chiffres hexadécimaux) et créent une condition d'exécution ON lorsque la condition de comparaison est vraie. Il existe trois types d'instructions de comparaison à symbole : LD (LOAD), AND et OR.</p>	<p>LD : non obligatoire AND, OR : obligatoire</p>
<p><b>Comparaison de temps</b> LD, AND, OR += DT, &lt;&gt; DT, &lt; DT, &lt;= DT, &gt; DT, &gt;= DT 341 (= DT) 342 (&lt;&gt; DT) 343 (&lt; DT) 344 (&lt;= DT) 345 (&gt; DT) 346 (&gt;= DT) (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)</p>	<p>LD (LOAD) :</p>  <p>AND :</p>  <p>OR :</p>  <p>C : Mot de contrôle S<sub>1</sub> : 1er mot temps présent S<sub>2</sub> : 1er mot temps de comparaison</p>	<p>Les instructions de comparaison de temps comparent deux valeurs de temps BCD et créent une condition d'exécution ON lorsque la condition de comparaison est vraie. Il existe trois types d'instructions de comparaison de temps : LD (LOAD), AND et OR. Les valeurs de temps (année, mois, jour, heure, minute et seconde) peuvent être masquées/démasquées dans la comparaison, ce qui permet de créer facilement des fonctions de temporisation de calendrier.</p>	<p>LD : non obligatoire AND, OR : obligatoire</p>
<p><b>UNSIGNED COMPARE</b> CMP !CMP*1 020</p>	 <p>S<sub>1</sub> : Donnée de comparaison 1 S<sub>2</sub> : Donnée de comparaison 2</p>	<p>Compare deux valeurs binaires non signées (constantes et/ou le contenu des mots spécifiés) et place le résultat dans les drapeaux arithmétiques à l'intérieur de la zone auxiliaire.</p> 	<p>Sortie Obligatoire</p>
<p><b>DOUBLE UNSIGNED COMPARE</b> CMPL 060</p>	 <p>S<sub>1</sub> : Donnée de comparaison 1 S<sub>2</sub> : Donnée de comparaison 2</p>	<p>Compare deux valeurs binaires non signées doubles (constantes et/ou le contenu des mots spécifiés) et place le résultat dans les drapeaux arithmétiques à l'intérieur de la zone auxiliaire.</p> 	<p>Sortie Obligatoire</p>

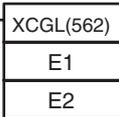
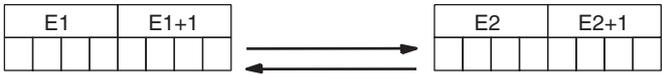
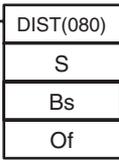
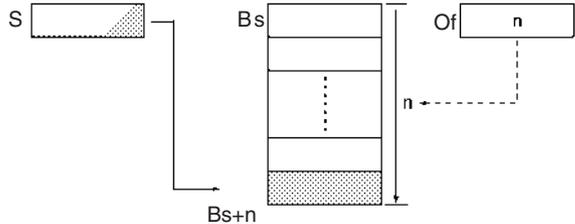
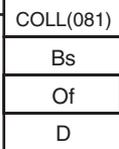
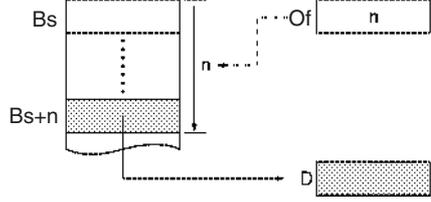
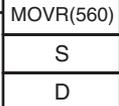
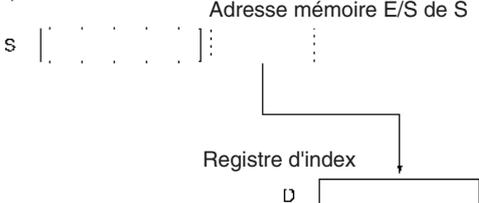
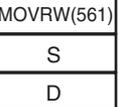
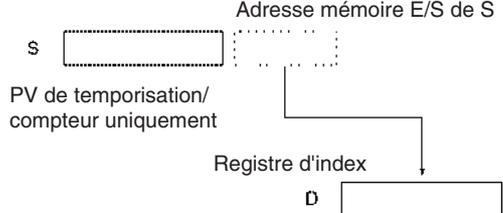
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>SIGNED BINARY COMPARE</b>  CPS !CPS*1 114	<p>S<sub>1</sub> : Donnée de comparaison 1 S<sub>2</sub> : Donnée de comparaison 2</p>	Compare deux valeurs binaires signées (constantes et/ou le contenu des mots spécifiés) et place le résultat dans les drapeaux arithmétiques à l'intérieur de la zone auxiliaire.  Comparaison de valeurs binaires non signées 	Sortie Obligatoire
<b>DOUBLE SIGNED BINARY COMPARE</b>  CPSL 115	<p>S<sub>1</sub> : Donnée de comparaison 1 S<sub>2</sub> : Donnée de comparaison 2</p>	Compare deux valeurs binaires signées doubles (constantes et/ou le contenu des mots spécifiés) et place le résultat dans les drapeaux arithmétiques à l'intérieur de la zone auxiliaire.  Comparaison de valeurs binaires non signées 	Sortie Obligatoire
<b>MULTIPLE COMPARE</b>  MCMP @MCMP 019	<p>S<sub>1</sub> : 1er mot de l'ensemble 1 S<sub>2</sub> : 1er mot de l'ensemble 2 R : Mot de résultat</p>	Compare 16 mots consécutifs avec 16 autres mots consécutifs et met sur ON le bit correspondant dans le mot de résultat lorsque les contenus ne sont pas égaux.  Comparaison 	Sortie Obligatoire
<b>TABLE COMPARE</b>  TCMP @TCMP 085	<p>S : Donnée source T : 1er mot de la table R : Mot de résultat</p>	Compare la donnée source avec le contenu de 16 mots et met sur ON le bit correspondant dans le mot de résultat lorsque les contenus sont égaux.  Comparaison 	Sortie Obligatoire
<b>UNSIGNED BLOCK COMPARE</b>  BCMP @BCMP 068	<p>S : Donnée source T : 1er mot de la table R : Mot de résultat</p>	Compare la donnée source avec 16 plages (définies par 16 limites inférieures et 16 limites supérieures) et met sur ON le bit correspondant dans le mot de résultat lorsque la donnée source est comprise dans la plage.  Plages Limite inférieure      Limite supérieure 	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>EXPANDED BLOCK COMPARE</b> BCMP2 @BCMP2 502</p> <p>(UC CS1-H, CJ1-H ou CS1D Ver. 2.0 ou supérieure uniquement) UC CJ1M (Pré-ver. 2.0 ou Ver. 2.0 ou supérieure)</p>	 <p>S : Donnée source T : 1er mot du bloc R : Mot de résultat</p>	<p>Compare les données source par rapport à 256 plages maximum (définies par des limites supérieure et inférieure) et met à ON le bit correspondant dans le mot de résultat lorsque les données sources sont à l'intérieur d'une plage.</p>  <p><b>Remarque :</b> A peut être inférieur ou égal à B ou supérieur à B.</p>	<p>Sortie Obligatoire</p>
<p><b>AREA RANGE COMPARE</b> ZCP @ZCP 088</p> <p>(CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	 <p>CD : Donnée de comparaison (1 mot) LL : Limite inférieure de la plage UL : Limite supérieure de la plage</p>	<p>Compare la valeur binaire 16 bits non signée de CD (contenu de mot ou constante) à la plage définie par LL et UL, puis sort les résultats vers les drapeaux arithmétiques dans la zone auxiliaire.</p>	<p>Sortie Obligatoire</p>
<p><b>DOUBLE AREA RANGE COMPARE</b> ZCPL @ZCPL 116</p> <p>(CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	 <p>CD : Donnée de comparaison (2 mots) LL : Limite inférieure de la plage UL : Limite supérieure de la plage</p>	<p>Compare la valeur binaire 32 bits non signée de CD et CD+1 (contenu de mot ou constante) à la plage définie par LL et UL, puis sort les résultats vers les drapeaux arithmétiques dans la zone auxiliaire.</p>	<p>Sortie Obligatoire</p>

### 3-6 Instructions de déplacement de données

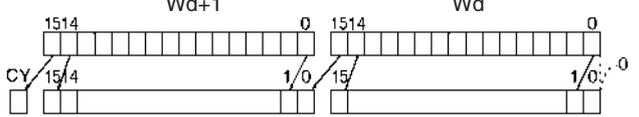
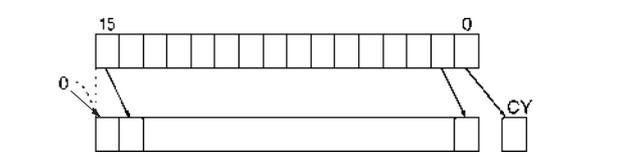
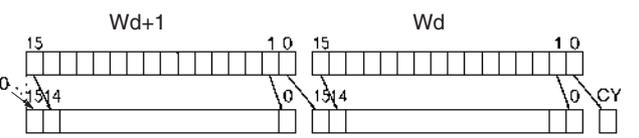
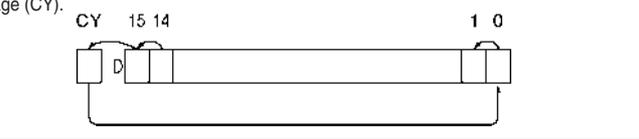
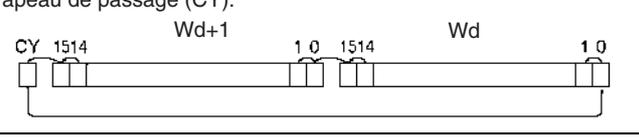
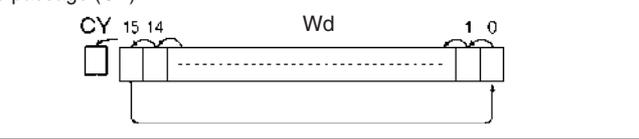
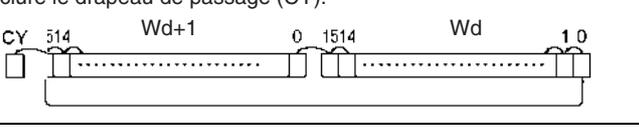
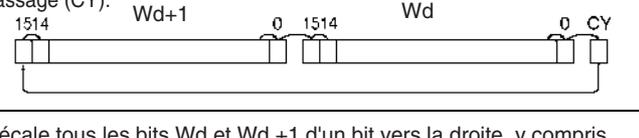
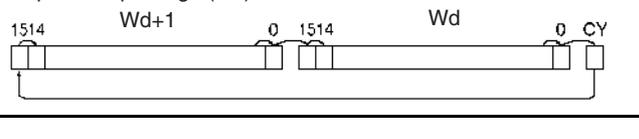
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>MOVE</b> MOV @MOV !MOV !@MOV 021	MOV(021) S D S : Source D : Destination	Transfère un mot de données vers le mot spécifié. 	Sortie Obligatoire
<b>DOUBLE MOVE</b> MOVL @MOVL 498	MOVL(498) S D S : 1er mot source D : 1er mot destination	Transfère deux mots de données vers les mots spécifiés. 	Sortie Obligatoire
<b>MOVE NOT</b> MVN @MVN 022	MVN(022) S D S : Source D : Destination	Transfère le complément d'un mot de données vers le mot spécifié. 	Sortie Obligatoire
<b>DOUBLE MOVE NOT</b> MVNL @MVNL 499	MVNL(499) S D S : 1er mot source D : 1er mot destination	Transfère le complément de deux mots vers les mots spécifiés. 	Sortie Obligatoire
<b>MOVE BIT</b> MOVB @MOVB 082	MOVB(082) S C D S : Mot ou donnée source C : Mot de contrôle D : Mot de destination	Transfère le bit spécifié. 	Sortie Obligatoire

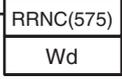
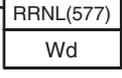
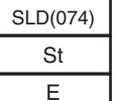
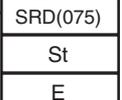
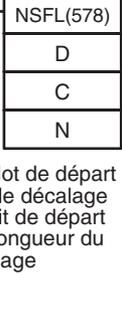
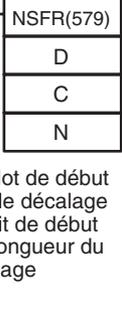
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<b>MOVE DIGIT</b> MOVD @MOVD 083	<table border="1" style="width: 100%; text-align: center;"> <tr><td>MOVD(083)</td></tr> <tr><td>S</td></tr> <tr><td>C</td></tr> <tr><td>D</td></tr> </table> <p>S : Mot ou donnée source                      C : Mot de contrôle                      D : Mot de destination</p>	MOVD(083)	S	C	D	Transfère le ou les chiffres spécifiés. (Chaque chiffre est constitué de 4 bits.) 	Sortie Obligatoire
MOVD(083)							
S							
C							
D							
<b>MULTIPLE BIT TRANSFER</b> XFRB @XFRB 062	<table border="1" style="width: 100%; text-align: center;"> <tr><td>XFRB(062)</td></tr> <tr><td>C</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> </table> <p>C : Mot de contrôle                      S : 1er mot source                      D : 1er mot de destination</p>	XFRB(062)	C	S	D	Transfère le nombre de bits consécutifs spécifié. 	Sortie Obligatoire
XFRB(062)							
C							
S							
D							
<b>BLOCK TRANSFER</b> XFER @XFER 070	<table border="1" style="width: 100%; text-align: center;"> <tr><td>XFER(070)</td></tr> <tr><td>N</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> </table> <p>N : Nombre de mots                      S : 1er mot source                      D : 1er mot de destination</p>	XFER(070)	N	S	D	Transfère le nombre de mots consécutifs spécifié. 	Sortie Obligatoire
XFER(070)							
N							
S							
D							
<b>BLOCK SET</b> BSET @BSET 071	<table border="1" style="width: 100%; text-align: center;"> <tr><td>BSET(071)</td></tr> <tr><td>S</td></tr> <tr><td>St</td></tr> <tr><td>E</td></tr> </table> <p>S : Mot source                      St : Mot de début                      E : Mot de fin</p>	BSET(071)	S	St	E	Copie le même mot dans une plage de mots 	Sortie Obligatoire
BSET(071)							
S							
St							
E							
<b>DATA EXCHANGE</b> XCHG @XCHG 073	<table border="1" style="width: 100%; text-align: center;"> <tr><td>XCHG(073)</td></tr> <tr><td>E1</td></tr> <tr><td>E2</td></tr> </table> <p>E1 : 1er mot échange                      E2 : Second mot échange</p>	XCHG(073)	E1	E2	Echange le contenu des deux mots spécifiés. 	Sortie Obligatoire	
XCHG(073)							
E1							
E2							

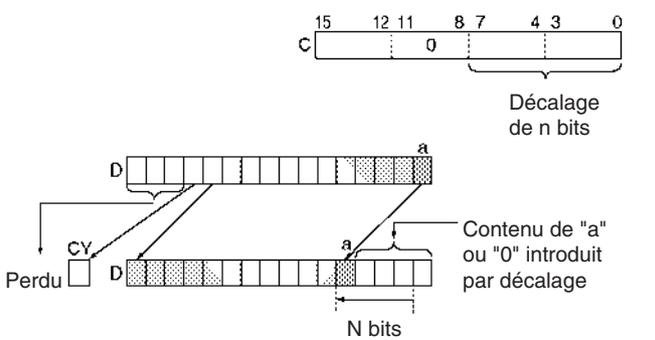
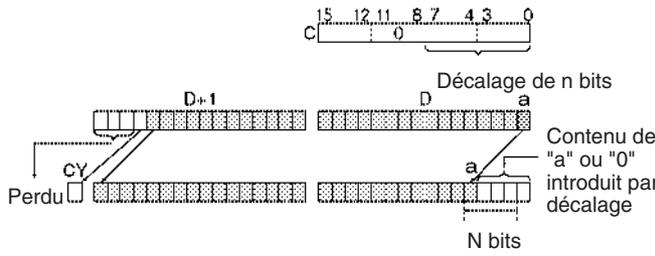
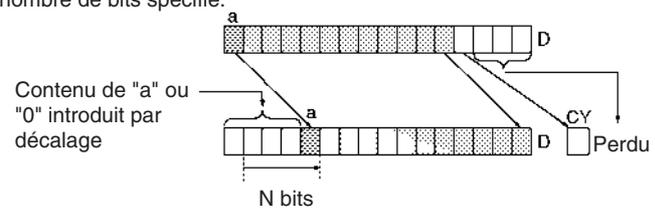
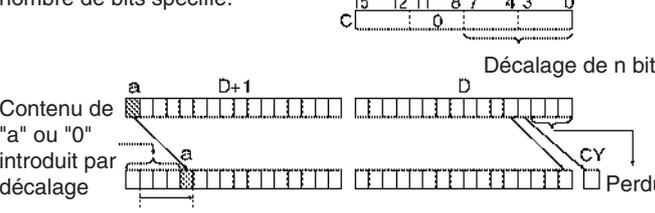
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DOUBLE DATA EXCHANGE</b> XCGL @XCGL 562	 <p>E1 : 1er mot échange E2 : Second mot échange</p>	Echange le contenu d'une paire de mots consécutifs avec une autre paire de mots consécutifs. 	Sortie Obligatoire
<b>SINGLE WORD DISTRIBUTE</b> DIST @DIST 080	 <p>S : Mot source Bs : Adresse destination de base Of : Décalage</p>	Transfère le mot source vers un mot de destination calculé en ajoutant une valeur de décalage à l'adresse de base. 	Sortie Obligatoire
<b>DATA COLLECT</b> COLL @COLL 081	 <p>Bs : Adresse de base source Of : Décalage D : Mot de destination</p>	Transfère le mot source (calculé en ajoutant une valeur de décalage à l'adresse de base) vers le mot de destination. 	Sortie Obligatoire
<b>MOVE TO REGISTER</b> MOVR @MOVR 560	 <p>S : Source (mot ou bit voulu) D : Destination (registre d'index)</p>	Définit l'adresse mémoire E/S interne du mot, du bit ou du drapeau de fin de temporisation/compteur spécifié dans le registre d'index spécifié. (Utilisez MOVRW(561) pour définir l'adresse mémoire E/S interne d'une PV de temporisation/compteur dans un registre d'index.) 	Sortie Obligatoire
<b>MOVE TIMER/COUNTER PV TO REGISTER</b> MOVRW @MOVRW 561	 <p>S : Source (numéro de TC voulu) D : Destination (registre d'index)</p>	Définit l'adresse mémoire E/S interne de la PV de temporisation/compteur spécifiée dans le registre d'index spécifié. (Utilisez MOVRW(560) pour définir l'adresse mémoire E/S interne d'un mot, d'un bit ou d'un drapeau de fin de temporisation/compteur) 	Sortie Obligatoire

### 3-7 Instructions de décalage de données

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>SHIFT REGISTER</b> SFT 010	<p>Entrée de données SFT(010) Entrée de décalage St E Entrée de réinitialisation</p> <p>St : Mot de début E : Mot de fin</p>	Utilise un registre à décalage <p>Perdu</p> <p>Etat de l'entrée de données pour chaque entrée de décalage</p>	Sortie Obligatoire
<b>REVERSIBLE SHIFT REGISTER</b> SFTR @SFTR 084	<p>SFTR(084) C St E</p> <p>C : Mot de contrôle St : Mot de début E : Mot de fin</p>	Crée un registre de décalage qui décale les données soit vers la droite soit vers la gauche. <p>Entrée de données</p> <p>Sens de décalage</p>	Sortie Obligatoire
<b>ASYNCHRONOUS SHIFT REGISTER</b> ASFT @ASFT 017	<p>ASFT(017) C St E</p> <p>C : Mot de contrôle St : Mot de début E : Mot de fin</p>	Décale toutes les données de mot non nulles dans la plage de mots spécifiée vers St ou vers E, en remplaçant les données de mot 0000Hex. <p>St</p> <p>Décalage</p> <p>Décalage activé</p> <p>Effacer</p> <p>E</p> <p>St</p> <p>Données nulles</p> <p>Données non nulles</p> <p>E</p>	Sortie Obligatoire
<b>WORD SHIFT</b> WSFT @WSFT 016	<p>WSFT(016) S St E</p> <p>S : Mot source St : Mot de début E : Mot de fin</p>	Décale les données entre St et E en unités de mots. <p>Perdu</p> <p>St</p> <p>S</p>	Sortie Obligatoire
<b>ARITHMETIC SHIFT LEFT</b> ASL @ASL 025	<p>ASL(025) Wd</p> <p>Wd : Mot</p>	Décale le contenu de Wd d'un bit vers la gauche. <p>15</p> <p>0</p> <p>CY</p>	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DOUBLE SHIFT LEFT</b> ASLL @ASLL 570	ASLL(570) Wd	Décale le contenu de Wd et Wd +1 d'un bit vers la gauche. 	Sortie Obligatoire
<b>ARITHMETIC SHIFT RIGHT</b> ASR @ASR 026	ASR(026) Wd	Décale le contenu de Wd d'un bit vers la droite. 	Sortie Obligatoire
<b>DOUBLE SHIFT RIGHT</b> ASRL @ASRL 571	ASRL(571) Wd	Décale le contenu de Wd et Wd +1 d'un bit vers la droite. 	Sortie Obligatoire
<b>ROTATE LEFT</b> ROL @ROL 027	ROL(027) Wd	Décale tous les bits Wd d'un bit vers la gauche, y compris le drapeau de passage (CY). 	Sortie Obligatoire
<b>DOUBLE ROTATE LEFT</b> ROLL @ROLL 572	ROLL(572) Wd	Décale tous les bits Wd et Wd +1 d'un bit vers la gauche, y compris le drapeau de passage (CY). 	Sortie Obligatoire
<b>ROTATE LEFT WITHOUT CARRY</b> RLNC @RLNC 574	RLNC(574) Wd	Décale tous les bits Wd d'un bit vers la gauche, sans inclure le drapeau de passage (CY). 	Sortie Obligatoire
<b>DOUBLE ROTATE LEFT WITHOUT CARRY</b> RLNL @RLNL 576	RLNL(576) Wd	Décale tous les bits Wd et Wd +1 d'un bit vers la gauche, sans inclure le drapeau de passage (CY). 	Sortie Obligatoire
<b>ROTATE RIGHT</b> ROR @ROR 028	ROR(028) Wd	Décale tous les bits Wd d'un bit vers la droite, y compris le drapeau de passage (CY). 	Sortie Obligatoire
<b>DOUBLE ROTATE RIGHT</b> RORL @RORL 573	RORL(573) Wd	Décale tous les bits Wd et Wd +1 d'un bit vers la droite, y compris le drapeau de passage (CY). 	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>ROTATE RIGHT WITHOUT CARRY</b>  RRNC @RRNC 575	 <p>RRNC(575) Wd</p> <p>Wd : Mot</p>	Décale tous les bits Wd d'un bit vers la droite, sans inclure le drapeau de passage (CY). Le contenu du bit le plus à droite de Wd se décale vers le bit de gauche et vers le drapeau de passage (CY).	Sortie Obligatoire
<b>DOUBLE ROTATE RIGHT WITHOUT CARRY</b>  RRNL @RRNL 577	 <p>RRNL(577) Wd</p> <p>Wd : Mot</p>	Décale tous les bits Wd et Wd + 1 d'un bit vers la droite, sans inclure le drapeau de passage (CY). Le contenu du bit le plus à droite de Wd + 1 est décalé vers le bit de gauche de Wd et vers le drapeau de passage (CY).	Sortie Obligatoire
<b>ONE DIGIT SHIFT LEFT</b>  SLD @SLD 074	 <p>SLD(074) St E</p> <p>St : Mot de début E : Mot de fin</p>	Décale les données d'un chiffre (4 octets) vers la gauche.	Sortie Obligatoire
<b>ONE DIGIT SHIFT RIGHT</b>  SRD @SRD 075	 <p>SRD(075) St E</p> <p>St : Mot de début E : Mot de fin</p>	Décale les données d'un chiffre (4 octets) vers la droite.	Sortie Obligatoire
<b>SHIFT N-BIT DATA LEFT</b>  NSFL @NSFL 578	 <p>NSFL(578) D C N</p> <p>D : Mot de départ pour le décalage C : Bit de départ N : Longueur du décalage</p>	Décale le nombre de bits spécifié vers la gauche.	Sortie Obligatoire
<b>SHIFT N-BIT DATA RIGHT</b>  NSFR @NSFR 579	 <p>NSFR(579) D C N</p> <p>D : Mot de départ pour le décalage C : Bit de départ N : Longueur du décalage</p>	Décale le nombre de bits spécifié vers la droite.	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution			
<b>SHIFT N-BITS LEFT</b>  NASL @NASL 580	<table border="1" style="width: 100%; text-align: center;"> <tr><td>NASL(580)</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p>D : Mot à décaler C : Mot de contrôle</p>	NASL(580)	D	C	Décale les 16 bits de données de mot spécifiées vers la gauche du nombre de bits spécifié. 	Sortie Obligatoire
NASL(580)						
D						
C						
<b>DOUBLE SHIFT N-BITS LEFT</b>  NSLL @NSLL 582	<table border="1" style="width: 100%; text-align: center;"> <tr><td>NSLL(582)</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p>D : Mot à décaler C : Mot de contrôle</p>	NSLL(582)	D	C	Décale les 32 bits de données de mot vers la gauche du nombre de bits spécifié. 	Sortie Obligatoire
NSLL(582)						
D						
C						
<b>SHIFT N-BITS RIGHT</b>  NASR @NASR 581	<table border="1" style="width: 100%; text-align: center;"> <tr><td>NASR(581)</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p>D : Mot à décaler C : Mot de contrôle</p>	NASR(581)	D	C	Décale les 16 bits de données de mot spécifiées vers la droite du nombre de bits spécifié. 	Sortie Obligatoire
NASR(581)						
D						
C						
<b>DOUBLE SHIFT N-BITS RIGHT</b>  NSRL @NSRL 583	<table border="1" style="width: 100%; text-align: center;"> <tr><td>NSRL(583)</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p>D : Mot à décaler C : Mot de contrôle</p>	NSRL(583)	D	C	Décale les 32 bits de données de mot spécifiées vers la droite du nombre de bits spécifié. 	Sortie Obligatoire
NSRL(583)						
D						
C						

### 3-8 Instructions d'incrémentation/décrémentation

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>INCREMENT BINARY</b>  ++ @++ 590	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">++(590)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : Mot</p>	Incrémente le contenu (4 chiffres hexadécimaux) du mot spécifié de 1. $\boxed{\text{Wd}} + 1 \longrightarrow \boxed{\text{Wd}}$	Sortie Obligatoire
<b>DOUBLE INCREMENT BINARY</b>  ++L @++L 591	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">++L(591)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : Mot</p>	Incrémente le contenu (8 chiffres hexadécimaux) des mots spécifiés de 1. $\boxed{\text{Wd}+1} \quad \boxed{\text{Wd}} + 1 \longrightarrow \boxed{\text{Wd}+1} \quad \boxed{\text{Wd}}$	Sortie Obligatoire
<b>DECREMENT BINARY</b>  -- @-- 592	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">(592)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : Mot</p>	Décrémente le contenu (4 chiffres hexadécimaux) du mot spécifié de 1. $\boxed{\text{Wd}} - 1 \longrightarrow \boxed{\text{Wd}}$	Sortie Obligatoire
<b>DOUBLE DECREMENT BINARY</b>  --L @--L 593	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">L(593)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : 1er mot</p>	Incrémente le contenu (8 chiffres hexadécimaux) des mots spécifiés de 1. $\boxed{\text{Wd}+1} \quad \boxed{\text{Wd}} - 1 \longrightarrow \boxed{\text{Wd}+1} \quad \boxed{\text{Wd}}$	Sortie Obligatoire
<b>INCREMENT BCD</b>  ++B @++B 594	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">++B(594)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : Mot</p>	Incrémente le contenu (4 chiffres BCD) du mot spécifié de 1. $\boxed{\text{Wd}} + 1 \longrightarrow \boxed{\text{Wd}}$	Sortie Obligatoire
<b>DOUBLE INCREMENT BCD</b>  ++BL @++BL 595	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">++BL(595)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : 1er mot</p>	Incrémente le contenu (8 chiffres BCD) des mots spécifiés de 1. $\boxed{\text{Wd}+1} \quad \boxed{\text{Wd}} + 1 \longrightarrow \boxed{\text{Wd}+1} \quad \boxed{\text{Wd}}$	Sortie Obligatoire
<b>DECREMENT BCD</b>  --B @--B 596	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">B(596)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : Mot</p>	Décrémente le contenu (4 chiffres BCD) du mot spécifié de 1. $\boxed{\text{Wd}} - 1 \longrightarrow \boxed{\text{Wd}}$	Sortie Obligatoire
<b>DOUBLE DECREMENT BCD</b>  --BL @--BL 597	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">BL(597)</div> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Wd</div> <p>Wd : 1er mot</p>	Décrémente le contenu (8 chiffres BCD) des mots spécifiés de 1. $\boxed{\text{Wd}+1} \quad \boxed{\text{Wd}} - 1 \longrightarrow \boxed{\text{Wd}+1} \quad \boxed{\text{Wd}}$	Sortie Obligatoire

### 3-9 Instructions mathématiques à symboles

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution																									
<b>SIGNED BINARY ADD WITHOUT CARRY</b>  + @+ 400	<table border="1"> <tr><td>+(400)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : Mot 1er terme addition Ad : Mot 2ème terme addition R : Mot de résultat</p>	+(400)	Au	Ad	R	<p>Ajoute 4 chiffres de données hexadécimales (mot unique) et/ou des constantes.</p> <table style="margin-left: auto; margin-right: auto;"> <tr><td>Au</td><td>(Binaire signé)</td></tr> <tr><td>+</td><td></td></tr> <tr><td>Ad</td><td>(Binaire signé)</td></tr> <tr><td colspan="2" style="text-align: center;">+</td></tr> <tr><td>CY</td><td>R (Binaire signé)</td></tr> </table> <p>CY passera à ON s'il existe une retenue.</p>	Au	(Binaire signé)	+		Ad	(Binaire signé)	+		CY	R (Binaire signé)	Sortie Obligatoire											
+(400)																												
Au																												
Ad																												
R																												
Au	(Binaire signé)																											
+																												
Ad	(Binaire signé)																											
+																												
CY	R (Binaire signé)																											
<b>DOUBLE SIGNED BINARY ADD WITHOUT CARRY</b>  +L @+L 401	<table border="1"> <tr><td>+L(401)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : 1er mot 1er terme de l'addition Ad : 1er mot 2ème terme addition R : 1er mot de résultat</p>	+L(401)	Au	Ad	R	<p>Ajoute 8 chiffres de données hexadécimales (mot double) et/ou des constantes.</p> <table style="margin-left: auto; margin-right: auto;"> <tr><td>Au+1</td><td>Au</td><td>(Binaire signé)</td></tr> <tr><td>+</td><td></td><td></td></tr> <tr><td>Ad+1</td><td>Ad</td><td>(Binaire signé)</td></tr> <tr><td colspan="3" style="text-align: center;">+</td></tr> <tr><td>CY</td><td>R+1</td><td>R (Binaire signé)</td></tr> </table> <p>CY passera à ON s'il existe une retenue.</p>	Au+1	Au	(Binaire signé)	+			Ad+1	Ad	(Binaire signé)	+			CY	R+1	R (Binaire signé)	Sortie Obligatoire						
+L(401)																												
Au																												
Ad																												
R																												
Au+1	Au	(Binaire signé)																										
+																												
Ad+1	Ad	(Binaire signé)																										
+																												
CY	R+1	R (Binaire signé)																										
<b>SIGNED BINARY ADD WITH CARRY</b>  +C @+C 402	<table border="1"> <tr><td>+C(402)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : Mot 1er terme addition Ad : Mot 2ème terme addition R : Mot de résultat</p>	+C(402)	Au	Ad	R	<p>Ajoute 4 chiffres de données hexadécimales (mot unique) et/ou des constantes avec le drapeau de passage (CY).</p> <table style="margin-left: auto; margin-right: auto;"> <tr><td>Au</td><td>(Binaire signé)</td></tr> <tr><td>+</td><td></td></tr> <tr><td>Ad</td><td>(Binaire signé)</td></tr> <tr><td colspan="2" style="text-align: center;">+</td></tr> <tr><td>CY</td><td></td></tr> <tr><td colspan="2" style="text-align: center;">+</td></tr> <tr><td>CY</td><td>R (Binaire signé)</td></tr> </table> <p>CY passera à ON s'il existe une retenue.</p>	Au	(Binaire signé)	+		Ad	(Binaire signé)	+		CY		+		CY	R (Binaire signé)	Sortie Obligatoire							
+C(402)																												
Au																												
Ad																												
R																												
Au	(Binaire signé)																											
+																												
Ad	(Binaire signé)																											
+																												
CY																												
+																												
CY	R (Binaire signé)																											
<b>DOUBLE SIGNED BINARY ADD WITH CARRY</b>  +CL @+CL 403	<table border="1"> <tr><td>+CL(403)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : 1er mot 1er terme de l'addition Ad : 1er mot 2ème terme addition R : 1er mot de résultat</p>	+CL(403)	Au	Ad	R	<p>Ajoute 8 chiffres de données hexadécimales (mot double) et/ou des constantes avec le drapeau de passage (CY).</p> <table style="margin-left: auto; margin-right: auto;"> <tr><td>Au+1</td><td>Au</td><td>(Binaire signé)</td></tr> <tr><td>+</td><td></td><td></td></tr> <tr><td>Ad+1</td><td>Ad</td><td>(Binaire signé)</td></tr> <tr><td colspan="3" style="text-align: center;">+</td></tr> <tr><td>CY</td><td></td><td></td></tr> <tr><td colspan="3" style="text-align: center;">+</td></tr> <tr><td>CY</td><td>R+1</td><td>R (Binaire signé)</td></tr> </table> <p>CY passera à ON s'il existe une retenue.</p>	Au+1	Au	(Binaire signé)	+			Ad+1	Ad	(Binaire signé)	+			CY			+			CY	R+1	R (Binaire signé)	Sortie Obligatoire
+CL(403)																												
Au																												
Ad																												
R																												
Au+1	Au	(Binaire signé)																										
+																												
Ad+1	Ad	(Binaire signé)																										
+																												
CY																												
+																												
CY	R+1	R (Binaire signé)																										
<b>BCD ADD WITHOUT CARRY</b>  +B @+B 404	<table border="1"> <tr><td>+B(404)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : Mot 1er terme addition Ad : Mot 2ème terme addition R : Mot de résultat</p>	+B(404)	Au	Ad	R	<p>Ajoute 4 chiffres de données BCD (mot unique) et/ou des constantes.</p> <table style="margin-left: auto; margin-right: auto;"> <tr><td>Au</td><td>(BCD)</td></tr> <tr><td>+</td><td></td></tr> <tr><td>Ad</td><td>(BCD)</td></tr> <tr><td colspan="2" style="text-align: center;">+</td></tr> <tr><td>CY</td><td>R (BCD)</td></tr> </table> <p>CY passera à ON s'il existe une retenue.</p>	Au	(BCD)	+		Ad	(BCD)	+		CY	R (BCD)	Sortie Obligatoire											
+B(404)																												
Au																												
Ad																												
R																												
Au	(BCD)																											
+																												
Ad	(BCD)																											
+																												
CY	R (BCD)																											

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<b>DOUBLE BCD ADD WITHOUT CARRY</b>  +BL @+BL 405	<table border="1"> <tr><td>+BL(405)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : 1er mot 1er terme de l'addition Ad : 1er mot 2ème terme addition R : 1er mot de résultat</p>	+BL(405)	Au	Ad	R	Ajoute 8 chiffres de données BCD (mot double) et/ou des constantes. $  \begin{array}{r}  \begin{array}{ c c } \hline \text{Au+1} & \text{Au} \\ \hline \end{array} \text{ (BCD)} \\  + \begin{array}{ c c } \hline \text{Ad+1} & \text{Ad} \\ \hline \end{array} \text{ (BCD)} \\  \hline  \begin{array}{ c c c } \hline \text{CY} & \text{R+1} & \text{R} \\ \hline \end{array} \text{ (BCD)}  \end{array}  $ CY passera à ON s'il existe une retenue.	Sortie Obligatoire
+BL(405)							
Au							
Ad							
R							
<b>BCD ADD WITH CARRY</b>  +BC @+BC 406	<table border="1"> <tr><td>+BC(406)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : Mot 1er terme addition Ad : Mot 2ème terme addition R : Mot de résultat</p>	+BC(406)	Au	Ad	R	Ajoute 4 chiffres de données BCD (mot unique) et/ou des constantes avec le drapeau de passage (CY). $  \begin{array}{r}  \begin{array}{ c } \hline \text{Au} \\ \hline \end{array} \text{ (BCD)} \\  + \begin{array}{ c } \hline \text{Ad} \\ \hline \end{array} \text{ (BCD)} \\  + \begin{array}{ c } \hline \text{CY} \\ \hline \end{array} \\  \hline  \begin{array}{ c c } \hline \text{CY} & \text{R} \\ \hline \end{array} \text{ (BCD)}  \end{array}  $ CY passera à ON s'il existe une retenue.	Sortie Obligatoire
+BC(406)							
Au							
Ad							
R							
<b>DOUBLE BCD ADD WITH CARRY</b>  +BCL @+BCL 407	<table border="1"> <tr><td>+BCL(407)</td></tr> <tr><td>Au</td></tr> <tr><td>Ad</td></tr> <tr><td>R</td></tr> </table> <p>Au : 1er mot 1er terme de l'addition Ad : 1er mot 2ème terme addition R : 1er mot de résultat</p>	+BCL(407)	Au	Ad	R	Ajoute 8 chiffres de données BCD (mot double) et/ou des constantes avec le drapeau de passage (CY). $  \begin{array}{r}  \begin{array}{ c c } \hline \text{Au+1} & \text{Au} \\ \hline \end{array} \text{ (BCD)} \\  + \begin{array}{ c c } \hline \text{Ad+1} & \text{Ad} \\ \hline \end{array} \text{ (BCD)} \\  + \begin{array}{ c } \hline \text{CY} \\ \hline \end{array} \\  \hline  \begin{array}{ c c c } \hline \text{CY} & \text{R+1} & \text{R} \\ \hline \end{array} \text{ (BCD)}  \end{array}  $ CY passera à ON s'il existe une retenue.	Sortie Obligatoire
+BCL(407)							
Au							
Ad							
R							
<b>SIGNED BINARY SUBTRACT WITHOUT CARRY</b>  - @- 410	<table border="1"> <tr><td>(410)</td></tr> <tr><td>Mi</td></tr> <tr><td>Su</td></tr> <tr><td>R</td></tr> </table> <p>Mi : Mot à soustraire Su : Mot soustracteur R : Mot de résultat</p>	(410)	Mi	Su	R	Ajoute 4 chiffres de données hexadécimales (mot unique) et/ou des constantes. $  \begin{array}{r}  \begin{array}{ c } \hline \text{Mi} \\ \hline \end{array} \text{ (Binaire signé)} \\  - \begin{array}{ c } \hline \text{Su} \\ \hline \end{array} \text{ (Binaire signé)} \\  \hline  \begin{array}{ c c } \hline \text{CY} & \text{R} \\ \hline \end{array} \text{ (Binaire signé)}  \end{array}  $ CY passera à ON s'il existe un emprunt.	Sortie Obligatoire
(410)							
Mi							
Su							
R							
<b>DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY</b>  -L @-L 411	<table border="1"> <tr><td>L(411)</td></tr> <tr><td>Mi</td></tr> <tr><td>Su</td></tr> <tr><td>R</td></tr> </table> <p>Mi : Mot à soustraire Su : Mot soustracteur R : Mot de résultat</p>	L(411)	Mi	Su	R	Ajoute 8 chiffres de données hexadécimales (mot double) et/ou des constantes. $  \begin{array}{r}  \begin{array}{ c c } \hline \text{Mi+1} & \text{Mi} \\ \hline \end{array} \text{ (Binaire signé)} \\  - \begin{array}{ c c } \hline \text{Su+1} & \text{Su} \\ \hline \end{array} \text{ (Binaire signé)} \\  \hline  \begin{array}{ c c c } \hline \text{CY} & \text{R+1} & \text{R} \\ \hline \end{array} \text{ (Binaire signé)}  \end{array}  $ CY passera à ON s'il existe un emprunt.	Sortie Obligatoire
L(411)							
Mi							
Su							
R							

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>SIGNED BINARY SUBTRACT WITH CARRY</b> -C @-C 412	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                         C(412)  <hr/>                         Mi  <hr/>                         Su  <hr/>                         R                     </div> <p>Mi : Mot à soustraire                      Su : Mot soustracteur                      R : Mot de résultat</p>	Ajoute 4 chiffres de données hexadécimales (mot unique) et/ou des constantes avec le drapeau de passage (CY). $  \begin{array}{r}  \text{Mi} \quad (\text{Binaire signé}) \\  \text{Su} \quad (\text{Binaire signé}) \\  \hline  \text{CY} \\  \text{CY} \quad \text{R} \quad (\text{Binaire signé})  \end{array}  $ <p>CY passera à ON s'il existe un emprunt.</p>	Sortie Obligatoire
<b>DOUBLE SIGNED BINARY SUBTRACT WITH CARRY</b> -CL @-CL 413	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                         CL(413)  <hr/>                         Mi  <hr/>                         Su  <hr/>                         R                     </div> <p>Mi : Mot à soustraire                      Su : Mot soustracteur                      R : Mot de résultat</p>	Ajoute 8 chiffres de données hexadécimales (mot double) et/ou des constantes avec le drapeau de passage (CY). $  \begin{array}{r}  \text{Mi+1} \quad \text{Mi} \quad (\text{Binaire signé}) \\  \text{Su+1} \quad \text{Su} \quad (\text{Binaire signé}) \\  \hline  \text{CY} \\  \text{CY} \quad \text{R+1} \quad \text{R} \quad (\text{Binaire signé})  \end{array}  $ <p>CY passera à ON s'il existe un emprunt.</p>	Sortie Obligatoire
<b>BCD SUBTRACT WITHOUT CARRY</b> -B @-B 414	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                         B(414)  <hr/>                         Mi  <hr/>                         Su  <hr/>                         R                     </div> <p>Mi : Mot à soustraire                      Su : Mot soustracteur                      R : Mot de résultat</p>	Ajoute 4 chiffres de données BCD (mot unique) et/ou des constantes. $  \begin{array}{r}  \text{Mi} \quad (\text{BCD}) \\  \text{Su} \quad (\text{BCD}) \\  \hline  \text{CY} \quad \text{R} \quad (\text{BCD})  \end{array}  $ <p>CY passera à ON s'il existe une retenue.</p>	Sortie Obligatoire
<b>DOUBLE BCD SUBTRACT WITHOUT CARRY</b> -BL @-BL 415	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                         BL(415)  <hr/>                         Mi  <hr/>                         Su  <hr/>                         R                     </div> <p>Mi : 1er mot à soustraire                      Su : 1er mot soustracteur                      R : 1er mot de résultat</p>	Ajoute 8 chiffres de données BCD (mot double) et/ou des constantes. $  \begin{array}{r}  \text{Mi +1} \quad \text{Mi} \quad (\text{BCD}) \\  \text{Su+1} \quad \text{Su} \quad (\text{BCD}) \\  \hline  \text{CY} \quad \text{R+1} \quad \text{R} \quad (\text{BCD})  \end{array}  $ <p>CY passera à ON s'il existe un emprunt.</p>	Sortie Obligatoire
<b>BCD SUBTRACT WITH CARRY</b> -BC @-BC 416	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                         BC(416)  <hr/>                         Mi  <hr/>                         Su  <hr/>                         R                     </div> <p>Mi : Mot à soustraire                      Su : Mot soustracteur                      R : Mot de résultat</p>	Ajoute 4 chiffres de données BCD (mot unique) et/ou des constantes avec le drapeau de passage (CY). $  \begin{array}{r}  \text{Mi} \quad (\text{BCD}) \\  \text{Su} \quad (\text{BCD}) \\  \hline  \text{CY} \\  \text{CY} \quad \text{R} \quad (\text{BCD})  \end{array}  $ <p>CY passera à ON s'il existe un emprunt.</p>	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution																	
<b>DOUBLE BCD SUBTRACT WITH CARRY</b> -BCL @-BCL 417	<table border="1"> <tr><td>BCL(417)</td></tr> <tr><td>Mi</td></tr> <tr><td>Su</td></tr> <tr><td>R</td></tr> </table> <p>Mi : 1er mot à soustraire                      Su : 1er mot soustracteur                      R : 1er mot de résultat</p>	BCL(417)	Mi	Su	R	Ajoute 8 chiffres de données BCD (mot double) et/ou des constantes avec le drapeau de passage (CY).  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">Mi + 1</td> <td style="border: 1px solid black; padding: 2px;">Mi</td> <td>(BCD)</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">Su + 1</td> <td style="border: 1px solid black; padding: 2px;">Su</td> <td>(BCD)</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">-</td> <td style="border: 1px solid black; padding: 2px;">CY</td> <td></td> </tr> </table> CY passera à ON s'il existe un emprunt.  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">CY</td> <td style="border: 1px solid black; padding: 2px;">R + 1</td> <td style="border: 1px solid black; padding: 2px;">R</td> <td>(BCD)</td> </tr> </table>	Mi + 1	Mi	(BCD)	Su + 1	Su	(BCD)	-	CY		CY	R + 1	R	(BCD)	Sortie Obligatoire
BCL(417)																				
Mi																				
Su																				
R																				
Mi + 1	Mi	(BCD)																		
Su + 1	Su	(BCD)																		
-	CY																			
CY	R + 1	R	(BCD)																	
<b>SIGNED BINARY MULTIPLY</b> * @* 420	<table border="1"> <tr><td>*(420)</td></tr> <tr><td>Md</td></tr> <tr><td>Mr</td></tr> <tr><td>R</td></tr> </table> <p>Md : Mot multiplicande                      Mr : Mot multiplicateur                      R : Mot de résultat</p>	*(420)	Md	Mr	R	Multiplie 4 chiffres de données hexadécimales signées et/ou des constantes.  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">Md</td> <td>(Binaire signé)</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">x</td> <td style="border: 1px solid black; padding: 2px;">Mr</td> <td>(Binaire signé)</td> </tr> </table> <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">R + 1</td> <td style="border: 1px solid black; padding: 2px;">R</td> <td>(Binaire signé)</td> </tr> </table>	Md	(Binaire signé)	x	Mr	(Binaire signé)	R + 1	R	(Binaire signé)	Sortie Obligatoire					
*(420)																				
Md																				
Mr																				
R																				
Md	(Binaire signé)																			
x	Mr	(Binaire signé)																		
R + 1	R	(Binaire signé)																		
<b>DOUBLE SIGNED BINARY MULTIPLY</b> *L @*L 421	<table border="1"> <tr><td>*L(421)</td></tr> <tr><td>Md</td></tr> <tr><td>Mr</td></tr> <tr><td>R</td></tr> </table> <p>Md : 1er mot multiplicande                      Mr : 1er mot multiplicateur                      R : 1er mot de résultat</p>	*L(421)	Md	Mr	R	Multiplie 8 chiffres de données hexadécimales signées et/ou des constantes.  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">Md + 1</td> <td style="border: 1px solid black; padding: 2px;">Md</td> <td>(Binaire signé)</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">x</td> <td style="border: 1px solid black; padding: 2px;">Mr + 1</td> <td style="border: 1px solid black; padding: 2px;">Mr</td> <td>(Binaire signé)</td> </tr> </table> <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">R + 3</td> <td style="border: 1px solid black; padding: 2px;">R + 2</td> <td style="border: 1px solid black; padding: 2px;">R + 1</td> <td style="border: 1px solid black; padding: 2px;">R</td> <td>(Binaire signé)</td> </tr> </table>	Md + 1	Md	(Binaire signé)	x	Mr + 1	Mr	(Binaire signé)	R + 3	R + 2	R + 1	R	(Binaire signé)	Sortie Obligatoire	
*L(421)																				
Md																				
Mr																				
R																				
Md + 1	Md	(Binaire signé)																		
x	Mr + 1	Mr	(Binaire signé)																	
R + 3	R + 2	R + 1	R	(Binaire signé)																
<b>UNSIGNED BINARY MULTIPLY</b> *U @*U 422	<table border="1"> <tr><td>*U(422)</td></tr> <tr><td>Md</td></tr> <tr><td>Mr</td></tr> <tr><td>R</td></tr> </table> <p>Md : Mot multiplicande                      Mr : Mot multiplicateur                      R : Mot de résultat</p>	*U(422)	Md	Mr	R	Multiplie 4 chiffres de données hexadécimales non signées et/ou des constantes.  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">Md</td> <td>(Binaire non signé)</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">x</td> <td style="border: 1px solid black; padding: 2px;">Mr</td> <td>(Binaire non signé)</td> </tr> </table> <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">R + 1</td> <td style="border: 1px solid black; padding: 2px;">R</td> <td>(Binaire non signé)</td> </tr> </table>	Md	(Binaire non signé)	x	Mr	(Binaire non signé)	R + 1	R	(Binaire non signé)	Sortie Obligatoire					
*U(422)																				
Md																				
Mr																				
R																				
Md	(Binaire non signé)																			
x	Mr	(Binaire non signé)																		
R + 1	R	(Binaire non signé)																		
<b>DOUBLE UNSIGNED BINARY MULTIPLY</b> *UL @*UL 423	<table border="1"> <tr><td>*UL(423)</td></tr> <tr><td>Md</td></tr> <tr><td>Mr</td></tr> <tr><td>R</td></tr> </table> <p>Md : 1er mot multiplicande                      Mr : 1er mot multiplicateur                      R : 1er mot de résultat</p>	*UL(423)	Md	Mr	R	Multiplie 8 chiffres de données hexadécimales non signées et/ou des constantes.  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">Md + 1</td> <td style="border: 1px solid black; padding: 2px;">Md</td> <td>(Binaire non signé)</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">x</td> <td style="border: 1px solid black; padding: 2px;">Mr + 1</td> <td style="border: 1px solid black; padding: 2px;">Mr</td> <td>(Binaire non signé)</td> </tr> </table> <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">R + 3</td> <td style="border: 1px solid black; padding: 2px;">R + 2</td> <td style="border: 1px solid black; padding: 2px;">R + 1</td> <td style="border: 1px solid black; padding: 2px;">R</td> <td>(Binaire non signé)</td> </tr> </table>	Md + 1	Md	(Binaire non signé)	x	Mr + 1	Mr	(Binaire non signé)	R + 3	R + 2	R + 1	R	(Binaire non signé)	Sortie Obligatoire	
*UL(423)																				
Md																				
Mr																				
R																				
Md + 1	Md	(Binaire non signé)																		
x	Mr + 1	Mr	(Binaire non signé)																	
R + 3	R + 2	R + 1	R	(Binaire non signé)																

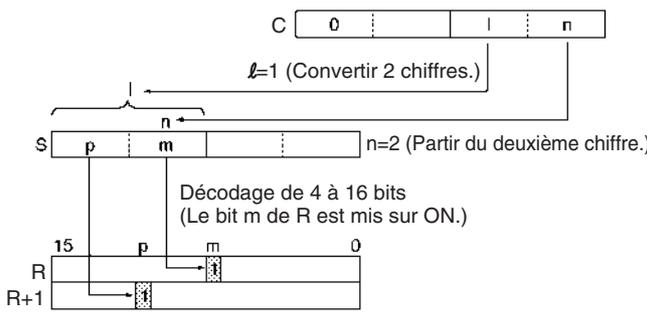
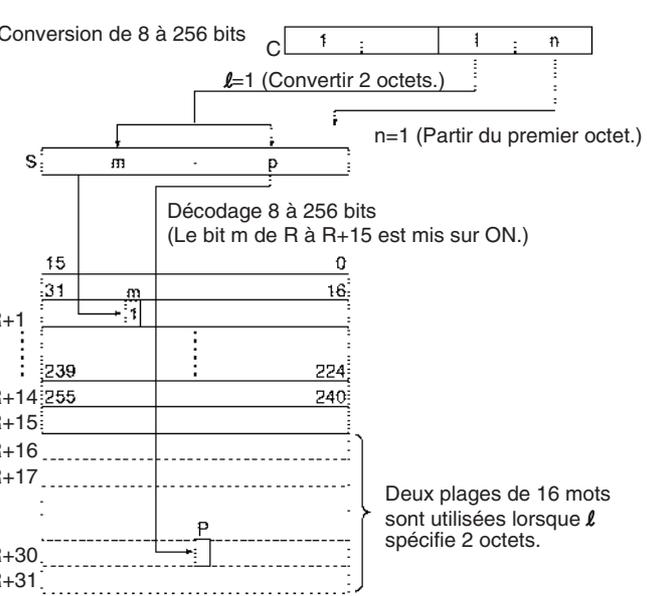
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>BCD MULTIPLY</b> *B @*B 424	*B(424) Md Mr R	Multiplie 4 chiffres de données BCD (mot unique) et/ou des constantes. $\begin{array}{r} \text{Md} \text{ (BCD)} \\ \times \text{Mr} \text{ (BCD)} \\ \hline \text{R+1} \quad \text{R} \text{ (BCD)} \end{array}$	Sortie Obligatoire
<b>DOUBLE BCD MULTIPLY</b> *BL @*BL 425	*BL(425) Md Mr R	Multiplie 8 chiffres de données BCD (mot double) et/ou des constantes. $\begin{array}{r} \text{Md+1} \quad \text{Md} \text{ (BCD)} \\ \times \text{Mr+1} \quad \text{Mr} \text{ (BCD)} \\ \hline \text{R+3} \quad \text{R+2} \quad \text{R+1} \quad \text{R} \text{ (BCD)} \end{array}$	Sortie Obligatoire
<b>SIGNED BINARY DIVIDE</b> / @/ 430	/(430) Dd Dr R	Ajoute 4 chiffres de données hexadécimales signées (mot unique) et/ou des constantes. $\begin{array}{r} \text{Dd} \text{ (Binaire signé)} \\ \div \text{Dr} \text{ (Binaire signé)} \\ \hline \text{R+1} \quad \text{R} \text{ (Binaire signé)} \end{array}$ Reste      Quotient	Sortie Obligatoire
<b>DOUBLE SIGNED BINARY DIVIDE</b> /L @/L 431	/L(431) Dd Dr R	Ajoute 8 chiffres de données hexadécimales signées (mot double) et/ou des constantes. $\begin{array}{r} \text{Dd+1} \quad \text{Dd} \text{ (Binaire signé)} \\ \div \text{Dr+1} \quad \text{Dr} \text{ (Binaire signé)} \\ \hline \text{R+3} \quad \text{R+2} \quad \text{R+1} \quad \text{R} \text{ (Binaire signé)} \end{array}$ Reste      Quotient	Sortie Obligatoire
<b>UNSIGNED BINARY DIVIDE</b> /U @/U 432	/U(432) Dd Dr R	Ajoute 4 chiffres de données hexadécimales non signées (mot unique) et/ou des constantes. $\begin{array}{r} \text{Dd} \text{ (Binaire non signé)} \\ \div \text{Dr} \text{ (Binaire non signé)} \\ \hline \text{R+1} \quad \text{R} \text{ (Binaire non signé)} \end{array}$ Reste      Quotient	Sortie Obligatoire

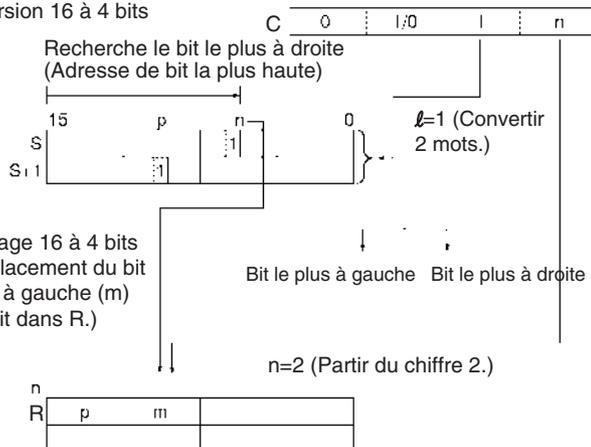
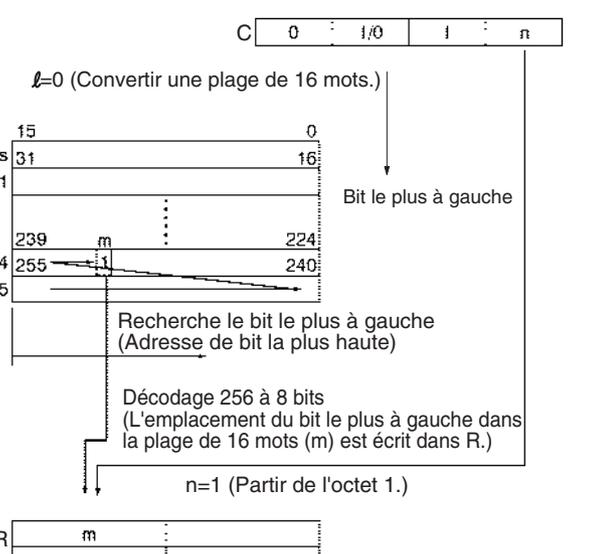
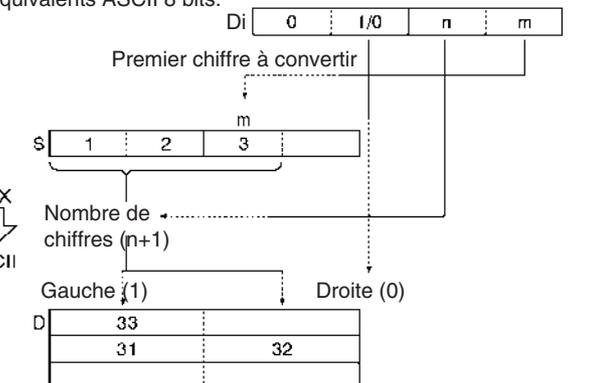
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DOUBLE UNSIGNED BINARY DIVIDE</b> /UL @/UL 433	/UL(433) Dd Dr R	Divise 8 chiffres de données hexadécimales non signées (mot double) et/ou des constantes. $\begin{array}{r} \text{Dd} + 1 \quad \text{Dd} \\ \hline \text{Dr} + 1 \quad \text{Dr} \\ \hline \text{R} + 3 \quad \text{R} + 2 \quad \text{R} + 1 \quad \text{R} \end{array}$ (Binaire non signé) ÷ (Binaire non signé) Reste                      Quotient	Sortie Obligatoire
<b>BCD DIVIDE</b> /B @/B 434	/B(434) Dd Dr R	Divise 4 chiffres de données BCD (mot unique) et/ou des constantes. $\begin{array}{r} \text{Dd} \\ \hline \text{Dr} \\ \hline \text{R} + 1 \quad \text{R} \end{array}$ (BCD) ÷ (BCD) (BCD) Reste                      Quotient	Sortie Obligatoire
<b>DOUBLE BCD DIVIDE</b> /BL @/BL 435	/BL(435) Dd Dr R	Divise 8 chiffres de données BCD (mot double) et/ou des constantes. $\begin{array}{r} \text{Dd} + 1 \quad \text{Dd} \\ \hline \text{Dr} + 1 \quad \text{Dr} \\ \hline \text{R} + 3 \quad \text{R} + 2 \quad \text{R} + 1 \quad \text{R} \end{array}$ (BCD) ÷ (BCD) (BCD) Reste                      Quotient	Sortie Obligatoire

### 3-10 Instructions de conversion

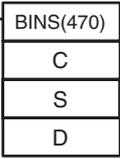
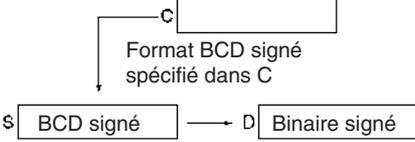
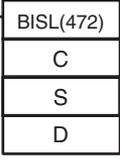
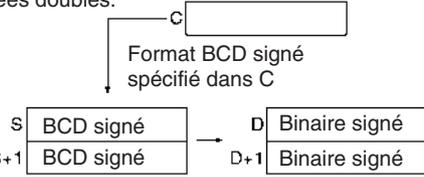
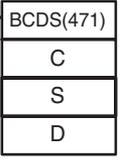
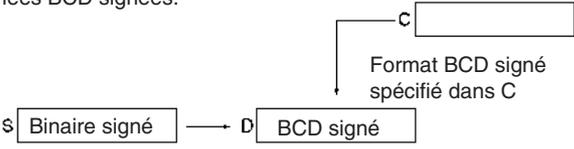
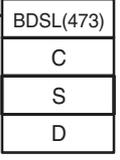
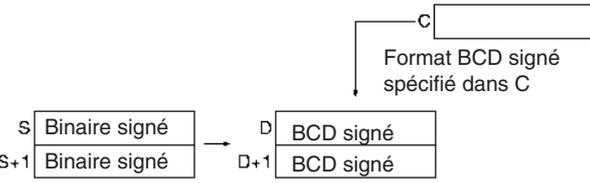
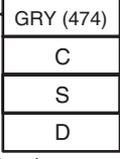
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>BCD-TO-BINARY</b> BIN @BIN 023	BIN(023) S R	Convertit des données BCD en données binaires. $S \text{ (BCD)} \rightarrow R \text{ (BIN)}$	Sortie Obligatoire
<b>DOUBLE BCD-TO-DOUBLE BINARY</b> BINL @BINL 058	BINL(058) S R	Convertit des données BCD à 8 chiffres en données hexadécimales à 8 chiffres (binaire 32 bits). $\begin{array}{r} S \text{ (BCD)} \rightarrow R \text{ (BIN)} \\ S+1 \text{ (BCD)} \rightarrow R+1 \text{ (BIN)} \end{array}$	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>BINARY-TO-BCD</b> BCD @BCD 024	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 BCD(024)  <hr/>                 S  <hr/>                 R             </div> <p>S : Mot source R : Mot de résultat</p>	Convertit un mot de données binaires en un mot de données BCD. $s \text{ (BIN)} \longrightarrow R \text{ (BCD)}$	Sortie Obligatoire
<b>DOUBLE BINARY-TO-DOUBLE BCD</b> BCDL @BCDL 059	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 BCDL(059)  <hr/>                 S  <hr/>                 R             </div> <p>S : 1er mot source R : 1er mot de résultat</p>	Convertit des données hexadécimales à 8 chiffres (binaire 32 bits) en données BCD à 8 chiffres. $s \text{ (BIN)} \longrightarrow R \text{ (BCD)}$ $s+1 \text{ (BIN)} \longrightarrow R+1 \text{ (BCD)}$	Sortie Obligatoire
<b>2'S COMPLEMENT</b> NEG @NEG 160	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 NEG(160)  <hr/>                 S  <hr/>                 R             </div> <p>S : Mot source R : Mot de résultat</p>	Calcule le complément 2 d'un mot de données hexadécimales. $\overline{(S)} \longrightarrow (R)$ <p style="text-align: center;">Complément 2 (Complément + 1)</p>	Sortie Obligatoire
<b>DOUBLE 2'S COMPLEMENT</b> NEGL @NEGL 161	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 NEGL(161)  <hr/>                 R             </div> <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule le complément 2 de deux mots de données hexadécimales. $\overline{(S+1, S)} \longrightarrow (R+1, R)$ <p style="text-align: center;">Complément 2 (Complément + 1)</p>	Sortie Obligatoire
<b>16-BIT TO 32-BIT SIGNED BINARY</b> SIGN @SIGN 600	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 SIGN(600)  <hr/>                 S  <hr/>                 R             </div> <p>S : Mot source R : 1er mot de résultat</p>	Développe une valeur binaire de 16 bits signée de façon à obtenir son équivalent 32 bits. <p>MSB = 1 : FFFF Hex MSB = 0 : 0000 Hex</p> <p>D+1      D D = Contenu de S</p>	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>DATA DECODER</b> MLPX @MLPX 076</p>	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MLPX(076)</p> <hr/> <p>S</p> <hr/> <p>C</p> <hr/> <p>R</p> </div> <p>S : Mot source C : Mot de contrôle R : 1er mot de résultat</p>	<p>Lit la valeur numérique dans le chiffre (ou l'octet) spécifié du mot source, met sur ON le bit correspondant dans le mot (ou la plage de 16 mots) de résultat et met sur OFF tous les autres bits dans le mot (ou la plage de 16 mots) de résultat.</p> <p>Conversion 4 à 16 bits</p>  <p>Conversion de 8 à 256 bits</p>  <p>Deux plages de 16 mots sont utilisées lorsque <math>l</math> spécifie 2 octets.</p>	<p>Sortie Obligatoire</p>

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<p><b>DATA ENCODER</b> DMPX @DMPX 077</p>	<table border="1" style="margin-left: 20px;"> <tr><td>DMPX(077)</td></tr> <tr><td>S</td></tr> <tr><td>R</td></tr> <tr><td>C</td></tr> </table> <p>S : 1er mot source R : Mot de résultat C : Mot de contrôle</p>	DMPX(077)	S	R	C	<p>Recherche l'emplacement du premier ou du dernier bit sur ON dans le mot (ou la plage de 16 mots) source et écrit cette valeur dans le chiffre (ou l'octet) spécifié du mot de résultat.</p> <p>Conversion 16 à 4 bits</p>  <p>Décodage 16 à 4 bits (L'emplacement du bit le plus à gauche (m) est écrit dans R.)</p> <p>Bit le plus à gauche Bit le plus à droite</p> <p>n=2 (Partir du chiffre 2.)</p> <p>Conversion 256 à 8 bits</p>  <p>l=0 (Convertir une plage de 16 mots.)</p> <p>Bit le plus à gauche</p> <p>Recherche le bit le plus à gauche (Adresse de bit la plus haute)</p> <p>Décodage 256 à 8 bits (L'emplacement du bit le plus à gauche dans la plage de 16 mots (m) est écrit dans R.)</p> <p>n=1 (Partir de l'octet 1.)</p>	<p>Sortie Obligatoire</p>
DMPX(077)							
S							
R							
C							
<p><b>ASCII CONVERT</b> ASC @ASC 086</p>	<table border="1" style="margin-left: 20px;"> <tr><td>ASC(086)</td></tr> <tr><td>S</td></tr> <tr><td>Di</td></tr> <tr><td>D</td></tr> </table> <p>S : Mot source Di : Désignateur de chiffres D : 1er mot de destination</p>	ASC(086)	S	Di	D	<p>Convertit les chiffres hexadécimaux 4 bits du mot source dans leurs équivalents ASCII 8 bits.</p>  <p>Di 0 I/O n m</p> <p>Premier chiffre à convertir</p> <p>Nombre de chiffres (n+1)</p> <p>HEX ↓ ASCII</p> <p>Gauche (1) Droite (0)</p>	<p>Sortie Obligatoire</p>
ASC(086)							
S							
Di							
D							

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution																																																																																																																																																								
<p><b>ASCII TO HEX</b> HEX @HEX 162</p>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">                 HEX(162)             </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px; width: 100%;">S</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px; width: 100%;">Di</div> <div style="border: 1px solid black; padding: 2px; width: 100%;">D</div> <p>S : 1er mot source Di : Désignateur de chiffres D : Mot de destination</p>	<p>Convertit jusqu'à 4 octets de données ASCII du mot source dans leurs équivalents hexadécimaux et écrit ces chiffres dans le mot de destination spécifié.</p> <p style="text-align: center;">C :0021</p> <div style="text-align: center;"> <table border="1" style="margin: 0 auto;"> <tr> <td style="padding: 2px;">Di</td> <td style="padding: 2px;">0</td> <td style="padding: 2px;">0/1</td> <td style="padding: 2px;">n</td> <td style="padding: 2px;">m</td> </tr> </table> <p>Premier octet à convertir</p> <table border="1" style="margin: 0 auto;"> <tr> <td style="padding: 2px;">S</td> <td style="padding: 2px;">33</td> <td style="padding: 2px;">32</td> </tr> <tr> <td style="padding: 2px;">S+1</td> <td style="padding: 2px;"></td> <td style="padding: 2px;">34</td> </tr> </table> <p>Gauche (1)      Droite (0)</p> <p>Nombre de chiffres (n+1)</p> <p>ASCII ↓ HEX</p> <table border="1" style="margin: 0 auto;"> <tr> <td style="padding: 2px;">D</td> <td style="padding: 2px;">4</td> <td style="padding: 2px;">3</td> <td style="padding: 2px;">2</td> </tr> </table> <p>Premier chiffre à écrire</p> </div>	Di	0	0/1	n	m	S	33	32	S+1		34	D	4	3	2	<p>Sortie Obligatoire</p>																																																																																																																																									
Di	0	0/1	n	m																																																																																																																																																							
S	33	32																																																																																																																																																									
S+1		34																																																																																																																																																									
D	4	3	2																																																																																																																																																								
<p><b>COLUMN TO LINE</b> LINE @LINE 063</p>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">                 LINE(063)             </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px; width: 100%;">S</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px; width: 100%;">N</div> <div style="border: 1px solid black; padding: 2px; width: 100%;">D</div> <p>S : 1er mot source N : Numéro de bit D : Mot de destination</p>	<p>Convertit une colonne de bits d'une plage de 16 mots (le même numéro de bit dans 16 mots consécutifs) dans les 16 bits du mot de destination.</p> <div style="text-align: center;"> <table border="1" style="margin: 0 auto;"> <tr> <td style="padding: 2px;">Bit 15</td> <td colspan="16"></td> <td style="padding: 2px;">Bit 00</td> </tr> <tr> <td></td> <td colspan="16" style="text-align: center;">N</td> <td></td> </tr> <tr> <td style="padding: 2px;">S</td> <td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">S+1</td> <td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">S+2</td> <td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">S+3</td> <td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">:</td> <td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td> <td></td> </tr> <tr> <td style="padding: 2px;">S+15</td> <td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td> <td></td> </tr> </table> <p>↓</p> <table border="1" style="margin: 0 auto;"> <tr> <td style="padding: 2px;">D</td> <td>0</td> <td>...</td> <td>0</td><td>1</td><td>1</td><td>1</td> </tr> </table> <p>Bit 15      Bit 00</p> </div>	Bit 15																	Bit 00		N																	S	0	0	0	1	1	1	1	0	0	0	1	0	0	0	0	1		S+1	1	1	0	1	0	0	1	0	0	1	1	1	0	0	0	1		S+2	0	0	0	1	1	0	1	1	0	0	1	0	0	1	1	1		S+3	1	0	0	0	0	0	1	1	0	0	0	0	0	1	1	1		:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:		S+15	0	1	1	0	0	0	0	1	1	0	0	0	1	0	1	0		D	0	...	0	1	1	1	<p>Sortie Obligatoire</p>	
Bit 15																	Bit 00																																																																																																																																										
	N																																																																																																																																																										
S	0	0	0	1	1	1	1	0	0	0	1	0	0	0	0	1																																																																																																																																											
S+1	1	1	0	1	0	0	1	0	0	1	1	1	0	0	0	1																																																																																																																																											
S+2	0	0	0	1	1	0	1	1	0	0	1	0	0	1	1	1																																																																																																																																											
S+3	1	0	0	0	0	0	1	1	0	0	0	0	0	1	1	1																																																																																																																																											
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:																																																																																																																																											
S+15	0	1	1	0	0	0	0	1	1	0	0	0	1	0	1	0																																																																																																																																											
D	0	...	0	1	1	1																																																																																																																																																					
<p><b>LINE TO COLUMN</b> COLM @COLM 064</p>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">                 COLM(064)             </div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px; width: 100%;">S</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px; width: 100%;">D</div> <div style="border: 1px solid black; padding: 2px; width: 100%;">N</div> <p>S : Mot source D : 1er mot de destination N : Numéro de bit</p>	<p>Convertit les 16 bits du mot source en une colonne de bits dans une plage de 16 mots de destination (le même numéro de bit dans 16 mots consécutifs).</p> <div style="text-align: center;"> <table border="1" style="margin: 0 auto;"> <tr> <td style="padding: 2px;">Bit 15</td> <td colspan="16"></td> <td style="padding: 2px;">Bit 00</td> </tr> <tr> <td style="padding: 2px;">S</td> <td>0</td> <td>...</td> <td>0</td><td>1</td><td>1</td><td>1</td> <td></td> </tr> </table> <p>↓</p> <table border="1" style="margin: 0 auto;"> <tr> <td style="padding: 2px;">Bit 15</td> <td colspan="16" style="text-align: center;">Bi</td> <td style="padding: 2px;">Bit 00</td> </tr> <tr> <td style="padding: 2px;">D</td> <td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">D+1</td> <td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">D+2</td> <td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">D+3</td> <td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td> <td></td> </tr> <tr> <td style="padding: 2px;">:</td> <td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td> <td></td> </tr> <tr> <td style="padding: 2px;">D+15</td> <td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td> <td></td> </tr> </table> </div>	Bit 15																	Bit 00	S	0	...	0	1	1	1		Bit 15	Bi																Bit 00	D	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0	1		D+1	1	1	0	1	0	0	1	0	0	1	1	1	0	0	0	1		D+2	0	0	0	1	1	0	1	1	0	0	1	0	0	1	1	1		D+3	1	0	0	0	0	0	1	1	0	0	0	0	0	1	1	1		:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:		D+15	0	1	1	1	0	0	0	1	1	0	0	0	1	0	1	0		<p>Sortie Obligatoire</p>
Bit 15																	Bit 00																																																																																																																																										
S	0	...	0	1	1	1																																																																																																																																																					
Bit 15	Bi																Bit 00																																																																																																																																										
D	0	0	0	0	1	1	1	0	0	0	1	0	0	0	0	1																																																																																																																																											
D+1	1	1	0	1	0	0	1	0	0	1	1	1	0	0	0	1																																																																																																																																											
D+2	0	0	0	1	1	0	1	1	0	0	1	0	0	1	1	1																																																																																																																																											
D+3	1	0	0	0	0	0	1	1	0	0	0	0	0	1	1	1																																																																																																																																											
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:																																																																																																																																											
D+15	0	1	1	1	0	0	0	1	1	0	0	0	1	0	1	0																																																																																																																																											

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>SIGNED BCD-TO-BINARY</b> BINS @BINS 470	 <p>C : Mot de contrôle S : Mot source D : Mot de destination</p>	Convertit un mot de données BCD signées en un mot de données binaires signées. 	Sortie Obligatoire
<b>DOUBLE SIGNED BCD-TO-BINARY</b> BISL @BISL 472	 <p>C : Mot de contrôle S : 1er mot source D : 1er mot de destination</p>	Convertit des données BCD signées doubles en données binaires signées doubles. 	Sortie Obligatoire
<b>SIGNED BINARY-TO-BCD</b> BCDS @BCDS 471	 <p>C : Mot de contrôle S : Mot source D : Mot de destination</p>	Convertit un mot de données binaires signées en un mot de données BCD signées. 	Sortie Obligatoire
<b>DOUBLE SIGNED BINARY-TO-BCD</b> BDSL @BDSL 473	 <p>C : Mot de contrôle S : 1er mot source D : 1er mot de destination</p>	Convertit des données binaires signées doubles en données BCD signées doubles. 	Sortie Obligatoire
<b>GRAY CODE CONVERSION</b> GRY 474 (UC série CS/CJ Ver. 2.0 ou supérieure uniquement, y compris les UC CS1-H, CJ1-H et CJ1M à partir du numéro de lot 030201)	 <p>C : Mot de contrôle S : Mot source D : 1er mot de destination</p>	Convertit les données de code Gray du mot spécifié en données binaires, données BCD ou données d'angle (°) à la résolution spécifiée.	Sortie Obligatoire

### 3-11 Instructions logiques

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution																			
<b>LOGICAL AND</b> ANDW @ANDW 034	<table border="1"> <tr><td>ANDW(034)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : Mot de résultat</p>	ANDW(034)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le AND logique des bits correspondants dans des mots uniques de données de mots et/ou des constantes.</p> $I_1 \cdot I_2 \rightarrow R$ <table border="1"> <thead> <tr> <th>I<sub>1</sub></th> <th>I<sub>2</sub></th> <th>R</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	I <sub>1</sub>	I <sub>2</sub>	R	1	1	1	1	0	0	0	1	0	0	0	0	Sortie Obligatoire
ANDW(034)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub>	I <sub>2</sub>	R																				
1	1	1																				
1	0	0																				
0	1	0																				
0	0	0																				
<b>DOUBLE LOGICAL AND</b> ANDL @ANDL 610	<table border="1"> <tr><td>ANDL(610)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : Mot de résultat</p>	ANDL(610)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le AND logique des bits correspondants dans des mots doubles de données de mots et/ou des constantes.</p> $(I_1, I_1+1) \cdot (I_2, I_2+1) \rightarrow (R, R+1)$ <table border="1"> <thead> <tr> <th>I<sub>1</sub>, I<sub>1</sub>+1</th> <th>I<sub>2</sub>, I<sub>2</sub>+1</th> <th>R, R+1</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	I <sub>1</sub> , I <sub>1</sub> +1	I <sub>2</sub> , I <sub>2</sub> +1	R, R+1	1	1	1	1	0	0	0	1	0	0	0	0	Sortie Obligatoire
ANDL(610)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub> , I <sub>1</sub> +1	I <sub>2</sub> , I <sub>2</sub> +1	R, R+1																				
1	1	1																				
1	0	0																				
0	1	0																				
0	0	0																				
<b>LOGICAL OR</b> ORW @ORW 035	<table border="1"> <tr><td>ORW(035)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : Mot de résultat</p>	ORW(035)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le OR logique des bits correspondants dans des mots uniques de données de mots et/ou des constantes.</p> $I_1 + I_2 \rightarrow R$ <table border="1"> <thead> <tr> <th>I<sub>1</sub></th> <th>I<sub>2</sub></th> <th>R</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	I <sub>1</sub>	I <sub>2</sub>	R	1	1	1	1	0	1	0	1	1	0	0	0	Sortie Obligatoire
ORW(035)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub>	I <sub>2</sub>	R																				
1	1	1																				
1	0	1																				
0	1	1																				
0	0	0																				
<b>DOUBLE LOGICAL OR</b> ORWL @ORWL 611	<table border="1"> <tr><td>ORWL(611)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : Mot de résultat</p>	ORWL(611)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le OR logique des bits correspondants dans des mots doubles de données de mots et/ou des constantes.</p> $(I_1, I_1+1) + (I_2, I_2+1) \rightarrow (R, R+1)$ <table border="1"> <thead> <tr> <th>I<sub>1</sub>, I<sub>1</sub>+1</th> <th>I<sub>2</sub>, I<sub>2</sub>+1</th> <th>R, R+1</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	I <sub>1</sub> , I <sub>1</sub> +1	I <sub>2</sub> , I <sub>2</sub> +1	R, R+1	1	1	1	1	0	1	0	1	1	0	0	0	Sortie Obligatoire
ORWL(611)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub> , I <sub>1</sub> +1	I <sub>2</sub> , I <sub>2</sub> +1	R, R+1																				
1	1	1																				
1	0	1																				
0	1	1																				
0	0	0																				
<b>EXCLUSIVE OR</b> XORW @XORW 036	<table border="1"> <tr><td>XORW(036)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : Mot de résultat</p>	XORW(036)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le OR exclusif logique des bits correspondants dans des mots uniques de données de mots et/ou des constantes.</p> $I_1 \cdot \bar{I}_2 + \bar{I}_1 \cdot I_2 \rightarrow R$ <table border="1"> <thead> <tr> <th>I<sub>1</sub></th> <th>I<sub>2</sub></th> <th>R</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	I <sub>1</sub>	I <sub>2</sub>	R	1	1	0	1	0	1	0	1	1	0	0	0	Sortie Obligatoire
XORW(036)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub>	I <sub>2</sub>	R																				
1	1	0																				
1	0	1																				
0	1	1																				
0	0	0																				

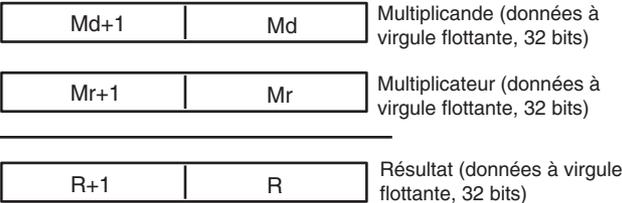
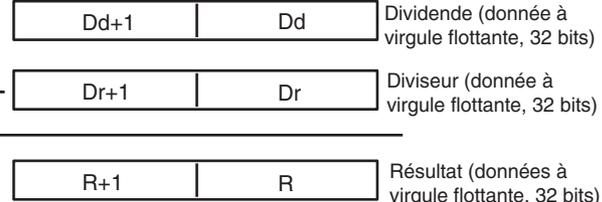
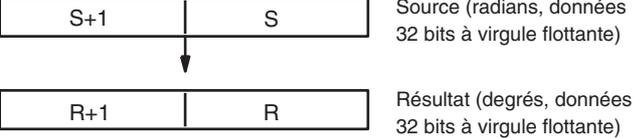
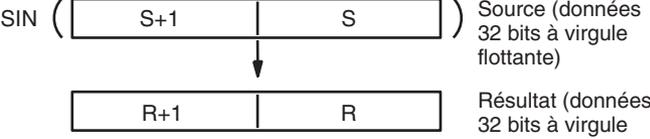
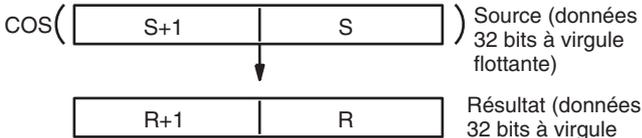
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution																			
<b>DOUBLE EXCLUSIVE OR</b> XORL @XORL 612	<table border="1"> <tr><td>XORL(612)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : Mot de résultat</p>	XORL(612)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le OR exclusif logique des bits correspondants dans des mots doubles de données de mots et/ou des constantes.</p> $(I_1.I_1+1). (I_2.I_2+1) + (I_1.I_1+1). (I_2.I_2+1) \rightarrow (R, R+1)$ <table border="1"> <thead> <tr> <th>I<sub>1</sub>.I<sub>1</sub>+1</th> <th>I<sub>2</sub>.I<sub>2</sub>+1</th> <th>R, R+1</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	I <sub>1</sub> .I <sub>1</sub> +1	I <sub>2</sub> .I <sub>2</sub> +1	R, R+1	1	1	0	1	0	1	0	1	1	0	0	0	Sortie Obligatoire
XORL(612)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub> .I <sub>1</sub> +1	I <sub>2</sub> .I <sub>2</sub> +1	R, R+1																				
1	1	0																				
1	0	1																				
0	1	1																				
0	0	0																				
<b>EXCLUSIVE NOR</b> XNRW @XNRW 037	<table border="1"> <tr><td>XNRW(037)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : Mot de résultat</p>	XNRW(037)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le NOR exclusif logique des mots uniques de données de mots et/ou des constantes correspondants.</p> $I_1.I_2 + \bar{I}_1.\bar{I}_2 \rightarrow R$ <table border="1"> <thead> <tr> <th>I<sub>1</sub></th> <th>I<sub>2</sub></th> <th>R</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </tbody> </table>	I <sub>1</sub>	I <sub>2</sub>	R	1	1	1	1	0	0	0	1	0	0	0	1	Sortie Obligatoire
XNRW(037)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub>	I <sub>2</sub>	R																				
1	1	1																				
1	0	0																				
0	1	0																				
0	0	1																				
<b>DOUBLE EXCLUSIVE NOR</b> XNRL @XNRL 613	<table border="1"> <tr><td>XNRL(613)</td></tr> <tr><td>I<sub>1</sub></td></tr> <tr><td>I<sub>2</sub></td></tr> <tr><td>R</td></tr> </table> <p>I<sub>1</sub> : Entrée 1 I<sub>2</sub> : Entrée 2 R : 1er mot de résultat</p>	XNRL(613)	I <sub>1</sub>	I <sub>2</sub>	R	<p>Effectue le NOR exclusif logique des bits correspondants dans des mots doubles de données de mots et/ou des constantes.</p> $(I_1.I_1+1). (I_2.I_2+1) + (\bar{I}_1.\bar{I}_1+1). (\bar{I}_2.\bar{I}_2+1) \rightarrow (R, R+1)$ <table border="1"> <thead> <tr> <th>I<sub>1</sub>.I<sub>1</sub>+1</th> <th>I<sub>2</sub>.I<sub>2</sub>+1</th> <th>R, R+1</th> </tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </tbody> </table>	I <sub>1</sub> .I <sub>1</sub> +1	I <sub>2</sub> .I <sub>2</sub> +1	R, R+1	1	1	1	1	0	0	0	1	0	0	0	1	Sortie Obligatoire
XNRL(613)																						
I <sub>1</sub>																						
I <sub>2</sub>																						
R																						
I <sub>1</sub> .I <sub>1</sub> +1	I <sub>2</sub> .I <sub>2</sub> +1	R, R+1																				
1	1	1																				
1	0	0																				
0	1	0																				
0	0	1																				
<b>COMPLEMENT</b> COM @COM 029	<table border="1"> <tr><td>COM(029)</td></tr> <tr><td>Wd</td></tr> </table> <p>Wd : Mot</p>	COM(029)	Wd	<p>Met sur OFF tous les bits sur ON et met sur ON tous les bits sur OFF dans Wd.  <math>Wd \rightarrow \bar{Wd} : 1 \rightarrow 0 \text{ et } 0 \rightarrow 1</math></p>	Sortie Obligatoire																	
COM(029)																						
Wd																						
<b>DOUBLE COMPLEMENT</b> COML @COML 614	<table border="1"> <tr><td>COML(614)</td></tr> <tr><td>Wd</td></tr> </table> <p>Wd : Mot</p>	COML(614)	Wd	<p>Met sur OFF tous les bits sur ON et met sur ON tous les bits sur OFF dans Wd et Wd+1.  <math>(Wd+1, Wd) \rightarrow (Wd+1, Wd)</math></p>	Sortie Obligatoire																	
COML(614)																						
Wd																						

### 3-12 Instructions mathématiques spéciales

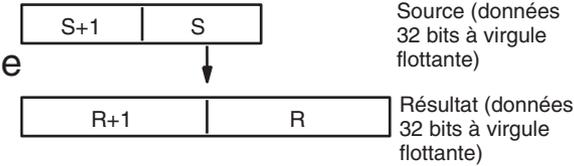
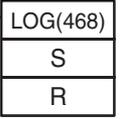
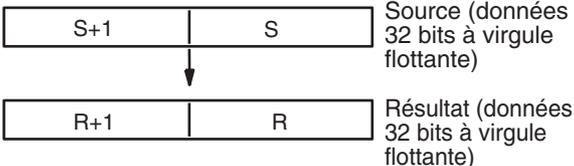
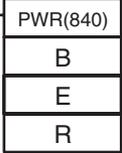
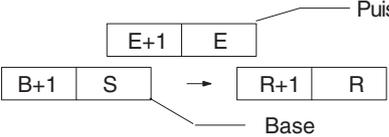
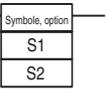
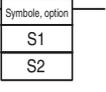
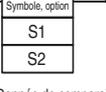
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>BINARY ROOT</b> ROT B @ ROT B 620	<p>S : 1er mot source R : Mot de résultat</p>	<p>Calcule la racine carrée du contenu binaire 32 bits des mots spécifiés et place la partie entière du résultat dans le mot de résultat spécifié.</p>	Sortie Obligatoire
<b>BCD SQUARE ROOT</b> ROOT @ ROOT 072	<p>S : 1er mot source R : Mot de résultat</p>	<p>Calcule la racine carrée d'un nombre binaire 8 chiffres et place la partie entière du résultat dans le mot de résultat spécifié.</p>	Sortie Obligatoire
<b>ARITHMETIC PROCESS</b> APR @ APR 069	<p>C : Mot de contrôle S : Donnée source R : Mot de résultat</p>	<p>Calcule le sinus, le cosinus ou une extrapolation linéaire des données sources. La fonction d'extrapolation linéaire permet d'établir une approximation d'une relation entre X et Y avec des segments de ligne.</p>	Sortie Obligatoire
<b>FLOATING POINT DIVIDE</b> FDIV @ FDIV 079	<p>Dd : 1er mot dividende Dr : 1er mot diviseur R : 1er mot de résultat</p>	<p>Divise un nombre de 7 chiffres à virgule flottante par un autre. Les nombres à virgule flottante sont exprimés dans une notation scientifique (7 chiffres pour la mantisse et 1 chiffre pour l'exposant).</p> <p style="text-align: center;">Quotient</p>	Sortie Obligatoire
<b>BIT COUNTER</b> BCNT @ BCNT 067	<p>N : Nombre de mots S : 1er mot source R : Mot de résultat</p>	<p>Compte le nombre total de bits sur ON dans le(s) mot(s) spécifié(s).</p>	Sortie Obligatoire

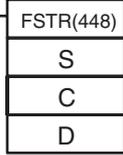
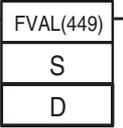
### 3-13 Instructions mathématiques à virgule flottante

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>FLOATING TO 16-BIT</b>  FIX @FIX 450	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 FIX(450)  <hr/>                 S  <hr/>                 R             </div> <p>S : 1er mot source R : Mot de résultat</p>	Convertit une valeur à virgule flottante 32 bits en donnée binaire signée 16 bits et place le résultat dans le mot de résultat spécifié. <div style="display: flex; align-items: center; margin-top: 10px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">S+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">S</div> <div style="margin-right: 10px;">Donnée à virgule flottante (32 bits)</div> </div> <div style="text-align: center; margin: 5px 0;">↓</div> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">R</div> <div style="margin-left: 100px;">Donnée binaire signée (16 bits)</div>	Sortie Obligatoire
<b>FLOATING TO 32-BIT</b>  FIXL @FIXL 451	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 FIXL(451)  <hr/>                 S  <hr/>                 R             </div> <p>S : 1er mot source R : 1er mot de résultat</p>	Convertit une valeur à virgule flottante 32 bits en donnée binaire signée 32 bits et place le résultat dans les mots de résultat spécifiés. <div style="display: flex; align-items: center; margin-top: 10px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">S+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">S</div> <div style="margin-right: 10px;">Donnée à virgule flottante (32 bits)</div> </div> <div style="text-align: center; margin: 5px 0;">↓</div> <div style="display: flex; justify-content: center; margin: 0 auto;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R</div> </div> <div style="margin-left: 100px;">Donnée binaire signée (32 bits)</div>	Sortie Obligatoire
<b>16-BIT TO FLOATING</b>  FLT @FLT 452	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 FLT(452)  <hr/>                 S  <hr/>                 R             </div> <p>S : Mot source R : 1er mot de résultat</p>	Convertit une valeur binaire signée 16 bits en donnée à virgule flottante 32 bits et place le résultat dans les mots de résultat <div style="display: flex; align-items: center; margin-top: 10px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">S</div> <div style="margin-right: 10px;">Donnée binaire signée (16 bits)</div> </div> <div style="text-align: center; margin: 5px 0;">↓</div> <div style="display: flex; justify-content: center; margin: 0 auto;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R</div> </div> <div style="margin-left: 100px;">Donnée à virgule flottante (32 bits)</div>	Sortie Obligatoire
<b>32-BIT TO FLOATING</b>  FLTL @FLTL 453	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 FLTL(453)  <hr/>                 S  <hr/>                 R             </div> <p>S : 1er mot source R : 1er mot de résultat</p>	Convertit une valeur binaire signée 32 bits en donnée à virgule flottante 32 bits et place le résultat dans les mots de résultat <div style="display: flex; align-items: center; margin-top: 10px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">S+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">S</div> <div style="margin-right: 10px;">Donnée binaire signée (32 bits)</div> </div> <div style="text-align: center; margin: 5px 0;">↓</div> <div style="display: flex; justify-content: center; margin: 0 auto;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R</div> </div> <div style="margin-left: 100px;">Donnée à virgule flottante (32 bits)</div>	Sortie Obligatoire
<b>FLOATING-POINT ADD</b>  +F @+F 454	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 +F(454)  <hr/>                 Au  <hr/>                 Ad  <hr/>                 R             </div> <p>Au : 1er mot 1er terme de l'addition AD : 1er mot 2ème terme addition R : 1er mot de résultat</p>	Ajoute deux nombres à virgule flottante 32 bits et place le résultat dans les mots de résultat spécifiés. <div style="margin-top: 10px;"> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Au+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Au</div> <div style="margin-right: 10px;">1er terme de l'addition (donnée à virgule flottante, 32 bits)</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <span style="margin-right: 5px;">+</span> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Ad+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Ad</div> <div style="margin-right: 10px;">2ème terme addition (donnée à virgule flottante, 32 bits)</div> </div> <hr style="width: 100%;"/> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R</div> <div style="margin-right: 10px;">Résultat (données à virgule flottante, 32 bits)</div> </div> </div>	Sortie Obligatoire
<b>FLOATING-POINT SUBTRACT</b>  -F @-F 455	<div style="border: 1px solid black; padding: 5px; width: fit-content;">                 -F(455)  <hr/>                 Mi  <hr/>                 Su  <hr/>                 R             </div> <p>Mi : 1er mot à soustraire Su : 1er mot soustracteur R : 1er mot de résultat</p>	Soustrait un nombre à virgule flottante 32 bits d'un autre nombre et place le résultat dans les mots de résultat spécifiés. <div style="margin-top: 10px;"> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Mi+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Mi</div> <div style="margin-right: 10px;">Mot à soustraire (données à virgule flottante, 32 bits)</div> </div> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <span style="margin-right: 5px;">-</span> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Su+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">Su</div> <div style="margin-right: 10px;">Soustracteur (données à virgule flottante, 32 bits)</div> </div> <hr style="width: 100%;"/> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R+1</div> <div style="border: 1px solid black; padding: 5px; margin-right: 10px;">R</div> <div style="margin-right: 10px;">Résultat (données à virgule flottante, 32 bits)</div> </div> </div>	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>FLOATING-POINT MULTIPLY</b> *F @*F 456	*F(456) Md Mr R Md : 1er mot multiplicande Mr : 1er mot multiplicateur R : 1er mot de résultat	Multiplie deux nombres à virgule flottante 32 bits et place le résultat dans les mots de résultat spécifiés. 	Sortie Obligatoire
<b>FLOATING-POINT DIVIDE</b> /F @/F 457	/F(457) Dd Dr R Dd : 1er mot dividende Dr : 1er mot diviseur R : 1er mot de résultat	Divise un nombre à virgule flottante 32 bits par un autre et place le résultat dans les mots de résultat spécifiés. 	Sortie Obligatoire
<b>DEGREES TO RADIANS</b> RAD @RAD 458	RAD(458) S R S : 1er mot source R : 1er mot de résultat	Convertit un nombre de 32 bits à virgule flottante de degrés en radians et place le résultat dans les mots de résultat spécifiés. 	Sortie Obligatoire
<b>RADIANS TO DEGREES</b> DEG @DEG 459	DEG(459) S R S : 1er mot source R : 1er mot de résultat	Convertit un nombre de 32 bits à virgule flottante de radians en degrés et place le résultat dans les mots de résultat spécifiés. 	Sortie Obligatoire
<b>SINE</b> SIN @SIN 460	SIN(460) S R S : 1er mot source R : 1er mot de résultat	Calcule le sinus d'un nombre de 32 bits à virgule flottante (en radians) et place le résultat dans les mots de résultat spécifiés. 	Sortie Obligatoire
<b>COSINE</b> COS @COS 461	COS(461) S R S : 1er mot source R : 1er mot de résultat	Calcule le cosinus d'un nombre de 32 bits à virgule flottante (en radians) et place le résultat dans les mots de résultat spécifiés. 	Sortie Obligatoire

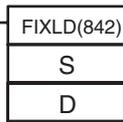
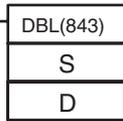
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>TANGENT</b> TAN @TAN 462	<p>S : 1er mot source R : 1er mot de résultat</p>	<p>Calcule la tangente d'un nombre de 32 bits à virgule flottante (en radians) et place le résultat dans les mots de résultat spécifiés.</p> $\text{TAN} \left( \begin{array}{ c c } \hline \text{S+1} & \text{S} \\ \hline \end{array} \right)$ <p>Source (données 32 bits à virgule flottante)</p> $\begin{array}{ c c } \hline \text{R+1} & \text{R} \\ \hline \end{array}$ <p>Résultat (données 32 bits à virgule flottante)</p>	Sortie Obligatoire
<b>ARC SINE</b> ASIN @ASIN 463	<p>S : 1er mot source R : 1er mot de résultat</p>	<p>Calcule l'arc sinus d'un nombre de 32 bits à virgule flottante et place le résultat dans les mots de résultat spécifiés (la fonction arc sinus est l'inverse de la fonction sinus ; elle renvoie l'angle qui correspond à une valeur de sinus donnée, comprise entre -1 et 1).</p> $\text{SIN}^{-1} \left( \begin{array}{ c c } \hline \text{S+1} & \text{S} \\ \hline \end{array} \right)$ <p>Source (données 32 bits à virgule flottante)</p> $\begin{array}{ c c } \hline \text{R+1} & \text{R} \\ \hline \end{array}$ <p>Résultat (données 32 bits à virgule flottante)</p>	Sortie Obligatoire
<b>ARC COSINE</b> ACOS @ACOS 464	<p>S : 1er mot source R : 1er mot de résultat</p>	<p>Calcule l'arc cosinus d'un nombre de 32 bits à virgule flottante et place le résultat dans les mots de résultat spécifiés (la fonction arc cosinus est l'inverse de la fonction cosinus ; elle renvoie l'angle qui donne une valeur de cosinus donnée, comprise entre -1 et 1).</p> $\text{COS}^{-1} \left( \begin{array}{ c c } \hline \text{S+1} & \text{S} \\ \hline \end{array} \right)$ <p>Source (données 32 bits à virgule flottante)</p> $\begin{array}{ c c } \hline \text{R+1} & \text{R} \\ \hline \end{array}$ <p>Résultat (données 32 bits à virgule flottante)</p>	Sortie Obligatoire
<b>ARC TANGENT</b> ATAN @ATAN 465	<p>S : 1er mot source R : 1er mot de résultat</p>	<p>Calcule l'arc tangente d'un nombre de 32 bits à virgule flottante et place le résultat dans les mots de résultats spécifiés (la fonction arc tangente est l'inverse de la fonction tangente ; elle retourne l'angle qui correspond à une valeur de tangente donnée).</p> $\text{TAN}^{-1} \left( \begin{array}{ c c } \hline \text{S+1} & \text{S} \\ \hline \end{array} \right)$ <p>Source (données 32 bits à virgule flottante)</p> $\begin{array}{ c c } \hline \text{R+1} & \text{R} \\ \hline \end{array}$ <p>Résultat (données 32 bits à virgule flottante)</p>	Sortie Obligatoire
<b>SQUARE ROOT</b> SQRT @SQRT 466	<p>S : 1er mot source R : 1er mot de résultat</p>	<p>Calcule la racine carrée d'un nombre de 32 bits à virgule flottante et place le résultat dans les mots de résultat spécifiés.</p> $\sqrt{\begin{array}{ c c } \hline \text{S+1} & \text{S} \\ \hline \end{array}}$ <p>Source (données 32 bits à virgule flottante)</p> $\begin{array}{ c c } \hline \text{R+1} & \text{R} \\ \hline \end{array}$ <p>Résultat (données 32 bits à virgule flottante)</p>	Sortie Obligatoire

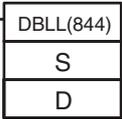
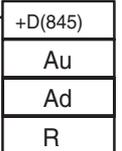
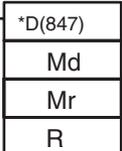
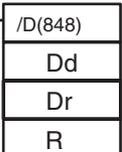
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>EXPONENT</b>  EXP @EXP 467	 <p>S : 1er mot source R : 1er mot de résultat</p>	<p>Calcule l'exponentiel naturel (base e) d'un nombre 32 bits à virgule flottante et place le résultat dans les mots de résultat spécifiés.</p>  <p>Source (données 32 bits à virgule flottante) Résultat (données 32 bits à virgule flottante)</p>	Sortie Obligatoire
<b>LOGARITHM</b>  LOG @LOG 468	 <p>S : 1er mot source R : 1er mot de résultat</p>	<p>Calcule le logarithme naturel (base e) d'un nombre 32 bits à virgule flottante et place le résultat dans les mots de résultat spécifiés.</p>  <p>Source (données 32 bits à virgule flottante) Résultat (données 32 bits à virgule flottante)</p>	Sortie Obligatoire
<b>EXPONENTIAL POWER</b>  PWR @PWR 840	 <p>B : 1er mot de base E : 1er mot exponentiel R : 1er mot de résultat</p>	<p>Elève un nombre 32 bits à virgule flottante à la puissance d'un autre nombre 32 bits à virgule flottante.</p>  <p>Puissance Base</p>	Sortie Obligatoire
<b>FLOATING SYMBOL COMPARISON (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b>  LD, AND ou OR + =F (329), <>F (330), <F (331), <=F (332), >F (333) ou >=F (334)	<p>Utilisation de LD :</p>  <p>Utilisation de AND :</p>  <p>Utilisation de OR :</p>  <p>S1 : Donnée de comparaison 1 S2 : Donnée de comparaison 2</p>	<p>Compare les données simple précision (32 bits) ou les constantes spécifiées et crée une condition d'exécution ON si le résultat de la comparaison est vrai.</p> <p>Trois sortes de symboles peuvent être utilisés avec les instructions de comparaison de symbole à virgule flottante : LD (Load), AND et OR.</p>	LD : non obligatoire  AND ou OR : obligatoire

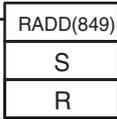
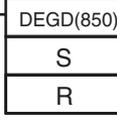
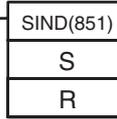
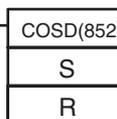
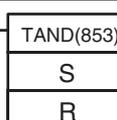
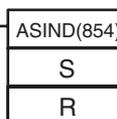
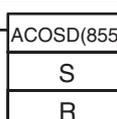
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>FLOATING-POINT TO ASCII (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> FSTR @FSTR 448	 <p>S : 1er mot source                      C : Mot de contrôle                      D : Mot de destination</p>	Convertit les données à virgule flottante simple précision (format à virgule décimale ou exponentiel 32 bits) spécifiées en chaîne de caractères (ASCII) et place le résultat dans le mot de destination.	Sortie obligatoire
<b>ASCII TO FLOATING-POINT (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> FVAL @FVAL 449	 <p>S : Mot source                      D : 1er mot de destination</p>	Convertit la chaîne de caractères (ASCII) spécifiée représentant les données à virgule flottante simple précision (format à virgule décimale ou exponentiel) en données à virgule flottante simple précision 32 bits et place le résultat dans les mots de destination.	Sortie obligatoire

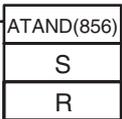
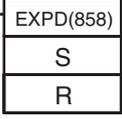
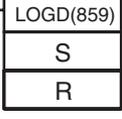
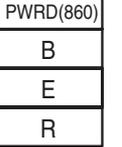
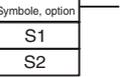
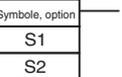
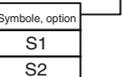
### 3-14 Instructions à virgule flottante double précision

Les instructions à virgule flottante double précision sont prises en charge uniquement par les UC CS1-H, CJ1-H, CJ1M et CS1D.

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DOUBLE FLOATING TO 16-BIT BINARY</b> FIXD @FIXD 841	 <p>S : 1er mot source                      D : Mot de destination</p>	Convertit les données à virgule flottante double précision (64 bits) spécifiées en données binaires 16 bits signées et place le résultat dans le mot de destination.	Sortie Obligatoire
<b>DOUBLE FLOATING TO 32-BIT BINARY</b> FIXLD @FIXLD 842	 <p>S : 1er mot source                      D : 1er mot de destination</p>	Convertit les données à virgule flottante double précision (64 bits) spécifiées en données binaires 32 bits signées et place le résultat dans les mots de destination.	Sortie Obligatoire
<b>16-BIT BINARY TO DOUBLE FLOATING</b> DBL @DBL 843	 <p>S : Mot source                      D : 1er mot de destination</p>	Convertit les données binaires 16 bits signées spécifiées en données à virgule flottante double précision (64 bits) et place le résultat dans les mots de destination.	Sortie Obligatoire

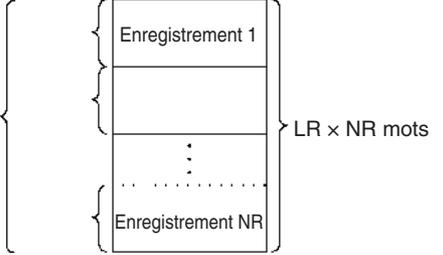
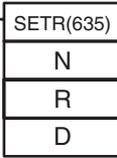
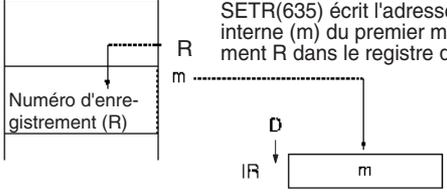
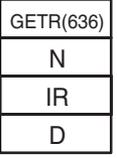
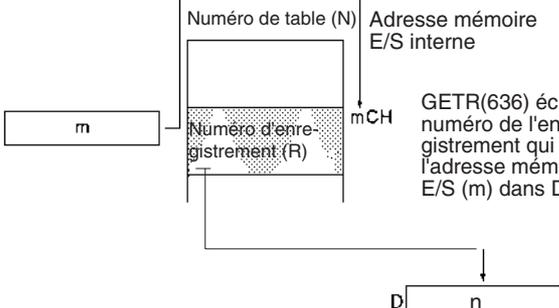
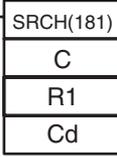
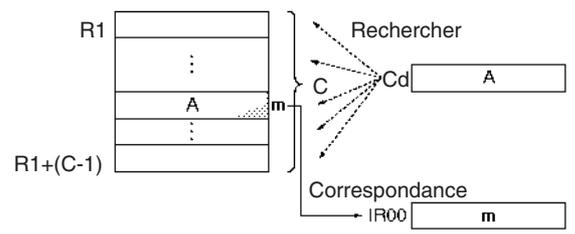
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>32-BIT BINARY TO DOUBLE FLOATING</b>  DBLL @DBLL 844	 <p><b>S</b> : 1er mot source <b>D</b> : 1er mot de destination</p>	Convertit les données binaires 32 bits signées spécifiées en données à virgule flottante double précision (64 bits) et place le résultat dans les mots de destination.	Sortie Obligatoire
<b>DOUBLE FLOATING-POINT ADD</b>  +D @+D 845	 <p><b>Au</b> : 1er mot 1er terme de l'addition <b>Ad</b> : 1er mot 2ème terme addition <b>R</b> : 1er mot de résultat</p>	Ajoute les valeurs à virgule flottante double précision spécifiées (64 bits chacune) et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE FLOATING-POINT SUBTRACT</b>  -D @-D 846	 <p><b>Mi</b> : 1er mot à soustraire <b>Su</b> : 1er mot soustracteur <b>R</b> : 1er mot de résultat</p>	Soustrait les valeurs à virgule flottante double précision (64 bits chacune) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE FLOATING-POINT MULTIPLY</b>  *D @*D 847	 <p><b>Md</b> : 1er mot multiplicande <b>Mr</b> : 1er mot multiplicateur <b>R</b> : 1er mot de résultat</p>	Multiplie les valeurs à virgule flottante double précision (64 bits chacune) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE FLOATING-POINT DIVIDE</b>  /D @/D 848	 <p><b>Dd</b> : 1er mot dividende <b>Dr</b> : 1er mot diviseur <b>R</b> : 1er mot de résultat</p>	Divise les valeurs à virgule flottante double précision (64 bits chacune) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire

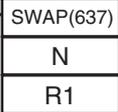
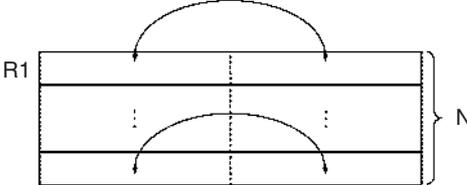
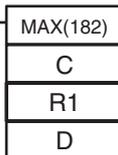
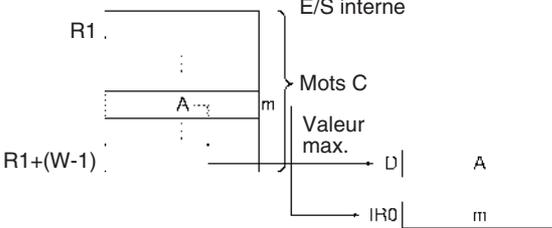
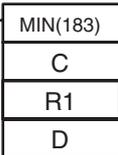
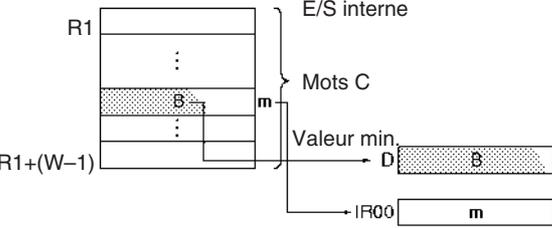
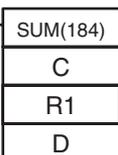
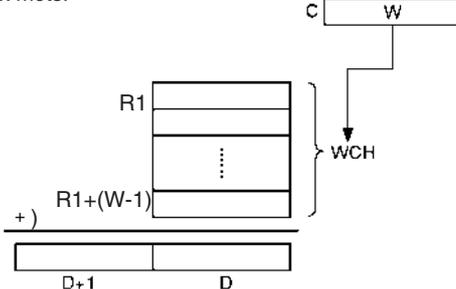
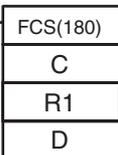
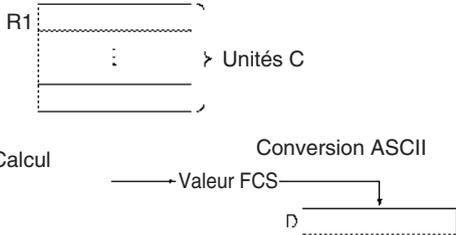
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DOUBLE DEGREES TO RADIANS</b> RADD @RADD 849	 <p>S : 1er mot source R : 1er mot de résultat</p>	Convertit les données à virgule flottante double précision (64 bits) spécifiées de degrés en radians et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE RADIANS TO DEGREES</b> DEGD @DEGD 850	 <p>S : 1er mot source R : 1er mot de résultat</p>	Convertit les données à virgule flottante double précision (64 bits) spécifiées de radians en degrés et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE SINE</b> SIND @SIND 851	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule le sinus de l'angle (radians) dans les données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE COSINE</b> COSD @COSD 852	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule le cosinus de l'angle (radians) dans les données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE TANGENT</b> TAND @TAND 853	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule la tangente de l'angle (radians) dans les données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE ARC SINE</b> ASIND @ASIND 854	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule l'angle (en radians) à partir de la valeur du sinus dans les données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat (la fonction arc sinus est l'inverse de la fonction sinus ; elle retourne l'angle qui correspond à une valeur de sinus donnée, comprise entre -1 et 1).	Sortie Obligatoire
<b>DOUBLE ARC COSINE</b> ACOSD @ACOSD 855	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule l'angle (en radians) à partir de la valeur du cosinus dans les données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat (la fonction arc cosinus est l'inverse de la fonction cosinus ; elle retourne l'angle qui correspond à une valeur de cosinus donnée, comprise entre -1 et 1).	Sortie Obligatoire

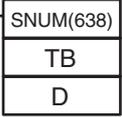
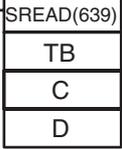
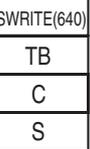
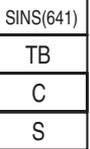
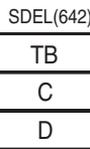
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DOUBLE ARC TANGENT</b> ATAND @ATAND 856	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule l'angle (en radians) à partir de la valeur de la tangente dans les données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat (la fonction arc tangente est l'inverse de la fonction tangente ; elle retourne l'angle qui correspond à une valeur de tangente donnée).	Sortie Obligatoire
<b>DOUBLE SQUARE ROOT</b> SQRTD @SQRTD 857	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule la racine carrée des données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE EXPONENT</b> EXPD @EXPD 858	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule l'exponentiel naturel (base e) des données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE LOGARITHM</b> LOGD @LOGD 859	 <p>S : 1er mot source R : 1er mot de résultat</p>	Calcule le logarithme naturel (base e) des données à virgule flottante double précision (64 bits) spécifiées et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE EXPONENTIAL POWER</b> PWRD @PWRD 860	 <p>B : 1er mot de base E : 1er mot exponentiel R : 1er mot de résultat</p>	Elève un nombre à virgule flottante double précision (64 bits) à la puissance d'un autre nombre à virgule flottante double précision et place le résultat dans les mots de résultat.	Sortie Obligatoire
<b>DOUBLE SYMBOL COMPARISON</b> LD, AND ou OR + =D (335), <>D (336), <D (337), <=D (338), >D (339), or >=D (340)	Utilisation de LD :  Utilisation de AND :  Utilisation de OR :  S1 : Donnée de comparaison 1 S2 : Donnée de comparaison 2	Compare les données double précision (64 bits) spécifiées et crée une condition d'exécution ON si le résultat de la comparaison est vrai. Ces trois types de symboles peuvent être utilisés avec les instructions de comparaison à symbole à virgule flottante : LD (Load), AND et OR.	LD : non obligatoire AND ou OR : obligatoire

### 3-15 Instructions de traitement de données de tableaux

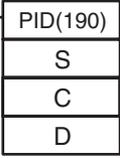
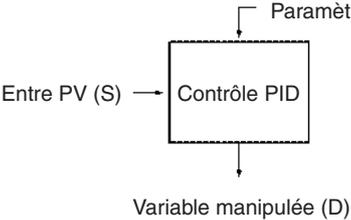
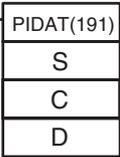
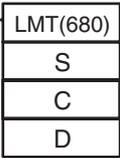
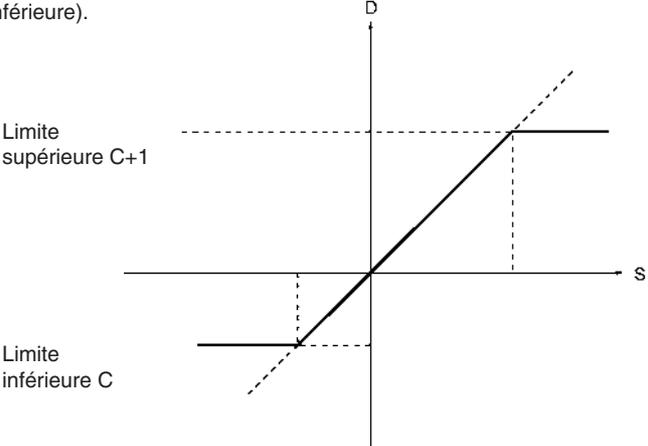
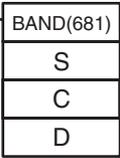
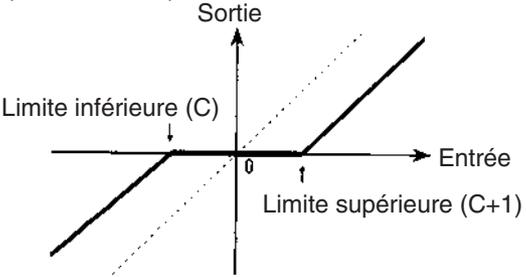
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution			
<b>SET STACK</b> SSET @SSET 630	<table border="1"> <tr><td>SSET(630)</td></tr> <tr><td>TB</td></tr> <tr><td>N</td></tr> </table> <p>TB : 1ère adresse de pile N : Nombre de mots</p>	SSET(630)	TB	N	Définit une pile de la longueur spécifiée commençant au mot spécifié et initialise (met à zéro) les mots dans la zone de données.  Adresse mémoire E/S interne  	Sortie Obligatoire
SSET(630)						
TB						
N						
<b>PUSH ONTO STACK</b> PUSH @PUSH 632	<table border="1"> <tr><td>PUSH(632)</td></tr> <tr><td>TB</td></tr> <tr><td>S</td></tr> </table> <p>TB : 1ère adresse de pile S : Mot source</p>	PUSH(632)	TB	S	Ecrit un mot de données dans la pile spécifiée.  Adresse mémoire E/S interne  Adresse mémoire E/S interne  	Sortie Obligatoire
PUSH(632)						
TB						
S						
<b>LAST IN FIRST OUT</b> LIFO @LIFO 634	<table border="1"> <tr><td>LIFO(634)</td></tr> <tr><td>TB</td></tr> <tr><td>D</td></tr> </table> <p>TB : 1ère adresse de pile D : Mot de destination</p>	LIFO(634)	TB	D	Lit le dernier mot de données écrit dans la pile spécifiée (les données de la pile les plus récentes).  Pointeur d'empilement  Adresse mémoire E/S interne  Adresse mémoire E/S interne  	Sortie Obligatoire
LIFO(634)						
TB						
D						
<b>FIRST IN FIRST OUT</b> FIFO @FIFO 633	<table border="1"> <tr><td>FIFO(633)</td></tr> <tr><td>TB</td></tr> <tr><td>D</td></tr> </table> <p>TB : 1ère adresse de pile D : Mot de destination</p>	FIFO(633)	TB	D	Lit le premier mot de données écrit dans la pile spécifiée (les données de la pile les plus anciennes).  Adresse mémoire E/S interne  Adresse mémoire E/S interne  	Sortie Obligatoire
FIFO(633)						
TB						
D						

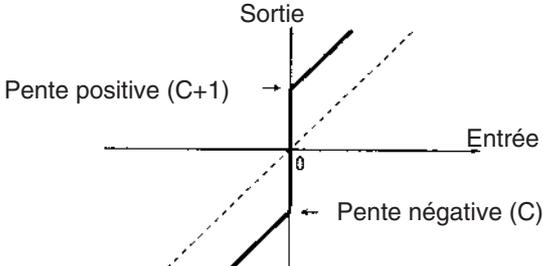
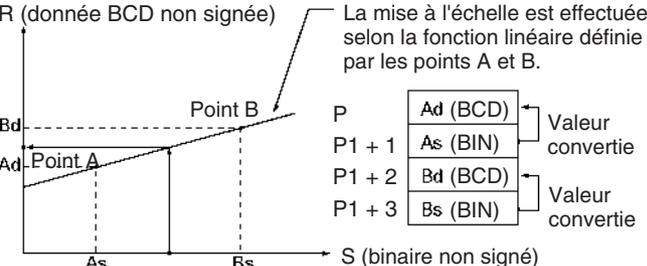
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DIMENSION RECORD TABLE</b> DIM @DIM 631	 <p>N : Numéro de table                      LR : Longueur de chaque enregistrement                      NR : Nombre d'enregistrements                      TB : 1er mot de table</p>	Définit une table d'enregistrements en déclarant la longueur de chaque enregistrement et le nombre d'enregistrement. Il est possible de définir jusqu'à 16 tables d'enregistrements. <p>Numéro de table (N)</p>  <p>Nombre d'enregistrements</p> <p>LR x NR mots</p>	Sortie Obligatoire
<b>SET RECORD LOCATION</b> SETR @SETR 635	 <p>N : Numéro de table                      R : Numéro d'enregistrement                      D : Registre d'index de destination</p>	Ecrit l'emplacement de l'enregistrement spécifié (l'adresse mémoire E/S interne du début de l'enregistrement) dans le registre d'index spécifié. <p>Adresse mémoire E/S interne</p>  <p>SETR(635) écrit l'adresse mémoire E/S interne (m) du premier mot de l'enregistrement R dans le registre d'index D.</p>	Sortie Obligatoire
<b>GET RECORD NUMBER</b> GETR @GETR 636	 <p>N : Numéro de table                      IR : Registre d'index                      D : Mot de destination</p>	Renvoie le numéro de l'enregistrement à l'adresse mémoire E/S interne contenue dans le registre d'index spécifié. <p>Adresse mémoire E/S interne</p>  <p>GETR(636) écrit le numéro de l'enregistrement qui inclut l'adresse mémoire E/S (m) dans D.</p>	Sortie Obligatoire
<b>DATA SEARCH</b> SRCH @SRCH 181	 <p>C : 1er mot de contrôle                      R1 : 1er mot de la plage                      Cd : Donnée de comparaison</p>	Recherche un mot de données dans une plage de mots. <p>Adresse mémoire E/S interne</p>  <p>Rechercher</p> <p>Correspondance</p>	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>SWAP BYTES</b> SWAP @SWAP 637	 <p>N : Nombre de mots R1 : 1er mot de la plage</p>	Permute l'octet le plus à gauche et l'octet le plus à droite dans tous les mots de la plage. Permute la position des octets. 	Sortie Obligatoire
<b>FIND MAXIMUM</b> MAX @MAX 182	 <p>C : 1er mot de contrôle R1 : 1er mot de la plage D : Mot de destination</p>	Recherche la valeur maximale dans la plage. Adresse mémoire E/S interne 	Sortie Obligatoire
<b>FIND MINIMUM</b> MIN @MIN 183	 <p>C : 1er mot de contrôle R1 : 1er mot de la plage D : Mot de destination</p>	Recherche la valeur minimale dans la plage. Adresse mémoire E/S interne 	Sortie Obligatoire
<b>SUM</b> SUM @SUM 184	 <p>C : 1er mot de contrôle R1 : 1er mot de la plage D : 1er mot de destination</p>	Ajoute les octets ou les mots dans la plage et place le résultat dans deux mots. Adresse mémoire E/S interne 	Sortie Obligatoire
<b>FRAME CHECKSUM</b> FCS @FCS 180	 <p>C : 1er mot de contrôle R1 : 1er mot de la plage D : 1er mot de destination</p>	Calcule la valeur FCS ASCII pour la plage spécifiée. Adresse mémoire E/S interne 	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>STACK SIZE READ (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> SNUM @SNUM 638	 <p>TB : Première adresse pile D : Mot de destination</p>	Compte le nombre de données de pile (nombre de mots) dans la pile spécifiée.	Sortie obligatoire
<b>STACK DATA READ (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> SREAD @SREAD 639	 <p>TB : Première adresse pile C : Valeur de décalage D : Mot de destinat</p>	Lit les données d'un élément de données spécifié dans la pile. La valeur de décalage indique l'emplacement de l'élément de données voulu (nombre d'éléments de données avant la position courante du pointeur).	Sortie obligatoire
<b>STACK DATA OVERWRITE (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> SWRIT @SWRIT 640	 <p>TB : Première adresse pile C : Valeur de décalage S : Donnée source</p>	Ecrit les données sources dans l'élément de données spécifié dans la pile (en remplaçant les données existantes). La valeur de décalage indique l'emplacement de l'élément de données voulu (nombre d'éléments de données avant la position courante du pointeur).	Sortie obligatoire
<b>STACK DATA INSERT (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> SINS @SINS 641	 <p>TB : Première adresse pile C : Valeur de décalage S : Donnée source</p>	Insère les données sources à l'emplacement spécifié dans la pile et décale le reste des données de la pile vers le bas. La valeur de décalage indique l'emplacement du point d'insertion (nombre d'éléments de données avant la position courante du pointeur).	Sortie obligatoire
<b>STACK DATA DELETE (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> SDEL @SDEL 642	 <p>TB : Première adresse pile C : Valeur de décalage D : Mot de destination</p>	Supprime l'élément de données à l'emplacement spécifié dans la pile et décale le reste des données de la pile vers le haut. La valeur de décalage indique l'emplacement du point de suppression (nombre d'éléments de données avant la position courante du pointeur).	Sortie obligatoire

### 3-16 Instructions de contrôle de données

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>PID CONTROL</b> PID 190	 <p>S : Mot d'entrée                      C : 1er mot paramètre                      D : Mot de sortie</p>	Exécute la commande PID en fonction des paramètres spécifiés. 	Sortie Obligatoire
<b>PID CONTROL WITH AUTOTUNING</b> PIDAT 191 (CS1-H, CJ1-H ou CJ1M uniquement)	 <p>S : Mot d'entrée                      C : 1er mot paramètre                      D : Mot de sortie</p>	Exécute la commande PID en fonction des paramètres spécifiés. Les constantes PID peuvent être réglées automatiquement par PIDAT(191).	Sortie obligatoire
<b>LIMIT CONTROL</b> LMT @LMT 680	 <p>S : Mot d'entrée                      C : 1er mot limite                      D : Mot de sortie</p>	Contrôle les données de sortie en fonction des données d'entrée (selon qu'elles sont comprises ou non entre les limites supérieure et inférieure). 	Sortie Obligatoire
<b>DEAD BAND CONTROL</b> BAND @BAND 681	 <p>S : Mot d'entrée                      C : 1er mot limite                      D : Mot de sortie</p>	Contrôle les données de sortie en fonction des données d'entrée (selon qu'elles sont comprises ou non dans la zone d'insensibilité). 	Sortie Obligatoire

Instruction Mnémomique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<b>DEAD ZONE CONTROL</b>  ZONE @ZONE 682	<table border="1" style="margin-left: 20px;"> <tr><td>ZONE(682)</td></tr> <tr><td>S</td></tr> <tr><td>C</td></tr> <tr><td>D</td></tr> </table> <p>S : Mot d'entrée C : 1er mot limite D : Mot de sortie</p>	ZONE(682)	S	C	D	Ajoute la pente spécifiée aux données d'entrée et sort le résultat. 	Sortie Obligatoire
ZONE(682)							
S							
C							
D							
<b>TIME-PROPORTIONAL OUTPUT</b>  TPO 685  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	<table border="1" style="margin-left: 20px;"> <tr><td>TPO (685)</td></tr> <tr><td>S</td></tr> <tr><td>C</td></tr> <tr><td>R</td></tr> </table> <p>S : Mot d'entrée C : 1er mot paramètre R : Bit de sortie d'impulsions</p>	TPO (685)	S	C	R	Entre le taux de service ou la variable manipulée à partir du mot spécifié, convertit le taux de service en une sortie proportionnelle au temps en fonction des paramètres spécifiés et sort le résultat à partir de la sortie spécifiée.	Sortie Obligatoire
TPO (685)							
S							
C							
R							
<b>SCALING</b>  SCL @SCL 194	<table border="1" style="margin-left: 20px;"> <tr><td>SCL(194)</td></tr> <tr><td>S</td></tr> <tr><td>P1</td></tr> <tr><td>R</td></tr> </table> <p>S : Mot source P1 : 1er mot paramètre R : Mot de résultat</p>	SCL(194)	S	P1	R	Convertit les données binaires non signées en données BCD non signées selon la fonction linéaire spécifiée. 	Sortie Obligatoire
SCL(194)							
S							
P1							
R							

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution													
<p><b>SCALING 2</b></p> <p>SCL2 @SCL2 486</p>	<table border="1" style="margin-left: 20px;"> <tr><td>SCL2(486)</td></tr> <tr><td>S</td></tr> <tr><td>P1</td></tr> <tr><td>R</td></tr> </table> <p>S : Mot source P1 : 1er mot paramètre R : Mot de résultat</p>	SCL2(486)	S	P1	R	<p>Convertit les données binaires signées en données BCD signées selon la fonction linéaire spécifiée. Vous pouvez entrer un décalage en définissant la fonction linéaire.</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p><b>Décalage positif</b></p> </div> <div style="text-align: center;"> <p><b>Décalage négatif</b></p> </div> </div> <div style="margin-top: 20px;"> <p><b>Décalage de 0000</b></p> <table border="1" style="margin-left: 20px;"> <tr><td>P1</td><td>Décalage</td><td>(Donnée binaire signée)</td></tr> <tr><td>P1 + 1</td><td>ΔY</td><td>(Donnée binaire signée)</td></tr> <tr><td>P1 + 2</td><td>ΔX</td><td>(Donnée BCD signée)</td></tr> </table> </div>	P1	Décalage	(Donnée binaire signée)	P1 + 1	ΔY	(Donnée binaire signée)	P1 + 2	ΔX	(Donnée BCD signée)	<p>Sortie Obligatoire</p>
SCL2(486)																
S																
P1																
R																
P1	Décalage	(Donnée binaire signée)														
P1 + 1	ΔY	(Donnée binaire signée)														
P1 + 2	ΔX	(Donnée BCD signée)														

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<p><b>SCALING 3</b></p> <p>SCL3 @SCL3 487</p>	<table border="1" style="margin-left: 20px;"> <tr><td>SCL3(487)</td></tr> <tr><td>S</td></tr> <tr><td>P1</td></tr> <tr><td>R</td></tr> </table> <p><b>S</b> : Mot source <b>P1</b> : 1er mot paramètre <b>R</b> : Mot de résultat</p>	SCL3(487)	S	P1	R	<p>Convertit les données BCD signées en données binaires signées selon la fonction linéaire spécifiée. Vous pouvez entrer un décalage en définissant la fonction linéaire.</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p><b>Décalage positif</b></p> </div> <div style="text-align: center;"> <p><b>Décalage négatif</b></p> </div> </div> <div style="text-align: center; margin-top: 20px;"> <p><b>Décalage de 0000</b></p> </div>	<p>Sortie Obligatoire</p>
SCL3(487)							
S							
P1							
R							
<p><b>AVERAGE</b></p> <p>AVG 195</p>	<table border="1" style="margin-left: 20px;"> <tr><td>AVG(195)</td></tr> <tr><td>S</td></tr> <tr><td>N</td></tr> <tr><td>R</td></tr> </table> <p><b>S</b> : Mot source <b>N</b> : Nombre de cycles <b>R</b> : Mot de résultat</p>	AVG(195)	S	N	R	<p>Calcule la valeur moyenne d'un mot d'entrée pour le nombre de cycles spécifié.</p> <p><b>S</b> : Mot source</p> <p><b>N</b> : Nombre de cycles</p>	<p>Sortie Obligatoire</p>
AVG(195)							
S							
N							
R							

### 3-17 Instructions de sous-programme

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>SUBROUTINE CALL</b>  SBS @SBS 091	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">                         SBS(091)  <hr style="width: 80%; margin: 0 auto;"/>                         N                     </div> <p>N : Numéro de sous-programme</p>	<p>Appelle le sous-programme correspondant au numéro spécifié et l'exécute.</p> <p style="text-align: right;">Condition d'exécution sur ON</p>	Sortie Obligatoire
<b>MACRO</b>  MCRO @MCRO 099	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">                         MCRO(099)  <hr style="width: 80%; margin: 0 auto;"/>                         N  <hr style="width: 80%; margin: 0 auto;"/>                         S  <hr style="width: 80%; margin: 0 auto;"/>                         D                     </div> <p>N : Numéro de sous-programme                      S : 1er mot paramètre entrée                      D : 1er mot paramètre sortie</p>	<p>Appelle le sous-programme correspondant au numéro spécifié et l'exécute en utilisant les paramètres d'entrée inclus dans S à S+3 et les paramètres de sortie inclus dans D à D+3.</p>	Sortie Obligatoire
<b>SUBROUTINE ENTRY</b>  SBN 092	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">                         SBN(092)  <hr style="width: 80%; margin: 0 auto;"/>                         N                     </div> <p>N : Numéro de sous-programme</p>	<p>Indique le début du sous-programme correspondant au numéro spécifié.</p>	Sortie Non obligatoire
<b>SUBROUTINE RETURN</b>  RET 093	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">                         RET(093)                     </div>	<p>Indique la fin d'un sous-programme.</p>	Sortie Non obligatoire

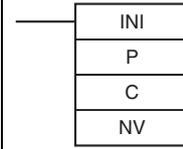
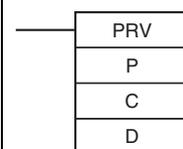
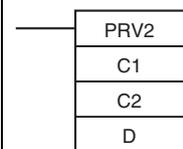
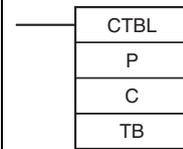
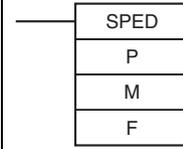
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution		
<b>GLOBAL SUBROUTINE CALL (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> GSBS 750	<table border="1"> <tr><td>GSBS(750)</td></tr> <tr><td>N</td></tr> </table> <p>N : Numéro de sous-programme</p>	GSBS(750)	N	Appelle le sous-programme correspondant au numéro spécifié et l'exécute.	Sortie Non obligatoire
GSBS(750)					
N					
<b>GLOBAL SUBROUTINE ENTRY (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> GSBN 751	<table border="1"> <tr><td>GSBN(751)</td></tr> <tr><td>N</td></tr> </table> <p>N : Numéro de sous-programme</p>	GSBN(751)	N	Indique le début du sous-programme correspondant au numéro spécifié.	Sortie Non obligatoire
GSBN(751)					
N					
<b>GLOBAL SUBROUTINE RETURN (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> GRET 752	<table border="1"> <tr><td>GRET(752)</td></tr> </table>	GRET(752)	Indique la fin d'un sous-programme.	Sortie Non obligatoire	
GRET(752)					

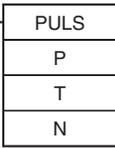
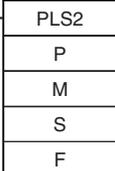
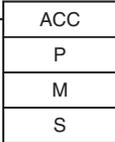
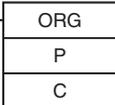
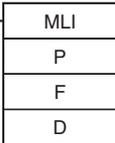
### 3-18 Instructions de traitement d'interruption

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution			
<b>SET INTERRUPT MASK (Non prise en charge par les UC CS1D pour les systèmes d'UC en duplex.)</b> MSKS @MSKS 690	<table border="1"> <tr><td>MSKS(690)</td></tr> <tr><td>N</td></tr> <tr><td>S</td></tr> </table> <p>N : Identifiant d'interruption S : Donnée interruption</p>	MSKS(690)	N	S	<p>Configure le traitement des interruptions pour les interruptions E/S ou les interruptions programmées. Les tâches d'interruption E/S et les tâches d'interruption programmées sont masquées (désactivées) lorsque l'API est mis sous tension pour la première fois. Vous pouvez utiliser MSKS(690) pour afficher ou masquer les interruptions E/S et définir les intervalles de temps des interruptions programmées.</p> <p>Cartes d'entrées d'interruption 0 à 3</p> <p>Interrupt E/S</p> <p>Masquer (1) ou démasquer (0) les entrées d'interruption 0 à 7.</p> <p>Interruption programmée</p> <p>Intervalle</p> <p>Définir le temps d'interruption programmée.</p>	Sortie Obligatoire
MSKS(690)						
N						
S						
<b>READ INTERRUPT MASK (Non prise en charge par les UC CS1D pour les systèmes à UC en duplex.)</b> MSKR @MSKR 692	<table border="1"> <tr><td>MSKR(692)</td></tr> <tr><td>N</td></tr> <tr><td>D</td></tr> </table> <p>N : Identifiant d'interruption D : Mot de destination</p>	MSKR(692)	N	D	Lit les paramètres de traitement des interruptions de courant qui ont été définis avec MSKS(690).	Sortie Obligatoire
MSKR(692)						
N						
D						

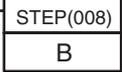
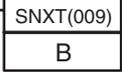
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution		
<p><b>CLEAR INTERRUPT</b> (Non prise en charge par les UC CS1D pour les systèmes à UC en duplex.)</p> <p>CLI @CLI 691</p>	<p>CLI(691)</p> <table border="1" style="margin-left: 20px;"> <tr><td style="text-align: center;">N</td></tr> <tr><td style="text-align: center;">S</td></tr> </table> <p>N : Identifiant d'interruption S : Donnée interruption</p>	N	S	<p>Remet à zéro ou conserve les entrées d'interruption sauvegardées pour les interruptions E/S ou définit le délai jusqu'à la première interruption programmée pour les interruptions programmées.</p> <p>N = 0 à 3</p> <p>N = 4 à 5</p>	<p>Sortie Obligatoire</p>
N					
S					
<p><b>DISABLE INTERRUPTS</b></p> <p>DI @DI 693</p>	<p>DI(693)</p>	<p>Désactive l'exécution de toutes les tâches d'interruption à l'exception de la tâche d'interruption de mise hors tension.</p>	<p>Sortie Obligatoire</p>		
<p><b>ENABLE INTERRUPTS</b></p> <p>EI 694</p>	<p>EI(694)</p>	<p>Active l'exécution de toutes les tâches d'interruption qui ont été désactivées avec DI(693).</p>	<p>Sortie Non obligatoire</p>		

### 3-19 Instructions de compteur à grande vitesse et de sortie d'impulsion (CJ1M-CPU21/22/23 uniquement)

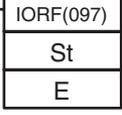
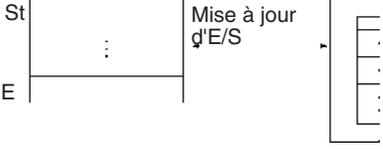
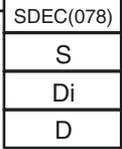
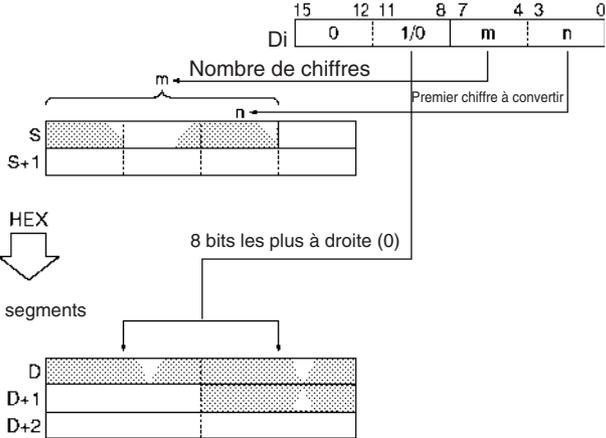
Instruction Mnémorique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>MODE CONTROL</b> INI @ INI 880	 <p><b>P</b> : identificateur de port  <b>C</b> : données de contrôle  <b>NV</b> : 1er mot avec nouvelle PV</p>	INI(880) permet de démarrer et d'interrompre la comparaison des valeurs cibles, de changer la valeur actuelle (PV) d'un compteur à grande vitesse, de changer la PV d'une entrée d'interruption (mode compteur), de changer la PV d'une sortie d'impulsion ou d'arrêter la sortie d'impulsion.	Sortie Obligatoire
<b>HIGH-SPEED COUNTER PV READ</b> PRV @ PRV 881	 <p><b>P</b> : identificateur de port  <b>C</b> : données de contrôle  <b>D</b> : 1er mot de destination</p>	PRV(881) permet de lire la valeur actuelle (PV) d'un compteur à grande vitesse, d'une sortie d'impulsion ou d'une entrée d'interruption (mode compteur).	Sortie Obligatoire
<b>COUNTER FREQUENCY CONVERT</b> PRV2 883 (UC CJ1M Ver. 2.0 ou supérieure uniquement)	 <p><b>C1</b> : données de contrôle  <b>C2</b> : impulsions/révolution  <b>D</b> : 1er mot de destination</p>	Lit l'entrée de la fréquence d'impulsion sur un compteur à grande vitesse et convertit la fréquence en vitesse de rotation (nombre de révolutions) ou convertit la valeur actuelle (PV) du compteur en nombre total de tours. Le résultat est dirigé vers les mots de destination sous la forme d'un nombre hexadécimal à 8 chiffres. Les impulsions ne peuvent être entrées qu'à partir du compteur à grande vitesse 0.	Sortie Obligatoire
<b>COMPARISON TABLE LOAD</b> CTBL @ CTBL 882	 <p><b>P</b> : identificateur de port  <b>C</b> : données de contrôle  <b>TB</b> : 1er mot du tableau de comparaison</p>	CTBL(882) permet de comparer la valeur actuelle (PV) d'un compteur à grande vitesse par rapport à des plages ou des valeurs cibles.	Sortie Obligatoire
<b>SPEED OUTPUT</b> SPED @ SPED 885	 <p><b>P</b> : identificateur de port  <b>M</b> : mode de sortie  <b>F</b> : 1er mot de fréquence d'impulsion</p>	SPED(885) permet de spécifier la fréquence et exécuter la sortie d'impulsion sans accélération ni décélération.	Sortie Obligatoire

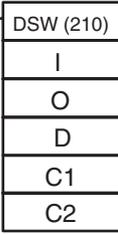
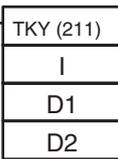
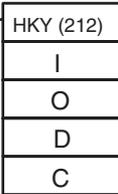
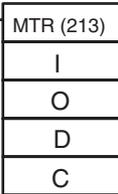
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>SET PULSES</b> PULS @PULS 886	 <p><b>P</b> : identificateur de port  <b>T</b> : type d'impulsion  <b>N</b> : nombre d'impulsions</p>	PULS(886) permet de définir le nombre d'impulsions pour la sortie d'impulsion.	Sortie Obligatoire
<b>PULSE OUTPUT</b> PLS2 @PLS2 887	 <p><b>P</b> : identificateur de port  <b>M</b> : mode de sortie  <b>S</b> : 1er mot du tableau des paramètres  <b>F</b> : 1er mot de la fréquence de démarrage</p>	PLS2(887) permet de définir la fréquence d'impulsion et les taux d'accélération/décélération, et d'exécuter la sortie d'impulsion avec accélération/décélération (avec des taux d'accélération/décélération différents). Seul le positionnement est possible.	Sortie Obligatoire
<b>ACCELERATION CONTROL</b> ACC @ACC 888	 <p><b>P</b> : identificateur de port  <b>M</b> : mode de sortie  <b>S</b> : 1er mot du tableau des paramètres</p>	ACC(888) permet de définir la fréquence d'impulsion et les taux d'accélération/décélération, et d'exécuter la sortie d'impulsion avec accélération/décélération (avec le même taux d'accélération/décélération). Le positionnement et la vitesse peuvent tous deux être contrôlés.	Sortie Obligatoire
<b>ORIGIN SEARCH</b> ORG @ORG 889	 <p><b>P</b> : identificateur de port  <b>C</b> : données de contrôle</p>	ORG(889) permet d'exécuter des recherches d'origine et des retours.	Sortie Obligatoire
<b>PULSE WITH VARIABLE DUTY FACTOR</b> MLI @ 891	 <p><b>P</b> : identificateur de port  <b>F</b> : fréquence  <b>D</b> : coefficient d'exploitation</p>	MLI(891) permet de sortir des impulsions avec un coefficient d'exploitation variable.	Sortie Obligatoire

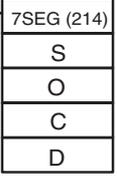
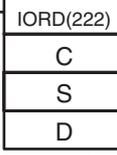
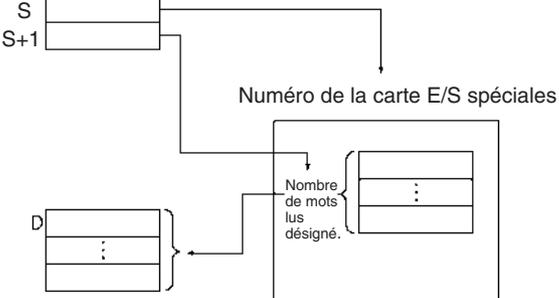
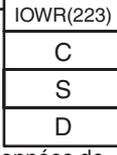
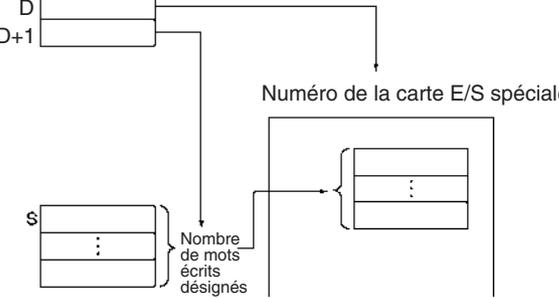
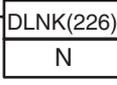
### 3-20 Instructions de pas

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>STEP DEFINE</b> STEP 008	 <p>B : Bit</p>	STEP(008) fonctionne de 2 façons, en fonction de sa position et selon qu'un bit de contrôle a été spécifié ou non. (1) Démarre un pas spécifique. (2) Termine la zone de programmation pas à pas (c'est-à-dire l'exécution par pas).	Sortie Obligatoire
<b>STEP START</b> SNXT 009	 <p>B : Bit</p>	SNXT(009) est utilisée de trois façons : (1) Pour démarrer l'exécution de la programmation pas à pas. (2) Pour passer au bit de contrôle de pas suivant. (3) Pour arrêter l'exécution de la programmation pas à pas.	Sortie Obligatoire

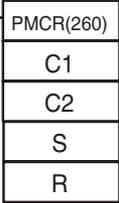
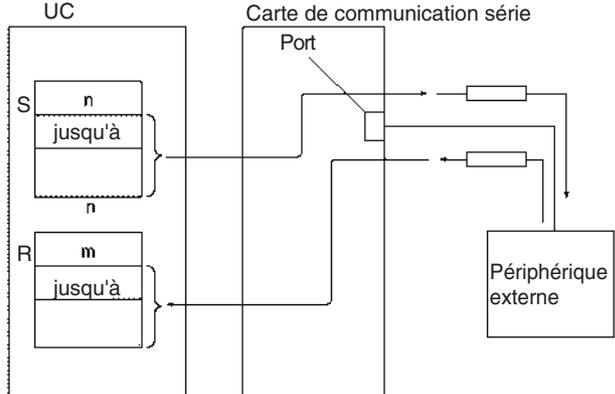
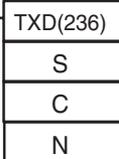
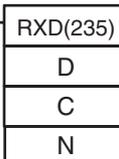
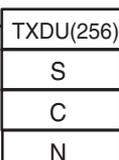
### 3-21 Instructions des cartes d'E/S standard

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>I/O REFRESH</b> IORF @ IORF 097	 <p>St : Mot de début E : Mot de fin</p>	Rafraîchit les mots E/S spécifiés.  Zone de bits d'E/S ou zone de bits de carte E/S spéciales  	Sortie Obligatoire
<b>7-SEGMENT DECODER</b> SDEC @ SDEC 078	 <p>S : Mot source Di : Désignateur de chiffres D : 1er mot de destination</p>	Convertit le contenu hexadécimal des chiffres désignés en code d'affichage 8 bits, 7 segments et le place dans les 8 bits supérieurs ou inférieurs des mots de destination spécifiés.  	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DIGITAL SWITCH INPUT</b>  DSW 210  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	 <p>I : Mot d'entrée de données (D0 à D3)                      O : Mot de sortie                      D : 1er mot de résultat                      C1 : Nombre de chiffres                      C2 : Mot système</p>	Lit la valeur définie sur un interrupteur digital externe (ou un interrupteur à roue codeuse) raccordé à une carte d'entrée ou une carte de sortie et enregistre les données BCD (4 ou 8 chiffres) dans les mots spécifiés.	Sortie Obligatoire
<b>TEN KEY INPUT</b>  TKY 211  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	 <p>I: Mot d'entrée de données                      D1 : 1er mot de registre                      D2 : Mot d'entrée touche</p>	Lit les données numériques à partir d'un clavier de dix touches raccordé à une carte d'entrée et enregistre jusqu'à 8 chiffres de données BCD dans les mots spécifiés.	Sortie Obligatoire
<b>HEXADECIMAL KEY INPUT</b>  HKY 212  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	 <p>I : Mot d'entrée de données                      O : Mot de sortie                      D : 1er mot de registre                      C : Mot système</p>	Lit des données numériques à partir d'un clavier hexadécimal raccordé à une carte d'entrée et une carte de sortie, et enregistre jusqu'à 8 chiffres de données hexadécimales dans les mots spécifiés.	Sortie Obligatoire
<b>MATRIX INPUT</b>  MTR 213  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	 <p>I : Mot d'entrée de données                      O : Mot de sortie                      D : 1er mot de destination                      C : Mot système</p>	Entre jusqu'à 64 signaux à partir d'une matrice de 8 × 8 connectée à une carte d'entrée et une carte de sortie (avec 8 points d'entrée et 8 points de sortie) et enregistre ces données 64 bits dans les 4 mots de destination.	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>7-SEGMENT DISPLAY OUTPUT</b></p> <p>7SEG 214</p> <p>(UC série CS/CJ Ver. 2.0 ou supérieure uniquement)</p>	 <p>S : 1er mot source O : Mot de sortie C : Données de contrôle D : Mot système</p>	<p>Convertit les données source (BCD 4 chiffres ou 8 chiffres) en données d'affichage à 7 segments et place ces données dans le mot de sortie spécifié.</p>	<p>Sortie Obligatoire</p>
<p><b>INTELLIGENT I/O READ</b></p> <p>IORD @IORD 222</p>	 <p>C : Données de contrôle S : Source du transfert et nombre de mots D : Destination du transfert et nombre de mots</p>	<p>Lit le contenu de la zone de mémoire pour la carte E/S spéciales ou la carte de bus UC (voir remarque).</p>  <p><b>Remarque</b> Les UC série CS/CJ Ver. 2.0 ou supérieure (y compris les UC CS1-H, CJ1-H et CJ1M du lot numéro 030418 ou ultérieur) peuvent lire à partir des cartes réseau.</p>	<p>Sortie Obligatoire</p>
<p><b>INTELLIGENT I/O WRITE</b></p> <p>IOWR @IOWR 223</p>	 <p>C : Données de contrôle S : Source du transfert et nombre de mots D : Destination du transfert et nombre de mots</p>	<p>Sort le contenu de la zone de mémoire E/S de l'UC sur la carte E/S spéciales ou la carte de bus UC (voir remarque).</p>  <p><b>Remarque</b> Les UC série CS/CJ Ver. 2.0 ou supérieure (y compris les UC CS1-H, CJ1-H et CJ1M du lot numéro 030418 ou ultérieur) peuvent écrire sur les cartes réseau.</p>	<p>Sortie Obligatoire</p>
<p><b>CPU BUS UNIT I/O REFRESH (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b></p> <p>DLNK @DLNK 226</p>	 <p>N : Numéro d'unité</p>	<p>Met immédiatement les E/S à jour sur la carte réseau avec le numéro de carte spécifié.</p>	<p>Sortie obligatoire</p>

### 3-22 Instructions de communications série

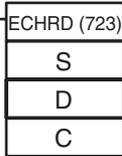
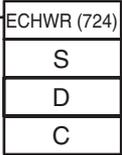
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>PROTOCOL MACRO</b>  PMCR @PMCR 260	 <p>C1 : Mot de contrôle 1                      C2 : Mot de contrôle 2                      S : 1er mot envoyé                      R : 1er mot reçu</p>	<p>Appelle et exécute une séquence de communication enregistrée sur une carte de communication série (série CS ou autre).</p> 	Sortie Obligatoire
<b>TRANSMIT</b>  TXD @TXD 236	 <p>S : 1er mot source                      C : Mot de contrôle                      N : Nombre d'octets                      0000 à 0100 hex                      (0 à 256 décimal)</p>	Sort le nombre d'octets de données spécifié du port RS-232C intégré à l'UC.	Sortie Obligatoire
<b>RECEIVE</b>  RXD @RXD 235	 <p>D : 1er mot de destination                      C : Mot de contrôle                      N : Nombre d'octets à sauvegarder                      0000 à 0100 hex                      (0 à 256 décimal)</p>	Lit le nombre d'octets de données spécifié à partir du port RS-232C intégré à l'UC.	Sortie Obligatoire
<b>TRANSMIT VIA SERIAL COMMUNICATIONS UNIT</b>  TXDU @TXDU 256	 <p>S : 1er mot source                      C : 1er mot de contrôle                      N : Nombre d'octets                      (0000 à 0256 BCD)</p>	Sort le nombre d'octets de données spécifié sans conversion du port série d'une carte de communications série avec une version de carte 1.2 ou supérieure. Les données sont sorties en mode sans protocole avec le code de début et le code de fin (le cas échéant) spécifiés dans la zone DM allouée.	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<b>RECEIVE VIA SERIAL COMMUNICATIONS UNIT</b> RXDU @RXDU 255	<table border="1"> <tr><td>RXDU(255)</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> <tr><td>N</td></tr> </table> <p>D : 1er mot de destination                      C : 1er mot de contrôle                      N : Nombre d'octets à stocker (0000 à 0100 hex)</p>	RXDU(255)	D	C	N	Lit le nombre d'octets de données spécifié, en partant du premier mot spécifié, sur le port série d'une carte de communications série avec une version de carte 1.2 ou supérieure. Les données sont lues en mode sans protocole avec le code de début et le code de fin (le cas échéant) spécifiés dans la zone de configuration DM allouée.	
RXDU(255)							
D							
C							
N							
<b>CHANGE SERIAL PORT SETUP</b> STUP @STUP 237	<table border="1"> <tr><td>STUP(237)</td></tr> <tr><td>C</td></tr> <tr><td>S</td></tr> </table> <p>C : Mot de contrôle (port)                      S : Premier mot source</p>	STUP(237)	C	S	Change les paramètres de communication d'un port série sur l'UC, la carte de communications série (carte réseau) ou la carte de communications série. STUP(237) permet ainsi de changer le mode protocole pendant le fonctionnement de l'API.	Sortie Obligatoire	
STUP(237)							
C							
S							

### 3-23 Instructions réseaux

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<b>NETWORK SEND</b> SEND @SEND 090	<table border="1"> <tr><td>SEND(090)</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p>S : 1er mot source                      D : 1er mot de destination                      C : 1er mot de contrôle</p>	SEND(090)	S	D	C	Transmet les données à un nœud du réseau.  	Sortie Obligatoire
SEND(090)							
S							
D							
C							
<b>NETWORK RECEIVE</b> RECV @RECV 098	<table border="1"> <tr><td>RECV(098)</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p>S : 1er mot source                      D : 1er mot de destination                      C : 1er mot de contrôle</p>	RECV(098)	S	D	C	Demande la transmission des données à partir d'un nœud du réseau et reçoit les données.  	Sortie Obligatoire
RECV(098)							
S							
D							
C							

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution				
<b>DELIVER COMMAND</b>  CMND @CMND 490	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>CMND(490)</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p><b>S</b> : 1er mot de commande  <b>D</b> : 1er mot de réponse  <b>C</b> : 1er mot de contrôle</p>	CMND(490)	S	D	C	Envoie la commande FINS et reçoit la réponse  	Sortie Obligatoire
CMND(490)							
S							
D							
C							
<b>EXPLICIT MESSAGE SEND</b>  EXPLT 720  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>EXPLT (720)</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p><b>S</b> : 1er mot envoi message  <b>D</b> : 1er mot message reçu  <b>C</b> : 1er mot de contrôle</p>	EXPLT (720)	S	D	C	Envoie un message explicite avec un code de service.	Sortie Obligatoire
EXPLT (720)							
S							
D							
C							
<b>EXPLICIT GET ATTRIBUTE</b>  EGATR 721  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>EGATR (721)</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table> <p><b>S</b> : 1er mot envoi message  <b>D</b> : 1er mot message reçu  <b>C</b> : 1er mot de contrôle message</p>	EGATR (721)	S	D	C	Lit les informations d'état avec un message explicite ' (Get Attribute Single, Service Code: 0E hex).	Sortie Obligatoire
EGATR (721)							
S							
D							
C							
<b>EXPLICIT SET ATTRIBUTE</b>  ESATR 722  (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>ESATR (722)</td></tr> <tr><td>S</td></tr> <tr><td>C</td></tr> </table> <p><b>S</b> : 1er mot envoi message  <b>C</b> : Premier mot contrôle</p>	ESATR (722)	S	C	Ecrit les informations d'état avec un message explicite (Set Attribute Single, Service Code : 0E hex)	Sortie Obligatoire	
ESATR (722)							
S							
C							

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>EXPLICIT WORD READ</b></p> <p>ECHRD 723</p> <p>(UC série CS/CJ Ver. 2.0 ou supérieure uniquement)</p>	 <p><b>S</b> : 1er mot source dans UC distante</p> <p><b>D</b> : 1er mot de destination dans UC locale</p> <p><b>C</b> : 1er mot de contrôle</p>	<p>Lit les données sur l'UC locale à partir d'une UC distante dans le réseau. (L'UC distante doit prendre en charge les messages explicites.)</p>	<p>Sortie Obligatoire</p>
<p><b>EXPLICIT WORD WRITE</b></p> <p>ECHWR 724</p> <p>(UC série CS/CJ Ver. 2.0 ou supérieure uniquement)</p>	 <p><b>S</b> : 1er mot source dans UC locale</p> <p><b>D</b> : 1er mot de des- tination dans UC distante</p> <p><b>C</b> : 1er mot de contrôle</p>	<p>Ecrit les données de l'UC locale sur une UC distante dans le réseau. (L'UC distante doit prendre en charge les messages explicites.)</p>	<p>Sortie Obligatoire</p>

### 3-24 Instructions de mémoire de fichiers

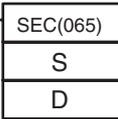
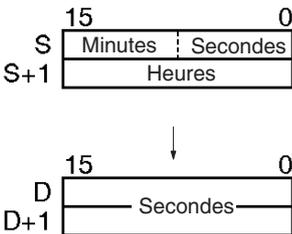
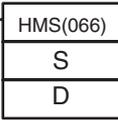
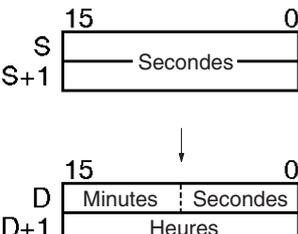
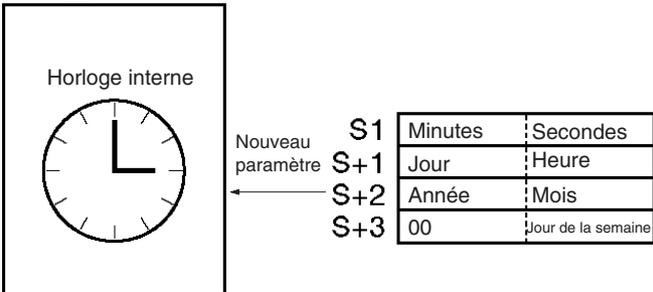
Instruction Mnémorique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution					
<p><b>READ DATA FILE</b> FREAD @FREAD 700</p>	<table border="1" style="margin-left: 20px;"> <tr><td style="padding: 2px;">FREAD(700)</td></tr> <tr><td style="padding: 2px;">C</td></tr> <tr><td style="padding: 2px;">S1</td></tr> <tr><td style="padding: 2px;">S2</td></tr> <tr><td style="padding: 2px;">D</td></tr> </table> <p style="margin-left: 20px;">C : Mot de contrôle S1 : 1er mot source S2 : Nom de fichier D : 1er mot de destination</p>	FREAD(700)	C	S1	S2	D	<p>Lit les données ou la quantité de données indiquées du fichier de données spécifié à l'intérieur de la mémoire de fichiers dans la zone de données spécifiée sur l'UC.</p> <p>Adresse de début de lecture spécifiée dans S1+2 et S1+3</p> <p>Fichier spécifié dans S2</p> <p>UC</p> <p>Nombre de mots spécifiés dans S1 et S1+1</p> <p>Carte mémoire ou mémoire de fichiers EM (Spécifiée par le 4ème chiffre de C.)</p> <p>Nombre de mots écrits dans D et D+1.</p> <p>Fichier spécifié dans S2</p> <p>Nombre de mots</p> <p>UC</p> <p>D</p> <p>D+1</p> <p>Carte mémoire ou mémoire de fichiers EM (Spécifiée par le 4ème chiffre de C.)</p>	<p>Sortie Obligatoire</p>
FREAD(700)								
C								
S1								
S2								
D								
<p><b>WRITE DATA FILE</b> FWRIT @FWRIT 701</p>	<table border="1" style="margin-left: 20px;"> <tr><td style="padding: 2px;">FWRIT(701)</td></tr> <tr><td style="padding: 2px;">C</td></tr> <tr><td style="padding: 2px;">D1</td></tr> <tr><td style="padding: 2px;">D2</td></tr> <tr><td style="padding: 2px;">S</td></tr> </table> <p style="margin-left: 20px;">C : Mot de contrôle D1 : 1er mot de destination D2 : Nom de fichier S : 1er mot source</p>	FWRIT(701)	C	D1	D2	S	<p>Ecrase ou ajoute des données du fichier de données spécifié dans la mémoire de fichiers avec les données spécifiées de la zone de données dans l'UC. Si le fichier spécifié n'existe pas, un nouveau fichier est créé avec ce nom.</p> <p>UC</p> <p>Adresse de début spécifiée dans S</p> <p>Mot de début spécifié dans D1+2 et D1+3</p> <p>Fichier spécifié dans D2</p> <p>Nombre de mots spécifiés dans D1 et D1+1</p> <p>Ecraser</p> <p>Carte mémoire ou mémoire de fichiers EM (Spécifiée par le 4ème chiffre de C.)</p> <p>UC</p> <p>Adresse de début spécifiée dans S</p> <p>Fin de fichier</p> <p>Fichier spécifié dans D2</p> <p>Donnée s</p> <p>Nombre de mots spécifiés dans D1 et D1+1</p> <p>Ajouter</p> <p>Carte mémoire ou mémoire de fichiers EM</p> <p>UC</p> <p>Adresse de début spécifiée dans S</p> <p>Début de fichier</p> <p>Fichier spécifié</p> <p>Nouveau fichier créé</p> <p>Nombre de mots spécifiés dans D1 et D1+1</p> <p>Carte mémoire ou mémoire de fichiers EM (Spécifiée par le 4ème chiffre de C.)</p>	<p>Sortie Obligatoire</p>
FWRIT(701)								
C								
D1								
D2								
S								

### 3-25 Instructions d'affichage

Instruction Mnémorique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution			
<b>DISPLAY MESSAGE</b> MSG @MSG 046	<table border="1"> <tr><td>MSG(046)</td></tr> <tr><td>N</td></tr> <tr><td>M</td></tr> </table> <p>N : Numéro du message                      M : 1er mot message</p>	MSG(046)	N	M	Lit les seize mots d'ASCII étendu spécifiés et affiche le message sur un équipement périphérique tel qu'une console de programmation.	Sortie Obligatoire
MSG(046)						
N						
M						

### 3-26 Instructions de temporisation

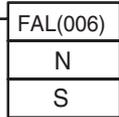
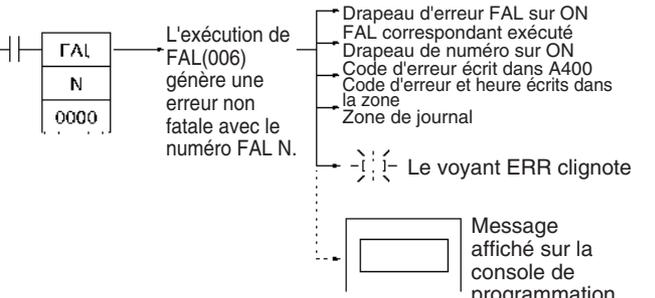
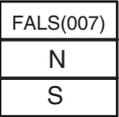
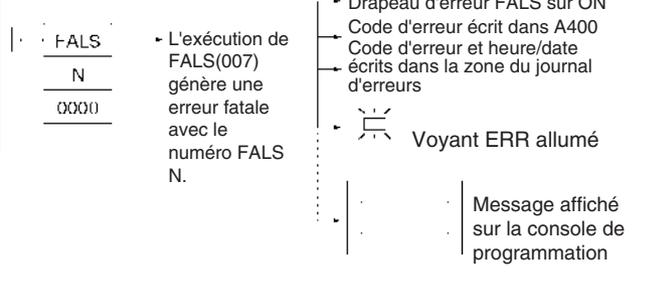
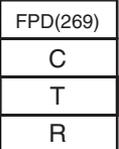
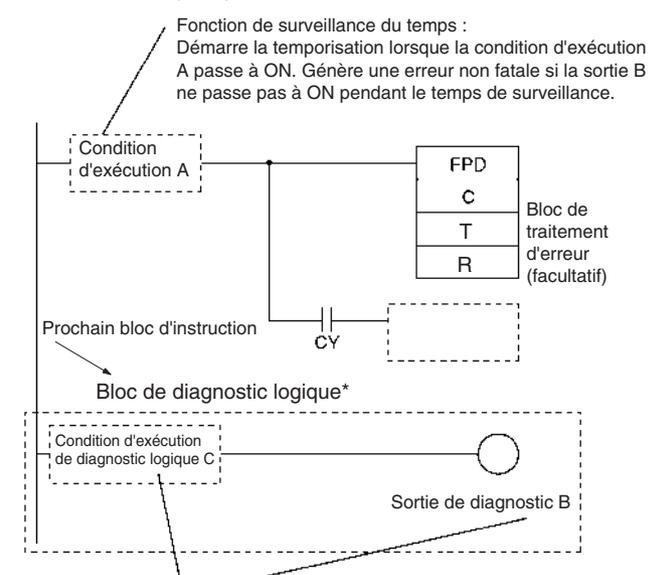
Instruction Mnémorique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution																																							
<b>CALENDAR ADD</b> CADD @CADD 730	<table border="1"> <tr><td>CADD(730)</td></tr> <tr><td>C</td></tr> <tr><td>T</td></tr> <tr><td>R</td></tr> </table> <p>C : 1er mot de calendrier                      T : 1er mot de temps                      R : 1er mot de résultat</p>	CADD(730)	C	T	R	Ajoute des données de temps aux données de calendrier dans les mots spécifiés. <table style="margin-left: 40px;"> <tr><td>15</td><td>87</td><td>0</td></tr> <tr><td>C</td><td>Minutes</td><td>Secondes</td></tr> <tr><td>C+1</td><td>Jour</td><td>Heure</td></tr> <tr><td>C+2</td><td>Année</td><td>Mois</td></tr> </table> <table style="margin-left: 40px;"> <tr><td>+</td></tr> <tr><td>15</td><td>87</td><td>0</td></tr> <tr><td>T</td><td>Minutes</td><td>Secondes</td></tr> <tr><td>T+1</td><td colspan="2">Heures</td></tr> </table> <table style="margin-left: 40px;"> <tr><td>↓</td></tr> <tr><td>15</td><td>87</td><td>0</td></tr> <tr><td>R</td><td>Minutes</td><td>Secondes</td></tr> <tr><td>R+1</td><td>Jour</td><td>Heure</td></tr> <tr><td>R+2</td><td>Année</td><td>Mois</td></tr> </table>	15	87	0	C	Minutes	Secondes	C+1	Jour	Heure	C+2	Année	Mois	+	15	87	0	T	Minutes	Secondes	T+1	Heures		↓	15	87	0	R	Minutes	Secondes	R+1	Jour	Heure	R+2	Année	Mois	Sortie Obligatoire
CADD(730)																																										
C																																										
T																																										
R																																										
15	87	0																																								
C	Minutes	Secondes																																								
C+1	Jour	Heure																																								
C+2	Année	Mois																																								
+																																										
15	87	0																																								
T	Minutes	Secondes																																								
T+1	Heures																																									
↓																																										
15	87	0																																								
R	Minutes	Secondes																																								
R+1	Jour	Heure																																								
R+2	Année	Mois																																								
<b>CALENDAR SUBTRACT</b> CSUB @CSUB 731	<table border="1"> <tr><td>CSUB(731)</td></tr> <tr><td>C</td></tr> <tr><td>T</td></tr> <tr><td>R</td></tr> </table> <p>C : 1er mot de calendrier                      T : 1er mot de temps                      R : 1er mot de résultat</p>	CSUB(731)	C	T	R	Soustrait des données de temps aux données de calendrier dans les mots spécifiés. <table style="margin-left: 40px;"> <tr><td>15</td><td>87</td><td>0</td></tr> <tr><td>C</td><td>Minutes</td><td>Secondes</td></tr> <tr><td>C+1</td><td>Jour</td><td>Heure</td></tr> <tr><td>C+2</td><td>Année</td><td>Mois</td></tr> </table> <table style="margin-left: 40px;"> <tr><td>15</td><td>87</td><td>0</td></tr> <tr><td>T</td><td>Minutes</td><td>Secondes</td></tr> <tr><td>T+1</td><td colspan="2">Heures</td></tr> </table> <table style="margin-left: 40px;"> <tr><td>↓</td></tr> <tr><td>15</td><td>87</td><td>0</td></tr> <tr><td>R</td><td>Minutes</td><td>Secondes</td></tr> <tr><td>R+1</td><td>Jour</td><td>Heure</td></tr> <tr><td>R+2</td><td>Année</td><td>Mois</td></tr> </table>	15	87	0	C	Minutes	Secondes	C+1	Jour	Heure	C+2	Année	Mois	15	87	0	T	Minutes	Secondes	T+1	Heures		↓	15	87	0	R	Minutes	Secondes	R+1	Jour	Heure	R+2	Année	Mois	Sortie Obligatoire	
CSUB(731)																																										
C																																										
T																																										
R																																										
15	87	0																																								
C	Minutes	Secondes																																								
C+1	Jour	Heure																																								
C+2	Année	Mois																																								
15	87	0																																								
T	Minutes	Secondes																																								
T+1	Heures																																									
↓																																										
15	87	0																																								
R	Minutes	Secondes																																								
R+1	Jour	Heure																																								
R+2	Année	Mois																																								

Instruction Mnémorique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>HOURS TO SECONDS</b> SEC @SEC 065	 <p>S : 1er mot source D : 1er mot de destination</p>	Convertit les données de temps au format heures/minutes/secondes en un temps équivalent en secondes uniquement. 	Sortie Obligatoire
<b>SECONDS TO HOURS</b> HMS @HMS 066	 <p>S : 1er mot source D : 1er mot de destination</p>	Convertit les données en secondes en un temps équivalent au format heures/minutes/secondes. 	Sortie Obligatoire
<b>CLOCK ADJUSTMENT</b> DATE @DATE 735	 <p>S : 1er mot source</p>	Règle le paramètre de l'horloge interne sur celui des mots sources spécifiés. UC 	Sortie Obligatoire

### 3-27 Instructions de débogage

Instruction Mnémorique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>TRACE MEMORY SAMPLING</b> TRSM 045		Lorsque TRSM(045) est exécuté, l'état d'un bit ou d'un mot présélectionné est échantillonné et enregistré dans la mémoire d'enregistrement. TRSM(045) peut être utilisé n'importe où dans le programme, autant de fois que vous le souhaitez.	Sortie Non obligatoire

### 3-28 Instructions de diagnostic d'erreur

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>FAILURE ALARM</b> FAL @FAL 006	 <p>N : Numéro FAL                      S : 1er mot de message ou code d'erreur à générer</p>	<p>Génère ou efface les erreurs non fatales définies par l'utilisateur. Les erreurs non fatales n'interrompent pas le fonctionnement de l'API. Génère également des erreurs non fatales avec le système.</p>  <p>L'exécution de FAL(006) génère une erreur non fatale avec le numéro FAL N.</p> <ul style="list-style-type: none"> <li>Drapeau d'erreur FAL sur ON</li> <li>FAL correspondant exécuté</li> <li>Drapeau de numéro sur ON</li> <li>Code d'erreur écrit dans A400</li> <li>Code d'erreur et heure écrits dans la zone</li> <li>Zone de journal</li> <li>Le voyant ERR clignote</li> <li>Message affiché sur la console de programmation</li> </ul>	Sortie Obligatoire
<b>SEVERE FAILURE ALARM</b> FALS 007	 <p>N : Numéro FALS                      S : 1er mot de message ou code d'erreur à générer</p>	<p>Génère des erreurs fatales définies par l'utilisateur. Les erreurs fatales arrêtent le fonctionnement de l'API. Génère également des erreurs fatales avec le système.</p>  <p>L'exécution de FALS(007) génère une erreur fatale avec le numéro FALS N.</p> <ul style="list-style-type: none"> <li>Drapeau d'erreur FALS sur ON</li> <li>Code d'erreur écrit dans A400</li> <li>Code d'erreur et heure/date écrits dans la zone du journal d'erreurs</li> <li>Voyant ERR allumé</li> <li>Message affiché sur la console de programmation</li> </ul>	Sortie Obligatoire
<b>FAILURE POINT DETECTION</b> FPD 269	 <p>C : Mot de contrôle                      T : Temps de surveillance                      R : 1er mot de registre</p>	<p>Permet de diagnostiquer une erreur au niveau d'un bloc d'instruction en surveillant le temps entre l'exécution de l'instruction FPD(269) et l'exécution d'un diagnostic et en retrouvant l'entrée qui empêche une sortie d'être mise sous tension.</p> <p>Fonction de surveillance du temps :                      Démarre la temporisation lorsque la condition d'exécution A passe à ON. Génère une erreur non fatale si la sortie B ne passe pas à ON pendant le temps de surveillance.</p>  <p>Prochain bloc d'instruction</p> <p>Bloc de diagnostic logique*</p> <p>Sortie de diagnostic B</p> <p>La fonction de diagnostic logique détermine quelle entrée dans C empêche la sortie B de passer à ON.</p>	Sortie Obligatoire

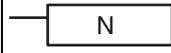
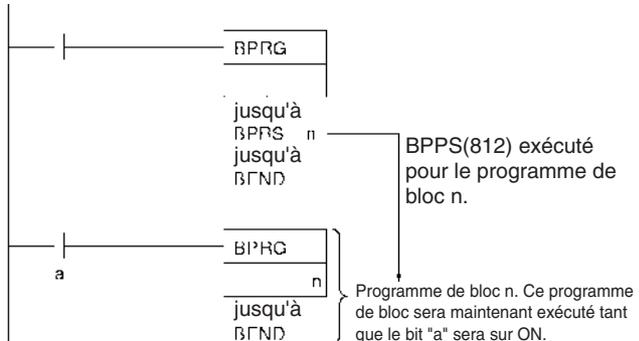
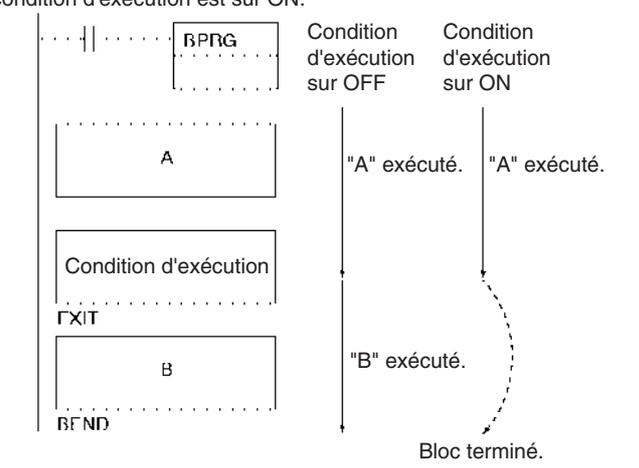
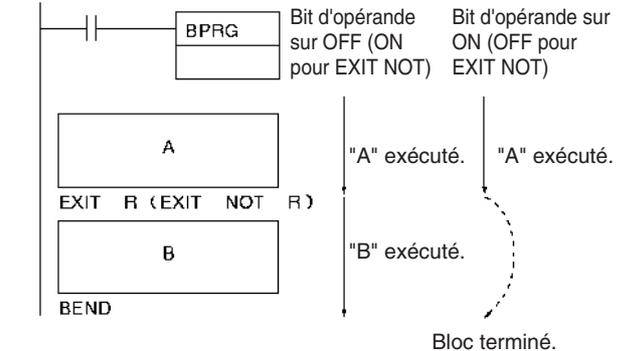
## 3-29 Autres instructions

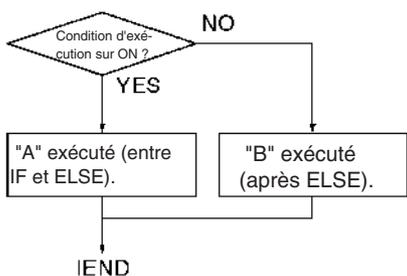
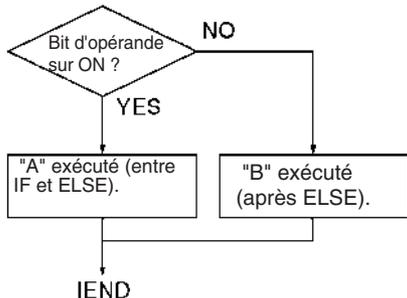
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution			
<b>SET CARRY</b> STC @STC 040	— <table border="1" style="display: inline-table;"><tr><td>STC(040)</td></tr></table>	STC(040)	Définit le drapeau de passage (CY).	Sortie Obligatoire		
STC(040)						
<b>CLEAR CARRY</b> CLC @CLC 041	— <table border="1" style="display: inline-table;"><tr><td>CLC(041)</td></tr></table>	CLC(041)	Met le drapeau de passage à OFF (CY).	Sortie Obligatoire		
CLC(041)						
<b>SELECT EM BANK</b> EMBC @EMBC 281	— <table border="1" style="display: inline-table;"><tr><td>EMBC(281)</td></tr><tr><td>N</td></tr></table> N : Numéro de banque EM	EMBC(281)	N	Change la banque EM courante.	Sortie Obligatoire	
EMBC(281)						
N						
<b>EXTEND MAXIMUM CYCLE TIME</b> WDT @WDT 094	— <table border="1" style="display: inline-table;"><tr><td>WDT(094)</td></tr><tr><td>T</td></tr></table> T : Param. temporisateur	WDT(094)	T	Prolonge le temps de cycle maximum, mais seulement pour le cycle dans lequel cette instruction est exécutée.	Sortie Obligatoire	
WDT(094)						
T						
<b>SAVE CONDITION FLAGS (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> CCS @CCS 282	— <table border="1" style="display: inline-table;"><tr><td>CCS(282)</td></tr></table>	CCS(282)	Sauvegarde l'état des drapeaux de condition.	Sortie Obligatoire		
CCS(282)						
<b>LOAD CONDITION FLAGS (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> CCL @CCL 283	— <table border="1" style="display: inline-table;"><tr><td>CCL(283)</td></tr></table>	CCL(283)	Lit l'état des drapeaux de condition qui ont été sauvegardés.	Sortie Obligatoire		
CCL(283)						
<b>CONVERT ADDRESS FROM CV (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> FRMCV @FRMCV 284	— <table border="1" style="display: inline-table;"><tr><td>FRMCV(284)</td></tr><tr><td>S</td></tr><tr><td>D</td></tr></table> S : Mot contenant l'adresse mémoire série CV D : Registre d'index destination	FRMCV(284)	S	D	Convertit une adresse mémoire d'API série CV en une adresse mémoire d'API série CS/CJ équivalente.	Sortie Obligatoire
FRMCV(284)						
S						
D						
<b>CONVERT ADDRESS TO CV (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</b> TOCV @TOCV 285	— <table border="1" style="display: inline-table;"><tr><td>TOCV(285)</td></tr><tr><td>S</td></tr><tr><td>D</td></tr></table> S : Registre d'index contenant l'adresse mémoire série CS D : Mot de destination	TOCV(285)	S	D	Convertit une adresse mémoire d'API série CS/CJ en une adresse mémoire d'API série CV équivalente.	Sortie Obligatoire
TOCV(285)						
S						
D						

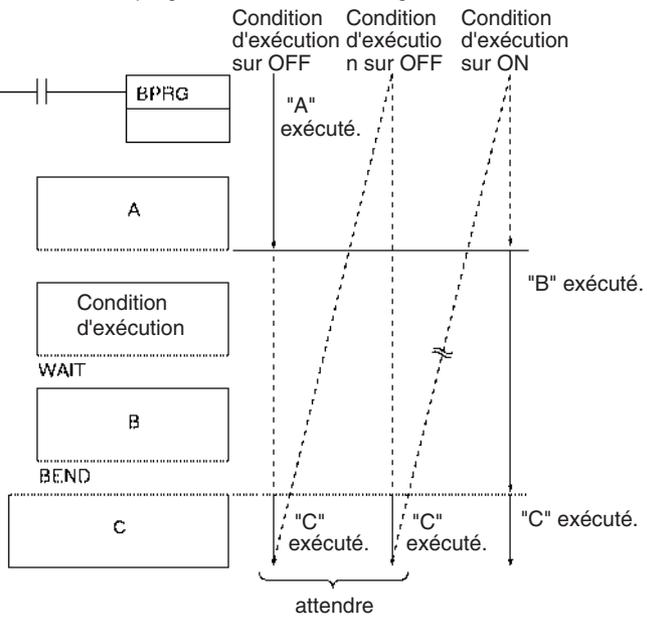
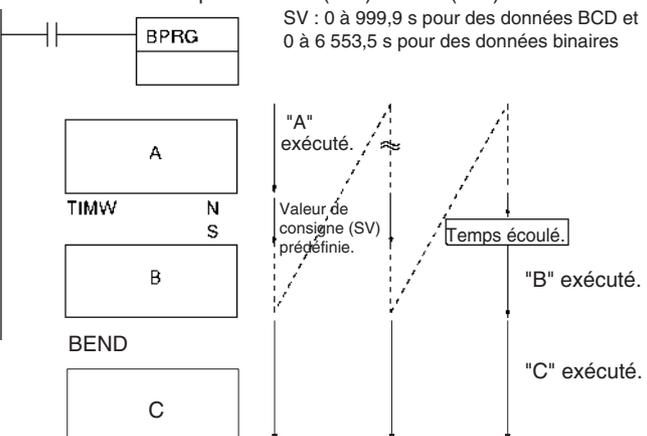
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>DISABLE PERIPHERAL SERVICING (UC CS1D pour les systèmes à une seule UC, CS1-H, CJ1-H ou CJ1M uniquement)</b> IOSP @IOSP 287		Désactive le traitement des périphériques pendant l'exécution du programme dans l'un des modes de traitement parallèle ou dans le mode prioritaire du traitement des périphériques.	Sortie Obligatoire
<b>ENABLE PERIPHERAL SERVICING (UC CS1D pour les systèmes à une seule UC, CS1-H, CJ1-H ou CJ1M uniquement)</b> IORS 288		Active le traitement des périphériques qui a été désactivé par IOSP(287) pour l'exécution du programme dans l'un des modes de traitement parallèle ou dans le mode prioritaire du traitement des périphériques.	Sortie Non obligatoire

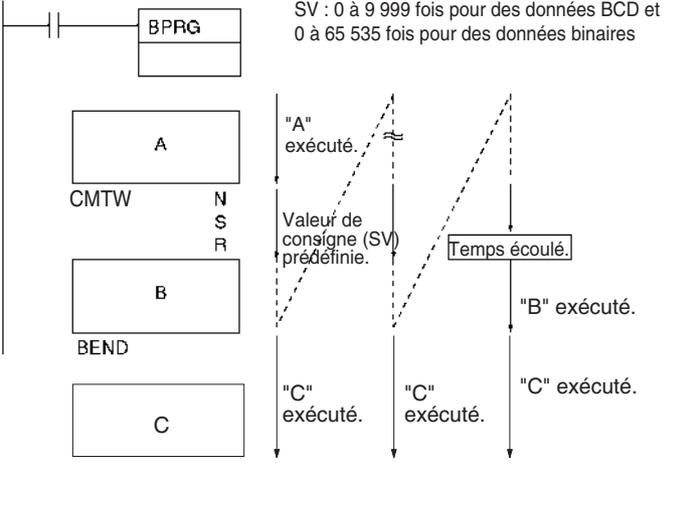
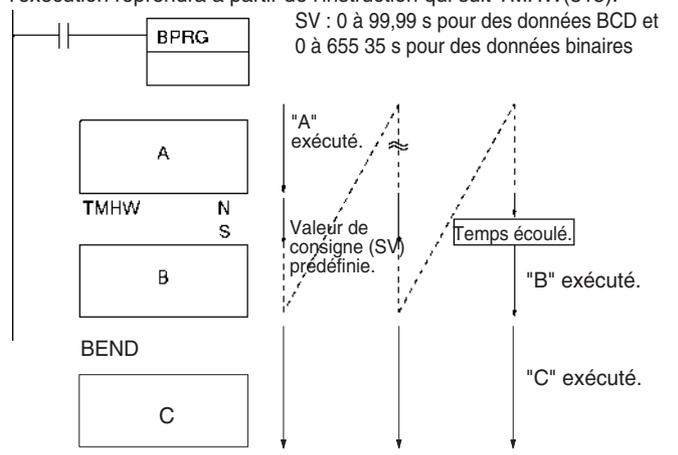
### 3-30 Instructions de programmation de bloc

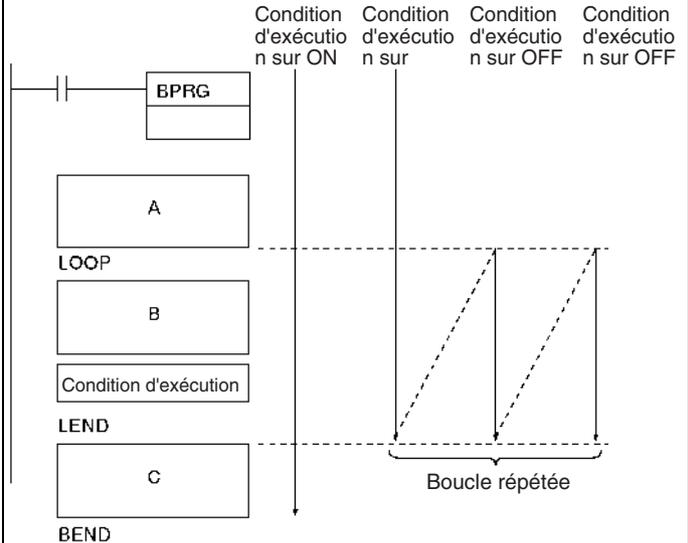
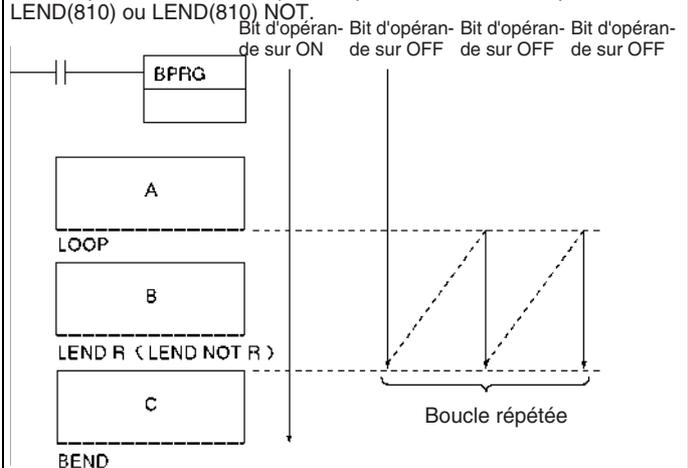
Instruction Mnémonique Code	Symbole/ Opérande	Fonctions	Emplacement Condition d'exécution
<b>BLOCK PROGRAM BEGIN</b> BPRG 096	 N : Numéro de programme de bloc	Définition d'une zone de programmation de bloc. A chaque BPRG(096) doit correspondre un BEND(801). 	Sortie Obligatoire
<b>BLOCK PROGRAM END</b> BEND 801		Définition d'une zone de programmation de bloc. A chaque BPRG(096) doit correspondre un BEND(801).	Programme de bloc Obligatoire
<b>BLOCK PROGRAM PAUSE</b> BPPS 811	 N : Numéro de bloc programme	Mise en pause et redémarrage du programme de bloc spécifié à partir d'un autre programme de bloc. 	Programme de bloc Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonctions	Emplacement Condition d'exécution
<b>BLOCK PROGRAM RESTART</b>  BPRS 812	BPRS (812)  N : Numéro de programme de bloc	Mise en pause et redémarrage du programme de bloc spécifié à partir d'un autre programme de bloc.  	Programme de bloc Obligatoire
<b>CONDITIONAL BLOCK EXIT</b>  EXIT 806	EXIT(806) B : opérande bit	EXIT(806) sans un bit d'opérande quitte le programme si la condition d'exécution est sur ON.  	Programme de bloc Obligatoire
<b>CONDITIONAL BLOCK EXIT</b>  EXIT 806	EXIT(806)B B : opérande bit	EXIT(806) sans un bit d'opérande quitte le programme si la condition d'exécution est sur ON.  	Programme de bloc Obligatoire
<b>CONDITIONAL BLOCK EXIT NOT</b>  EXIT NOT 806	EXIT NOT(806) B B : opérande bit	EXIT(806) sans un bit d'opérande quitte le programme si la condition d'exécution est à OFF.	Programme de bloc Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonctions	Emplacement Condition d'exécution
<b>CONDITIONAL BLOCK BRANCHING</b>  IF 802	IF (802)	<p>Si la condition d'exécution est sur ON, les instructions entre IF(802) et ELSE(803) seront exécutées et si la condition d'exécution est sur OFF, les instructions entre ELSE(803) et IEND(804) seront exécutées.</p> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 5px; margin-right: 20px;">Condition d'exécution</div> <div style="margin-right: 20px;">IF</div> <div style="border: 1px solid black; padding: 5px; margin-right: 20px;">A</div> <div style="margin-right: 20px;">ELSE</div> <div style="border: 1px solid black; padding: 5px; margin-right: 20px;">B</div> <div style="margin-right: 20px;">IEND</div> </div> 	Programme de bloc Obligatoire
<b>CONDITIONAL BLOCK BRANCHING</b>  IF 802	IF (802) B B : opérande bit	<p>Si le bit d'opérande est sur ON, les instructions entre IF(802) et ELSE(803) seront exécutées. Si le bit d'opérande est sur OFF, les instructions entre ELSE(803) et IEND(804) seront exécutées.</p> <div style="display: flex; align-items: center;"> <div style="margin-right: 20px;">IF R (IF NOT R)</div> <div style="border: 1px solid black; padding: 5px; margin-right: 20px;">A</div> <div style="margin-right: 20px;">ELSE</div> <div style="border: 1px solid black; padding: 5px; margin-right: 20px;">B</div> <div style="margin-right: 20px;">IEND</div> </div> 	Programme de bloc Obligatoire
<b>CONDITIONAL BLOCK BRANCHING (NOT)</b>  IF NOT 802	IF (802) NOT B B : opérande bit	<p>Les instructions comprises entre IF(802) et ELSE(803) seront exécutées et, si le bit d'opérande est à ON, les instructions entre ELSE(803) et IEND(804) sont exécutées si le bit d'opérande est à OFF.</p>	Programme de bloc Obligatoire
<b>CONDITIONAL BLOCK BRANCHING (ELSE)</b>  ELSE 803	---	<p>Si l'instruction ELSE(803) est omise et que le bit d'opérande est à ON, les instructions entre IF(802) et IEND(804) seront exécutées.</p>	Programme de bloc Obligatoire
<b>CONDITIONAL BLOCK BRANCHING END</b>  IEND 804	---	<p>Si le bit d'opérande est à OFF, seules les instructions après IEND(804) seront exécutées.</p>	Programme de bloc Obligatoire

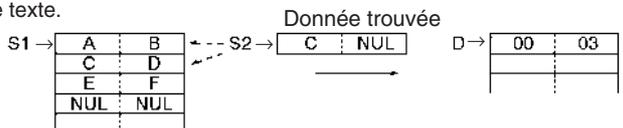
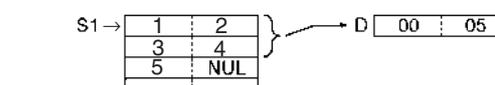
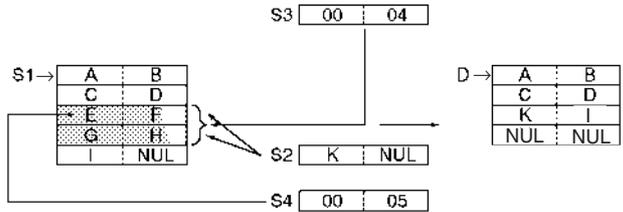
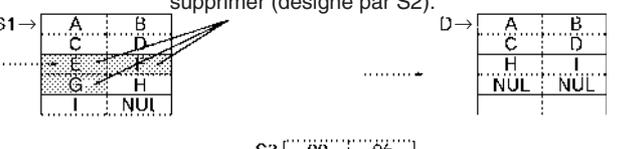
Instruction Mnémonique Code	Symbole/ Opérande	Fonctions	Emplacement Condition d'exécution
<p><b>ONE CYCLE AND WAIT</b></p> <p>WAIT 805</p>	<p>WAIT(805)</p>	<p>Si la condition d'exécution est sur ON pour WAIT(805), les autres instructions du programme de bloc seront ignorées.</p>  <p>Condition d'exécution sur OFF Condition d'exécution sur OFF Condition d'exécution sur ON</p> <p>"A" exécuté.</p> <p>"B" exécuté.</p> <p>"C" exécuté.</p> <p>"C" exécuté.</p> <p>attendre</p>	<p>Programme de bloc Obligatoire</p>
<p><b>ONE CYCLE AND WAIT</b></p> <p>WAIT 805</p>	<p>WAIT(805) B B : opérande de bit</p>	<p>Si le bit d'opérande est à OFF (ON pour WAIT NOT(805)), le reste des instructions dans le programme de bloc sera ignoré. Dans le cycle suivant, aucun élément du programme de bloc ne sera exécuté, sauf pour la condition d'exécution de WAIT(805) ou WAIT(805) NOT. Lorsque la condition d'exécution passera à ON (OFF pour WAIT(805) NOT), l'instruction de WAIT(805) ou WAIT(805) NOT à la fin du programme sera exécutée.</p>	<p>Programme de bloc Obligatoire</p>
<p><b>ONE CYCLE AND WAIT (NOT)</b></p> <p>WAIT NOT 805</p>	<p>WAIT(805) NOTB B : opérande de bit</p>	<p>Si le bit d'opérande est à OFF (ON pour WAIT NOT(805)), les autres instructions du programme de bloc seront ignorées. Dans le cycle suivant, aucun élément du programme de bloc ne sera exécuté, sauf pour la condition d'exécution de WAIT(805) ou WAIT(805) NOT. Lorsque la condition d'exécution passera à ON (OFF pour WAIT(805) NOT), l'instruction de WAIT(805) ou WAIT(805) NOT à la fin du programme sera exécutée.</p>	<p>Programme de bloc Obligatoire</p>
<p><b>TIMER WAIT</b></p> <p>TIMW 813 (BCD)</p> <p>TIMWX 816 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>TIMW(813) N SV</p> <p>TIMWX(816) N SV</p> <p>N : numéro de temporisation SV : valeur de consigne</p>	<p>Retarde l'exécution du programme de bloc jusqu'à la fin du délai spécifié. À l'expiration de la temporisation, l'exécution reprend à partir de l'instruction qui suit TIMW(813)/TIMWX(816).</p>  <p>SV : 0 à 999,9 s pour des données BCD et 0 à 6 553,5 s pour des données binaires</p> <p>"A" exécuté.</p> <p>Valeur de consigne (SV) prédéfinie.</p> <p>Temps écoulé.</p> <p>"B" exécuté.</p> <p>"C" exécuté.</p>	<p>Programme de bloc Obligatoire</p>

Instruction Mnémonique Code	Symbole/ Opérande	Fonctions	Emplacement Condition d'exécution
<p><b>COUNTER WAIT</b> CNTW 814 (BCD)</p> <p>CNTWX 817 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>CNTW(814) N SV</p> <hr/> <p>CNTWX(817) N SV</p> <p>N : numéro de compteur SV : valeur de consigne I : Entrée de comptage</p> <hr/> <p>N : Numéro du compteur SV : valeur de consigne I : entrée de comptage</p>	<p>Retarde l'exécution du reste du programme de bloc jusqu'à ce que le nombre spécifié ait été atteint. L'exécution reprendra à partir de l'instruction qui suit CNTW(814)/CNTWX(817) lorsque le décompte sera terminé.</p> <p>SV : 0 à 9 999 fois pour des données BCD et 0 à 65 535 fois pour des données binaires</p> 	<p>Programme de bloc Obligatoire</p>
<p><b>HIGH-SPEED TIMER WAIT</b> TMHW 815 (BCD)</p> <p>TMHWX 818 (Binaire) (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)</p>	<p>TMHW(815) N SV</p> <hr/> <p>TMHWX(818) N SV</p> <p>N : numéro de temporisation SV : valeur de consigne</p> <hr/> <p>N : numéro de temporisation SV : valeur de consigne</p>	<p>Retarde l'exécution du reste du programme de bloc jusqu'à l'expiration du délai spécifié. À l'expiration de la temporisation, l'exécution reprendra à partir de l'instruction qui suit TMHW(815).</p> <p>SV : 0 à 99,99 s pour des données BCD et 0 à 655 35 s pour des données binaires</p> 	<p>Programme de bloc Obligatoire</p>

Instruction Mnémonique Code	Symbole/ Opérande	Fonctions	Emplacement Condition d'exécution
<p><b>LOOP</b></p> <p>LOOP 809</p>	<p>---</p>	<p>LOOP(809) désigne le début du programme en boucle.</p> 	<p>Programme de bloc Obligatoire</p>
<p><b>LEND</b></p> <p>LEND 810</p>	<p>LEND (810)</p>	<p>LEND(810) ou LEND(810) NOT spécifie la fin de la boucle. Lorsque LEND(810) ou LEND(810) NOT est atteint, l'exécution du programme se boucle sur l'instruction LOOP(809) précédente jusqu'à ce que le bit d'opérande de LEND(810) ou LEND(810) NOT passe à ON ou à OFF (respectivement) ou jusqu'à ce que la condition d'exécution de LEND(810) passe à ON.</p>	<p>Programme de bloc Obligatoire</p>
<p><b>LEND</b></p> <p>LEND 810</p>	<p>LEND (810) B B : opérande bit</p>	<p>Si le bit d'opérande est sur OFF pour LEND(810) (ou sur ON pour LEND(810) NOT), l'exécution de la boucle est répétée à partir de l'instruction qui suit LOOP(809). Si le bit d'opérande est sur ON pour LEND(810) (ou sur OFF pour LEND(810) NOT), la boucle est interrompue et l'exécution reprend à partir de l'instruction qui suit LEND(810) ou LEND(810) NOT.</p>  <p><b>Remarque</b> L'état du bit d'opérande serait inversé pour LEND(810) NOT.</p>	<p>Programme de bloc Obligatoire</p>
<p><b>LEND NOT</b></p> <p>LEND NOT 810</p>	<p>LEND(810) NOT B : opérande bit</p>	<p>LEND(810) ou LEND(810) NOT spécifie la fin de la boucle. Lorsque LEND(810) ou LEND(810) NOT est atteint, l'exécution du programme se boucle sur le précédent LOOP(809) jusqu'à ce que le bit d'opérande de LEND(810) ou LEND(810) NOT passe à ON ou à OFF (respectivement) ou jusqu'à ce que la condition d'exécution de LEND(810) passe à ON.</p>	<p>Programme de bloc Obligatoire</p>

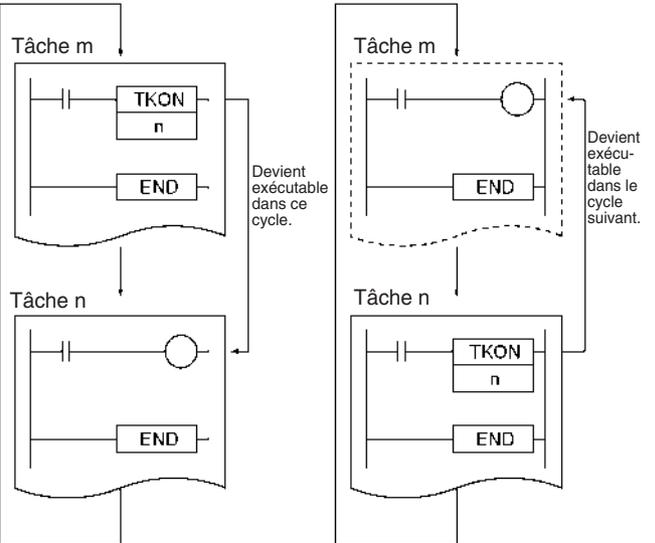
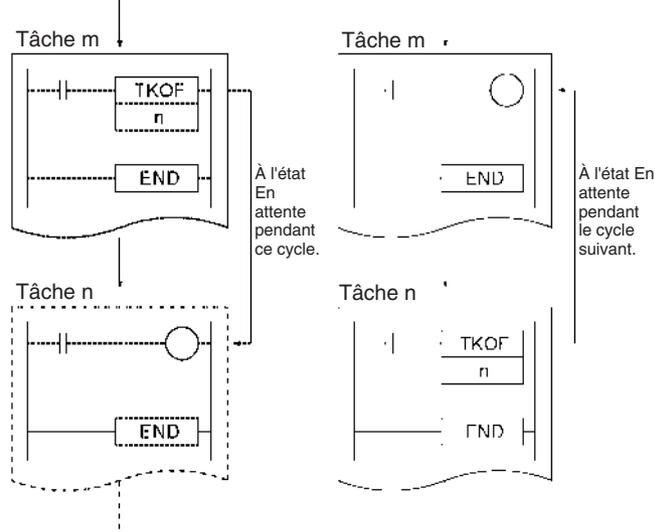
### 3-31 Instructions de traitement des chaînes de texte

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution					
<b>MOV STRING</b> MOV\$ @MOV\$ 664	<table border="1"> <tr><td>MOV\$(664)</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> </table> <p>S : 1er mot source D : 1er mot de destination</p>	MOV\$(664)	S	D	Transfère une chaîne de texte. 	Sortie Obligatoire		
MOV\$(664)								
S								
D								
<b>CONCATENATE STRING</b> +\$ @+\$ 656	<table border="1"> <tr><td>+\$(656)</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1 : Chaîne 1 S2 : Chaîne 2 D : Premier mot de destination</p>	+\$(656)	S1	S2	D	Lie une chaîne de texte à une autre chaîne de texte. 	Sortie Obligatoire	
+\$(656)								
S1								
S2								
D								
<b>GET STRING LEFT</b> LEFT\$ @LEFT\$ 652	<table border="1"> <tr><td>LEFT\$(652)</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1 : Premier mot de la chaîne S2 : Nombre de caractères D : Premier mot de destination</p>	LEFT\$(652)	S1	S2	D	Extrait un nombre de caractères désigné à partir de la gauche (début) d'une chaîne de texte. 	Sortie Obligatoire	
LEFT\$(652)								
S1								
S2								
D								
<b>GET STRING RIGHT</b> RGHT\$ @RGHT\$ 653	<table border="1"> <tr><td>RGHT\$(653)</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1 : Premier mot de la chaîne S2 : Nombre de caractères D : Premier mot de destination</p>	RGHT\$(653)	S1	S2	D	Lit un nombre de caractères désigné à partir de la droite (fin) d'une chaîne de texte. 	Sortie Obligatoire	
RGHT\$(653)								
S1								
S2								
D								
<b>GET STRING MIDDLE</b> MID\$ @MID\$ 654	<table border="1"> <tr><td>MID\$(654)</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>S3</td></tr> <tr><td>D</td></tr> </table> <p>S1 : Premier mot de la chaîne S2 : Nombre de caractères S3 : Position de départ D : Premier mot de destination</p>	MID\$(654)	S1	S2	S3	D	Lit un nombre de caractères désigné à partir d'une position quelconque au milieu d'une chaîne de texte. 	Sortie Obligatoire
MID\$(654)								
S1								
S2								
S3								
D								

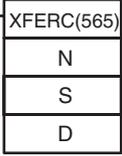
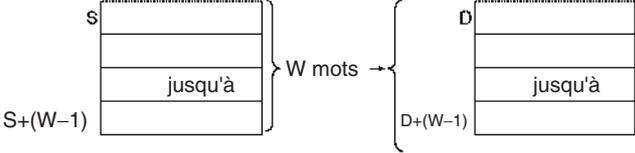
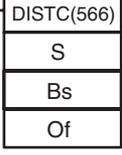
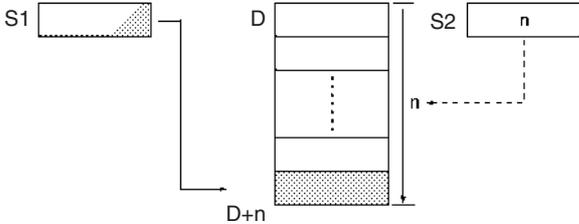
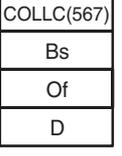
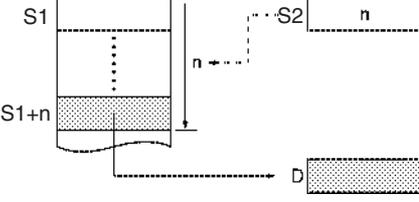
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution						
<b>FIND IN STRING</b> FIND @FIND\$ 660	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">FIND\$(660)</td></tr> <tr><td style="text-align: center;">S1</td></tr> <tr><td style="text-align: center;">S2</td></tr> <tr><td style="text-align: center;">D</td></tr> </table> <p>S1 : Premier mot de la chaîne source                      S2 : Premier mot de la chaîne de texte trouvée                      D : Premier mot de destination</p>	FIND\$(660)	S1	S2	D	Recherche une chaîne de texte désignée à l'intérieur d'une chaîne de texte. Donnée trouvée 	Sortie Obligatoire		
FIND\$(660)									
S1									
S2									
D									
<b>STRING LENGTH</b> LENS\$ @LENS\$ 650	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">LENS\$(650)</td></tr> <tr><td style="text-align: center;">S</td></tr> <tr><td style="text-align: center;">D</td></tr> </table> <p>S : Premier mot de la chaîne                      D : 1er mot de destination</p>	LENS\$(650)	S	D	Calcule la longueur d'une chaîne de texte. 	Sortie Obligatoire			
LENS\$(650)									
S									
D									
<b>REPLACE IN STRING</b> RPLC\$ @RPLC\$ 661	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">RPLC\$(661)</td></tr> <tr><td style="text-align: center;">S1</td></tr> <tr><td style="text-align: center;">S2</td></tr> <tr><td style="text-align: center;">S3</td></tr> <tr><td style="text-align: center;">S4</td></tr> <tr><td style="text-align: center;">D</td></tr> </table> <p>S1 : Premier mot de la chaîne                      S2 : Premier mot de la chaîne de remplacement                      S3 : Nombre de caractères                      S4 : Position de départ                      D : Premier mot de destination</p>	RPLC\$(661)	S1	S2	S3	S4	D	Remplace une chaîne de texte par une chaîne de texte désignée à partir d'une position désignée. 	Sortie Obligatoire
RPLC\$(661)									
S1									
S2									
S3									
S4									
D									
<b>DELETE STRING</b> DEL\$ @DEL\$ 658	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">DEL\$(658)</td></tr> <tr><td style="text-align: center;">S1</td></tr> <tr><td style="text-align: center;">S2</td></tr> <tr><td style="text-align: center;">S3</td></tr> <tr><td style="text-align: center;">D</td></tr> </table> <p>S1 : Premier mot de la chaîne                      S2 : Nombre de caractères                      S3 : Position de départ                      D : Premier mot de destination</p>	DEL\$(658)	S1	S2	S3	D	Supprime une chaîne de texte désignée au milieu d'une chaîne de texte. Nombre de caractères à supprimer (désigné par S2). 	Sortie Obligatoire	
DEL\$(658)									
S1									
S2									
S3									
D									

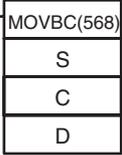
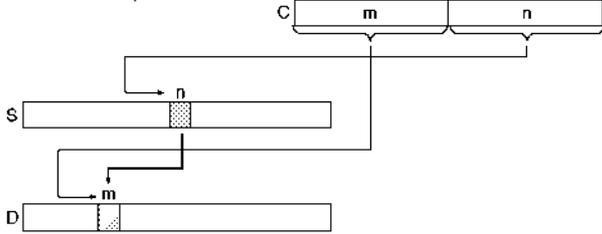
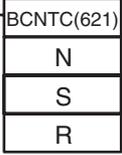
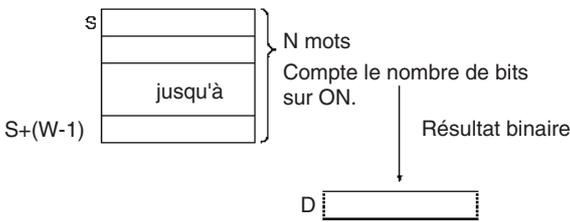
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution									
<b>EXCHANGE STRING</b>  XCHG\$ @XCHG\$ 665	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">XCHG\$(665)</td></tr> <tr><td style="text-align: center;">Ex1</td></tr> <tr><td style="text-align: center;">Ex2</td></tr> </table> <p>Ex1 : 1er mot échange 1 Ex2 : 1er mot échange 2</p>	XCHG\$(665)	Ex1	Ex2	Remplace une chaîne de texte désignée par une autre chaîne de texte désignée. 	Sortie Obligatoire						
XCHG\$(665)												
Ex1												
Ex2												
<b>CLEAR STRING</b>  CLR\$ @CLR\$ 666	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">CLR\$(666)</td></tr> <tr><td style="text-align: center;">S</td></tr> </table> <p>S : Premier mot de la chaîne</p>	CLR\$(666)	S	Remet à zéro une chaîne de texte entière avec NUL (00 hex). 	Sortie Obligatoire							
CLR\$(666)												
S												
<b>INSERT INTO STRING</b>  INS\$ @INS\$ 657	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">INS\$(657)</td></tr> <tr><td style="text-align: center;">S1</td></tr> <tr><td style="text-align: center;">S2</td></tr> <tr><td style="text-align: center;">S3</td></tr> <tr><td style="text-align: center;">D</td></tr> </table> <p>S1 : Premier mot de la chaîne d'origine S2 : Premier mot de la chaîne insérée S3 : Position de départ D : Premier mot de destination</p>	INS\$(657)	S1	S2	S3	D	Supprime une chaîne de texte désignée au milieu d'une chaîne de texte. 	Sortie Obligatoire				
INS\$(657)												
S1												
S2												
S3												
D												
<b>Comparaison de chaînes</b>  LD, AND, OR + =\$, <>\$, <\$, <=\$, >\$, >=\$ 670 (=\$) 671 (<>\$) 672 (<\$) 673 (<=\$) 674 (>\$) 675 (>=\$)	<p><b>LD</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">Symbole</td></tr> <tr><td style="text-align: center;">S1</td></tr> <tr><td style="text-align: center;">S2</td></tr> </table> <p><b>AND</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">Symbole</td></tr> <tr><td style="text-align: center;">S1</td></tr> <tr><td style="text-align: center;">S2</td></tr> </table> <p><b>OR</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center;">Symbole</td></tr> <tr><td style="text-align: center;">S1</td></tr> <tr><td style="text-align: center;">S2</td></tr> </table> <p>S1 : Chaîne 1 S2 : Chaîne 2</p>	Symbole	S1	S2	Symbole	S1	S2	Symbole	S1	S2	Les instructions de comparaison de chaînes (=\$, <>\$, <\$, <=\$, >\$, >=\$) comparent deux chaînes de texte à partir du début, en se basant sur la valeur des codes ASCII. Si le résultat de la comparaison est vrai, une condition d'exécution ON est créée pour un LOAD, un AND ou un OR.	LD : non obligatoire AND, OR : obligatoire
Symbole												
S1												
S2												
Symbole												
S1												
S2												
Symbole												
S1												
S2												

### 3-32 Instructions de contrôle des tâches

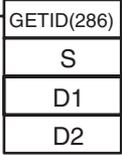
Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<p><b>TASK ON</b> TKON @TKON 820</p>	<div style="border: 1px solid black; padding: 5px; width: fit-content;">             TKON(820)  <hr/>             N         </div> <p>N : Numéro de tâche</p>	<p>Rend la tâche spécifiée exécutable.</p> <p>Le numéro de la tâche spécifiée est supérieur au numéro de la tâche locale (<math>m &lt; n</math>). Le numéro de la tâche spécifiée est inférieur au numéro de la tâche locale (<math>m &gt; n</math>).</p> 	<p>Sortie Obligatoire</p>
<p><b>TASK OFF</b> TKOF @TKOF 821</p>	<div style="border: 1px solid black; padding: 5px; width: fit-content;">             TKOF(821)  <hr/>             N         </div> <p>N : Numéro de tâche</p>	<p>Met la tâche spécifiée à l'état En attente.</p> <p>Le numéro de la tâche spécifiée est supérieur au numéro de la tâche locale (<math>m &lt; n</math>). Le numéro de la tâche spécifiée est inférieur au numéro de la tâche locale (<math>m &gt; n</math>).</p> 	<p>Sortie Obligatoire</p>

### 3-33 Instructions de conversion des modèles (UC ver. 3.0 ou supérieure uniquement)

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>BLOCK TRANSFER</b> XFERC @XFERC 565	 <p>N : Nombre de mots                      S : 1er mot source                      D : 1er mot de destination</p>	<p>Transfère le nombre de mots consécutifs spécifié.</p> 	Sortie Obligatoire
<b>SINGLE WORD DISTRIBUTE</b> DISTC @DISTC 566	 <p>S : Mot source                      Bs : Adresse destination de base                      Of : Décalage</p>	<p>Transfère le mot source vers un mot de destination calculé en ajoutant une valeur de décalage à l'adresse de base.</p>  <p>Peut également écrire sur une pile (opération pousser sur pile).</p>	Sortie Obligatoire
<b>DATA COLLECT</b> COLLC @COLLC 567	 <p>Bs : Adresse de base source                      Of : Décalage (BCD)                      D : Mot de destination</p>	<p>Transfère le mot source (calculé en ajoutant une valeur de décalage à l'adresse de base) vers le mot de destination.</p>  <p>Peut également lire les données d'une pile (opération lire sur pile).</p>	Sortie Obligatoire

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>MOVE BIT</b> MOVBC @MOVBC 568	 <p>S : Mot ou donnée source                      C : Mot de contrôle (BCD)                      D : Mot de destination</p>	Transfère le bit spécifié. 	Sortie Obligatoire
<b>BIT COUNTER</b> BCNTC @BCNTC 621	 <p>N : Nombre de mots (BCD)                      S : 1er mot source                      R : Mot de résultat</p>	Compte le nombre total de bits sur ON dans le(s) mot(s) spécifié(s). 	Sortie Obligatoire

### 3-34 Instructions spéciales des blocs de fonction

Instruction Mnémonique Code	Symbole/ Opérande	Fonction	Emplacement Condition d'exécution
<b>GET VARIABLE ID</b> GETID @GETID 286	 <p>S : variable ou adresse                      D1 : code de l'identifiant                      D2 : Mot de destination</p>	Sort le code et l'adresse mot du type de variable de commande FINS (zone données) pour la variable ou l'adresse spécifiée. Cette instruction est généralement utilisée pour obtenir l'adresse affectée à une variable dans un bloc de fonction.	Sortie Obligatoire



# CHAPITRE 4

## Tâches

Cette section décrit le fonctionnement des tâches.

4-1	Caractéristiques des tâches . . . . .	158
4-1-1	Présentation. . . . .	158
4-1-2	Tâches et programmes . . . . .	159
4-1-3	Fonctionnement de base de l'UC . . . . .	160
4-1-4	Types de tâches. . . . .	162
4-1-5	Conditions et paramètres d'exécution des tâches . . . . .	164
4-1-6	Etat des tâches cycliques . . . . .	165
4-1-7	Transitions entre les états . . . . .	166
4-2	Utilisation de tâches . . . . .	167
4-2-1	TASK ON et TASK OFF . . . . .	167
4-2-2	Limites des instructions relatives aux tâches . . . . .	170
4-2-3	Drapeaux relatifs aux tâches. . . . .	171
4-2-4	Conception de tâches . . . . .	175
4-2-5	Sous-programmes globaux . . . . .	176
4-3	Tâche d'interruption . . . . .	177
4-3-1	Types de tâches d'interruption . . . . .	177
4-3-2	Priorité des tâches d'interruption . . . . .	184
4-3-3	Drapeaux de tâche d'interruption et mots . . . . .	185
4-3-4	Précautions en matière d'application . . . . .	186
4-4	Fonctionnement du périphérique de programmation pour les tâches . . . . .	189
4-4-1	Utilisation de plusieurs tâches cycliques . . . . .	189
4-4-2	Fonctionnement du périphérique de programmation . . . . .	189

## 4-1 Caractéristiques des tâches

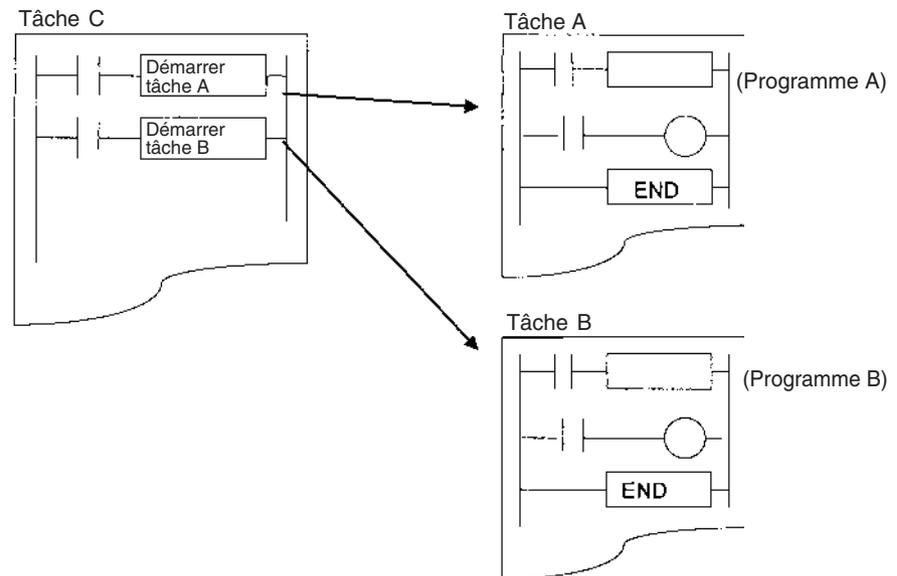
### 4-1-1 Présentation

Les opérations de contrôle des API série CS/CJ peuvent être réparties par fonctions, périphériques contrôlés, processus, développeurs ou selon n'importe quel autre critère, et chaque opération peut être programmée dans une unité distincte appelée « tâche ». L'utilisation des tâches présente les avantages suivants :

- 1,2,3...**
1. Des programmes peuvent être développés simultanément par plusieurs personnes.  
Les parties des programmes conçues séparément peuvent être assemblées très facilement pour former un programme utilisateur unique.
  2. Les programmes peuvent être standardisés dans des modules.  
Plus particulièrement, les fonctions de périphérique de programmation suivantes sont combinées afin de développer des programmes correspondant à des modules standard autonomes plutôt que des programmes conçus pour des systèmes spécifiques (machines, périphériques). Autrement dit, les programmes développés séparément par plusieurs personnes sont prêts à être combinés.
    - Programmation avec des symboles
    - Désignation globale et locale des symboles
    - Affectation automatique de symboles locaux à des adresses
  3. Meilleure réponse générale.  
La réponse générale est améliorée car le système est subdivisé en un programme de contrôle général et en plusieurs programmes de contrôle individuels, ce qui permet d'exécuter uniquement des programmes spécifiques en fonction des besoins.
  4. Révision et débogage simplifiés.
    - Le débogage est nettement plus efficace car les tâches peuvent être développées séparément par plusieurs personnes, puis révisées et mises au point par une tâche individuelle.
    - La maintenance est simple car seule la tâche qui doit être révisée sera remaniée en fonction des changements de spécification ou autres modifications.
    - Le débogage est plus efficace : en effet, il est possible de déterminer facilement si une adresse est spécifique ou globale et il suffit de contrôler une seule fois les adresses entre les programmes pendant le débogage car les symboles sont désignés globalement ou localement et les symboles locaux sont alloués automatiquement à des adresses par l'intermédiaire de périphériques de programmation.
  5. Basculement aisé entre les programmes.  
Une instruction de contrôle des tâches dans le programme peut être utilisée pour exécuter des tâches (programmes) spécifiques à un produit lorsqu'il faut changer d'opération.

## 6. Programmes utilisateur facilement compréhensibles.

Les programmes sont structurés en blocs, ce qui simplifie considérablement la compréhension du programme pour les sections qui seraient normalement traitées avec des instructions telles que JUMP.



## 4-1-2 Tâches et programmes

- Vous pouvez contrôler jusqu'à 288 programmes (tâches). Les différents programmes sont affectés à des tâches selon une relation 1:1. Les tâches sont regroupées par types, de la manière suivante :
- Tâches cycliques
- Tâches d'interruption

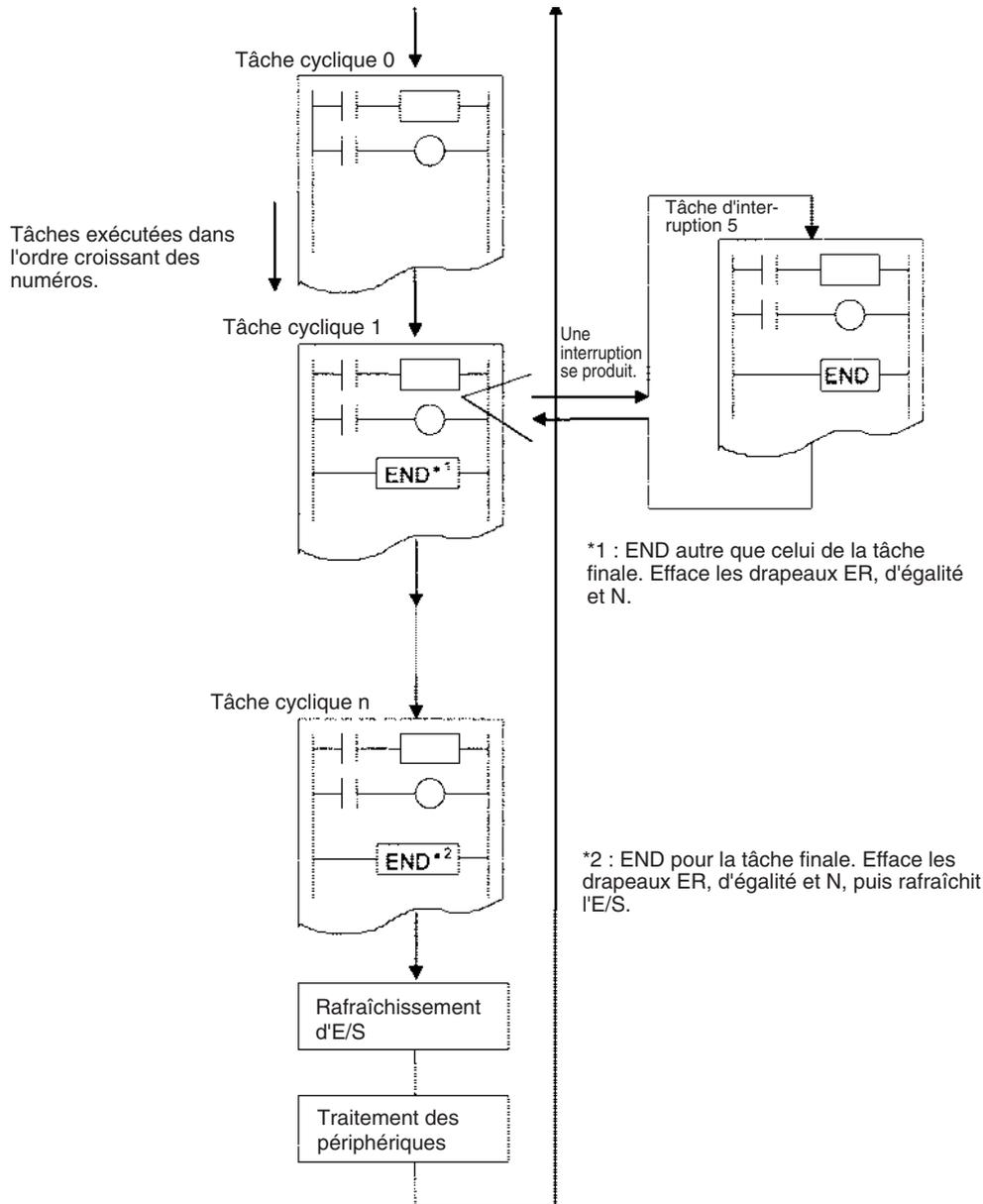
**Remarque**

1. Vous pouvez créer jusqu'à 32 tâches cycliques et 256 tâches d'interruption, pour un total de 288 tâches maximum. Chaque tâche possède son propre numéro unique, compris entre 0 et 31 pour les tâches cycliques et entre 0 et 255 pour les tâches d'interruption.
2. Avec les UC CS1-H, CJ1-H, CJ1M ou CS1D, vous pouvez exécuter les tâches d'interruption (numéros de tâches d'interruption 0 à 255) en tant que tâches cycliques en les démarrant avec TKON. Elles sont alors appelées « tâches cycliques extra ». Si vous employez des tâches cycliques extra, le nombre total de tâches cycliques pouvant être utilisées est de 288.
3. Les UC CJ1 ne prennent pas actuellement en charge les tâches d'interruption d'E/S et externes. Le nombre maximal de tâches pour une UC CJ1 est donc de 35 (32 tâches cycliques et 3 tâches d'interruption). Le nombre total de programmes pouvant être créés et gérés est également de 35.

Chaque programme affecté à une tâche doit se terminer par une instruction END(001). La mise à jour E/S ne sera effectué qu'après l'exécution de toutes les tâches programmées dans un cycle.

### 4-1-3 Fonctionnement de base de l'UC

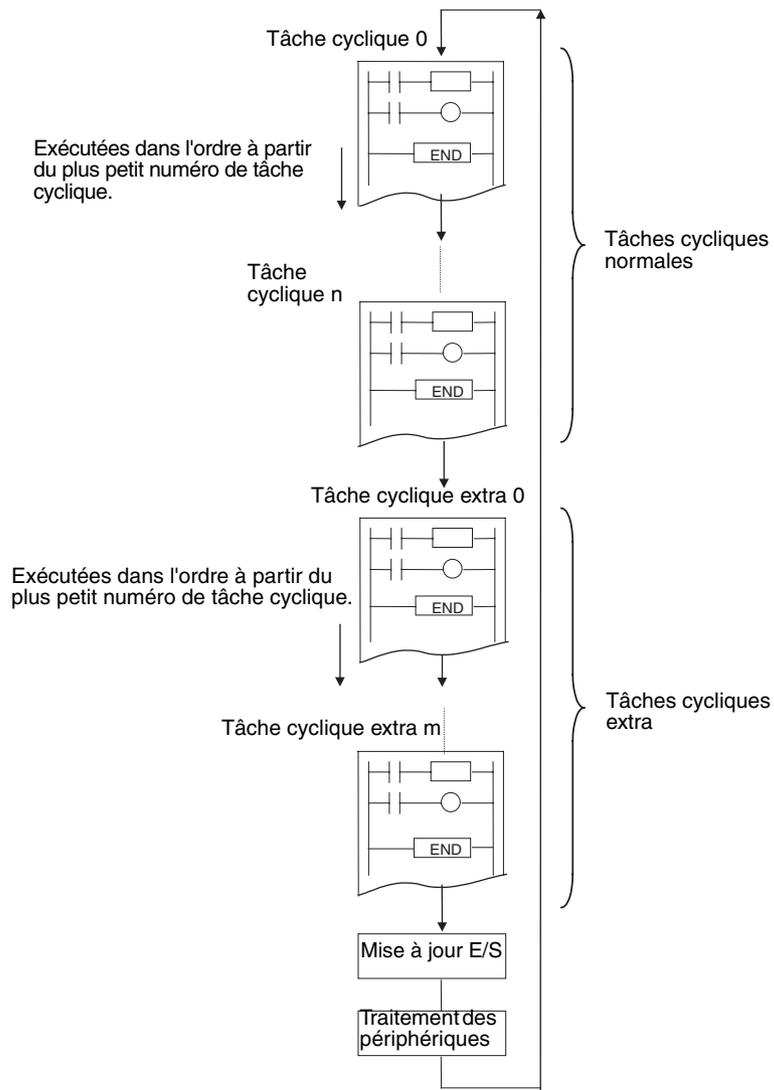
L'UC exécutera les tâches cycliques (y compris les tâches cycliques extra, UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement) en partant du plus petit numéro. Elle interrompra également l'exécution des tâches cycliques pour exécuter une tâche d'interruption, si une interruption a lieu.



**Remarque** Tous les drapeaux de condition (ER, CY, Equals, AER, etc.) et les conditions d'instruction (verrouillage ON, etc.) sont remis à zéro au début d'une tâche. Par conséquent, les drapeaux de condition ne peuvent pas être lus et les instructions INTERLOCK/INTERLOCK CLEAR (IL/ILC), JUMP/JUMP END (JMP/JME) ou SUBROUTINE CALL/SUBROUTINE ENTRY (SBS/SBN) ne peuvent pas être divisées entre les deux tâches.

Avec les UC CS1-H, CJ1-H, CJ1M ou CS1D, vous pouvez exécuter les tâches d'interruption en tant que tâches cycliques en les démarrant avec TKON. Elles sont alors appelées « tâches cycliques extra ». Les tâches cycliques extra (numéros de tâches d'interruption 0 à 255) sont exécutées à partir du plus

petit numéro de tâche, une fois que l'exécution de la tâche cyclique normale (numéros de tâches celiac 0 à 31) est terminée.



## 4-1-4 Types de tâches

Les tâches sont globalement classées en tant que tâches cycliques ou tâches d'interruption. Les tâches d'interruption sont ensuite réparties entre les tâches d'interruption de mise hors tension, les tâches d'interruption programmées, les tâches d'interruption d'E/S (série CS uniquement) et les tâches d'interruption externes (série CS uniquement). Les tâches d'interruption peuvent également être exécutées en tant que tâches cycliques extra.

**Remarque** Avec les UC CS1-H, CJ1-H, CJ1M ou CS1D, vous pouvez exécuter les tâches d'interruption en tant que tâches cycliques en les démarrant avec TKON. Elles sont alors appelées « tâches cycliques extra ».

### Tâches cycliques

Une tâche cyclique à l'état READY sera exécutée une fois par cycle (du début du programme jusqu'à l'instruction END(001)) en fonction de l'ordre numérique (en partant de la tâche correspondant au plus petit numéro). Le nombre maximum de tâches cycliques est de 32 (numéros de tâches cycliques : 00 à 31).

**Remarque** Avec les UC CS1-H, CJ1-H, CJ1M ou CS1D, vous pouvez exécuter les tâches d'interruption (numéros de tâches d'interruption 0 à 255) en tant que tâches cycliques en procédant comme pour les tâches cycliques normales (numéros de tâches 0 à 31). Si vous utilisez des tâches cycliques extra, le nombre total de tâches cycliques pouvant être utilisées est de 288.

### Tâches d'interruption

Une tâche d'interruption sera exécutée si une interruption a lieu, même si une tâche cyclique (y compris des tâches cycliques extra) est en cours d'exécution. La tâche d'interruption sera exécutée à n'importe quel moment du cycle, y compris pendant l'exécution du programme utilisateur, la mise à jour d'E/S ou le traitement des périphériques, lorsque la condition d'exécution pour l'interruption sera remplie.

Dans le cas des UC CS1-H, CJ1-H, CJ1M ou CS1D, les tâches d'interruption peuvent être exécutées en tant que tâches cycliques (les UC CS1D pour les systèmes à UC en duplex ne prennent pas en charge les interruptions. Avec ces dernières, les tâches d'interruption peuvent être utilisées uniquement en tant que tâches cycliques extra).

Les entrées d'interruption intégrées et les entrées des compteurs à grande vitesse sur une UC CJ1M peuvent être utilisées pour activer des tâches d'interruption. Pour des informations détaillées, reportez-vous au *Manuel d'utilisation des E/S intégrées, série CJ*.

#### **Tâche d'interruption de mise hors tension**

La tâche d'interruption de mise hors sera exécutée si l'alimentation de l'UC est coupée. Une seule tâche d'interruption de mise hors tension peut être programmée (numéro de tâche d'interruption : 1).

**Remarque** La tâche d'interruption de mise hors tension doit s'exécuter avant l'expiration du délai suivant, sinon la tâche est automatiquement fermée.

10 ms – (temps de retard de détection de mise hors tension)

Le temps de retard de détection de mise hors tension est défini dans la configuration de l'API.

#### **Tâches d'interruption programmées**

Une tâche d'interruption programmée sera exécutée à intervalle en fonction de la temporisation interne de l'UC. Le nombre maximum de tâches d'interruption programmées est de 2 (numéros de tâches d'interruption : 2 et 3).

**Remarque** L'instruction SET INTERRUPT MASK (MSKS(690)) permet de définir l'interruption pour une tâche d'interruption programmée. Les délais d'interruption peuvent être définis par incréments de 10 ms ou de 1,0 ms dans la configuration de l'API.

#### **Tâches d'interruptions d'E/S**

Une tâche d'interruption d'E/S sera exécutée si une entrée de carte d'entrée d'interruption passe à ON. Le nombre maximum de tâches d'interruption d'E/S est de 32 (numéros de tâches d'interruption : 100 à 131). La carte d'entrée d'interruption doit être montée sur le rack UC. Pour les UC CJ1-H, la carte doit être connectée comme étant l'une des cinq unités (emplacements 0 à 4) présentes à côté de l'UC. Pour les UC CJ1M, la carte doit être connectée comme étant l'une des trois unités (emplacements 0 à 2) présentes à côté de l'UC. Les

cartes d'interruption d'E/S installées à un autre emplacement ne peuvent pas être utilisées pour demander l'exécution de tâches d'interruption d'E/S.

Les interruptions d'E/S ne sont pas prises en charge par les UC CJ1.

**Tâches d'interruption externes**

Une tâche d'interruption externe sera exécutée à la demande d'un programme utilisateur d'une carte d'E/S spéciales, d'une carte réseau ou d'une carte interne (série CS uniquement). Les cartes d'E/S spéciales et les cartes réseau, cependant, doivent être montées sur le rack UC. La carte d'E/S spéciales ou la carte réseau doit être montée sur le rack UC. Pour les UC CJ1-H, la carte doit être connectée comme étant l'une des cinq unités (emplacements 0 à 4) présentes à côté de l'UC. Pour les UC CJ1M, la carte doit être connectée comme étant l'une des trois unités (emplacements 0 à 2) présentes à côté de l'UC. Les cartes installées à un autre emplacement ne peuvent pas être utilisées pour générer des interruptions externes.

Le nombre maximum de tâches d'interruption externes est de 256 (numéros de tâches d'interruption : 0 à 255). Si une tâche d'interruption externe a le même numéro qu'une tâche d'interruption de mise hors tension, programmée ou E/S, la tâche sera exécutée pour chaque condition (les deux conditions fonctionneront avec la logique OR), mais fondamentalement, les numéros de tâches ne doivent pas être dupliqués.

Les interruptions d'E/S ne sont pas prises en charge par les UC CJ1.

**Tâches cycliques extra (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement)**

Une tâche d'interruption peut être exécutée à chaque cycle, exactement comme une tâche cyclique normale. Les tâches cycliques extra (numéros de tâches d'interruption 0 à 255) sont exécutées à partir du plus petit numéro de tâche une fois que l'exécution de la tâche cyclique normale (numéros de tâches cycliques 0 à 31) est terminée. Le nombre maximum de tâches cycliques extra est de 256 (numéros de tâches d'interruption : 0 à 255). Les tâches d'interruption cycliques diffèrent cependant des tâches cycliques normales dans la mesure où elles sont déclenchées par l'instruction TKON(820). De plus, les instructions TKON(820) et TKOF ne peuvent pas être utilisées dans les tâches cycliques extra ; autrement dit, les tâches cycliques normales et les autres tâches cycliques extra ne peuvent pas être contrôlées à partir d'une tâche cyclique extra.

Si une tâche cyclique extra porte le même numéro qu'une tâche d'interruption de mise hors tension, programmée ou E/S, la tâche d'interruption sera exécutée pour l'une ou l'autre condition (les deux conditions fonctionneront avec la logique OR). N'utilisez pas de tâches d'interruption à la fois en tant que tâches d'interruption normales et tâches cycliques extra.

**Remarque**

1. La tâche d'interruption de mise hors tension du paragraphe 1) ci-dessus est prioritaire et sera exécutée lors de la mise hors tension, même si une autre tâche d'interruption est en cours d'exécution.
2. Si une autre tâche d'interruption est en cours d'exécution lorsqu'une interruption programmée, E/S ou externe survient, ces tâches d'interruption ne seront pas exécutées avant la fin de la tâche d'interruption en cours. Si plusieurs interruptions ont lieu simultanément, les tâches d'interruption seront exécutées l'une après l'autre en commençant au numéro de tâche d'interruption le plus petit.
3. Le tableau suivant répertorie les différences entre les tâches cycliques normales et les tâches cycliques extra.

Élément	Tâches cycliques extra	Tâches cycliques normales
Activation au démarrage	Configuration impossible.	Définies à partir du CX-Programmer
Utilisation d'instructions TKON/TKOF	Impossible.	Possible.
Drapeaux de tâches	Non pris en charge.	Pris en charge (les numéros de tâches cycliques 00 à 31 correspondent aux drapeaux de tâche TK00 à TK31).

Élément	Tâches cycliques extra	Tâches cycliques normales
Drapeau d'exécution de la tâche initiale (A20015) et drapeau de démarrage de tâche (A20014)	Non pris en charge.	Pris en charge.
Valeurs du registre d'index (IR) et du registre de données (DR)	Non définies au démarrage de la tâche (comme les tâches d'interruption normales). Les valeurs au début de chaque cycle sont indéfinies. Définissez toujours les valeurs avant de les utiliser. Les valeurs définies dans le cycle précédent ne peuvent pas être lues.	Non définies au début du fonctionnement. Les valeurs définies dans le cycle précédent peuvent être lues.

- Les UC CJ1 ne prennent pas en charge les tâches d'interruption d'E/S ni les tâches d'interruption externes.

### 4-1-5 Conditions et paramètres d'exécution des tâches

Le tableau suivant décrit les conditions d'exécution des tâches, les paramètres associés et les états.

Tâche		N°	Condition d'exécution	Paramètre associé
Tâches cycliques		0 à 31	Exécutée une fois par cycle si l'état est READY (configuré pour démarrer initialement ou démarré avec l'instruction TKON(820)) lorsque le droit d'exécution est obtenu.	Aucun
Tâches d'interruption	Tâche d'interruption de mise hors tension	Tâche d'interruption 1	S'exécute lorsque l'UC est mise hors tension.	<ul style="list-style-type: none"> <li>Interruption de mise hors tension activée dans la configuration de l'API.</li> </ul>
	Tâches d'interruption programmées 0 et 1	Tâches d'interruption 2 et 3	S'exécute chaque fois que la période prédéfinie s'est écoulée en fonction de la temporisation interne de l'UC.	<ul style="list-style-type: none"> <li>Le délai d'interruption programmé est défini (0 à 9999) par l'intermédiaire de l'instruction SET INTERRUPT MASK (MSKS).</li> <li>L'unité d'interruption programmée (10 ms ou 1,0 ms) est définie dans la configuration de l'API.</li> </ul>
	Tâches d'interruption d'E/S 00 à 31	Tâches d'interruption 100 à 131	S'exécute lorsqu'une entrée de carte d'entrée d'interruption sur le rack UC passe à ON.	<ul style="list-style-type: none"> <li>Les masques des entrées désignées sont annulés par l'intermédiaire de l'instruction SET INTERRUPT MASK (MSKS).</li> </ul>
	Tâches d'interruption externes 0 à 255	Tâches d'interruption 0 à 255	S'exécute à la demande d'un programme utilisateur dans une carte d'E/S spéciales ou une carte réseau sur le rack UC ou à la demande d'un programme utilisateur dans une carte interne (série CS uniquement).	Aucun (toujours activé)
Tâches cycliques extra (CS1-H, CJ1-H, CJ1M ou CS1D uniquement)		Tâches d'interruption 0 à 255	Exécutées une fois par cycle si l'état est READY (démarré avec l'instruction TKON(820)) lorsque le droit d'exécution est obtenu.	Aucun (toujours activé)

- Remarque**
- La carte d'entrée d'interruption doit être montée sur le rack UC. Pour les UC CJ1-H, la carte doit être connectée comme étant l'une des cinq unités (emplacements 0 à 4) présentes à côté de l'UC. Pour les UC CJ1M, la carte doit être connectée comme étant l'une des trois unités (emplacements 0 à 2) présentes à côté de l'UC. Les cartes d'interruption d'E/S installées à un autre emplacement ne peuvent pas être utilisées pour demander l'exécution de tâches d'interruption d'E/S.

2. La carte d'E/S spéciales ou la carte réseau doit être montée sur le rack UC. Pour les UC CJ1-H, la carte doit être connectée comme étant l'une des cinq unités (emplacements 0 à 4) présentes à côté de l'UC. Pour les UC CJ1M, la carte doit être connectée comme étant l'une des trois unités (emplacements 0 à 2) présentes à côté de l'UC. Les cartes installées à un autre emplacement ne peuvent pas être utilisées pour générer des interruptions externes.
3. Le nombre de tâches cycliques et de tâches d'interruption est limité lorsque l'opération de remise à zéro de la mémoire est exécutée avec une console de programmation.
  - Seule la tâche cyclique 0 peut être créée.  
Il n'est pas possible de créer les tâches cycliques 1 à 31 avec une console de programmation, mais ces tâches peuvent être modifiées si elles ont déjà été créées avec CX-Programmer.
  - Seules les tâches d'interruption 1, 2, 3 et 100 à 131 (série CS uniquement) peuvent être créées.  
Les tâches d'interruption 0, 4 à 99 et 132 à 255 ne peuvent pas être créées avec une console de programmation (à l'exception des tâches 140 à 143 qui peuvent être créées pour les UC CJ1M), mais ces tâches peuvent être modifiées si elles ont déjà été créées avec CX-Programmer.

## 4-1-6 Etat des tâches cycliques

Cette section décrit les états des tâches cycliques, y compris des tâches cycliques extra (prises en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement).

Les tâches cycliques peuvent avoir quatre états : Désactivé, READY (prêt), RUN (exécutable) et En attente (WAIT).

### Etat Désactivé (INI)

Une tâche avec un état Désactivé n'est pas exécutée. Toutes les tâches cycliques ont un état Désactivé en mode PROGRAM. Une tâche cyclique qui est passée de cet état à un autre état ne peut pas revenir à cet état sans repasser en mode PROGRAM.

### Etat READY

Il est possible de définir un attribut de tâche pour contrôler à quel moment la tâche passera à l'état READY. L'attribut peut être défini pour activer la tâche avec l'instruction TASK ON ou au démarrage de l'opération RUN.

#### **Tâches activées par instruction**

Une instruction TASK ON (TKON(820)) est utilisée pour faire passer une tâche cyclique activée par instruction de l'état Désactivé ou En attente à l'état READY.

#### **Tâches activées par opération**

Une tâche cyclique activée par opération passe de l'état Désactivé à l'état READY lorsque le mode de fonctionnement passe du mode PROGRAM au mode RUN ou MONITOR. Cela s'applique uniquement aux tâches cycliques normales.

#### **Remarque**

Vous pouvez, à l'aide d'un périphérique de programmation, configurer une ou plusieurs tâches de telle sorte qu'elles passent à l'état READY lorsque l'opération est démarrée pour les numéros de tâches 0 à 31. Ce paramétrage n'est pas possible, cependant, avec les tâches cycliques extra.

### Etat RUN

Une tâche cyclique qui est READY passe à l'état RUN et elle est exécutée lorsqu'elle obtient le droit d'exécution.

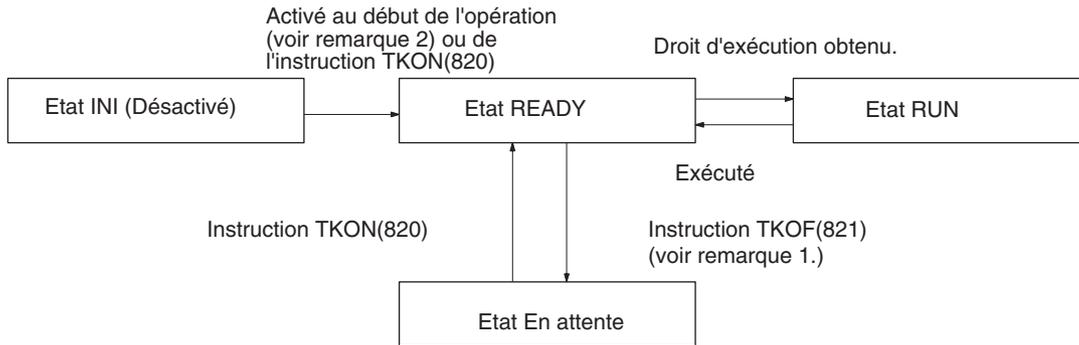
### Etat En attente

Vous pouvez utiliser une instruction TASK OFF (TKOF(821)) pour faire passer une tâche cyclique de l'état Désactivé à l'état En attente.

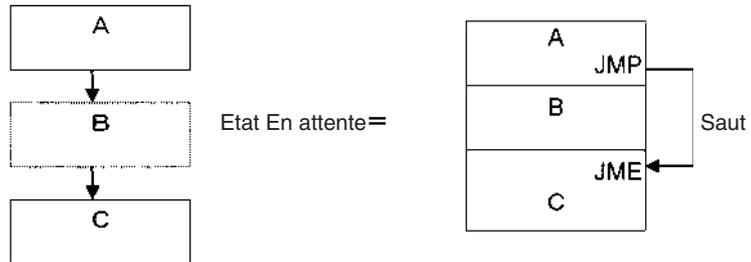
**Remarque** CX-Programmer version 4.0 (ou une version supérieure) permet d'assurer une surveillance en ligne des programmes de tâches pour les API série CS/CJ afin de déterminer s'ils sont en cours d'exécution ou arrêtés. Les indications d'état sur CX-Programmer sont les suivantes :

- Running : la tâche est à l'état READY ou RUN (rien ne permet de faire la distinction entre ces deux états).
- Stopped : la tâche est à l'état INI ou WAIT (rien ne permet de faire la distinction entre ces deux états).

### 4-1-7 Transitions entre les états

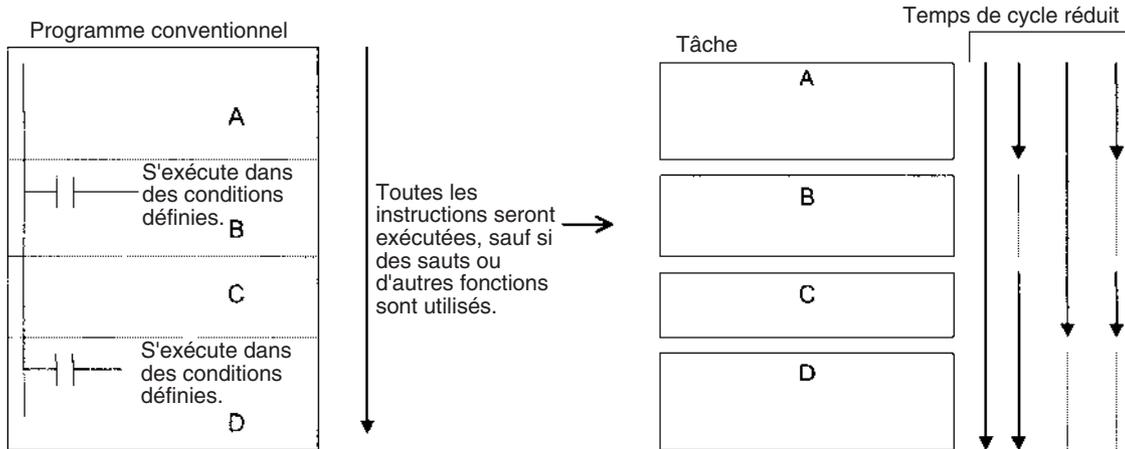


- Remarque**
1. Une tâche à l'état RUN sera mise à l'état En attente par l'instruction TKOF(821), même si l'instruction TKOF(821) est exécutée à l'intérieur de cette tâche.
  2. L'activation au début de l'opération n'est possible que pour les tâches cycliques normales. Elle n'est pas possible pour les tâches cycliques extra.
- L'état En attente fonctionne exactement de la même façon qu'un saut (JMP-JME). L'état de sortie associé à la tâche En attente sera conservé.



A l'état En attente, les instructions ne seront pas exécutées ; par conséquent, le temps d'exécution des instructions n'augmentera pas. Vous pouvez créer la

programmation qui ne doit pas être exécutée en permanence dans des tâches et lui affecter l'état En attente pour réduire le temps de cycle.



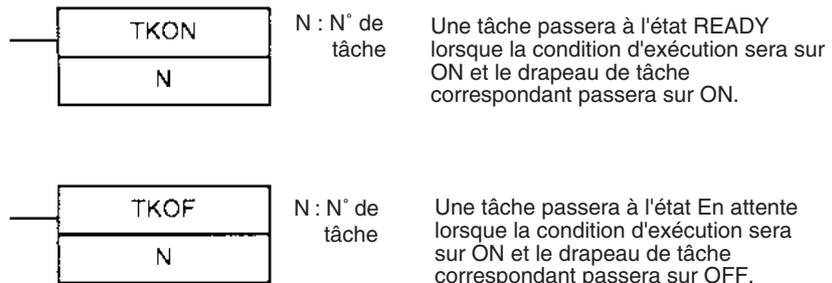
**Remarque** L'état En attente signifie simplement qu'une tâche sera ignorée pendant l'exécution des tâches. Le fait de passer à l'état En attente n'arrêtera pas le programme.

## 4-2 Utilisation de tâches

### 4-2-1 TASK ON et TASK OFF

Les instructions TASK ON (TKON(820)) et TASK OFF (TKOF(821)) permettent de faire passer une tâche cyclique (y compris une tâche cyclique extra) de l'état READY à l'état En attente et inversement à partir d'un programme.

**Remarque** Les tâches cycliques extra sont prises en charge uniquement par les UC CS1-H, CJ1-H, CJ1M et CS1D.

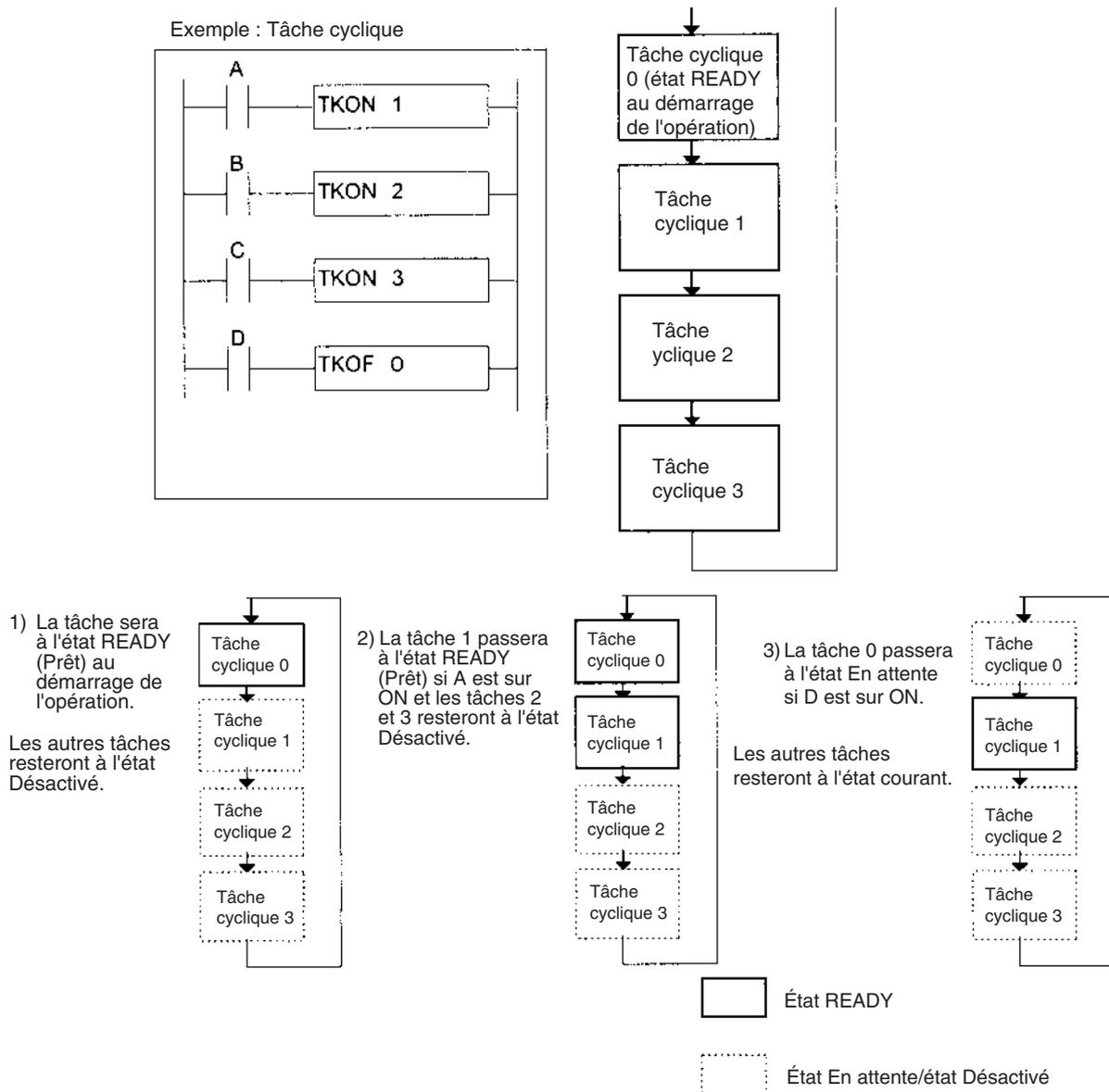


**Remarque :** les drapeaux de tâche ne sont pas opérationnels pour les tâches cycliques extra.

Les instructions TASK ON et TASK OFF permettent de faire passer à tout moment une tâche cyclique de l'état READY à l'état En attente et inversement. Une tâche cyclique qui est à l'état READY ou En attente conservera cet état dans les cycles suivants.

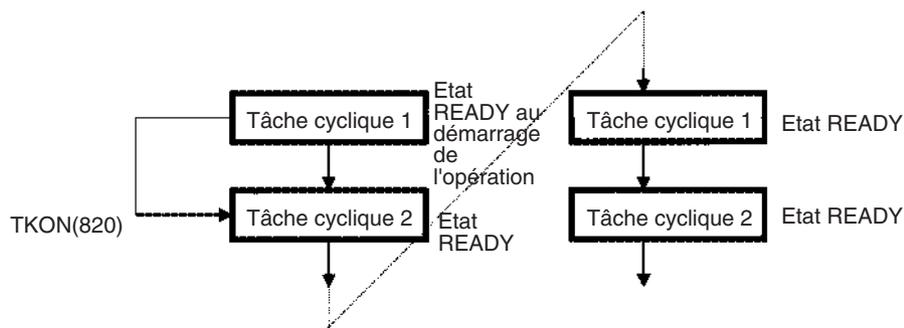
Les instructions TASK ON et TASK OFF peuvent être utilisées uniquement avec des tâches cycliques, et non avec des tâches d'interruption.

**Remarque** Une tâche cyclique au moins doit être à l'état READY dans chaque cycle. Si aucune tâche cyclique n'est à l'état READY, le drapeau d'erreur de tâche (A29512) passera à ON et l'UC cessera de fonctionner.

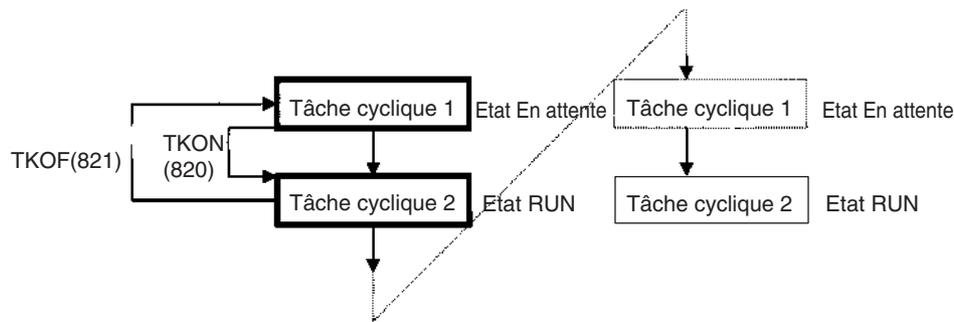


### Tâches et cycle d'exécution

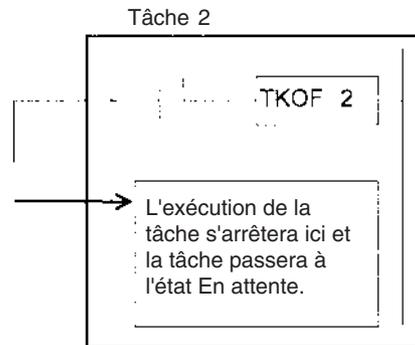
Une tâche cyclique (y compris une tâche cyclique extra) qui est à l'état READY conservera cet état dans les cycles suivants.



Une tâche cyclique qui est à l'état En attente conservera cet état dans les cycles suivants. La tâche devra être activée avec l'instruction TKON(820) pour passer de l'état En attente à l'état READY.



Si une instruction TKOF(821) est exécutée pour la tâche dans laquelle elle se trouve, l'exécution de la tâche s'interrompra à l'endroit où l'instruction est exécutée et la tâche passera à l'état En attente.



### **Numéros de tâches cycliques et cycle d'exécution (y compris les tâches cycliques extra)**

Si la tâche  $m$  fait passer la tâche  $n$  à ON et  $m > n$ , la tâche  $n$  passera à l'état READY dans le cycle suivant.

**Exemple :** si la tâche 5 fait passer la tâche 2 à ON, la tâche 2 passera à l'état READY dans le cycle suivant.

Si la tâche  $m$  fait passer la tâche  $n$  à ON et  $m > n$ , la tâche  $n$  passera à l'état READY au cours du même cycle.

**Exemple :** si la tâche 2 fait passer la tâche 5 à ON, la tâche 5 passera à l'état READY au cours du même cycle.

Si la tâche  $m$  met la tâche  $n$  à l'état En attente  $m > n$ , la tâche  $n$  passera à l'état En attente dans le cycle suivant.

**Exemple :** si la tâche 5 met la tâche 2 à l'état En attente, la tâche 2 passera à l'état En attente dans le cycle suivant.

Si la tâche  $m$  met la tâche  $n$  à l'état En attente et  $m < n$ , la tâche  $n$  passera à l'état En attente au cours du même cycle.

**Exemple :** si la tâche 2 met la tâche 5 à l'état En attente, la tâche 5 passera à l'état En attente au cours du même cycle.

### **Rapport entre les tâches et la mémoire E/S**

Il existe deux façons d'utiliser les registres d'index (IR) et les registres de données (DR) : 1) indépendamment par tâche ou 2) Partagés par toutes les tâches (mode pris en charge par les UC CS1-H, CJ1-H, CJ1M et CS1D uniquement).

Avec des registres indépendants, l'IRO utilisé par la tâche cyclique 1, par exemple, diffère de l'IRO utilisé par la tâche cyclique 2. Avec des registres partagés, l'IRO utilisé par la tâche cyclique 1, par exemple, est le même que celui utilisé par la tâche cyclique 2.

Le paramètre qui détermine si les registres sont indépendants ou partagés est configuré à partir du CX-Programmer.

- Les autres mots et bits dans la mémoire E/S sont partagés par toutes les tâches. CIO 001000, par exemple, est le même bit pour la tâche cyclique 1 et la tâche cyclique 2. Veuillez par conséquent à toujours utiliser dans la programmation des zones de mémoire E/S autres que les zones IR et DR, car les valeurs modifiées avec une tâche seront utilisées par d'autres tâches.

Mémoire E/S	Rapport avec les tâches
Zones CIO, auxiliaire, de mémoire de données et toutes les autres zones de mémoire à l'exception des zones IR et DR (voir remarque 1).	Partagées avec les autres tâches.
Registres d'index (IR) et registres de données (DR) (voir remarque 2).	Utilisés séparément pour chaque tâche

**Remarque**

1. La banque EM actuelle est également partagée par les tâches. Par conséquent, si le numéro de la banque EM actuelle est modifié avec la tâche cyclique 1, par exemple, le nouveau numéro de la banque EM actuelle sera également valide pour la tâche cyclique 2.
2. Les valeurs IR et DR ne sont pas définies au démarrage des tâches d'interruption (y compris des tâches cycliques extra). Si IR et DR sont utilisées dans une tâche d'interruption, ces valeurs doivent être définies par les instructions MOVR/MOVRW (MOVE TO REGISTER et MOVE TIMER/COUNTER PV TO REGISTER) dans la tâche d'interruption. Après l'exécution de la tâche d'interruption, IR et DR reprennent automatiquement les valeurs qu'elles avaient avant l'interruption.

**Rapport entre les tâches et le fonctionnement des temporisations**

Les valeurs actuelles des temporisations pour TIM, TIMX, TIMH, TIMHX, TMHH, TMHHX, TIMW, TIMWX, TMHW et TMHWX programmées avec les numéros de temporisation 0000 à 2047 seront mises à jour même si la tâche est commutée ou si la tâche contenant la temporisation passe à l'état En attente ou de nouveau à l'état READY.

Si la tâche contenant TIM passe à l'état En attente et repasse à l'état READY, le drapeau de fin passera à ON si l'instruction TIM est exécutée lorsque la valeur actuelle est 0 (les drapeaux de fin de temporisation sont mis à jour uniquement à l'exécution de l'instruction). Si l'instruction TIM est exécutée alors que la valeur actuelle n'est pas encore égale à 0, la valeur actuelle continuera d'être mise à jour comme elle l'était lorsque la tâche était à l'état READY.

- Les valeurs actuelles des temporisations programmées avec les numéros de temporisation 2048 à 4098 seront conservées pendant que la tâche sera à l'état En attente.

**Rapport entre les tâches et les drapeaux de condition**

Tous les drapeaux de condition seront remis à zéro avant l'exécution de chaque tâche. Par conséquent, l'état du drapeau de condition à la fin de la tâche 1 ne peut pas être lu dans la tâche 2. Avec une UC CS1-H, CJ1-H, CJ1M ou CS1D, toutefois, vous pouvez utiliser CCS(282) et CCL(283) pour lire l'état du drapeau de condition dans une autre partie du programme, par exemple, dans une autre tâche.

**Remarque**

Lorsque l'état des drapeaux de condition est surveillé à partir d'une console de programmation, celle-ci affiche l'état des drapeaux à la fin du cycle, c'est-à-dire leur état à la fin de la dernière tâche du cycle.

**4-2-2 Limites des instructions relatives aux tâches****Instructions requises dans la même tâche**

Les instructions suivantes doivent être placées dans la même tâche. Si vous essayez de séparer des instructions entre deux tâches, le drapeau ER passera à ON et les instructions ne seront pas exécutées.

Mnémonique	Instruction
JMP/JME	JUMP/JUMP END
CJP/JME	CONDITIONAL JUMP/JUMP END
CJPN/JME	CONDITIONAL JUMP NOT/CONDITIONAL JUMP END
JMP0/JME0	MULTIPLE JUMP END

Mnémonique	Instruction
FOR/NEXT	FOR/NEXT
IL/ILC	INTERLOCK/INTERLOCK CLEAR
SBS/RET :	SUBROUTINE CALL/SUBROUTINE ENTRY/SUBROUTINE RETURN
MCRO/SBN/RET	MACRO/SUBROUTINE ENTRY/SUBROUTINE RETURN
BPRG/BEND	BLOCK PROGRAM BEGIN/BLOCK PROGRAM END
STEP S/STEP	STEP DEFINE

### Instructions non autorisées dans les tâches d'interruption

Les instructions suivantes ne peuvent pas être placées dans les tâches d'interruption. Si vous essayez d'exécuter l'une de ces instructions dans une tâche d'exécution, le drapeau ER passera à ON et l'instruction ne sera pas exécutée. Les instructions suivantes peuvent être utilisées si une tâche d'interruption est utilisée en tant que tâche extra.

Mnémonique	Instruction
TKON(820)	TASK ON
TKOF(821)	TASK OFF
STEP	STEP DEFINE
SNXT	STEP NEXT
STUP	CHANGE SERIAL PORT SETUP
DI	DISABLE INTERRUPTS
EI	ENABLE INTERRUPTS

Le fonctionnement des instructions suivantes est imprévisible dans une tâche d'interruption : TIMER : TIM et TIMX((550), HIGH-SPEED TIMER : TIMH(015) et TIMHX(551), ONE-MS TIMER : TMHH(540) et TMHHX(552), ACCUMULATIVE TIMER : TTIM(087) et TTIMX(555), MULTIPLE OUTPUT TIMER : MTIM(543) et MTIMX(554), LONG TIMER : TIML(542) et TIMLX(553), TIMER WAIT : TIMW(813) et TIMWX(816), HIGH-SPEED TIMER WAIT : TMHW(815) et TMHWX(817), PID CONTROL : PID(190), FAILURE POINT DETECTION : FPD(269) et CHANGE SERIAL PORT SETUP : STUP(237).

Les instructions suivantes ne peuvent pas être utilisées dans la tâche d'interruption de mise hors tension (elles ne seront pas exécutées, même si elles sont utilisées, et le drapeau d'erreur ne passera pas à ON) :

READ DATA FILE : FREAD(700), WRITE DATA FILE : FWRT(701), NETWORK SEND : SEND(090), NETWORK RECEIVE : RECV(098), DELIVER COMMAND : CMND(490), PROTOCOL MACRO : PMCR(260).

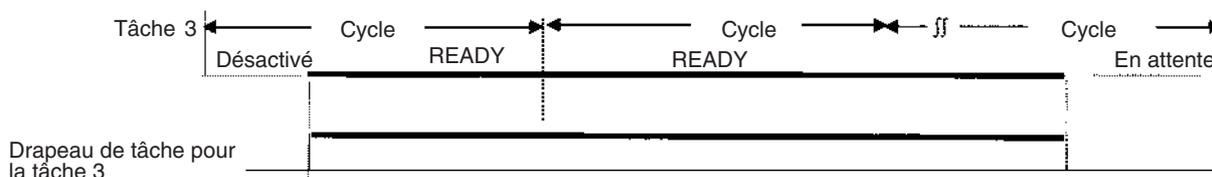
## 4-2-3 Drapeaux relatifs aux tâches

### Drapeaux relatifs aux tâches cycliques

Les drapeaux suivants fonctionnent uniquement pour les tâches cycliques normales. Ils ne sont pas opérationnels pour les tâches cycliques extra.

#### Drapeaux de tâches (TK00 à TK31)

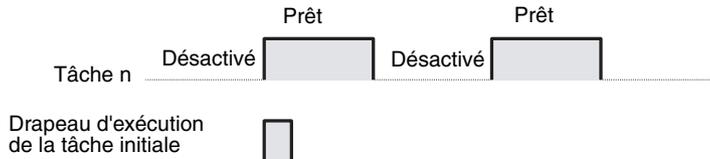
Un drapeau de tâche passe à ON lorsqu'une tâche cyclique est à l'état READY et à OFF lorsque la tâche est à l'état Désactivé (INI) ou En attente (WAIT). Les numéros de tâches 00 à 31 correspondent aux drapeaux de tâches TK00 à TK31.



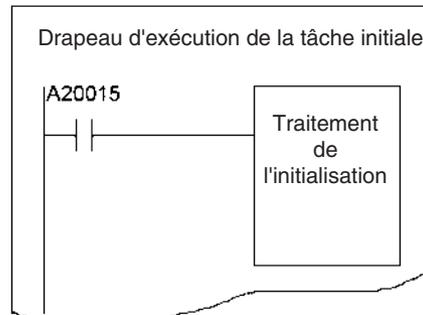
**Remarque** Les drapeaux de tâches sont utilisés uniquement avec des tâches cycliques, et non avec des tâches d'interruption. Avec une tâche d'interruption, A44115 sera activé si la tâche s'exécute après le début de l'opération et le numéro de la tâche d'interruption qui a nécessité un temps de traitement maximum sera stocké dans un format hexadécimal à deux chiffres dans les drapeaux A44100 à A44107.

**Drapeau d'exécution de la tâche initiale (A20015)**

Le drapeau d'exécution de la tâche initiale passe à ON lorsque les tâches cycliques passent de l'état Désactivé (INI) à l'état READY, lorsque les tâches obtiennent le droit d'exécution et à la première exécution des tâches. Il passe à OFF lorsque la première exécution des tâches est terminée.



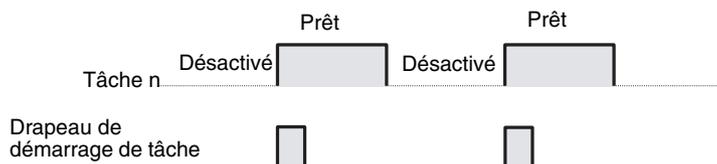
Le drapeau d'exécution de la tâche initiale indique si les tâches cycliques sont exécutées pour la première fois ou non. Ce drapeau peut ainsi être utilisé pour exécuter le traitement d'initialisation à l'intérieur des tâches.



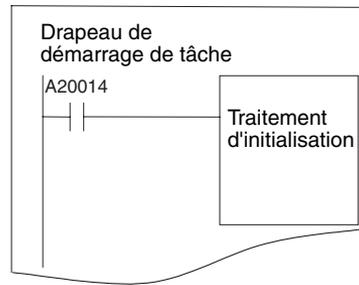
**Remarque** Bien qu'une tâche cyclique à l'état En attente repasse à l'état READY via l'instruction TKON(820), cela n'est pas considéré comme une exécution initiale et le drapeau d'exécution de la tâche initiale (20015) ne passe pas à ON. Le drapeau d'exécution de la tâche initiale (20015) ne passe pas non plus à ON si une tâche cyclique passe de l'état Désactivé à l'état RUN ou si elle est mise à l'état En attente par une autre tâche via l'instruction TKOF(821) avant l'obtention du droit d'exécution réel.

**Drapeau de démarrage de tâche (UC A20014, CS1-H, CJ1-H, CJ1M ou CS1D uniquement)**

Le drapeau de démarrage de tâche peut être utilisé pour exécuter le traitement d'initialisation à chaque démarrage du cycle de la tâche. Il passe à OFF lorsque l'état de la tâche cyclique passe de Désactivé (INI) ou En attente (WAIT) à READY (alors que le drapeau d'exécution de la tâche initiale passe à ON uniquement lorsque l'état passe de Désactivé (INI) à READY).



Le drapeau de démarrage de tâche peut être utilisé pour effectuer le traitement d'initialisation lorsqu'une tâche passe de l'état En attente à l'état RUN, c'est-à-dire lorsqu'une tâche à l'état En attente est activée avec l'instruction TRON(820).



### Drapeaux relatifs à toutes les tâches

#### **Drapeau d'erreur de tâche (A29512)**

Le drapeau d'erreur de tâche passera à ON si l'une des erreurs de tâche suivantes survient.

- Aucune tâche cyclique (y compris les tâches cycliques extra) n'est à l'état READY pendant un cycle.
- Le programme affecté à une tâche cyclique (y compris à une tâche cyclique extra) n'existe pas (cette situation ne se produira pas avec CX-Programmer ou une console de programmation).
- Aucun programme n'est affecté à une tâche d'interruption activée.

#### **Numéro de tâche lorsque le programme est arrêté (A294)**

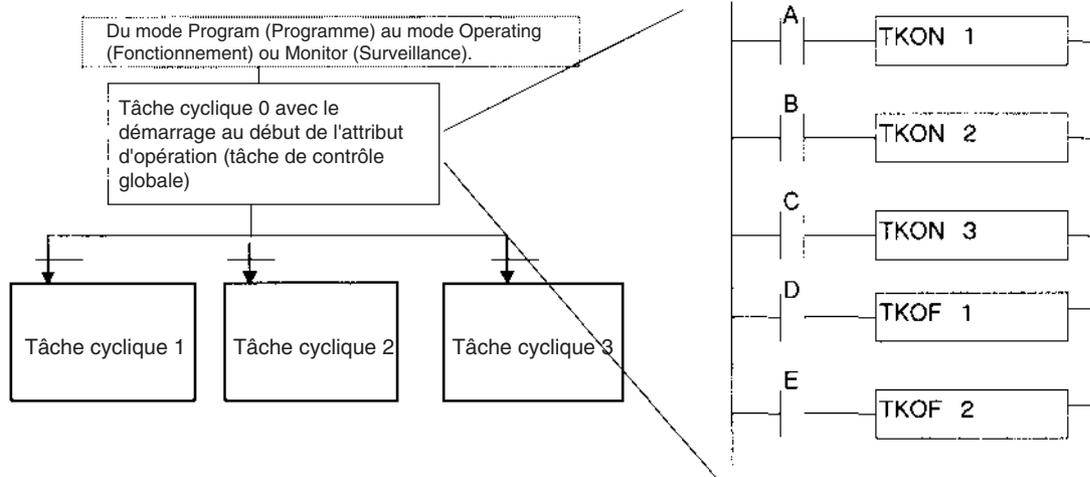
Le type de tâche et le numéro de la tâche actuelle, lorsque l'exécution d'une tâche s'interrompt suite à une erreur du programme, sont sauvegardés comme suit :

Type	A294
<b>Tâches cycliques</b>	0000 à 001F hex. (numéros de tâches 0 à 31)
<b>Tâche d'interruption</b>	8000 à 80FF hex. (correspondent aux numéros de tâches d'interruption 0 à 255)

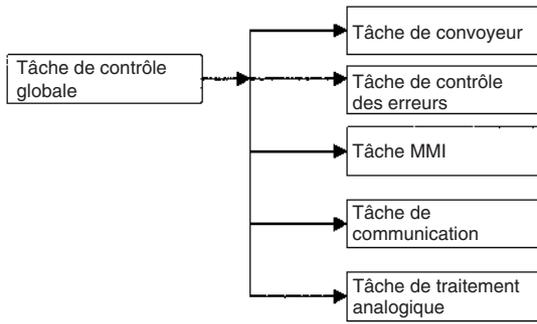
Cette information permet de déterminer facilement l'emplacement de l'erreur fatale et elle est supprimée lorsque l'erreur est corrigée. L'adresse du programme où la tâche s'est arrêtée est sauvegardée dans A298 (bits situés à l'extrême droite de l'adresse du programme) et dans A299 (bits situés à l'extrême gauche de l'adresse du programme).

### Exemples de tâches

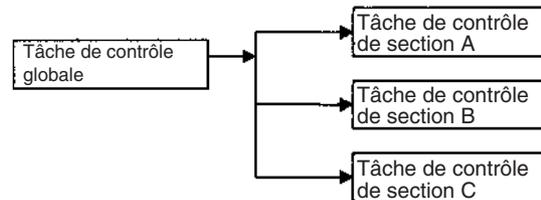
Une tâche de contrôle globale configurée pour passer à l'état READY au début de l'opération est généralement utilisée pour contrôler l'état READY/En attente de toutes les autres tâches cycliques (y compris des tâches cycliques extra). Naturellement, n'importe quelle tâche cyclique peut contrôler l'état READY/En attente d'une autre tâche cyclique comme l'exige l'application.



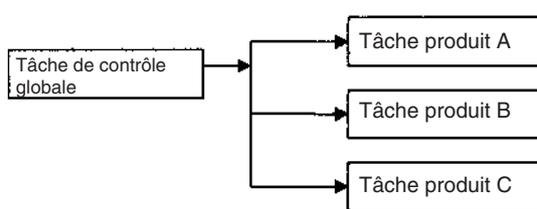
Tâches séparées par fonction



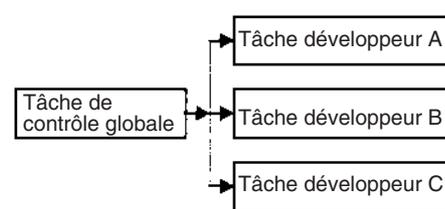
Tâches séparées par section contrôlée



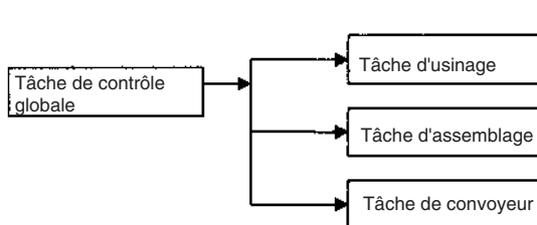
Tâches séparées par produit



Tâches séparées par développeur



Tâches séparées par procédé



Des combinaisons des classifications ci-dessus sont également possibles (classification par fonction et processus, etc.).

## 4-2-4 Conception de tâches

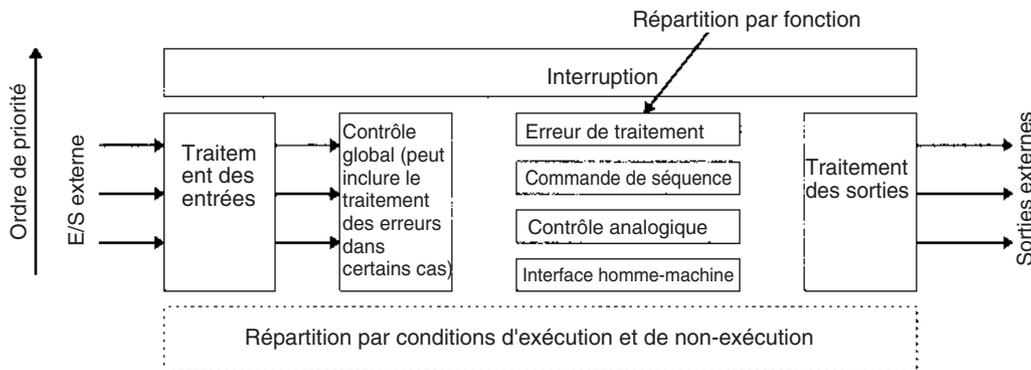
Nous vous recommandons de respecter les directives suivantes pour la conception des tâches.

- 1,2,3...**
1. Utilisez les normes suivantes pour étudier la séparation des tâches.
    - a. Récapitulez les conditions spécifiques pour l'exécution et la non-exécution.
    - b. Récapitulez la présence ou l'absence E/S externes.
    - c. Récapitulez les fonctions.

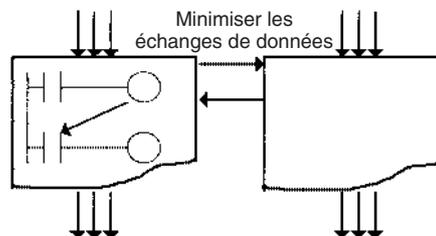
Limitez au maximum les échanges de données entre les tâches pour la commande de séquence, le contrôle analogique, l'interface homme-machine, le traitement des erreurs et autres process, afin de conserver un niveau d'autonomie élevé.

- d. Récapitulez l'exécution dans l'ordre de priorité.

Divisez le traitement entre tâches cycliques et tâches d'interruption.



2. Veillez à segmenter et à concevoir les programmes de façon à garantir l'autonomie et à limiter au maximum la quantité de données échangée entre les tâches (programmes).



3. D'une manière générale, utilisez une tâche de contrôle globale pour contrôler l'état READY/En attente des autres tâches.
4. Affectez les plus petits numéros aux tâches auxquelles est associé le niveau de priorité le plus haut.  
Exemple : affectez à la tâche de contrôle un numéro plus petit qu'aux tâches de traitement.
5. Affectez des numéros plus petits aux tâches d'interruption avec un niveau de priorité élevé.
6. Une tâche avec l'état READY sera exécutée dans les cycles ultérieurs tant que la tâche proprement dite ou une autre tâche ne la fera pas passer à l'état En attente. Veillez à insérer une instruction TKOF(821) (TASK OFF) pour les autres tâches si le traitement doit être ramifié entre les tâches.
7. Utilisez le drapeau d'exécution de la tâche initiale (A20015) ou le drapeau de démarrage de tâche (A20014) dans la condition d'exécution des

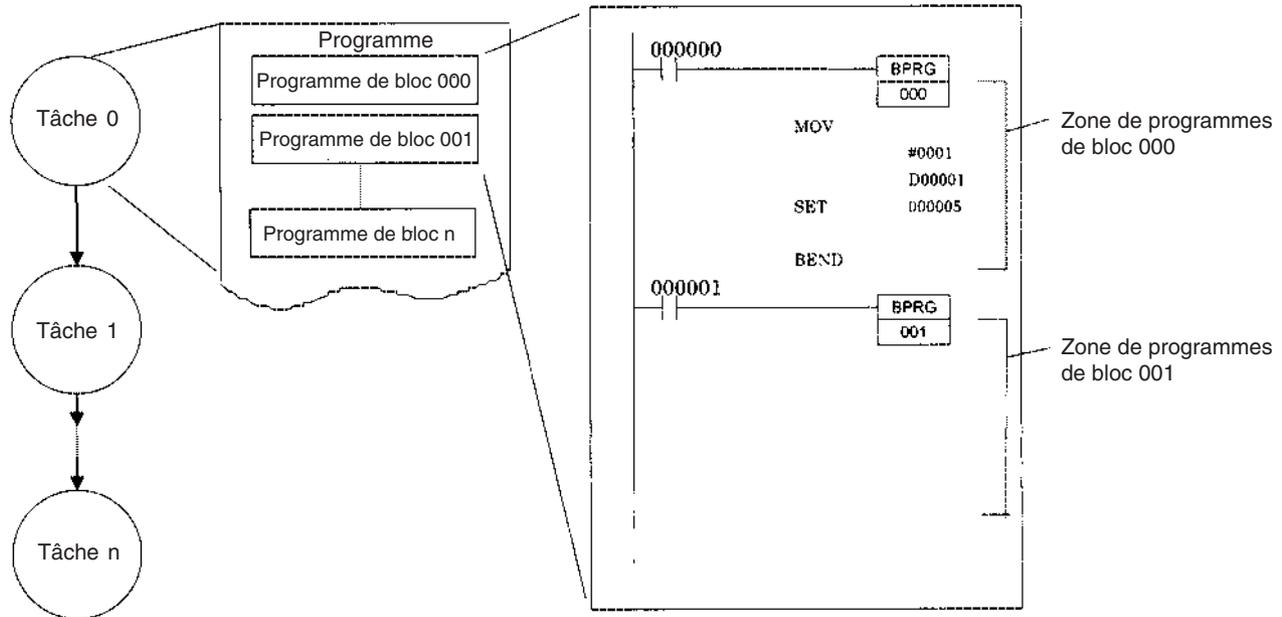
instructions pour initialiser les tâches. Le drapeau d'exécution de la tâche initiale sera sur ON pendant la première exécution de chaque tâche. Le drapeau de démarrage de tâche chaque fois qu'une tâche passe à l'état READY.

8. Affectez la mémoire E/S dans la mémoire partagée par les tâches et la mémoire utilisée uniquement pour les tâches individuelles, puis groupez la mémoire E/S utilisée uniquement pour les tâches individuelles par tâche.

**Rapport entre les tâches et les programmes de bloc**

Vous pouvez créer jusqu'à 128 programmes de bloc dans les tâches. Ce total vaut pour l'ensemble des tâches. L'exécution de chaque programme de bloc est contrôlée à partir du schéma contact, mais les instructions à l'intérieur du programme de bloc sont écrites à l'aide de mnémoniques. En d'autres termes, un programme de bloc est constitué d'une instruction de schéma contact combinée à un code mnémotechnique.

L'utilisation d'un programme de bloc simplifie l'écriture d'un flux logique (branchement conditionnel, exécution de processus pas à pas, etc.) qui pourrait s'avérer difficile avec des schémas contacts. Les programmes de bloc se trouvent en bas de la hiérarchie des programmes et les unités importantes de programme représentées par la tâche peuvent être réparties en petites unités de programme comme des programmes de bloc fonctionnant avec la même condition d'exécution (condition ON).



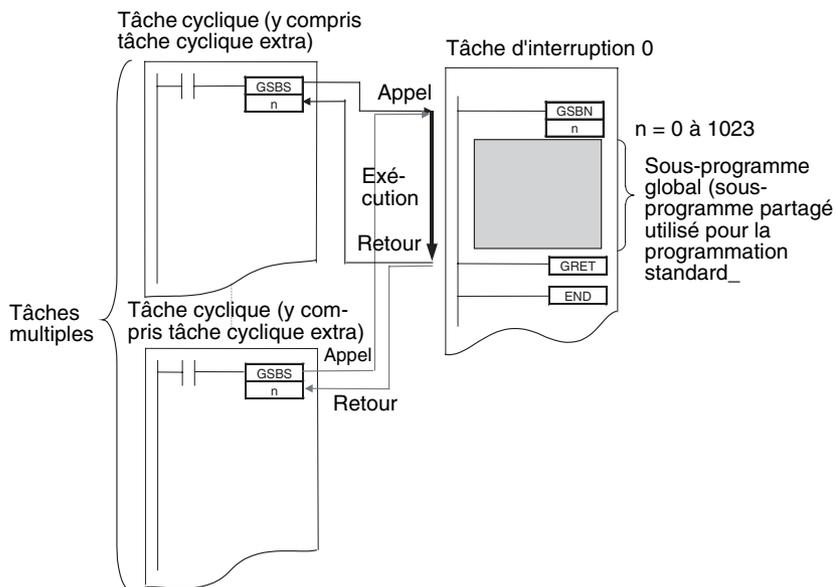
**4-2-5 Sous-programmes globaux**

Les sous-programmes globaux peuvent être appelés à partir de plusieurs tâches. Ils sont pris en charge par les UC CS1-H, CJ1-H, CJ1M et CS1D uniquement.

Avec les UC CS1 ou CJ1, un sous-programme dans une tâche ne peut pas être appelé à partir d'autres tâches. Avec les UC CS1-H, CJ1-H, CJ1M ou CS1D, cependant, il est possible de créer des sous-programmes globaux dans une tâche d'interruption numéro 0 et ces sous-programmes peuvent être appelés à partir de tâches cycliques (y compris des tâches cycliques extra).

L'instruction GSBS permet d'appeler un sous-programme global. Le numéro du sous-programme doit être compris entre 0 et 1023. Le sous-programme global est défini à la fin d'une tâche d'interruption numéro 0 (juste avant END(001)), entre les instructions GSBN et GRET.

Les sous-programmes globaux peuvent être utilisés pour créer une bibliothèque de sections de programme standard que vous appellerez quand vous en aurez besoin.



### 4-3 Tâche d'interruption

#### 4-3-1 Types de tâches d'interruption

Les tâches d'interruption peuvent être exécutées à tout moment du cycle si l'une des conditions suivantes est remplie.

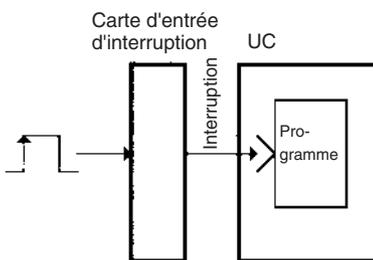
Les entrées d'interruption intégrées et les entrées des compteurs à grande vitesse sur une UC CJ1M peuvent être utilisées pour activer des tâches d'interruption. Pour des informations détaillées, reportez-vous au *Manuel d'utilisation des E/S intégrées, série CJ*.

**Remarque**

Les UC CS1D pour les systèmes d'UC en duplex ne prennent pas en charge les interruptions. Avec les UC CS1D, les tâches d'interruption ne peuvent être utilisées que comme tâches cycliques extra.

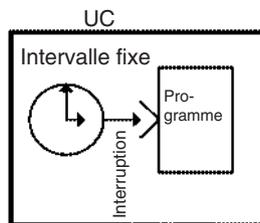
**Interruptions d'E/S (série CS uniquement)**

La tâche d'interruption d'E/S est exécutée lorsque l'entrée de la carte d'entrée d'interruption est sur ON.



**Interruptions programmées**

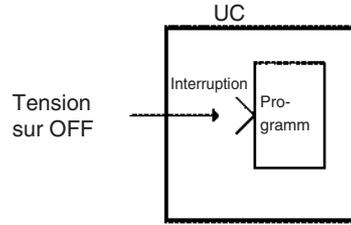
Une tâche d'interruption programmée est exécutée à intervalles fixes.



**Interruption de mise hors tension**

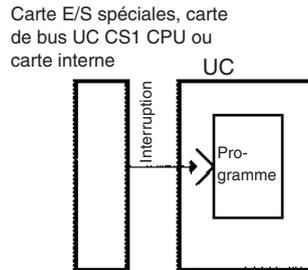
La tâche d'interruption de mise hors tension est exécutée lors de la mise hors tension.

**Remarque** Le temps d'exécution pour la tâche de mise hors tension doit être inférieur à 10 ms (temps de retard de détection de mise hors tension).



**Interruptions externes (série CS uniquement)**

Une tâche d'interruption externe sera exécutée lorsqu'une interruption sera demandée par une carte d'E/S spéciales, une carte réseau ou une carte interne (série CS uniquement). La carte d'E/S spéciales ou la carte réseau CJ, cependant, doit être montée sur le rack UC pour demander l'exécution d'une tâche d'interruption externe.



**Liste des tâches d'interruption**

Type	N° de tâche	Condition d'exécution	Procédure de configuration	Nombre d'interruptions	Exemples d'application
Interruption d'E/S 00 à 31	100 à 131	Entrée de la carte d'entrée d'interruption ON sur le rack UC (voir remarque 1).	Utilisez l'instruction MSKS (SET INTERRUPT MASK) pour affecter les entrées des cartes d'entrées d'interruption sur le rack UC.	32 points	Augmentation de la vitesse de réponse vers des entrées spécifiques
Interruptions programmées 0 et 1	2 et 3	Programmée (intervalles fixes)	Utilisez l'instruction MSKS (SET INTERRUPT MASK) pour définir l'intervalle des interruptions. Voir Unités de temps des interruptions programmées dans la configuration de l'API.	2 points	Surveillance de l'état de fonctionnement à intervalles réguliers.
Interruption de mise hors tension	1	Lorsque l'alimentation est mise hors tension (après le temps de détection de la mise hors tension par défaut + le temps de retard de détection de mise hors tension)	Voir Temps de retard de détection de mise hors tension dans la configuration de l'API.	1 point	Exécution d'un traitement d'urgence lorsque l'alimentation est coupée.
Interruptions externes 0 à 255	0 à 255	A la demande d'une carte d'E/S spéciales ou d'une carte réseau sur le rack UC ou d'une carte interne (série CS uniquement) (voir remarque 2).	Aucune (toujours valide)	256 points	Exécution du traitement requise par les cartes d'E/S spéciales, les cartes réseau et la carte interne.

**Remarque** 1. La carte d'entrée d'interruption doit être montée sur le rack UC. Pour les UC CJ1-H, la carte doit être connectée comme étant l'une des cinq unités (emplacements 0 à 4) présentes à côté de l'UC. Pour les UC CJ1M, la carte doit être connectée comme étant l'une des trois unités (emplacements 0 à 2)

présentes à côté de l'UC. Les cartes d'interruption d'E/S installées à un autre emplacement ne peuvent pas être utilisées pour demander l'exécution de tâches d'interruption d'E/S.

2. La carte d'E/S spéciales ou la carte réseau doit être montée sur le rack UC. Pour les UC CJ1-H, la carte doit être connectée comme étant l'une des cinq unités (emplacements 0 à 4) présentes à côté de l'UC. Pour les UC CJ1M, la carte doit être connectée comme étant l'une des trois unités (emplacements 0 à 2) présentes à côté de l'UC. Les cartes installées à un autre emplacement ne peuvent pas être utilisées pour générer des interruptions externes.
3. Les UC CJ1 pour les systèmes d'UC en duplex ne prennent pas en charge les tâches d'interruption d'E/S et externes.
4. Les UC CS1D pour les systèmes d'UC en duplex ne prennent pas en charge les interruptions. Avec les UC CS1D, les tâches d'interruption peuvent être utilisées en tant que tâches cycliques extra uniquement ; aucun autre type de tâche d'interruption ne peut être utilisé.

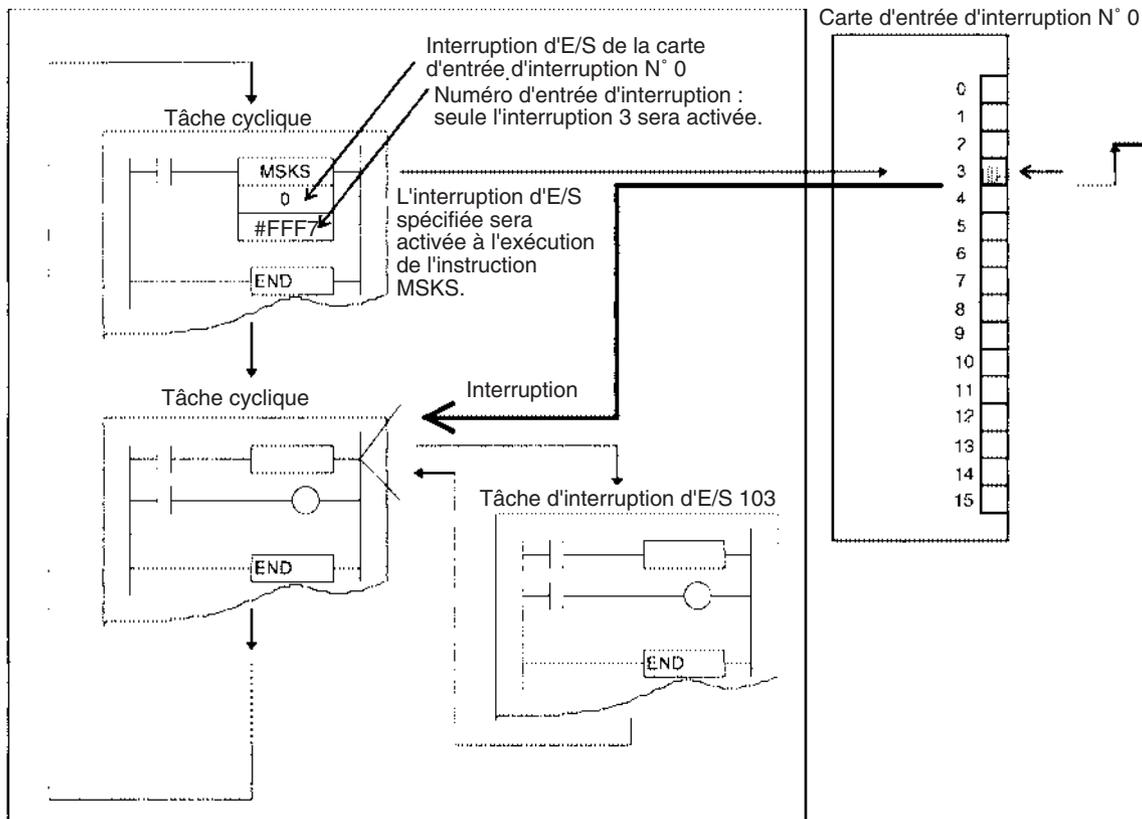
### Tâches d'interruptions d'E/S : Tâches 100 à 131

Les tâches d'interruption d'E/S sont désactivées par défaut au démarrage de l'exécution des tâches cycliques. Pour activer les interruptions d'E/S, exécutez l'instruction MSKS (SET INTERRUPT MASK) dans une tâche cyclique pour le numéro d'interruption correspondant à la carte d'entrée d'interruption.

**Exemple :** L'exemple suivant décrit l'exécution de la tâche d'interruption d'E/S 103 lorsque l'entrée d'interruption N° 3 de la carte d'entrée d'interruption N° 0 (celle des deux cartes 0 et 1 située le plus à gauche) est à ON.

**Remarque**

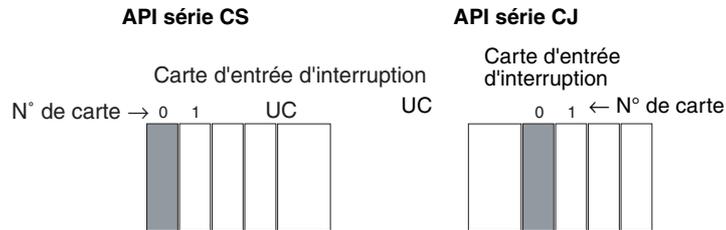
N'activez pas les tâches d'interruption d'E/S dont vous n'avez pas besoin. Si l'entrée d'interruption est déclenchée par le bruit et qu'il n'existe aucune tâche d'interruption correspondante, une erreur fatale (erreur de tâche) provoquera un arrêt du programme.



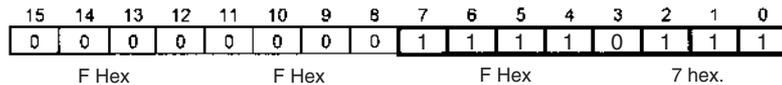
Numéros de cartes d'entrées d'interruption, Numéros d'entrée et numéros de tâches d'interruption d'E/S

N° de carte d'entrée d'interruption (voir remarque)	N° d'entrée	Tâche d'interruption d'E/S
0	0 à 15	100 à 115
1	0 à 15	116 à 131

**Remarque** Pour les API série CS, les numéros de cartes d'entrées d'interruption vont de 0 à 1 en partant de la gauche du rack UC. Pour les API série CJ, les numéros de cartes d'entrées d'interruption vont de 0 à 1 en partant de l'UC.



**Opérande S (le second opérande) de MSKS :** Les bits de FFF7 hex. correspondent aux entrées d'interruption de la carte d'entrée d'interruption. Les numéros d'entrée d'interruption 0 à 15 correspondent aux bits 0 à 15.



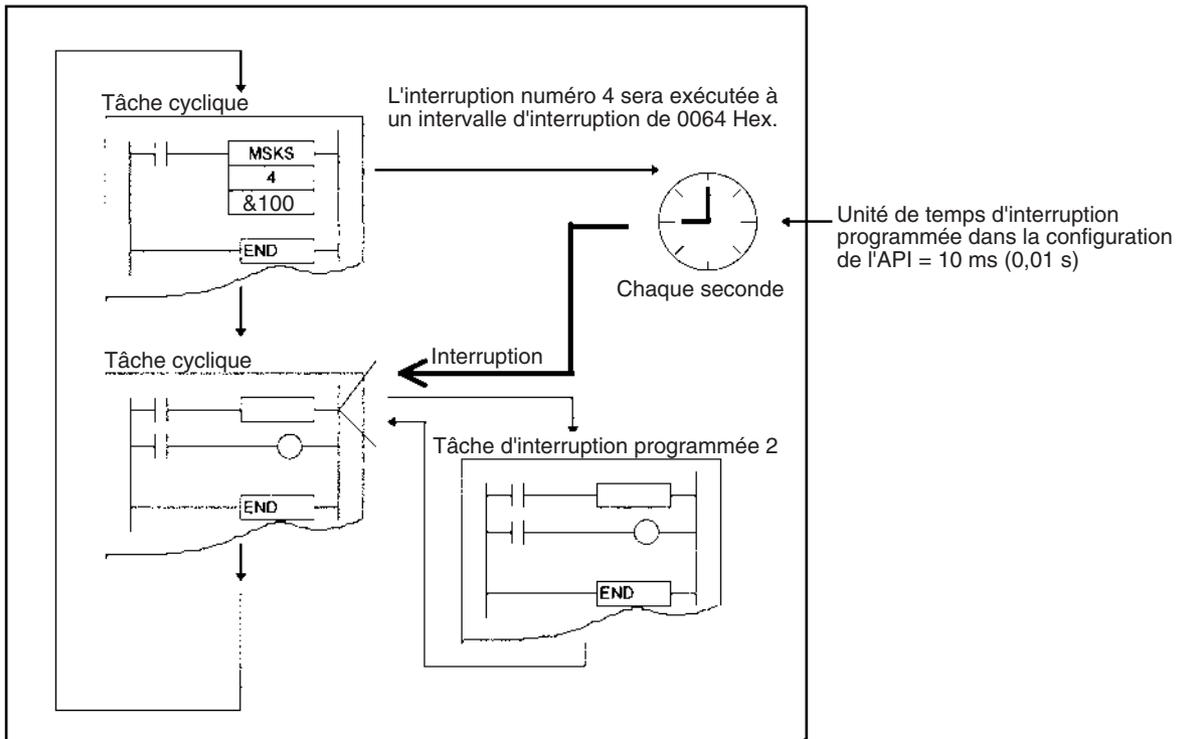
**Tâches d'interruption programmées : tâches 2 et 3**

Les tâches d'interruption programmées sont désactivées dans la configuration de l'API par défaut au début de l'exécution des tâches cycliques. Procédez comme suite pour activer les tâches d'interruption programmées.

- 1,2,3...**
1. Exécutez l'instruction MSKS (SET INTERRUPT MASK) à partir d'une tâche cyclique et définissez le temps (cycle) pour l'interruption programmée spécifiée.
  2. Définissez l'unité de temps de l'interruption programmée dans la configuration de l'API.

**Remarque** Le paramètre temps de l'interruption influe sur la tâche cyclique : plus il est court, plus la tâche s'exécute fréquemment et plus le temps de cycle est long.

**Exemple :** Les exemples suivants présentent une tâche d'interruption programmée pour être exécutée 2 fois par seconde.



**Numéros d'interruption et numéro de tâche d'interruption programmée**

N° d'interruption	Tâche d'interruption programmée
4	2
5	3

**Configuration de l'API**

Adresse	Nom	Description	Paramètres	Configuration par défaut
Bits 0 à 3 sur 195	Unités de temps des interruptions programmées	Définit l'unité de temps pour les interruptions programmées, de façon à exécuter les tâches d'interruption à intervalles fixes.	00 hex. : 10 ms 01 hex. : 1,0 ms 02 hex. : 0,1 ms (UC CJ1M uniquement)	00 hex. :

**Tâche d'interruption de mise hors tension : Tâche 1**

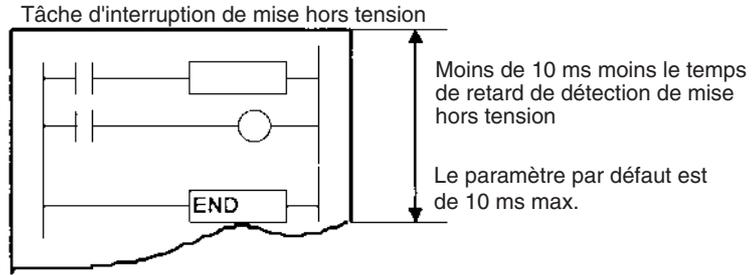
La tâche d'interruption de mise hors tension est désactivée dans la configuration de l'API par défaut au début de l'exécution de la tâche cyclique.

La tâche d'interruption de mise hors tension peut être activée dans la configuration de l'API.

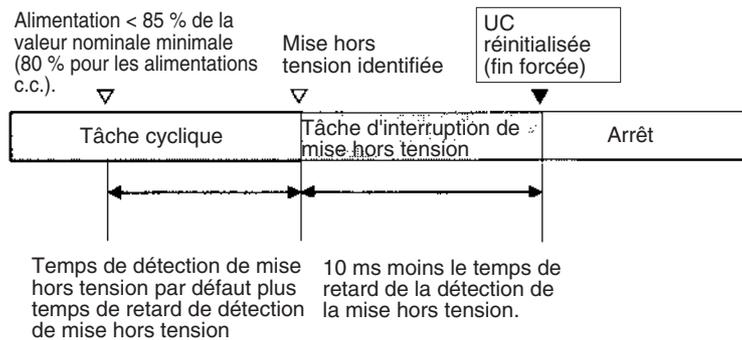
Dans la configuration de l'API par défaut, la tâche d'interruption de mise hors tension sera interrompue au bout de 10 ms. La tâche d'interruption de mise hors tension doit être exécutée en moins de 10 ms.

Si un temps de retard de détection de mise hors tension est défini dans la configuration de l'API, la tâche d'interruption de mise hors tension s'arrêtera au bout de 10 ms moins le temps de retard de détection de mise hors tension défini dans la configuration de l'API. Dans ce cas, la tâche d'interruption de mise hors tension devra s'exécuter en moins de 10 ms moins le temps de retard de détection de mise hors tension défini dans la configuration de l'API.

**Exemple :** si le temps de retard de détection de mise hors tension est défini à 4 ms dans la configuration de l'API, le temps d'exécution doit être inférieur à 10 moins 4 ms, soit 6 ms.

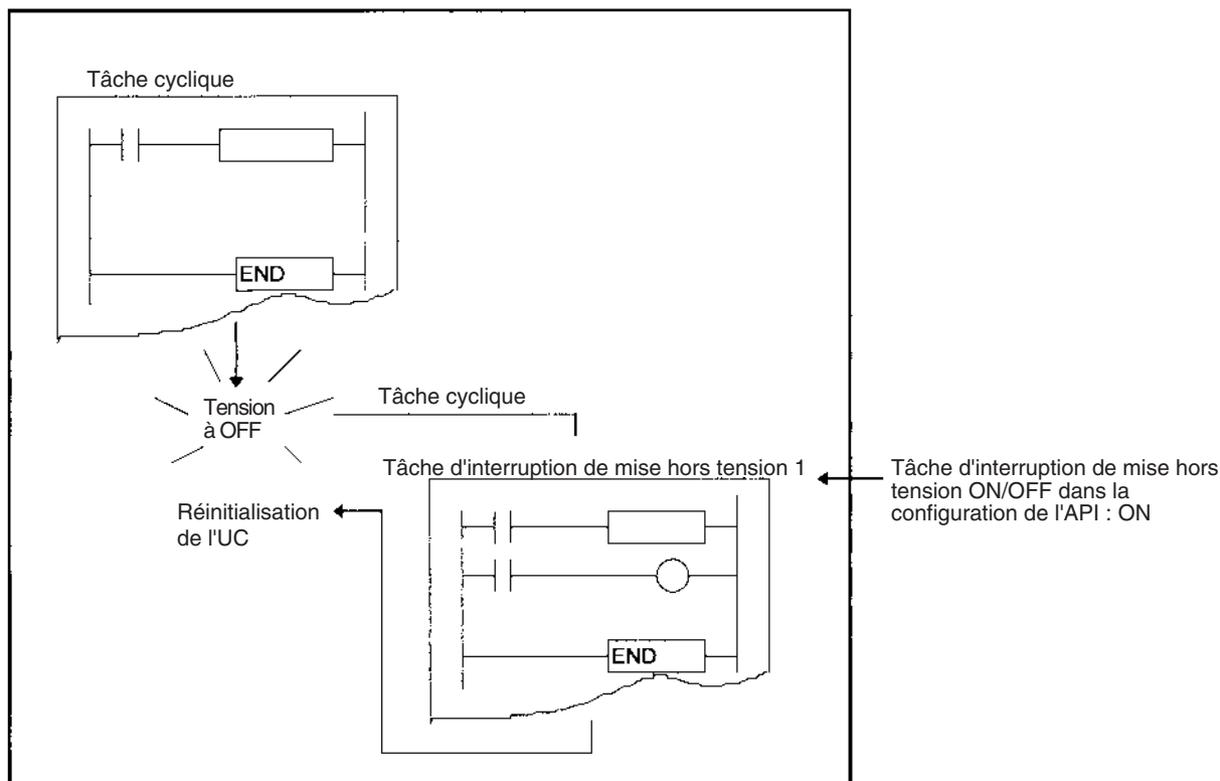


**Remarque** Une condition de mise hors tension est identifiée lorsque l'alimentation tombe au-dessous de 85 % de la tension nominale minimale (80 % pour les alimentations c.c.) et le délai avant l'exécution réelle de la tâche d'interruption de mise hors tension correspond au temps de détection de la mise hors tension par défaut (10 à 25 ms pour les alimentations c.a. et 2 à 5 ms pour les alimentations c.c.) plus le temps de retard de détection de mise hors tension défini dans la configuration de l'API (0 à 10 ms). Les tâches cycliques seront exécutées pendant cette durée.



**Remarque** Assurez-vous que la tâche d'interruption de mise hors tension peut être exécutée en moins de 10 ms moins le temps de retard de détection de mise hors tension défini dans la configuration de l'API. Les autres instructions ne seront pas exécutées après ce délai. La tâche d'interruption de mise hors tension ne sera pas exécutée si l'alimentation est coupée pendant l'édition en ligne. Outre les instructions qui ne peuvent être utilisées dans aucune tâche d'interruption (reportez-vous au *Manuel de référence des instructions* pour plus d'informations), les instructions suivantes ne peuvent pas être utilisées dans la tâche d'interruption de mise hors tension : READ DATA FILE FREAD(700), WRITE DATA FILE : FWRT(701), NETWORK SEND : SEND(090), NETWORK RECEIVE : RECV(098), DELIVER COMMAND : CMND(490), TRANSMIT : TXD(236), RECEIVE : RXD(235) et PROTOCOL MACRO : PMCR(260).

Exécution d'une tâche d'interruption de mise hors tension



Configuration de l'API pour la tâche d'interruption de mise hors tension (numéro de tâche : 1)

Adresse	Nom	Description	Paramètres	Configuration par défaut
Bit 15 à +225	Tâche d'interruption de mise hors tension	Si le bit 15 de +225 est à ON, une tâche d'interruption de mise hors tension démarrera si l'alimentation est coupée.	0 : OFF : 1 : ON	0
Bits 0 à 7 à +225	Temps de retard de détection de mise hors tension	La mise hors tension est identifiée lorsque ce délai plus le temps de détection de la mise hors tension par défaut (10 à 25 ms pour les alimentations c.a. et 2 à 5 ms pour les alimentations c.c.) expire.	00 à 0A hex. : 0 à 10 ms (unités de 1 ms)	00 hex.

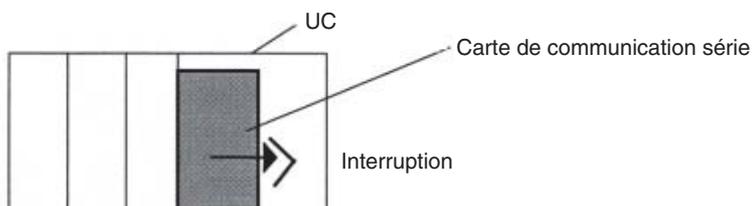
Tâches d'interruption externes : Tâches 0 à 255

Des tâches d'interruption externes peuvent être reçues à tout moment.

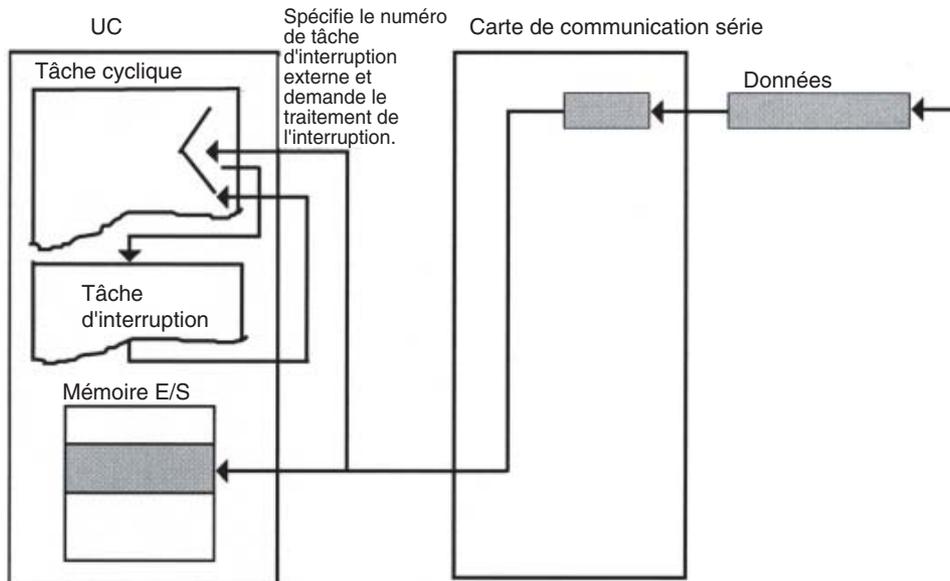
Le traitement des interruptions est effectué sur l'UC dans les API contenant une carte interne (série CS uniquement), des cartes d'E/S spéciales ou des cartes réseau. Il n'est pas nécessaire d'effectuer des réglages sur l'UC, sauf si le programme contient une tâche d'interruption externe pour un numéro de tâche particulier.

Les interruptions externes ne sont pas prises en charge par les UC CJ1.

**Exemple :** l'exemple suivant décrit une interruption externe générée par une carte de communications série CS1W-SCB□1.



Lorsque le message de réponse de la carte de communications série est configuré pour un message d'interruption (numéro fixe) ou un message d'interruption (numéro de cas de réception), la carte demande l'exécution d'une tâche d'interruption externe dans l'UC après avoir reçu des données de son port série et écrit ces données dans la mémoire E/S de l'UC.



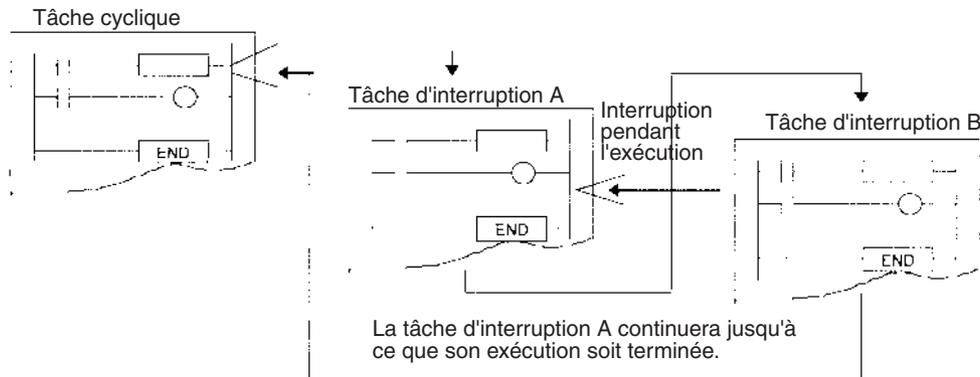
- Remarque**
1. Lorsque le message de réponse est configuré pour un message d'interruption (numéro fixe), la carte demande l'exécution de la tâche d'interruption correspondant au numéro de tâche prédéfini.
  2. Lorsque le message de réponse est configuré pour un message d'interruption (numéro de cas de réception), le numéro de la tâche d'interruption externe est calculé avec la formule spécifiée et la carte demande l'exécution de la tâche d'interruption correspondant à ce numéro de tâche.
  3. Si une tâche d'interruption externe (0 à 255) a le même numéro qu'une tâche de mise hors tension (tâche 1), qu'une tâche d'interruption programmée (tâche 2 ou 3) ou qu'une tâche d'interruption d'E/S (100 à 131), la tâche d'interruption sera exécutée pour l'une ou l'autre condition d'interruption (la condition d'interruption externe ou l'autre condition d'interruption). En principe, les numéros de tâches ne doivent jamais être dupliqués.

### 4-3-2 Priorité des tâches d'interruption

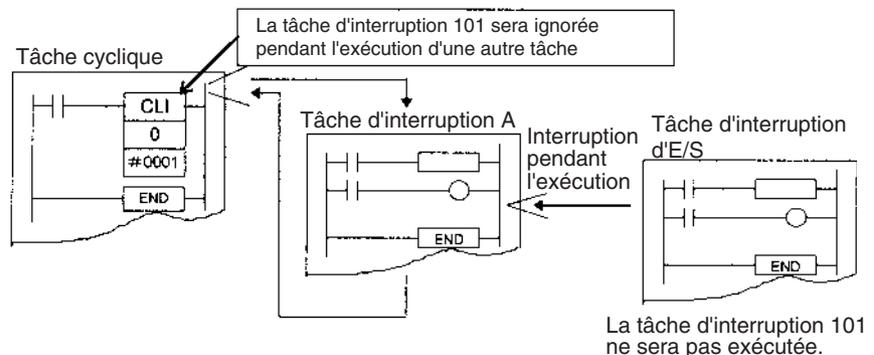
L'exécution d'une autre tâche d'interruption sera arrêtée pour que la tâche d'interruption de mise hors tension puisse être exécutée. L'UC sera réinitialisée, mais la tâche d'interruption terminée ne sera pas exécutée après l'exécution de la tâche d'interruption de mise hors tension.

**Interruption pendant l'exécution d'une tâche d'interruption**

Si une interruption survient pendant l'exécution d'une autre tâche d'interruption, la tâche d'interruption ne sera pas exécutée avant que l'exécution de l'interruption d'origine ne soit terminée.

**Remarque**

Si vous ne souhaitez pas qu'une tâche d'interruption d'E/S avec un numéro spécifique soit sauvegardée et exécutée pour une UC série CS lorsqu'elle survient pendant l'exécution d'une autre tâche d'exécution, exécutez l'instruction CLI (CLEAR INTERRUPT) à partir de l'autre tâche d'interruption pour EFFACER le numéro d'interruption sauvegardé en interne. Les interruptions programmées et les interruptions externes ne peuvent pas être annulées.

**Plusieurs interruptions exécutées simultanément**

Les tâches d'interruption autre que tâches d'interruption de mise hors tension sont exécutées dans l'ordre de priorité suivant lorsque plusieurs interruptions surviennent simultanément.

Tâches d'interruption d'E/S (série CS uniquement) > tâches d'interruption externes (série CS uniquement) > tâches d'interruption programmées

Les différents types de tâches d'interruption seront exécutés dans l'ordre à partir du plus petit numéro s'il en existe plusieurs.

**Remarque**

Une seule interruption sera enregistrée en mémoire par tâche d'interruption et aucune interruption ne sera enregistrée pour une interruption qui est déjà en cours d'exécution. Etant donné le faible niveau de priorité des interruptions programmées et dans la mesure où une seule interruption est enregistrée à la fois, une interruption programmée peut être ignorée.

**4-3-3 Drapeaux de tâche d'interruption et mots****Temps maximal de traitement de la tâche d'interruption (A440)**

Le temps de traitement maximal d'une tâche d'interruption est sauvegardé en données binaires (unités de 0,1 ms) et il est remis à zéro au début de l'opération.

**Tâche d'interruption avec temps de traitement maximal (A441)**

Le numéro de tâche d'interruption avec un temps de traitement maximal est sauvegardé en données binaires. Les valeurs hexadécimales de 8000 à 80FF correspondent aux numéros de tâches 00 à FF hex.

A44115 passera à ON lorsque la première interruption surviendra après le début de l'opération. Le temps de traitement maximal pour les tâches d'interruption suivantes sera sauvegardé dans les deux chiffres de droite en hexadécimal et remis à zéro au début de l'opération.

**Drapeau d'erreur de tâche d'interruption (erreur non fatale) (A40213)**

Si la détection des erreurs de tâche d'interruption définie sur ON dans la configuration de l'API, le drapeau correspondant passera à ON en cas d'erreur de tâche d'interruption.

**Drapeau d'erreur de tâche d'interruption (A42615)/numéro de tâche générant l'erreur de tâche d'interruption (A42600 à 42611)**

Si A40213 est à ON, les données suivantes seront sauvegardées dans A42615 et A42600 à A42611.

A40213	Description des erreurs de tâche d'interruption	A42615	A42600 à 42611
Erreur de tâche d'interruption (si la détection des tâches d'interruption est mise à ON dans la configuration de l'API)	Si une tâche d'interruption est exécutée plus de 10 ms pendant la mise à jour de la carte d'E/S spéciales C200H ou des E/S déportées du BUS SYSMAC (série CS uniquement).	OFF	Le numéro de tâche d'interruption sera sauvegardé dans 12 bits de données binaires (tâche d'interruption 0 à 255 : 000 à OFF hex.).
	Lorsque vous essayez de mettre à jour des E/S pour un grand nombre de mots avec l'instruction IORF à partir d'une tâche d'interruption alors qu'une carte d'E/S spéciales est actualisée par une mise à jour E/S cyclique.	ON	Le numéro de carte de la carte d'E/S spéciales mise à jour sera sauvegardée dans 12 bits de données binaires (carte N° 0 à 95 : 000 à 05F hex.).

**Numéro de tâche lorsque le programme est arrêté (A294)**

Le type et le numéro courant de la tâche lorsqu'un programme s'arrête suite à une erreur seront sauvegardés aux emplacements suivants.

Type	A294
Tâche d'interruption	8000 à 80FF hex. (correspond à la tâche d'interruption N° 0 à 255)
Tâche cyclique	0000 à 001F hex. (correspond à la tâche N° 0 à 31)

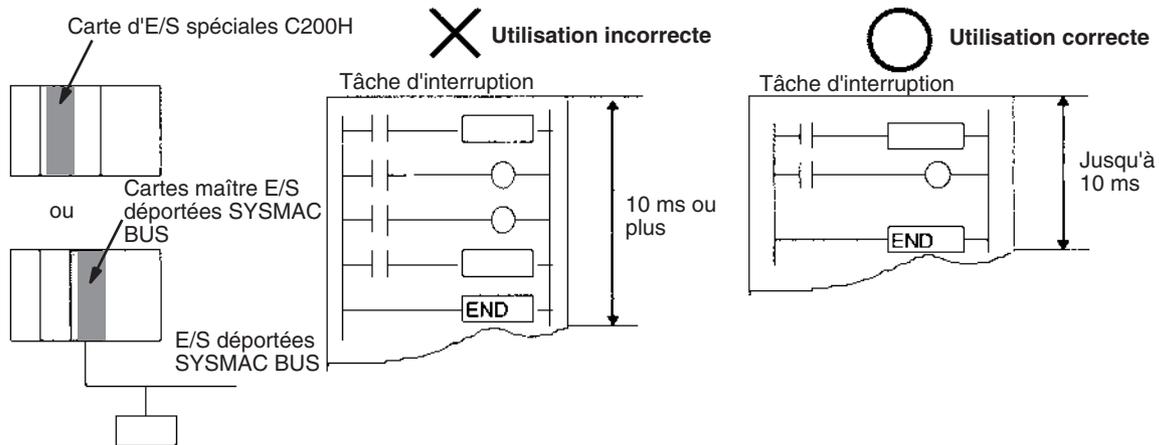
**4-3-4 Précautions en matière d'application****Temps d'exécution long avec cartes d'E/S spéciales C200H ou BUS SYSMAC (série CS uniquement)**

Assurez-vous que toutes les tâches d'interruption (E/S, programmées, de mise hors tension et externes) sont exécutées dans les 10 ms lorsque vous utilisez des cartes d'E/S spéciales C200H ou des E/S déportées du BUS SYSMAC.

Si une tâche d'interruption est exécutée plus de 10 ms pendant la mise à jour de la carte d'E/S spéciales C200H ou des E/S déportées du BUS SYSMAC, une erreur de tâche d'interruption surviendra, A40206 (drapeau d'erreur de carte d'E/S spéciales) passera à ON et la mise à jour E/S sera interrompue pour les cartes d'E/S spéciales. L'UC continuera cependant de fonctionner.

Si la détection des erreurs de tâche d'interruption est à ON dans la configuration de l'API, A40213 (drapeau d'erreur de tâche d'interruption) passera à ON lorsqu'une erreur de tâche d'interruption surviendra et le numéro de la tâche

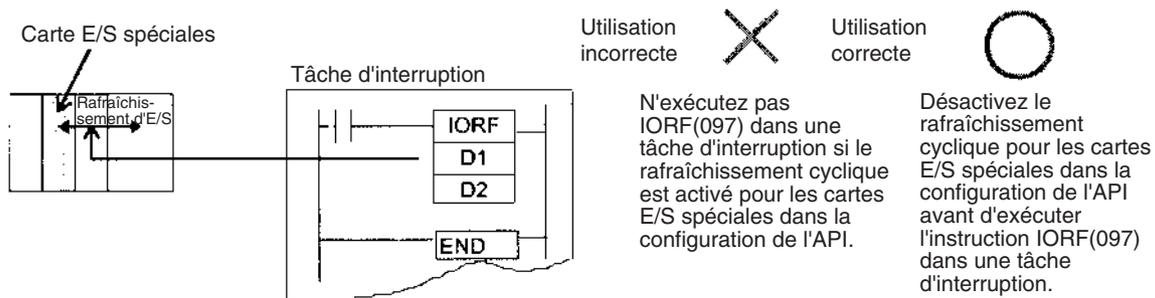
incriminée sera sauvegardé dans A426 (erreur de tâche d'interruption, numéro de tâche). L'UC continuera cependant de fonctionner.



**Exécution de IORF pour une carte d'E/S spéciales**

Si une instruction IORF(097) doit être exécutée à partir d'une tâche d'interruption pour d'une carte d'E/S spéciales, veillez à mettre la mise à jour cyclique à OFF pour la carte d'E/S spéciales (avec le numéro de carte) dans la configuration de l'API.

Une erreur de tâche d'interruption se produira si vous essayez de mettre à jour une carte d'E/S spéciales avec une instruction IORF(097) à partir d'une tâche d'interruption pendant que cette CARTE est mise à jour par d'autres moyens : mise à jour E/S cyclique ou tâches de mise à jour E/S (IORF(097) ou instructions de mise à jour immédiate (!)). Si la détection des erreurs de tâche d'interruption est à ON dans la configuration de l'API lorsqu'une erreur de tâche d'interruption survient, A40213 (drapeau d'erreur de tâche d'interruption) passera à ON et le numéro de la carte d'E/S spéciales pour laquelle la mise à jour E/S a été dupliquée sera sauvegardé dans A426 (erreur de tâche d'interruption, numéro de tâche). L'UC continuera de fonctionner.



**Remarque** Les bits situés à gauche de A426 (erreur de tâche d'interruption, numéro de tâche) permettent de déterminer les erreurs de tâche d'interruption qui sont survenues parmi les erreurs ci-dessus. (Bit 15 : erreur d'exécution de 10 ms ou supérieure si 0, erreur de mise à jour multiple si 1)

**Paramètres de configuration de l'API**

Adresse	Nom	Description	Paramètres	Configuration par défaut
Bit 14 à +128	Détection des erreurs de tâche d'interruption	Spécifie si les erreurs de tâches d'interruption doivent être détectées ou non. Le drapeau d'erreur de tâche d'interruption (A40213) sera à ON lorsque la détection sera activée.	0 : Détection activée, 1 : Détection désactivée	0

**Drapeaux/mots de zone auxiliaire associés**

Nom	Adresse	Description
Drapeau d'erreur de tâche d'interruption	A40213	<p>Passé à ON si une tâche d'interruption est exécutée plus de 10 ms pendant la mise à jour de la carte d'E/S spéciales C200H ou des E/S déportées du BUS SYSMAC, mais l'UC continue de fonctionner.</p> <p>Le voyant ERR/ALM s'allume sur le panneau avant (série CS uniquement).</p> <p>Passé à ON si vous essayez de mettre à jour une carte d'E/S spéciales avec une instruction IORF à partir d'une tâche d'interruption pendant que cette carte est actualisée par une mise à jour E/S cyclique.</p>
Erreur de tâche d'interruption, numéro de tâche	A426	<p>Contient le numéro de la tâche d'interruption ou le numéro de la carte d'E/S spéciales en cours de mise à jour.</p> <p>(Le bit 15 est à OFF lorsque l'exécution d'une tâche d'interruption exige 10 ms ou plus et à ON lorsque la mise à jour de la carte d'E/S spéciales dupliquée a eu lieu.)</p>

**Désactivation des interruptions**

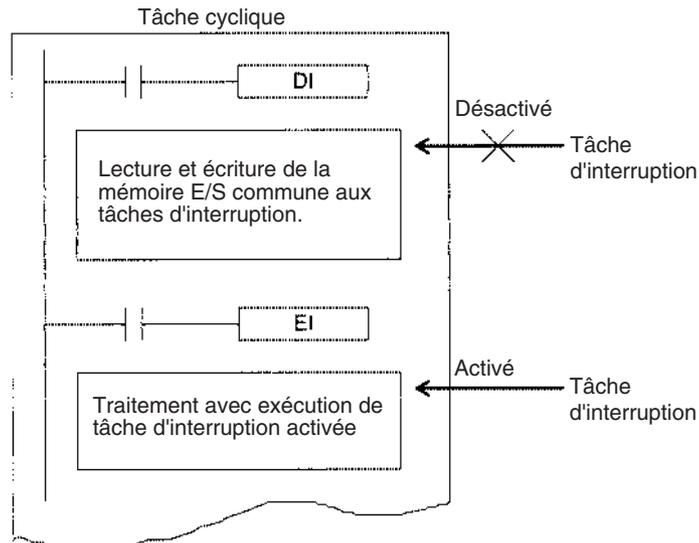
Le traitement sera interrompu et la tâche d'interruption sera exécutée dans les cas suivants.

- Lorsqu'une instruction est en cours d'exécution
- Pendant la mise à jour de la carte d'E/S standard, de la carte réseau, de la carte interne (série CS uniquement) ou des E/S déportées du BUS SYSMAC (série CS uniquement)
- Pendant une opération HOST LINK

**Concurrence de données entre les tâches cycliques et les tâches d'interruption**

Les données ne doivent pas être concurrentes si une tâche cyclique (y compris des tâches cycliques extra) et une tâche d'interruption lisent et écrivent les mêmes adresses mémoire E/S. Procédez comme suit pour désactiver les interruptions pendant que les instructions des tâches cycliques accèdent à la mémoire.

- Juste avant une opération de lecture ou d'écriture par une instruction de tâche cyclique, désactivez l'exécution des tâches d'interruption avec une instruction DI (DISABLE INTERRUPT).
- Utilisez une instruction EI (ENABLE INTERRUPT) juste après le traitement pour activer le traitement des tâches d'interruption.

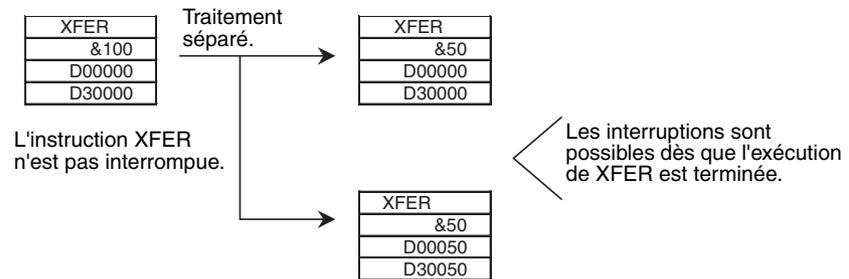


Des problèmes de concurrence de données peuvent survenir si DI(693) et EI(694) sont utilisées pour désactiver les tâches d'interruption pendant l'exécution d'une instruction qui exige la réception et le traitement d'une réponse (par exemple, une instruction de réseau ou de communication série).

**Remarque**

Avec les UC CS1-H, CJ1-H, CJ1M ou CS1D CPU Unit, l'exécution des instructions BIT COUNTER (BCNT), BLOCK SET (BSET) et BLOCK TRANSFER (XFER) ne sera pas interrompue pour exécuter la tâche d'interruption : l'exécu-

tion de l'instruction sera achevée avant d'exécuter la tâche d'interruption, retardant ainsi la réponse de l'interruption. Pour éviter cela, divisez le traitement des données pour ces instructions en plusieurs instructions, comme indiqué ci-après pour XFER.



## 4-4 Fonctionnement du périphérique de programmation pour les tâches

### 4-4-1 Utilisation de plusieurs tâches cycliques

Utilisez CX-Programmer pour créer plusieurs tâches cycliques (y compris des tâches cycliques extra). Il n'est pas possible de créer de nouvelles tâches cycliques avec une console de programmation. Veillez à utiliser un CX-Programmer pour affecter le type de tâche et le numéro de tâche pour les programmes créés.

- Plusieurs tâches cycliques créées et transférées vers une UC à partir de CX-Programmer peuvent être surveillées ou modifiées à partir d'une console de programmation.
- Vous pouvez facilement créer une tâche cyclique et une ou plusieurs tâches d'interruption avec la console de programmation en utilisant la fonction All Clear (Effacer tout) et en spécifiant Interrupt Tasks. Seules les tâches d'interruption 1 (interruption de mise hors tension), 2 et 3 (interruptions programmées) et 100 à 131 (interruptions d'E/S) peuvent être créées avec une console de programmation. Avec une UC CJ1M, cependant, les tâches d'interruption 140 à 143 (pour les entrées intégrées) peuvent également être créées. La tâche cyclique 0 démarrera lorsque l'API commencera de fonctionner.

### 4-4-2 Fonctionnement du périphérique de programmation

#### CX-Programmer

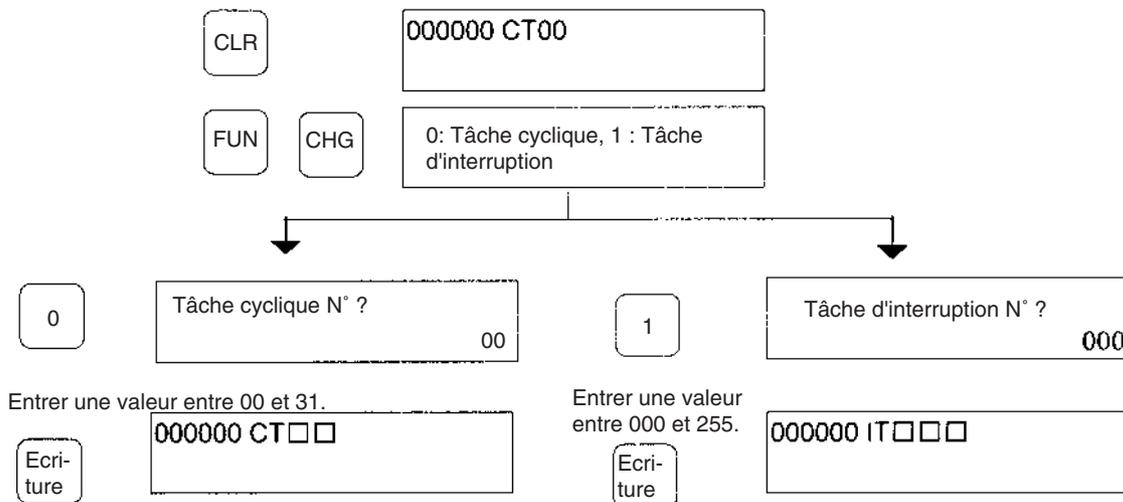
Spécifiez les attributs type et numéro de tâche pour chaque programme.

- 1,2,3... 1. Sélectionnez **View/Properties** (Affichage/Propriétés) ou cliquez sur le bouton droit et sélectionnez **Properties** dans le menu contextuel afin d'afficher le programme auquel une tâche sera affectée.
2. Sélectionnez l'onglet **General** et sélectionnez le type et le numéro de la tâche (**Task Type** et **Task No.**). Pour la tâche cyclique, activez la case à cocher **Operation start**.



**Console de programmation**

Une tâche est traitée comme l'ensemble du programme dans la console de programmation. Accédez à un programme et modifiez-le avec une console de programmation en spécifiant CT00 à CT31 pour une tâche cyclique ou IT001 à IT255 pour une tâche d'interruption.



- Remarque**
1. Une console de programmation ne peut pas créer de nouvelles tâches cycliques.
  2. Actuellement, les UC série CJ ne prennent pas en charge les tâches d'interruption d'E/S ou externes. Seules les tâches IT001 à IT003 peuvent être spécifiés.

# CHAPITRE 5

## Fonctions de mémoire de fichiers

Cette section décrit les fonctions utilisées pour gérer la mémoire de fichiers.

5-1	Mémoire de fichiers .....	192
5-1-1	Types de mémoire de fichiers .....	193
5-1-2	Données de fichiers .....	195
5-1-3	Fichiers .....	196
5-1-4	Description des procédures d'utilisation des fichiers .....	207
5-1-5	Applications .....	209
5-2	Traitement des fichiers .....	211
5-2-1	Périphériques de programmation (y compris les Consoles de programmation) .....	211
5-2-2	Commandes FINS .....	215
5-2-3	FREAD(700), FWRT(701) et CMND(490) .....	216
5-2-4	Remplacement du programme entier en cours d'opération .....	221
5-2-5	Transfert automatique au démarrage .....	228
5-2-6	Fonction de sauvegarde simple .....	234
5-3	Utilisation de la mémoire de fichiers .....	247
5-3-1	Initialisation des supports .....	247
5-3-2	Procédures de fonctionnement .....	249
5-3-3	Interruptions de l'alimentation pendant l'accès à la mémoire de fichiers .....	253

## 5-1 Mémoire de fichiers

Les unités de la série CS/CJ prennent en charge la mémoire de fichiers. Les supports suivants peuvent être utilisés en tant que mémoire pour le stockage de fichiers.

**1,2,3...**

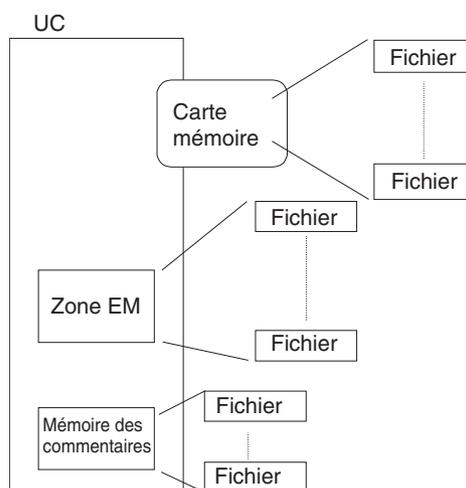
1. Cartes mémoire
2. Intervalle spécifié dans la zone EM appelée mémoire de fichiers EM

**Remarque** Les UC CJ1M ne possèdent pas de zone EM et ne permettent donc pas d'utiliser la mémoire de fichiers EM.

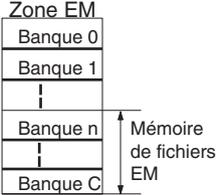
Les deux types de mémoire peuvent être employés pour sauvegarder intégralement le programme utilisateur, la mémoire E/S et la zone de paramètres dans des fichiers.

3. Mémoire des commentaires (dans la mémoire flash interne de l'UC)

Pour plus d'informations sur la sélection de la mémoire de fichiers, reportez-vous à la section 5-1-5 *Applications*, page 209.



### 5-1-1 Types de mémoire de fichiers

Catégorie	Type	Capacité	Modèle	Données de fichiers reconnues par l'UC	Opérations autorisées sur les fichiers
Cartes mémoire 	Mémoire flash	30 Mo 64 Mo	HMC-EF372 HMC-EF672	1) Programme utilisateur complet 2) Plage spécifiée dans la mémoire E/S 3) Données de la zone de paramètres (configuration de l'API et autres paramètres) Voir remarque 4.	Toutes sont possibles. (Voir page 207 pour plus de détails.) La fonction de transfert automatique au démarrage ne peut pas transférer les données de la mémoire de fichiers EM. (Voir page 207 pour plus de détails.)
Mémoire de fichiers EM 	RAM	Capacité de zone EM des UC Série CS CS1H-CPU67H : 832 Ko (banques 0 à C : E0_00000 à EC_00000) Série CJ CJ1H-CPU66H : 832 Ko (banques 0 à C : E0_00000 à EC_00000)	De la banque spécifiée dans la zone EM de la mémoire E/S à la dernière banque (spécifiée dans la configuration de l'API)	Commentaires des équations logiques et commentaires CX-Programmer Noms des sections, commentaires sur les sections, commentaires de programme de CX-Programmer. Tables de symboles globales, tables de symboles local, données de configuration de zone d'attribution automatique de CX-Programmer.	Activé lors du transfert de projets à l'aide de CX-Programmer ver. 5.0 ou supérieure. et pendant les opérations de sauvegarde simples.
Mémoires de commentaires	Mémoire flash interne de l'UC	Fichiers de commentaires : CPU66H/67H : 128 kilo-octets Autres UC : 64 kilo-octets Fichiers d'index de programmes : CPU66H/67H : 128 kilo-octets Autres UC : 64 kilo-octets Fichiers de tableaux de symboles : CPU45H/65H/66H/67H : 128 kilo-octets Autres UC : 64 kilo-octets	UC avec version 3.0 ou supérieure	Commentaires des équations logiques et commentaires CX-Programmer Noms des sections, commentaires sur les sections, commentaires de programme de CX-Programmer. Tables de symboles globales, tables de symboles local, données de configuration de zone d'attribution automatique de CX-Programmer.	Activé lors du transfert de projets à l'aide de CX-Programmer ver. 5.0 ou supérieure. et pendant les opérations de sauvegarde simples.

**Remarque**

1. Reportez-vous à la section 5-2 *Traitement des fichiers* pour plus d'informations sur l'installation et le retrait des cartes mémoire.
2. Initialisez la carte mémoire ou la mémoire de fichiers EM avant de l'utiliser pour la première fois. Reportez-vous à la section 5-3 *Utilisation de la mémoire de fichiers* pour plus d'informations sur l'initialisation.
3. L'adaptateur de carte mémoire HMC-AP001 permet d'installer une carte mémoire dans l'emplacement de cartes API d'un ordinateur pour l'utiliser comme un périphérique de stockage.
4. Lorsque CX-Programmer est utilisé, l'UC peut identifier les tableaux de symboles (y compris les commentaires d'E/S) et les commentaires. La destination du transfert est la carte mémoire, si une carte mémoire est installée, ou la mémoire de fichiers EM en l'absence d'une carte mémoire.

**Précautions relatives aux cartes mémoire**

Avant d'utiliser une carte mémoire, vérifiez les points suivants.

### Format

Les cartes mémoire sont formatées avant la livraison. Il n'est donc pas nécessaire de les formater après leur acquisition. Pour les formater une fois qu'elles ont été utilisées, utilisez CX-Programmer ou une console de programmation sur l'UC.

Si une carte mémoire est formatée directement dans un ordinateur portable ou un autre ordinateur, l'UC peut ne pas la reconnaître. Dans ce cas, vous ne serez pas en mesure de l'utiliser, même si elle est reformatée dans l'UC.

### Nombre de fichiers dans le répertoire racine

Le nombre de fichiers pouvant être placés dans le répertoire racine d'une carte mémoire est limité (de même qu'il existe une limite pour un disque dur). Bien que la limite dépende du type et du format de la carte mémoire, elle sera comprise entre 128 et 512 fichiers. Lorsque vous utilisez des applications qui génèrent régulièrement des fichiers-journaux ou d'autres fichiers, enregistrez ces fichiers dans un sous-répertoire plutôt que dans le répertoire racine.

Des sous-répertoires peuvent être créés sur un ordinateur ou à l'aide de l'instruction CMND(490). Reportez-vous à la section 3-25-5 *DELIVER COMMAND : CMND(490)* dans les *Référence des instructions, série CS/CJ* pour un exemple d'utilisation de l'instruction CMND(490).

### Nombre d'écritures

En règle générale, le nombre d'opérations d'écriture pouvant être exécutées pour une mémoire Flash n'est pas limité. En ce qui concerne les cartes mémoire, cependant, une limite de 100 000 opérations d'écriture a été définie pour des questions de garantie. Par exemple, si une écriture est effectuée toutes les 10 minutes sur la carte mémoire, plus de 100 000 opérations d'écriture auront lieu en 2 ans.

### Taille de fichier minimale

Si de nombreux fichiers de petite taille (tels ceux contenant uniquement quelques mots de données de zone DM) sont stockés sur la carte mémoire, il ne sera pas possible d'utiliser toute la capacité de cette carte. Par exemple, si vous utilisez une carte mémoire avec une taille d'unité d'allocation de 4096 octets, au moins 4096 octets de mémoire seront utilisés pour chaque fichier, quelle que soit sa taille. Si vous avez 10 mots de données de zone DM sur la carte mémoire, 4096 octets de mémoire seront utilisés, bien que la taille du fichier ne dépasse pas 68 octets. L'utilisation de fichiers de petite taille réduit considérablement le taux d'utilisation de la carte mémoire. Mais si vous réduisez la taille de l'unité d'allocation pour augmenter le taux d'utilisation, la vitesse d'accès sera également réduite.

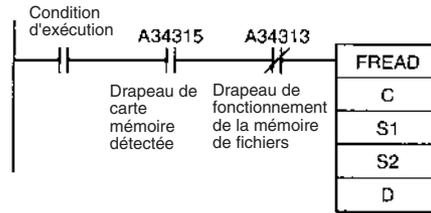
Vous pouvez contrôler la taille de l'unité d'allocation à partir d'une invite de commandes DOS avec CHKDSK. La procédure spécifique n'est pas décrite dans le présent document. Pour plus d'informations sur les tailles des unités d'allocation, reportez-vous aux références générales de l'ordinateur.

### Précautions pour l'accès aux cartes mémoire

Lorsque l'API accède à la carte mémoire, l'indicateur BUSY (occupé) s'allume sur l'UC. Prenez les précautions suivantes.

- 1,2,3... 1. Ne mettez pas l'UC hors tension lorsque le voyant BUSY est allumé. La carte mémoire pourrait devenir inutilisable.
2. Ne retirez jamais la carte mémoire de l'UC lorsque le témoin BUSY est allumé. Appuyez sur le bouton de mise hors tension de la carte mémoire et attendez que le témoin BUSY s'éteigne avant de la retirer. Si vous ne respectez pas cette règle, la carte risque de devenir inutilisable.
3. Insérer la carte mémoire, l'étiquette tournée vers la droite. N'essayez pas de l'insérer dans l'autre sens. Cela pourrait endommager la carte ou l'UC.

- Une fois la carte mémoire insérée, l'UC a besoin de quelques secondes pour l'identifier. Lorsque vous accédez à une carte mémoire juste après avoir mis l'alimentation sous tension ou lorsque vous insérez la carte mémoire, programmez une condition NC pour le drapeau de carte mémoire détectée (A34315) comme condition d'entrée, en suivant les indications ci-dessous.

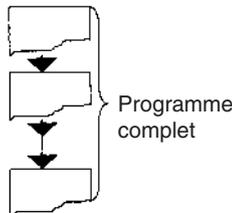


### 5-1-2 Données de fichiers

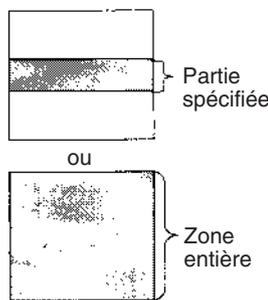
Le tableau suivant indique les types de fichiers qui peuvent être écrits.

Type de fichier	Opération
<ul style="list-style-type: none"> <li>Fichiers de programmes</li> <li>Fichiers de données</li> <li>Fichiers de paramètres</li> </ul>	Vous pouvez utiliser un périphérique de programmation (CX-Programmer ou console de programmation), des commandes FINS, des instructions de schéma contact, des opérations de sauvegarde simple ou des bits de contrôle spéciaux dans la mémoire de l'UC pour accéder à ces fichiers (en lecture, en écriture, etc.).
<ul style="list-style-type: none"> <li>Fichier de tables de symboles</li> <li>Fichiers de commentaires</li> <li>Fichiers d'index de programmes</li> </ul>	Vous pouvez utiliser le CX-Programmer et des opérations de sauvegarde simple pour accéder à ces fichiers (en lecture, en écriture, etc.).

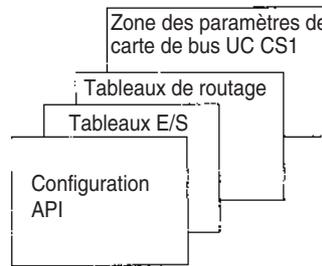
**Programme utilisateur :**  
fichier de programme  
Programme entier, y compris les attributs des tâches



**Plage spécifiée dans la mémoire E/S : fichiers de données**  
Plage entière ou partie spécifiée d'une zone de mémoire



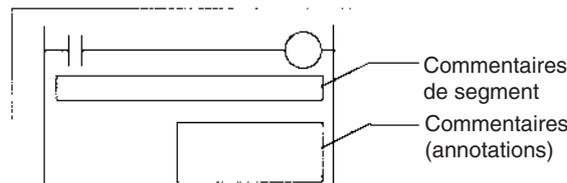
**Données de zone de paramètres : fichier de paramètres**  
Paramètres initiaux utilisés dans l'UC.



**Fichiers de tables de symboles**  
Tables de variables utilisées par le CX-Programmer

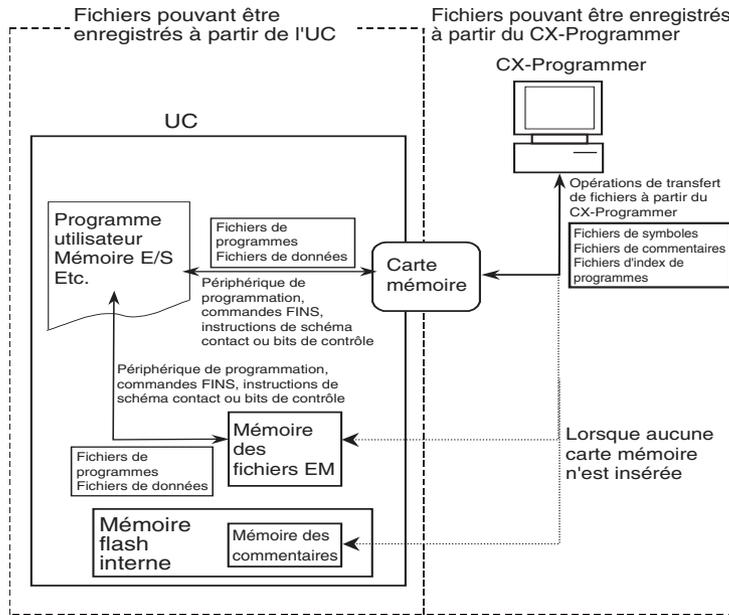
Symboles, adresses, types de données, commentaires d'E/S

**Fichiers de commentaires**  
Commentaires utilisés par le CX-Programmer



**Fichiers d'index de programmes**  
Informations sections (utilisées uniquement par CX-Programmer)

Noms des sections, commentaires sur les sections



**Remarque** Les tableaux de symboles (symboles, adresses et commentaire d'E/S) peuvent être traités comme des fichiers à partir du CX-Programmer.

Fichier	Nom de fichier	Extension	Contenu
Fichier de tableau de symboles	SYMBOLS	.SYM	Symboles généraux et locaux
Fichier de commentaires	COMMENTS	.CMT	Commentaires des équations logiques et commentaires (annotations)
Fichiers d'index de programmes	PROGRAM	.IDX	Noms des sections et commentaires sur les sections

Les opérations de transfert de fichiers peuvent être effectuées pour les projets à partir du CX-Programmer. Ce dernier permet de transférer tous les fichiers susmentionnés (fichiers de tableaux de symboles, fichiers de commentaires, fichiers d'index de programmes) entre l'UC et une carte mémoire ou entre EM et la mémoire de fichiers (les transferts de fichiers d'index de programmes sont pris en charge à partir de la version 2.0). Les fichiers de tableaux de symboles et les fichiers de commentaires peuvent également être transférés entre CX-Programmer, la RAM de l'ordinateur et un périphérique de stockage de données avec la version 1.2 de CX-Programmer ou une version supérieure.

### 5-1-3 Fichiers

Les fichiers sont au format DOS et peuvent donc être utilisés comme des fichiers normaux sur un ordinateur Windows.

Ils sont identifiés par des noms et des extensions de fichiers, comme indiqué dans le tableau suivant. Un nom de fichier peut comporter les caractères suivants : Lettres A à Z, chiffres 0 à 9, !, &, \$, #, \, {, }, -, ^, (, ) et \_

Vous pouvez utiliser les caractères suivants dans les noms de fichiers : ,, ,, /, ¥, ?, \*, ", :, ;, <, >, =, +, espace et les caractères sur 2 octets.

L'extension d'un nom de fichier dépend du type du fichier enregistré. Les fichiers de données ont l'extension IOM, TXT, CSV ou IOR. (extensions TXT, CSV et IOR : non prises en charge par les UC CS1 série CS antérieures à EV1.) Les fichiers de programmes ont l'extension OBJ et les fichiers de paramètres l'extension STD. L'emplacement d'un fichier en mémoire peut être spécifié dans le répertoire et un répertoire peut inclure jusqu'à 5 niveaux de sous-répertoires (en comptant le répertoire racine).

## Types, noms et extensions de fichiers

L'UC peut gérer trois types de fichiers (lecture et écriture).

- **Fichiers universels**

Ces fichiers sont accessibles (en lecture ou en écriture) avec des périphériques de programmation, des commandes FINS, des instructions ou des opérations de bit de contrôle de zone auxiliaire. Les noms de fichiers peuvent être définis librement par l'utilisateur.

- **Fichiers transférés automatiquement au démarrage**

Ces fichiers sont transférés automatiquement de la carte mémoire vers l'UC lors de la mise sous tension. Le nom du fichier est AUTOEXEC ou ATEXEC□□ lorsque vous transférez des fichiers de paramètres. Sinon, le nom du fichier est REPLACE (UC série CS/CJ Ver. 2.0 ou supérieure uniquement).

- **Fichiers de sauvegarde** (non pris en charge par les UC CS1 série CS antérieures à EV1)

Ces fichiers sont transférés entre la carte mémoire et l'UC par la fonction de sauvegarde. Les noms de fichiers ont un format fixe : BACKUP□□.

**Remarque** Utilisez la carte mémoire, la mémoire des fichiers EM ou la mémoire des commentaires pour les fichiers de tables de symboles, les fichiers de commentaires et les fichiers d'index de programmes.

- **Systèmes de fichiers du CX-Programmer**

Ces fichiers sont générés automatiquement lors d'un transfert effectué avec CX-Programmer. Les noms des fichiers sont fixes.

### Fichiers universels

Le tableau suivant présente les noms et les extensions des fichiers universels.

Type	Nom <sup>1</sup>	Extension	Description	Explication	
Fichier de programme	*****	.OBJ	Programme utilisateur complet	• Toutes les tâches cycliques et d'interruption, ainsi que les données des tâches pour une UC.	
Fichier de zone de paramètres	*****	.STD	Configuration de l'API, table d'E/S enregistrées, tables de routage, paramètres de carte réseau <sup>3</sup> , etc.	• Inclut tous les paramètres initiaux associés à une UC. • L'utilisateur n'est pas obligé d'établir la distinction des données des paramètres dans le fichier par type.	
Fichier de données	*****	.IOM	Plage spécifiée dans la mémoire E/S	<ul style="list-style-type: none"> <li>• Données du premier au dernier mot en unités de mots (16 bits) situées dans une zone.</li> <li>• La zone peut être l'une des zones suivantes : CIO, HR, WR, AR, DM ou EM.</li> </ul>	Format binaire
		.TXT			Format TXT <sup>2</sup> (non délimité ou délimité par des tabulations)
		.CSV			Format CSV <sup>2</sup> (délimité par des virgules)

**Remarque**

1. Les noms des fichiers, représentés plus haut par « \*\*\*\*\* », sont constitués de 8 caractères ASCII.
2. Formats de fichiers TXT et CSV : non pris en charge par les UC CS1 série CS antérieures à EV1.
3. Les tables de liaison de données offrent un exemple de paramètres de carte réseau. Pour les autres données de configuration, consultez les manuels d'utilisation des différentes cartes.

### Fichiers transférés automatiquement au démarrage

La colonne *Fichier* indique les fichiers qui doivent être présents dans la carte mémoire pour activer le transfert automatique au démarrage.

Il existe deux méthodes pour transférer automatiquement les fichiers au démarrage : transfert avec un fichier de zone de paramètres et transfert sans fichier de zone de paramètres.

## Transfert avec un fichier de zone de paramètres

Type	Nom <sup>1</sup>	Extension	Description	Explication	Fichier
Fichier de programme	AUTOEXEC	.OBJ	Programme utilisateur complet	<ul style="list-style-type: none"> <li>Le fichier ne doit pas obligatoirement se trouver sur la carte mémoire, même lorsque le transfert automatique au démarrage est spécifié.</li> <li>Tous les programmes de tâches cycliques et d'interruption, ainsi que les données des tâches pour une UC.</li> <li>Le transfert ne sera pas possible, sauf si la carte mémoire contient également un fichier de zone de paramètres (AUTOEXEC.STD).</li> </ul>	Obligatoire
Fichier de zone de paramètres	AUTOEXEC	.STD	Configuration de l'API, table d'E/S enregistrées, tables de routage, paramètres de carte réseau <sup>3</sup> , etc.	<p>Le fichier doit obligatoirement se trouver sur la carte mémoire lorsque le transfert automatique au démarrage est spécifié. Inclut tous les paramètres initiaux associés à une UC.</p> <p>L'utilisateur n'a pas à établir de distinction entre les données des paramètres dans le fichier par type.</p> <p>Les données des paramètres initiaux seront automatiquement sauvegardées à différents emplacements de l'UC au démarrage. Le fichier de zone de paramètres ne sera pas transféré si la carte mémoire contient un fichier de programme intitulé REPLACE.OBJ.</p>	Obligatoire
Fichiers de données	AUTOEXEC	.IOM	Données de mémoire E/S (Contient le nombre de mots de données spécifié à partir de D20000.)	<ul style="list-style-type: none"> <li>Sauvegardez les données DM à partir de D20000 dans un fichier nommé AUTOEXEC.IOM.</li> <li>Au démarrage, toutes les données du fichier seront transférées dans la zone DM à partir de D20000.</li> <li>Le fichier ne doit pas obligatoirement se trouver sur la carte mémoire lorsque la fonction de transfert automatique au démarrage est utilisée.</li> </ul>	---
	ATEXEC DM	.IOM	Données de mémoire E/S <sup>2</sup> (Contient le nombre de mots de données spécifié à partir de D00000.)	<ul style="list-style-type: none"> <li>Sauvegardez les données DM à partir de D00000 dans un fichier nommé ATEXEC DM.IOM.</li> <li>Au démarrage, toutes les données du fichier seront transférées dans la zone DM à partir de D00000.</li> <li>Le fichier ne doit pas obligatoirement se trouver sur la carte mémoire lorsque la fonction de transfert automatique au démarrage est utilisée.</li> </ul> <p><b>Remarque</b> Les données de ce fichier ont un niveau de priorité supérieur si elles recouvrent les données DM contenues dans AUTOEXEC.IOM.</p>	---
	ATEXECE□	.IOM	Données de zone EM (banque □) <sup>2</sup> (Contient le nombre de mots de données spécifié à partir de E□_00000.)	<ul style="list-style-type: none"> <li>Sauvegardez les données pour la banque EM □ à partir de E□_00000 dans un fichier nommé ATEXECE□.IOM. Le numéro de banque maximum dépend du modèle de l'UC utilisée.</li> <li>Au démarrage, toutes les données du fichier seront transférées dans la banque EM □ à partir de E□_00000.</li> <li>Le fichier ne doit pas obligatoirement se trouver sur la carte mémoire lorsque la fonction de transfert automatique au démarrage est utilisée.</li> </ul>	---

## Transfert sans fichier de zone de paramètres

Type	Nom <sup>1</sup>	Extension	Description	Explication	Fichier
Fichier de programme	REPLACE Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement.	.OBJ	Programme utilisateur complet	<ul style="list-style-type: none"> <li>Le contenu est identique à celui du fichier AUTOEXEC.OBJ.</li> <li>Ce fichier sera transféré au démarrage, même s'il n'existe pas de fichier de zone de paramètres (AUTOEXEC.STD).</li> </ul>	Obligatoire
Fichier de zone de paramètres	Non obligatoire.	---	---	Le fichier de zone de paramètres ne sera pas transféré, quel que soit le nom du fichier.	---
Fichier de données	REPLACE Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement.	.IOM	Données de mémoire E/S (Contient le nombre de mots de données spécifié à partir de D20000.)	<ul style="list-style-type: none"> <li>Le contenu est identique à celui du fichier AUTOEXEC.IOM</li> <li>Ce fichier sera transféré au démarrage si la carte mémoire contient également un fichier de programme nommé REPLACE.OBJ.</li> </ul>	---
	REPLCDM Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement.	.IOM	Données de mémoire E/S (Contient le nombre de mots de données spécifié à partir de D00000.)	<ul style="list-style-type: none"> <li>Le contenu est identique à celui du fichier ATEXECDM.IOM.</li> <li>Ce fichier sera transféré au démarrage si la carte mémoire contient également un fichier de programme nommé REPLACE.OBJ.</li> </ul>	---
	REPLCE□ Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement.	.IOM	Données de zone EM (banque □) (Contient le nombre de mots de données spécifié à partir de E□_00000.)	<ul style="list-style-type: none"> <li>Le contenu est identique à celui du fichier ATEXECE□.IOM.</li> <li>Ce fichier sera transféré au démarrage si la carte mémoire contient également un fichier de programme nommé REPLACE.OBJ.</li> </ul>	---

**Remarque**

- Assurez-vous que les noms des fichiers à transférer automatiquement au démarrage sont AUTOEXEC ou ATEXEC□□.
- Les fichiers ATEXECDM.IOM et ATEXECE□.IOM : non pris en charge par les UC CS1 série CS antérieures à EV1.
- Les tables de liaison de données offrent un exemple de paramètres de carte réseau. Pour les autres données de configuration, consultez les manuels d'utilisation des différentes cartes.

**Fichiers de sauvegarde  
(non pris en charge par  
les UC CS1 série CS  
antérieures à EV1)**

Les fichiers dans le tableau suivant sont créés automatiquement lorsque les données sont transférées vers et depuis la carte mémoire pendant l'opération de sauvegarde.

Type	Nom <sup>1</sup>	Extension	Description	Explication
Fichier de données	BACKUP	.IOM	Mots de zone DM affectés aux cartes d'E/S spéciales, cartes réseau et cartes internes (série CS uniquement)	<ul style="list-style-type: none"> <li>Contient les données DM de D20000 à D32767.</li> <li>Ce fichier doit être présent sur la carte mémoire lorsque les données sont lues sur la carte pendant la sauvegarde.</li> </ul>
	BACKUIO	.IOR	Zones de données de mémoire E/S	<ul style="list-style-type: none"> <li>Contient toutes les données des zones de données CIO, WR, HR et AR, ainsi que les drapeaux de fin de temporisation/compteur et les valeurs actuelles.<sup>2</sup></li> <li>Ce fichier doit être présent sur la carte mémoire lorsque les données sont lues sur la carte pendant la sauvegarde.</li> </ul>
	BACKUPDM	.IOM	Zone universelle DM	<ul style="list-style-type: none"> <li>Contient les données DM de D00000 à D19999.</li> <li>Ce fichier doit être présent sur la carte mémoire lorsque les données sont lues sur la carte pendant la sauvegarde.</li> </ul>
	BACKUPE□	.IOM	Zone universelle EM	<p>Contient toutes les données EM pour la banque EM □ avec des adresses comprises entre E□_00000 et E□_32767. (Le numéro de banque maximum dépend du modèle de l'UC utilisée.)</p> <p>Ce fichier doit être présent sur la carte mémoire lorsque les données sont lues sur la carte pendant la sauvegarde.</p> <ul style="list-style-type: none"> <li>Lorsque les données sont sauvegardées sur la carte mémoire, toutes les données de chaque banque EM sont automatiquement enregistrées dans un fichier distinct.</li> </ul>
Fichier de programme	BACKUP	.OBJ	Programme utilisateur complet	<ul style="list-style-type: none"> <li>Contient tous les programmes de tâches cycliques et d'interruption, ainsi que les données des tâches pour une UC.</li> <li>Ce fichier doit être présent sur la carte mémoire lorsque les données sont lues sur la carte pendant la sauvegarde.</li> </ul>
Fichier de paramètres		.STD	Configuration de l'API, table d'E/S enregistrées, tables de routage, paramètres de carte réseau <sup>3</sup> , etc.	<ul style="list-style-type: none"> <li>Inclut tous les paramètres initiaux associés à une UC.</li> <li>L'utilisateur n'a pas à établir de distinction entre les données des paramètres dans le fichier par type.</li> <li>Ce fichier doit être présent sur la carte mémoire lorsque les données sont lues sur la carte pendant la sauvegarde.</li> </ul>
Fichiers de sauvegarde unité/carte (UC CS1-H, CJ1-H ou CJ1M uniquement)	BACKUP□□ (où □□ désigne l'adresse d'unité de l'unité/carte sauvegardée)	.PRM	Données pour une unité ou une carte spécifique	<ul style="list-style-type: none"> <li>Contrôlez les données de sauvegarde à partir d'une unité ou d'une carte. Voir la section 5-2-6 <i>Fonction de sauvegarde simple</i> pour plus d'informations.</li> </ul>
Fichiers de tables de symboles (Voir remarque 1.)	BKUPSYM	.SYM	Tables de symboles globales, tables de symboles locales et données de configuration de zone affectées automatiquement	Inclut les données suivantes dans la table de symboles globale/locale du CX-Programmer : variables, adresses, types de données, commentaires d'E/S Inclut les données définies dans les paramètres d'adressage automatique des API du CX-Programmer.
Fichiers de commentaires (Voir remarque 1.)	BKUPCMT	.CMT	Commentaires des équations logiques et commentaires	Commentaires des équations logiques et commentaires du CX-Programmer.
Fichiers d'index de programmes (Voir remarque 1.)	BKUPPRG	.IDX	Noms de sections, commentaires de sections et commentaires de programmes du CX-Programmer.	Données des séparateurs de sections du CX-Programmer (l'emplacement du séparateur dépend toutefois du séparateur de sections dans le programme.)

- Remarque**
1. Les fichiers de sauvegarde suivants peuvent être créés uniquement lorsque vous utilisez des UC série CS/CJ avec la version de carte 3.0 ou supérieure. Fichiers de tables de symboles, fichiers de commentaires et fichiers d'index de programmes  
Ces fichiers sont créés automatiquement à partir des fichiers de la carte mémoire, de la mémoire des fichiers EM ou de la mémoire des commentaires.
  2. Les tables de liaison de données sont un exemple de paramètres de carte de bus UC. Pour les autres données de configuration, consultez les manuels d'utilisation des différentes cartes.

**Fichiers système du CX-Programmer**

Ces fichiers sont générés automatiquement lorsque des données sont téléchargées avec la version 5.0 ou une version supérieure du CX-Programmer. Les noms des fichiers sont fixes.

Lorsque vous utilisez la version 5.0 ou une version supérieure du CX-Programmer avec une version de carte 3.0 ou supérieure, vous pouvez sélectionner l'une des options de mémoire suivantes comme destination pour ces fichiers système lors du transfert des projets.

- Carte mémoire
- Mémoire des fichiers EM
- Mémoire des commentaires (dans la mémoire flash de l'UC)

**Remarque** Avec la version 4.0 ou une version antérieure du CX-Programmer, ces fichiers ne peuvent pas être stockés dans la mémoire des commentaires, même si une UC avec la version de carte 3.0 ou supérieure est utilisée.

Type	Nom	Extension	Description	Explication	
Fichiers de tables de symboles	SYMBOLS	.SYM	Tables de symboles globales et tables de symboles locales	Ces fichiers sont créés automatiquement lorsque des données sont téléchargées avec CX-Programmer.	Inclut les données suivantes dans la table de symboles globale/locale du CX-Programmer : Variables, adresses, types de données, commentaires d'E/S Inclut les données définies dans les paramètres d'adressage automatique des API du CX-Programmer.
Fichiers de commentaires	COMMENTS	.CMT	Commentaires des équations logiques et commentaires (annotations)		Commentaires des équations logiques et commentaires du CX-Programmer.
Fichiers d'index de programmes	PROGRAM	.IDX	Noms des sections, commentaires sur les sections		Données des séparateurs de sections du CX-Programmer (l'emplacement du séparateur dépend toutefois du séparateur de sections dans le programme.) Remarque : CX-Programmer version 2.0 ou supérieure uniquement.

**Remarque** Avec la version 1.2 ou une version supérieure du CX-Programmer, les fichiers des tables de symboles et les fichiers de commentaires du tableau précédent peuvent être transférés en ligne entre le CX-Programmer et la RAM du PC, et entre la RAM du PC et le périphérique de stockage.

## Répertoires

Les API série CS/CJ permettent d'accéder aux fichiers situés dans des sous-répertoires, mais les Consoles de programmation n'ont accès qu'aux fichiers présents du répertoire racine. Le chemin d'accès d'un répertoire comporte au maximum 65 caractères. Veillez à ne pas dépasser le nombre maximum de caractères lorsque vous créez des sous-répertoires sur la carte mémoire avec un programme tel que Windows.

## Tailles des fichiers

Les équations du tableau suivant permettent de calculer la taille des fichiers en octets.

Type de fichier	Taille de fichier
Fichiers de données (.IOM)	(Nombre de mots × 2) + 48 octets Exemple : Zone DM entière (D00000 à D32767) (32 768 mots × 2) + 48 = 65 584 octets
Fichiers de données (.TXT ou .CSV)	La taille de fichier dépend du nombre de séparateurs et de retours chariot utilisés. Le code d'un séparateur occupe un octet et le code d'un retour chariot deux octets. Exemple 1 : Mots non délimités, sans retour chariot 123456789ABCDEF012345678 occupe 24 octets. Exemple 2 : Mots délimités, retour chariot tous les 2 champs 1234,5678.␣ 9ABC,DEF0.␣ 1234,5678.␣ occupe 33 octets. Exemple 3 : Mots doubles délimités, retour chariot tous les 2 champs 56781234,DEF01234.␣ 56781234.␣ occupe 29 octets.
Fichiers de programme (.OBJ)	(Nombre de pas utilisés × 4) + 48 octets (Voir remarque.)
Fichiers de paramètres (.STD)	16 048 octets

### Remarque

Pour calculer le nombre de pas dans le fichier de programme, soustrayez les pas UM disponibles du nombre total de pas UM. Ces valeurs sont indiquées dans le rapport de référence croisée du CX-Programmer. Consultez le *Manuel d'utilisation du CX-Programmer* pour plus de détails.

## Fichiers de données

### Fichiers universels

#### 1,2,3...

1. Les fichiers de données universels portent les extensions IOM, TXT ou CSV. (Fichiers TXT et CSV : non pris en charge par les UC CS1 série CS antérieures à EV1.)

Extension	Format de données	Contenu		Mots/ champ
.IOM	Binaire	Format de données série CS/CJ		---
.TXT (Voir remarques.)	Mots non délimités	Format ASCII	Vous obtenez ce format en convertissant les champs de un mot de la mémoire E/S (4 chiffres hexadécimaux) en ASCII et en groupant les champs sans séparateurs. Les enregistrements peuvent être délimités par des retours chariot.	1 mot
	Mots doubles non délimités		Vous obtenez ce format en convertissant les champs de deux mots de la mémoire E/S (8 chiffres hexadécimaux) en ASCII et en groupant les champs sans séparateurs. Les enregistrements peuvent être délimités par des retours chariot.	2 mots
	Mots délimités par des tabulations		Vous obtenez ce format en convertissant les champs de un mot de la mémoire E/S (4 chiffres hexadécimaux) en ASCII et en délimitant les champs par des tabulations. Les enregistrements peuvent être séparés par des retours chariot.	1 mot
	Mots délimités par des tabulations		Vous obtenez ce format en convertissant les champs de deux mots de la mémoire E/S (8 chiffres hexadécimaux) en ASCII et en délimitant les champs par des tabulations. Les enregistrements peuvent être séparés par des retours chariot.	2 mots
.CSV (Voir remarques.)	Mots délimités par des virgules		Vous obtenez ce format en convertissant les champs de un mot de la mémoire E/S (4 chiffres hexadécimaux) en ASCII et en délimitant les champs par des virgules. Les enregistrements peuvent être séparés par des retours chariot.	1 mot
	Mots doubles délimités par des virgules		Vous obtenez ce format en convertissant les champs de deux mots de la mémoire E/S (8 chiffres hexadécimaux) en ASCII et en délimitant les champs par des virgules. Les enregistrements peuvent être séparés par des retours chariot.	2 mots

- Remarque a)** Lecture et écriture des fichiers de données TXT et CSV :  
Les fichiers de données TXT et CSV ne peuvent être lus et écrits qu'avec FREAD(700) et FWRT(701).
- b)** Précautions relatives aux caractères :  
Les données ne peuvent pas être enregistrées correctement dans la mémoire E/S si le fichier TXT ou CSV contient des caractères autres que des caractères hexadécimaux (0 à 9, A à F ou a à f.)
- c)** Précautions relatives à la taille des fichiers :  
Lorsque des mots sont utilisés, les données ne peuvent pas être enregistrées correctement dans la mémoire E/S si le fichier TXT ou CSV contient des champs autres que des champs à 4 chiffres hexadécimaux. De même, avec des mots doubles, les données ne peuvent pas être enregistrées correctement si le fichier contient des champs autres que des champs à 8 chiffres hexadécimaux.
- d)** Ordre de stockage :  
Lorsque des mots sont utilisés, les données de la mémoire E/S sont converties en ASCII et stockées dans des champs de un mot, en allant de l'adresse mémoire E/S la plus basse vers l'adresse la plus haute.  
Lorsque des mots doubles sont utilisés, les données de la mémoire E/S sont converties en ASCII et stockées dans des champs de deux mots, en allant de l'adresse mémoire E/S la plus basse vers l'adresse la plus haute. (Dans les champs de deux mots, le mot avec l'adresse la plus haute est enregistré en premier et le mot avec l'adresse la plus basse en second.)
- e)** Séparateurs :  
En l'absence de séparateurs, les champs sont regroupés les uns à la suite des autres, puis enregistrés. Lorsque les champs sont délimités par des virgules, ces dernières sont insérées entre les champs avant l'enregistrement. Lorsque les champs sont délimités par des tabulations, les codes de tabulation sont insérés entre

les champs avant l'enregistrement. Lorsque des séparateurs (virgules ou tabulations) sont spécifiés dans FREAD(700), les données sont lues comme des données délimitées avec des séparateurs de mots uniques (virgules ou tabulations).

**f) Retours chariot :**

Les données sont regroupées les unes à la suite des autres lorsque les retours chariot ne sont pas utilisés.

Lorsqu'ils sont utilisés, un code de retour chariot est inséré après le nombre de champs spécifié. Il n'est pas possible de spécifier un décalage par rapport au début du fichier (mot de début de lecture ou mot de début d'écriture) dans les instructions FREAD(700)/FWRITE(701) si le fichier contient des retours chariot.

**g) Nombre de champs :**

Le volume total de données dans le champ dépend du nombre de champs (nombre d'éléments d'écriture) spécifiés dans les instructions FWRITE(701) et du nombre de mots/champ. Il y a un mot/champ lorsque des mots sont utilisés et deux mots/champ lorsque des mots doubles sont utilisés.

- Les fichiers de données ne contiennent pas d'informations indiquant quelles sont les données mémorisées, c'est-à-dire quelle est la zone de mémoire enregistrée. Veuillez par conséquent attribuer aux fichiers des noms qui illustrent leur contenu, comme indiqué dans les exemples suivants, afin de faciliter la gestion des fichiers.

**Exemples :** D00100.IOM, CIO0020.IOM

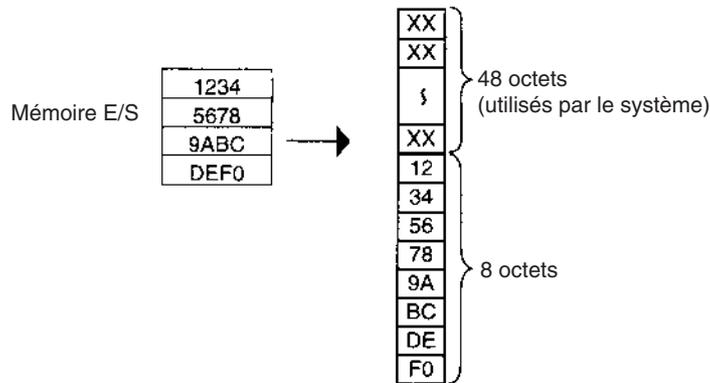
Les données à partir du début du fichier seront écrites à partir de l'adresse spécifiée dans la mémoire E/S, même si les données initialement écrites dans le fichier de données (IOM, TXT ou CSV) ne proviennent pas de la même zone. Par exemple, si les données CIO d'un fichier sont écrites dans la zone DM à partir d'un périphérique de programmation, les données seront lues dans la zone DM de l'UC sans aucune indication que la zone est différente.

**Remarque**

Les fichiers de données aux formats TXT et CSV contiennent des données hexadécimales (0 à 9, A à F) qui permettent d'échanger les données numériques de la mémoire E/S avec des tableurs.

**Structure des fichiers de données IOM**

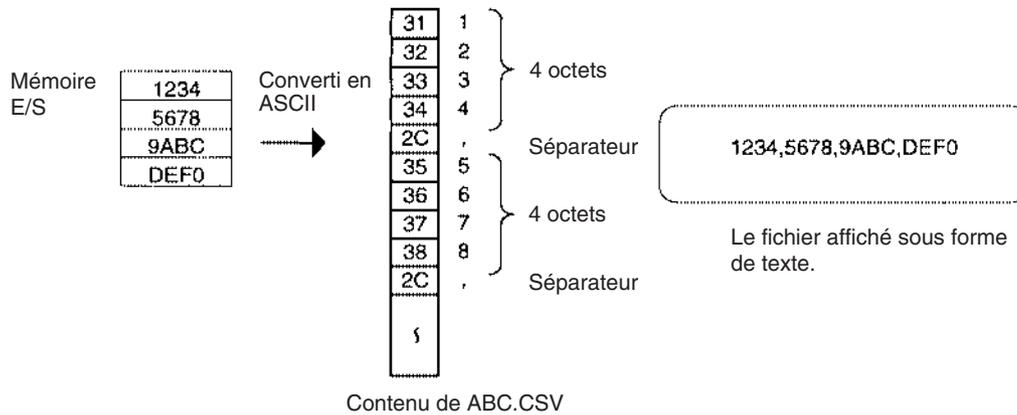
L'illustration suivante présente la structure des données binaires d'un fichier de données (ABC.IOM) contenant quatre mots de la mémoire E/S : 1234 hex., 5678 hex., 9ABC hex. et DEF0 hex. L'utilisateur, cependant, n'a pas à se soucier du format des données dans les opérations normales.



Contenu de ABC.IOM

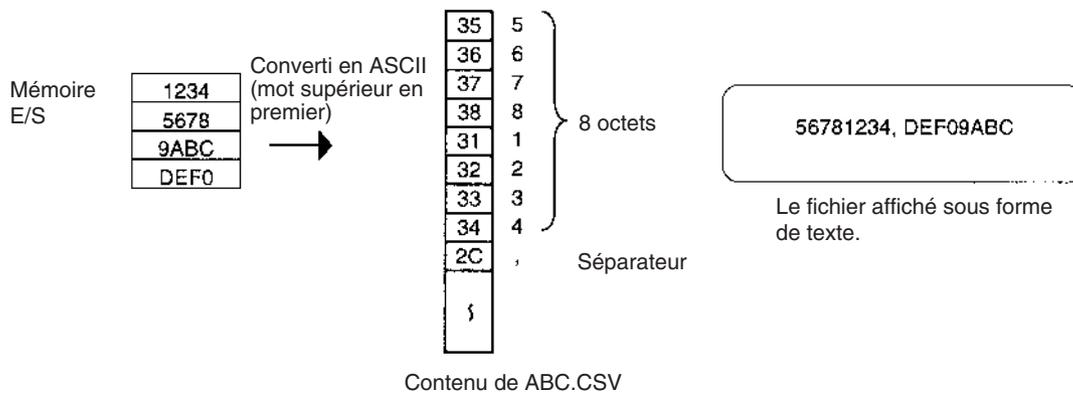
**Structure des fichiers de données CSV/TXT (mot unique)**

L'illustration suivante présente la structure des données binaires d'un fichier de données CSV (ABC.CSV) avec des champs à mot unique contenant quatre mots de la mémoire E/S : 1234 hex., 5678 hex., 9ABC hex. et DEF0 hex. Le fichier TXT avec des champs à mot unique a la même structure.



**Structure des fichiers de données CSV/TXT (double mot)**

L'illustration suivante présente la structure des données d'un fichier de données CSV (ABC.CSV) avec des champs à double mot contenant quatre mots de la mémoire E/S : 1234 hex., 5678 hex., 9ABC hex. et DEF0 hex. Le fichier TXT avec des champs à double mot a la même structure.



**Création de fichiers de données avec un logiciel de tableur**

Utilisez la procédure suivante pour créer des fichiers de données TXT et CSV avec un logiciel de tableur tel que Microsoft Excel.

- Configurez le contenu des cellules sur caractères.
- Entrez respectivement 4 ou 8 caractères dans chaque cellule selon que vous utilisez des champs à mot unique ou à double mot. Par exemple, si des champs à mot unique sont utilisés, entrez 000A, et non uniquement A.
- Veillez à entrer uniquement des caractères hexadécimaux (0 à 9, A à F ou a à f) dans les cellules. Vous ne pouvez pas utiliser d'autres caractères ou codes.

Lorsque vous voulez stocker des chiffres hexadécimaux dans la mémoire E/S, il est utile de convertir les entrées décimales du tableur en valeurs hexadécimales. Pour effectuer cette conversion, procédez comme suit.

**1,2,3...**

1. Sélectionnez **Add-Ins...** dans le menu Tools.
2. Sélectionnez **Analysis ToolPak** dans le menu Add-Ins.
3. Sélectionnez **Function** dans le menu Insert Menu au niveau de la cellule où la fonction sera utilisée.
4. Sélectionnez **DEC2HEX (number, digits)** pour Engineering dans Category Field.
5. Lors d'une conversion en valeur hexadécimale à 4 chiffres, entrez les informations suivantes pour la variable number : IF(0<=emplacement de cellule, emplacement de cellule, 65 535+emplacement de cellule)

Lors d'une conversion en valeur hexadécimale à 8 chiffres, entrez les informations suivantes pour la variable number : IF(0<=emplacement de cellule,emplacement de cellule, 4 294 967 296+emplacement de cellule)

• **Exemple 1** : Entrée de valeurs décimales non négatives.

Elément	Conversion de valeurs décimales non signées en valeurs hexadécimales à 4 chiffres	Conversion de valeurs décimales non signées en valeurs hexadécimales à 8 chiffres																																																
Fonction utilisée	DEC2HEX( <i>emplacement_cellule</i> ,4)	DEC2HEX( <i>emplacement_cellule</i> ,8)																																																
Exemple	<p>La valeur décimale 10 est convertie en valeur hexadécimale à 4 chiffres : 000A.</p> <table border="1"> <thead> <tr> <th colspan="2">B2</th> <th colspan="2">=DEC2HEX(B1,4)</th> </tr> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Non-negative source decimal:</td> <td>10</td> <td></td> </tr> <tr> <td>2</td> <td>Converted 4-digit hexadecimal:</td> <td>000A</td> <td></td> </tr> <tr> <td>3</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	B2		=DEC2HEX(B1,4)			A	B	C	1	Non-negative source decimal:	10		2	Converted 4-digit hexadecimal:	000A		3				4				<p>La valeur décimale 10 est convertie en valeur hexadécimale à 8 chiffres : 0000000A.</p> <table border="1"> <thead> <tr> <th colspan="2">B2</th> <th colspan="2">=DEC2HEX(B1,8)</th> </tr> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Non-negative source decimal:</td> <td>10</td> <td></td> </tr> <tr> <td>2</td> <td>Converted 8-digit hexadecimal:</td> <td>0000000A</td> <td></td> </tr> <tr> <td>3</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	B2		=DEC2HEX(B1,8)			A	B	C	1	Non-negative source decimal:	10		2	Converted 8-digit hexadecimal:	0000000A		3				4			
B2		=DEC2HEX(B1,4)																																																
	A	B	C																																															
1	Non-negative source decimal:	10																																																
2	Converted 4-digit hexadecimal:	000A																																																
3																																																		
4																																																		
B2		=DEC2HEX(B1,8)																																																
	A	B	C																																															
1	Non-negative source decimal:	10																																																
2	Converted 8-digit hexadecimal:	0000000A																																																
3																																																		
4																																																		

• **Exemple 2** : Entrée de valeurs décimales signées.

Elément	Conversion de valeurs décimales signées en valeurs hexadécimales à 4 chiffres	Conversion de valeurs décimales signées en valeurs hexadécimales à 8 chiffres																																																
Fonction utilisée	DEC2HEX(IF(0<= <i>emplacement_cellule</i> , <i>emplacement_cellule</i> ,65536+ <i>emplacement_cellule</i> ),4)	DEC2HEX(IF(0<= <i>emplacement_cellule</i> , <i>emplacement_cellule</i> ,4 294 967 296+ <i>emplacement_cellule</i> ),8)																																																
Exemple	<p>La valeur décimale -10 est convertie en valeur hexadécimale à 4 chiffres : FFF6.</p> <table border="1"> <thead> <tr> <th colspan="2">B2</th> <th colspan="2">=&lt;DEC2HEX(IF(0&lt;B1,B1,65536+B1),4)</th> </tr> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Signed source decimal:</td> <td>-10</td> <td></td> </tr> <tr> <td>2</td> <td>Converted 4-digit hexadecimal:</td> <td>FFF6</td> <td></td> </tr> <tr> <td>3</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	B2		=<DEC2HEX(IF(0<B1,B1,65536+B1),4)			A	B	C	1	Signed source decimal:	-10		2	Converted 4-digit hexadecimal:	FFF6		3				4				<p>La valeur décimale -10 est convertie en valeur hexadécimale à 8 chiffres : FFFFFFF6.</p> <table border="1"> <thead> <tr> <th colspan="2">B2</th> <th colspan="2">=&lt;DEC2HEX(IF(0&lt;B1,B1,4294967296+B1),8)</th> </tr> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Signed source decimal:</td> <td>-10</td> <td></td> </tr> <tr> <td>2</td> <td>Converted 8-digit hexadecimal:</td> <td>FFFFFFF6</td> <td></td> </tr> <tr> <td>3</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	B2		=<DEC2HEX(IF(0<B1,B1,4294967296+B1),8)			A	B	C	1	Signed source decimal:	-10		2	Converted 8-digit hexadecimal:	FFFFFFF6		3				4			
B2		=<DEC2HEX(IF(0<B1,B1,65536+B1),4)																																																
	A	B	C																																															
1	Signed source decimal:	-10																																																
2	Converted 4-digit hexadecimal:	FFF6																																																
3																																																		
4																																																		
B2		=<DEC2HEX(IF(0<B1,B1,4294967296+B1),8)																																																
	A	B	C																																															
1	Signed source decimal:	-10																																																
2	Converted 8-digit hexadecimal:	FFFFFFF6																																																
3																																																		
4																																																		

**Fichiers de données transférés automatiquement au démarrage**

Trois types de fichiers sont transférés automatiquement au démarrage lorsque la fonction de transfert automatique au démarrage est utilisée.

- AUTOEXEC.IOM ou REPLACE.IOM : mots DM affectés aux cartes d'E/S spéciales et aux cartes internes.  
Le contenu de ce fichier est transféré dans la zone DM à partir de D20000 lors de la mise sous tension.
- ATEXECDM.IOM ou REPLCDM.IOM : Mots universels DM  
Le contenu de ce fichier est transféré dans la zone DM à partir de D20000 lors de la mise sous tension.
- ATEXECE□.IOM ou REPLCE□.IOM : Mots universels EM  
Le contenu de ce fichier est transféré dans la zone EM à partir de E□\_00000 lors de la mise sous tension.

Lorsque vous créez les fichiers de données répertoriés plus haut, spécifiez toujours la première adresse indiquée ci-dessus (D20000, D00000 ou E□\_00000) et assurez-vous que la taille du fichier ne dépasse pas la capacité de la zone de données spécifiée.

L'ensemble des données de chaque fichier seront toujours transférées à partir de la première adresse spécifiée (D20000, D00000 ou E□\_00000).

**Remarque**

1. Lorsque vous créez les fichiers AUTOEXEC.IOM, ATEXECDM.IOM et ATEXECE□.IOM ou les fichiers REPLACE.IOM, REPLCDM.IOM ou REPLCE□.IOM à partir d'un périphérique de programmation (console de programmation ou CX-Programmer), spécifiez toujours la première adresse correcte (D20000, D00000 ou E□\_00000) et assurez-vous que la taille du fichier ne dépasse pas la capacité de la zone DM ou de la banque EM spécifiée. Le contenu du fichier sera toujours transféré à partir de la première adresse appropriée (D20000, D00000 ou E□\_00000), même si un autre mot de début est spécifié, ce qui pourrait entraîner le recouvrement du contenu de cette partie de la zone DM ou de la banque EM par des données incorrectes. De plus, si la capacité de la zone DM ou de la banque EM est dépassée (comme cela est possible lorsque vous

définissez les paramètres à partir du CX-Programmer), les données restantes seront écrites dans la banque EM 0 (si la zone DM est saturée) ou la banque suivante (si une banque Em est saturée).

2. Lorsque vous utilisez CX-Programmer, vous pouvez spécifier un fichier de données qui dépassera l'adresse maximale de la zone DM (D32767) ou l'adresse maximale de la zone EM (E□\_32767). Si le fichier AUTOEXEC.IOM dépasse les limites de la zone DM, les données restantes seront écrites dans la zone EM, à partir de E0\_00000 et jusqu'à la banque finale, en suivant l'ordre des adresses mémoire et des banques. Il est ainsi possible de transférer automatiquement des données vers les zones DM et EM au démarrage. De même, si le fichier ATEXECE□.IOM est plus grand qu'une banque EM, les données restantes seront écrites dans les banques EM suivantes.
3. Vous pouvez modifier les configurations système pour les cartes d'E/S spéciales, les cartes réseau et la carte interne (série CS uniquement) en utilisant d'autres fichiers AUTOEXEC.IOM, contenant des paramètres différents pour la zone de carte d'E/S spéciales (D20000 à D29599), la zone de carte réseau (D30000 à D31599) et la zone de carte interne (série CS uniquement, D32000 à D32099). Les cartes mémoire permettent ainsi de créer des bibliothèques de données de configuration du système destinées aux cartes d'E/S spéciales, aux cartes réseau et aux cartes internes (série CS uniquement) pour des systèmes ou des périphériques différents.

#### Fichiers de données de sauvegarde

La fonction de sauvegarde crée 4 types de fichiers de données, comme indiqué ci-après.

Pour sauvegarder les données, mettez la broche 7 de l'interrupteur DIP de l'UC sur ON et la broche 8 sur OFF, insérez la carte mémoire, puis maintenez l'interrupteur d'alimentation de la carte mémoire enfoncé pendant trois secondes. Les quatre fichiers de sauvegarde (BACKUP.IOM, BACKUPIO.IOR, BACKUPDM.IOM et BACKUPE□.IOM) seront créés automatiquement et écrits sur la carte mémoire.

Les quatre fichiers de sauvegarde sont utilisés exclusivement par la fonction de sauvegarde, bien que trois de ces fichiers (BACKUP.IOM, BACKUPDM.IOM et BACKUPE□.IOM) puissent être créés avec des opérations du périphérique de programmation (BACKUPIO.IOR ne peut pas être créé de cette façon).

#### 5-1-4 Description des procédures d'utilisation des fichiers

Le tableau suivant résume les 6 méthodes que vous pouvez utiliser pour lire et écrire des fichiers.

Lecture : transfère les fichiers de la mémoire de fichiers vers l'UC.

Ecriture : transfère les fichiers de l'UC vers la mémoire de fichiers.

OK : Possible; ---: Impossible

Procédure d'utilisation		Support	Nom de fichier	Description	Programme complet	Données de zone de données (Voir remarque 3.)	Données de zone de paramètres	Tables de symboles, fichiers de commentaires, fichiers d'index de programmes (Voir remarque 6.)
Périphérique de programmation	CX-Programmer ou console de programmation	Carte mémoire Mémoire des fichiers EM	Tout nom de fichier correct	Lecture	OK	OK	OK	---
				Ecriture	OK	OK	OK	---
				Autres opérations (Voir remarque 2.)	OK	OK	OK	---
Périphérique de programmation	CX-Programmer uniquement	Carte mémoire Mémoire des fichiers EM, mémoire des commentaires	Système de fichiers du CX-Programmer (nom de fichier fixe)	Lecture	---	---	---	OK
				Ecriture	---	---	---	OK (Voir remarque 6.)
				Autres opérations (Voir remarque 2.)	---	---	---	---
Commande FINS (Voir remarque 1.)		Carte mémoire Mémoire des fichiers EM	Tout nom de fichier correct	Lecture	OK	OK	OK	---
				Ecriture	OK	OK	OK	---
				Autres opérations (Voir remarque 2.)	OK (Voir remarque 4.)	OK	OK	---
Instructions FREAD(700) et FWRIT(701)		Carte mémoire Mémoire des fichiers EM	Tout nom de fichier correct	Lecture des données d'un fichier.	---	OK	---	---
				Ecriture de données dans un fichier.	---	OK	---	---
L'opération de bit de contrôle de zone auxiliaire remplace tout le programme en cours d'opération (non pris en charge par les UC CS1 série CS antérieures à EV1).		Carte de mémoire	Tout nom de fichier correct	Lecture	OK	---	---	---
Transfert automatique au démarrage		Carte mémoire	Transfert automatique avec les fichiers de démarrage (AUTOEXEC, ATEXC□□ ou REPLACE (Voir remarque 5.))	Lecture	OK	OK	OK	---
				Ecriture	---	---	---	---
Opération de sauvegarde (non prise en charge par les UC CS1 série CS antérieures à EV1).		Carte mémoire	Fichiers de sauvegarde simple (BACKUP□□)	Lecture	OK	OK	OK	OK
				Ecriture	OK	OK	OK	OK (Voir remarque 6.)

- Remarque**
1. Les commandes FINS pour les opérations sur la mémoire de fichiers peuvent être envoyées depuis des ordinateurs connectés via une liaison Host Link, une autre API connectée à un réseau (avec CMND(490)) ou le programme de l'API locale (avec CMND(490)). (Pour les UC CS1 série CS antérieures à EV1, les opérations sur la mémoire de fichiers ne peuvent pas être exécutées avec CMND(490) sur la même UC que celle pour laquelle les opérations sur la mémoire de fichiers sont exécutées avec d'autres instructions.
  2. Autres opérations : formatage de la mémoire de fichiers, lecture de données de fichiers, écriture de données de fichiers, modification de noms de fichiers, lecture de données de la mémoire de fichiers, suppression de fichier, copie de fichier, création de sous-répertoire et modification de nom de fichier.
  3. Les fichiers de données aux formats TXT et CSV ne peuvent être lus et écrits qu'avec les instructions FREAD(700) et FWRIT(701). La lecture et l'écriture de ces fichiers ne sont pas possibles avec un périphérique de programmation.

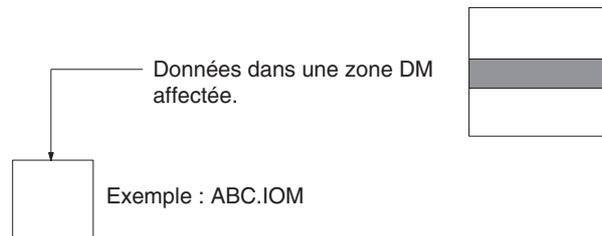
4. Les versions V1.2 et supérieures du CX-Programmer peuvent être utilisées pour transférer des fichiers de programmes (.OBJ) entre la RAM de l'ordinateur et un périphérique de stockage.
5. Avec une UC série CS/CJ Ver. 2.0, les fichiers peuvent être transférés automatiquement vers l'UC au démarrage sans qu'un fichier de paramètres ne soit stocké sur la carte mémoire. Pour ce faire, remplacez le nom du fichier de programme par REPLACE.OBJ. Il est également possible de transférer des fichiers de données avec REPLACE.OBJ en utilisant les noms de fichiers suivants : REPLACE.IOM, REPLCDM.IOM et REPLCE□.IOM.
6. Lorsque vous transférez des projets vers une UC série CS/CJ avec une version de carte 3.0 ou supérieure à partir d'un CX-Programmer version 5.0 ou supérieure, les tables de symboles, les fichiers de commentaires et les fichiers d'index de programmes peuvent être stockés à l'intérieur de la mémoire des commentaires dans la mémoire flash interne de l'UC (uniquement en l'absence de carte mémoire ou de mémoire de fichiers EM, ou si aucune zone n'est disponible).  
Les fichiers de sauvegarde des tables de symboles, des fichiers de commentaires et des fichiers d'index de programmes qui sont stockés sur la carte mémoire, la mémoire de fichiers EM ou la mémoire des commentaires seront automatiquement créés et stockés sur la carte mémoire.

### 5-1-5 Applications

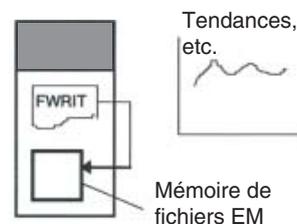
La mémoire de fichiers peut être utilisée pour les applications suivantes.

#### Fichiers de données

Dans cette application, les paramètres des données de la zone DM (pour les cartes d'E/S spéciales, les cartes réseau et les cartes internes (série CS uniquement)) sont stockées dans la carte mémoire. Si le fichier de données est nommé AUTOEXEC.IOM, les paramètres stockés dans le fichier seront automatiquement transférés à la mise sous tension.



Dans cette application, les données des fonctionnement (tendances, contrôle qualité et autres données) générées pendant l'exécution du programme sont stockées dans la mémoire de fichiers EM avec l'instruction WRITE DATA FILE (FWRIT(701)).



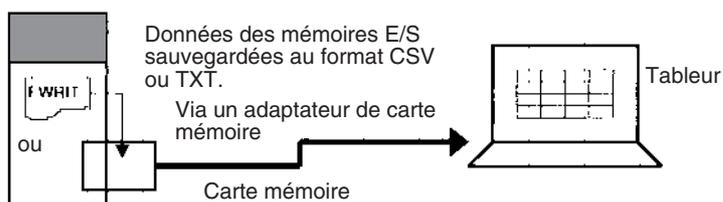
#### Remarque

Il est préférable de stocker les données souvent consultées, telles les données des tendances, dans la mémoire de fichiers EM plutôt que sur une carte mémoire.

#### Fichiers de données ASCII (.TXT et .CSV)

Les données de production qui ont été sauvegardées sur la carte mémoire dans le format TXT ou CSV peuvent être transférées vers un PC par l'inter-

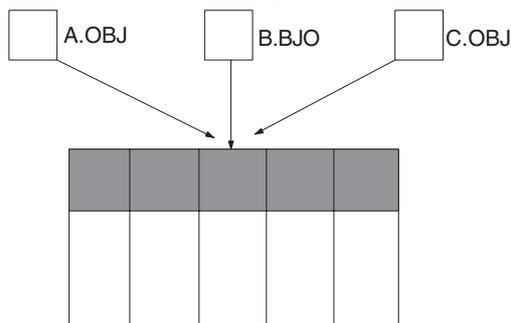
médiaire d'un adaptateur de carte mémoire et modifiées avec un tableur (non pris en charge par les UC CS1 série CS antérieures à EV1).



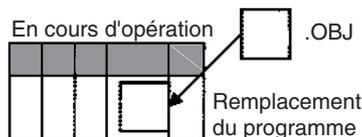
Inversement, les données telles que les paramètres des cartes d'E/S spéciales peuvent être créés avec un tableur dans le format TXT ou CSV, stockées sur une carte mémoire et lues sur l'UC par FREAD(700) (non pris en charge par les UC CS1 série CS antérieures à EV1).

**Fichiers de programmes (.OBJ)**

Dans cette application, les programmes qui contrôlent des processus différents sont stockés sur des cartes mémoire individuelles. Vous pouvez changer l'ensemble de la configuration de l'API (programme, configuration de l'API, etc.) en insérant une carte mémoire différente et en utilisant la fonction de démarrage automatique au démarrage.

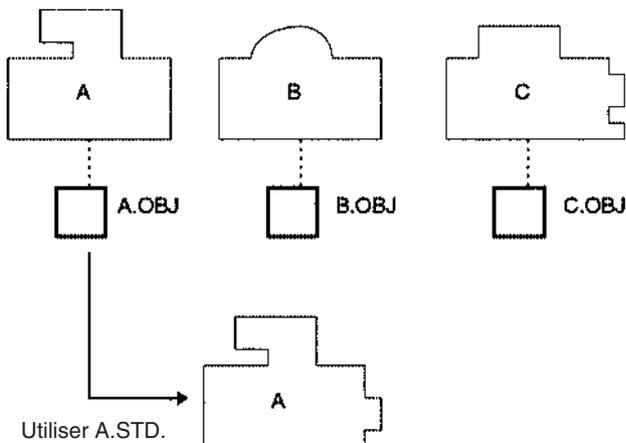


L'intégralité du programme peut être remplacée en cours d'opération à partir du programme lui-même (sans Périphérique de programmation) avec un bit de contrôle de zone auxiliaire (non pris en charge par les UC CS1 série CS antérieures à EV1).



**Fichiers de zone de paramètres (.STD)**

Dans cette application, la configuration de l'API, les tables de routage, la table d'E/S et les autres données destinées à des périphériques ou des machines particuliers sont stockés sur des cartes mémoire. Vous pouvez transférer les données vers un autre périphérique ou une autre machine simplement en changeant de carte mémoire.



**Fichiers de sauvegarde**

La fonction de sauvegarde permet de stocker toutes les données de l'UC (l'ensemble de la mémoire E/S, du programme et de la zone de paramètres) sur la carte mémoire sans périphérique de programmation. Si un problème survient sur les données de l'UC, les données sauvegardées peuvent être restaurées immédiatement (non pris en charge par les UC CS1 série CS antérieures à EV1).

**Fichiers de tableaux de symboles**

Vous pouvez utiliser CX-Programmer pour sauvegarder des symboles de programmes et des commentaires d'E/S dans des fichiers de tableaux de symboles intitulés SYMBOLS.SYM sur les cartes mémoire ou dans la mémoire de fichiers EM.

Lorsque vous utilisez la version 5.0 ou une version supérieure du CX-Programmer associé à une UC série CS/CJ avec une version de carte 3.0 ou supérieure, les fichiers de tables de symboles peuvent être enregistrés à l'intérieur de la mémoire des commentaires dans la mémoire flash interne de l'UC si aucune carte mémoire ou mémoire de fichiers EM n'est fournie, ou si aucune zone de mémoire n'est disponible.

**Fichiers de commentaires**

Vous pouvez utiliser CX-Programmer pour sauvegarder les commentaires des équations logiques du programme dans des fichiers de commentaires intitulés COMMENTS.CMT sur les cartes mémoire ou dans la mémoire de fichiers EM.

Lorsque vous utilisez une UC série CS/CJ avec une version de carte 3.0 ou supérieure, les fichiers de commentaires peuvent être enregistrés à l'intérieur de la mémoire des commentaires dans la mémoire flash interne de l'UC, qu'une carte mémoire ou une mémoire de fichiers EM soient disponibles ou non.

## 5-2 Traitement des fichiers

Les procédures suivantes permettent d'effectuer des opérations de lecture, d'écriture et autres sur les fichiers avec les méthodes suivantes.

- Périphériques de programmation
- Commandes FINS
- Instructions FREAD(700), FWRT(701) et CMND(490) dans le programme utilisateur (CMND(490) : non pris en charge par les UC CS1 série CS antérieures à EV1.)
- Remplacement du programme entier avec les bits de contrôle de zone auxiliaire (non pris en charge par les UC CS1 série CS antérieures à EV1)
- Transfert automatique au démarrage
- Fonction de sauvegarde (non prise en charge par les UC CS1 série CS antérieures à EV1)

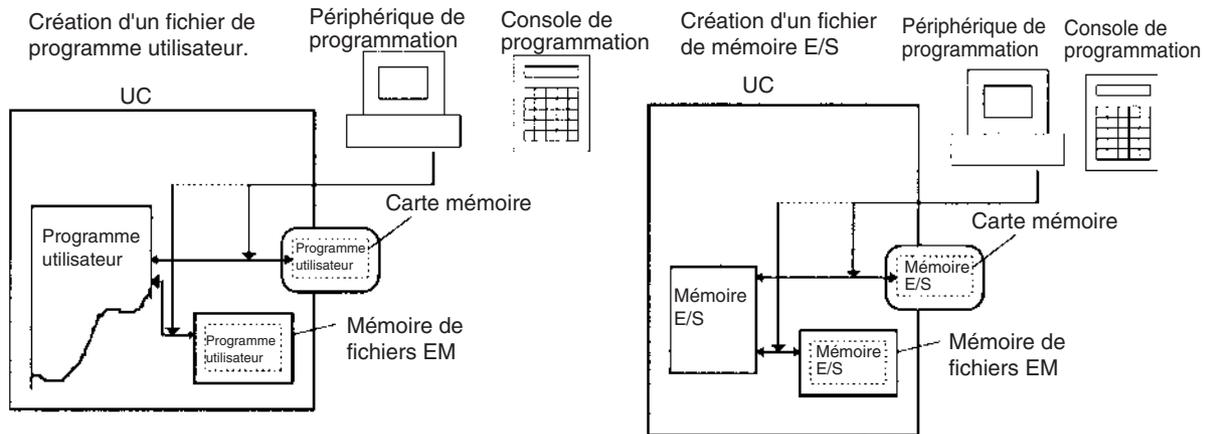
### 5-2-1 Périphériques de programmation (y compris les Consoles de programmation)

Les opérations suivantes sont disponibles avec les périphériques de programmation.

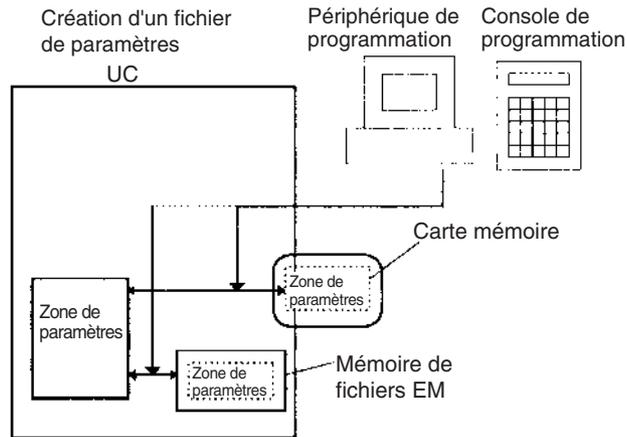
Opération		CX-Programmer	Console de programmation
Lecture de fichiers (transfert de la mémoire de fichiers vers l'UC)		OK	OK
Ecriture de fichiers (transfert de l'UC vers la mémoire de fichiers)		OK (voir remarque.)	OK (voir remarque.)
Comparaison de fichiers (comparaison des fichiers dans l'UC et la mémoire de fichiers)		Impossible	OK
Formatage de la mémoire de fichiers	Cartes mémoire	OK	OK
	Fichiers EM	OK	OK
Modification des noms de fichiers		OK	Impossible
Lecture des données de la mémoire de fichiers		OK	Impossible

Opération	CX-Programmer	Console de programmation
Suppression de fichiers	OK	OK
Copie de fichiers	OK	Impossible
Suppression/création de sous-répertoires	OK	Impossible

**Remarque** Avec l'UC série CS/CJ Ver. 2.0 ou supérieure, vous pouvez utiliser une protection en lecture par mot de passe pour interdire l'écriture d'un fichier de programme dans la mémoire de fichiers (par exemple, une carte mémoire ou la mémoire de fichiers EM).



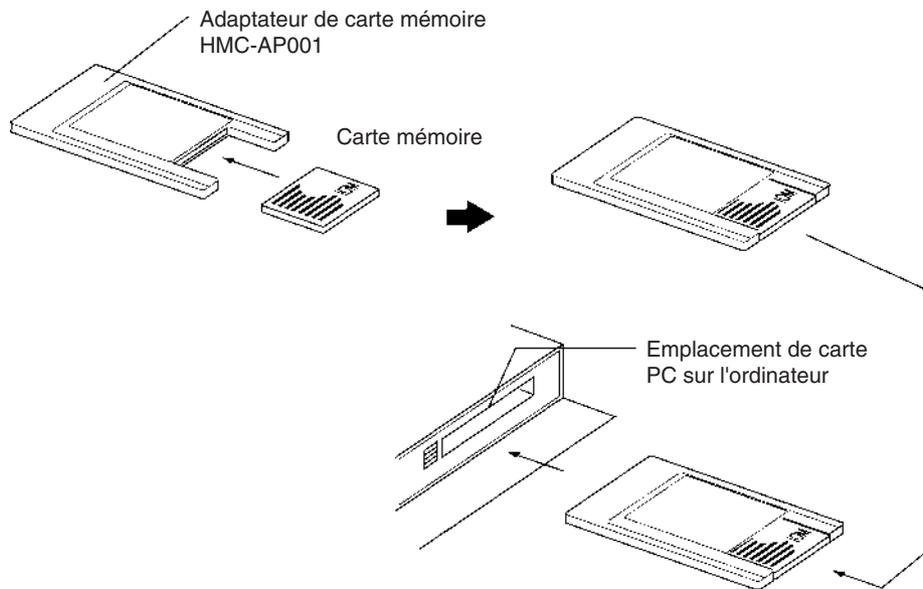
- Remarque**
1. Créez les étiquettes de volume requises avec l'Explorateur Windows.
  2. La mémoire de fichiers utilise le formatage rapide de Windows. Si une erreur de formatage a lieu pour les cartes mémoire, elles peuvent être formatées avec la commande de formatage normal de Windows.
  3. L'heure et la date d'écriture des fichiers pour les transferts entre l'UC et la mémoire de fichiers seront données par l'horloge de l'UC.



**Remarque** Avec l'UC série CS/CJ Ver. 2.0 ou supérieure et CX-Programmer Ver. 4.0 ou supérieure, la création d'un fichier de programme de sauvegarde (.OBJ) peut être interdite en tant qu'option lorsque vous enregistrez un mot de passe pour l'ensemble du programme utilisateur ou pour des tâches spécifiques. Pour plus de détails, voir *Protection en écriture de programmes* dans la section 1-4-2 *Protection contre la lecture améliorée grâce aux mots de passe* du *Manuel d'utilisation des API, série CS* ou du *Manuel d'utilisation des API, série CJ*.

Une carte mémoire peut être installée dans un emplacement de cartes API sur un PC avec l'adaptateur de carte mémoire HMC-AP001 (vendu séparément). Lorsque vous installez une carte mémoire sur l'ordinateur, d'autres

programmes, tel l'Explorateur Windows, peuvent accéder en lecture et en écriture aux fichiers présents sur la carte.



**CX-Programmer**

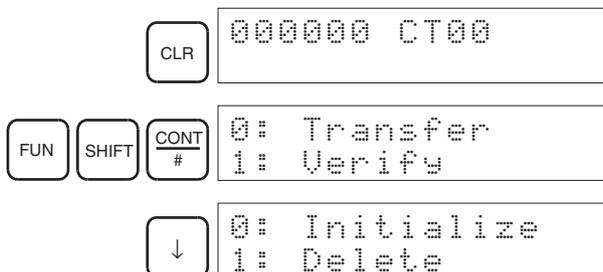
Pour les opérations sur la mémoire de fichiers, utilisez la procédure suivante.

- 1,2,3...
1. Double-cliquez sur l'icône Memory Card dans la fenêtre de projet lorsque l'UC est en ligne. La fenêtre Memory Card s'affiche à l'écran.
  2. Pour effectuer un transfert de l'UC vers la mémoire de fichiers, sélectionnez la zone de programme, la zone de mémoire E/S ou la zone de paramètres dans l'espace de travail du projet, sélectionnez **Transfer** dans la mémoire de fichiers, puis sélectionnez le transfert vers la carte mémoire ou la mémoire de fichiers EM.
- or** Pour effectuer un transfert de la mémoire de fichiers vers l'UC, sélectionnez le fichier dans la mémoire de fichiers, puis déplacez-le (avec la méthode glisser-déplacer) vers la zone de programme, la zone de mémoire E/S ou la zone de paramètres dans l'espace de travail du projet.

**Remarque**

Utilisez les opérations de transfert de projet pour créer et lire des fichiers de tableaux de symboles (SYMBOLS.SYM) et des fichiers de commentaires (COMMENTS.CMT) sur CX-Programmer.

**Console de programmation**



Vous pouvez exécuter les opérations suivantes.

Elément 1	Elément 2	Elément 3	Elément 4	Elément 5
0 : Envoyer	0 : API vers carte mémoire	Sélectionner OBJ, CIO, HR, WR, AR, DM, EM ou STD.	Définir les adresses de début et de fin de transfert	Type de support, nom de fichier
	1 : carte mémoire vers API	Sélectionner OBJ, CIO, HR, WR, AR, DM, EM ou STD.	Définir les adresses de début et de fin de transfert	Type de support, nom de fichier

Elément 1	Elément 2	Elément 3	Elément 4	Elément 5
1 : Vérifier		Sélectionner OBJ, CIO, HR, WR, AR, DM, EM ou STD.	Définir les adresses de début et de fin de comparaison	Type de support, nom de fichier
2 : Initialiser		Entrer 9713 (carte mémoire) ou 8426 (mémoire de fichiers EM).	---	---
3 : Supprimer		Sélectionner OBJ, CIO, HR, WR, AR, DM, EM ou STD.	Type de support, nom de fichier	---

**Remarque** Les types de fichiers sont répertoriées dans le tableau suivant.

Symbole	Type de fichier	
OBJ	Fichier de programme (.OBJ)	
CIO	Fichier de données (.IOM)	Zone CIO
HR		Zone HR
WR		Zone WR
AR		Zone auxiliaire
DM		Zone DM
EMO_		Zone EM
STD		Fichiers de paramètres (.STD)

**Précautions lors de la comparaison des données après le transfert des fichiers de paramètres**

Des erreurs de vérification peuvent se produire sur la console de programmation lorsque vous comparez les données des paramètres entre les fichiers avant le transfert et les données après le transfert, si les fichiers de paramètres (.STD) créés dans une UC série CJ sont enregistrés sur la carte mémoire d'une autre UC série CJ avec une version de carte différente. Le tableau suivant présente l'occurrence des erreurs pour les différentes combinaisons de versions de cartes.

UC source		Etat initial des affectations d'E/S	UC destination			
			Version de carte de l'UC vers laquelle les fichiers de paramètres seront transférés			
			Pré-ver. 2.0	Version de carte 2.0	Version de carte 3.0 ou supérieure	
Version de carte de l'UC dans laquelle les fichiers de paramètres ont été créés	Pré-ver. 2.0	Affectation automatique	Vérification possible	Vérification possible	Erreur de vérification	
		Spécifié par l'utilisateur			Erreur de vérification	
	Version de carte 2.0	Affectation automatique			Erreur de vérification	
		Spécifié par l'utilisateur			Vérification possible	
	Version de carte 3.0 ou supérieure	Affectation automatique			Erreur de vérification	Vérification possible
		Spécifié par l'utilisateur			Vérification possible	

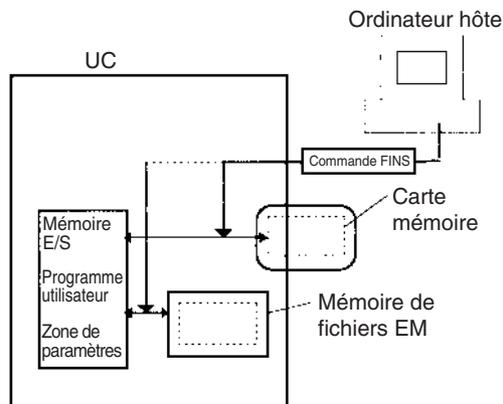
UC source	Etat initial des affectations d'E/S	UC destination			
		Version de carte de l'UC vers laquelle les fichiers pour le transfert automatique au démarrage seront envoyés			
		Pré-ver. 2.0	Version de carte 2.0	Version de carte 3.0 ou supérieure	
Version de carte de l'UC utilisée afin de créer les fichiers pour le transfert automatique au démarrage	Pré-ver. 2.0	Affectation automatique	Bascule sur spécifié par l'utilisateur	(Idem) Affectation automatique	Bascule sur spécifié par l'utilisateur
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	Bascule sur affectation automatique (Voir remarque 1.)	(Idem) Spécifié par l'utilisateur
	Version de carte 2.0	Affectation automatique	Bascule sur spécifié par l'utilisateur	(Idem) Affectation automatique	Bascule sur opération utilisateur
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur
	Version de carte 3.0 ou supérieure	Affectation automatique	Bascule sur spécifié par l'utilisateur	(Idem) Affectation automatique	(Idem) Affectation automatique
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur

### 5-2-2 Commandes FINS

L'UC peut exécuter les opérations suivantes sur la mémoire de fichiers lorsqu'il reçoit la commande FINS appropriée. Ces opérations sont similaires aux fonctions du périphérique de programmation.

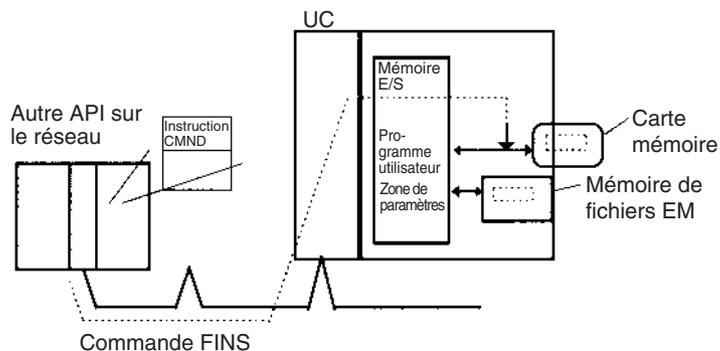
#### Commandes FINS via Host Link

Un ordinateur connecté via un système Host Link peut envoyer une commande FINS avec un en-tête et un code final Host Link.

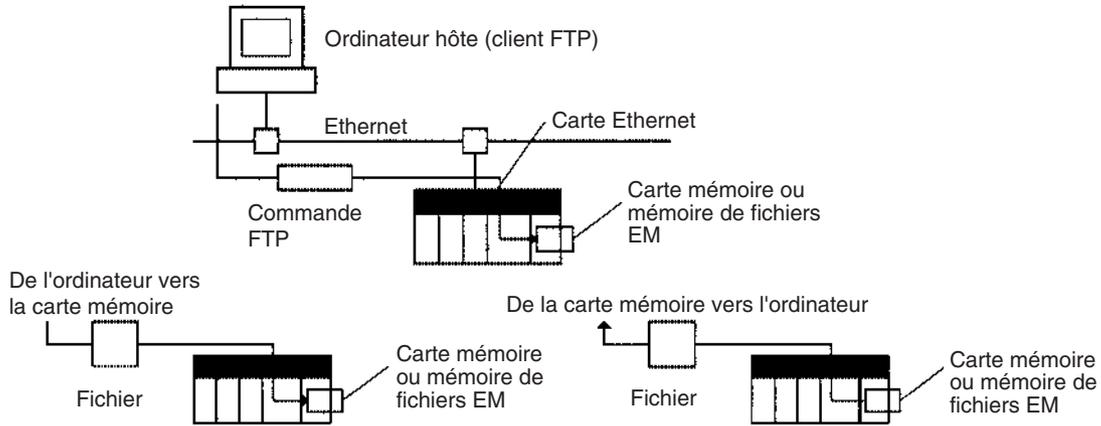


#### Commande FINS d'une autre API réseau

Une autre API sur un réseau peut envoyer une commande FINS avec CMND(490).



**Remarque** Un ordinateur sur un réseau Ethernet peut lire et écrire la mémoire de fichiers (cartes mémoire ou mémoire de fichiers EM) sur une UC via une carte Ethernet. Les données des fichiers peuvent être échangées si l'ordinateur hôte fonctionne en tant que client FTP et l'API série CS/CJ en tant que serveur FTP.



Vous pouvez utiliser les commandes FINS suivantes pour exécuter différentes fonctions (lecture et écriture de fichiers, etc.).

Commande	Nom	Description
2201 hex.	FILE NAME READ	Lit les données de la mémoire de fichiers
2202 hex.	SINGLE FILE READ	Lit une longueur de données de fichier spécifiée à partir d'une position donnée dans un fichier.
2203 hex.	SINGLE FILE WRITE	Ecrit une longueur de données de fichier spécifiée à partir d'une position donnée dans un fichier.
2204 hex.	FILE MEMORY FORMAT	Formate (initialise) la mémoire de fichiers.
2205 hex.	FILE DELETE	Supprime les fichiers spécifiés stockés dans la mémoire de fichiers.
2207 hex.	FILE COPY	Copie des fichiers d'une mémoire de fichiers vers une autre mémoire de fichiers.
2208 hex.	FILE NAME CHANGE	Change un nom de fichier.
220A hex.	MEMORY AREA FILE TRANSFER	Transfère ou compare des données entre la zone de mémoire E/S et la mémoire de fichiers.
220B hex.	PARAMETER AREA FILE TRANSFER	Transfère ou compare les données entre la zone de paramètres et la mémoire de fichiers.
220C hex.	PROGRAM AREA FILE TRANSFER	Transfère ou compare les données entre la zone UM (mémoire utilisateur) et la mémoire de fichiers.
2215 hex.	CREATE/DELETE SUBDIRECTORY	Crée et supprime des sous-répertoires.

**Remarque** L'heure de l'horloge interne de l'UC est utilisée pour dater les fichiers créés dans la mémoire de fichiers avec les commandes 220A, 220B, 220C et 2203.

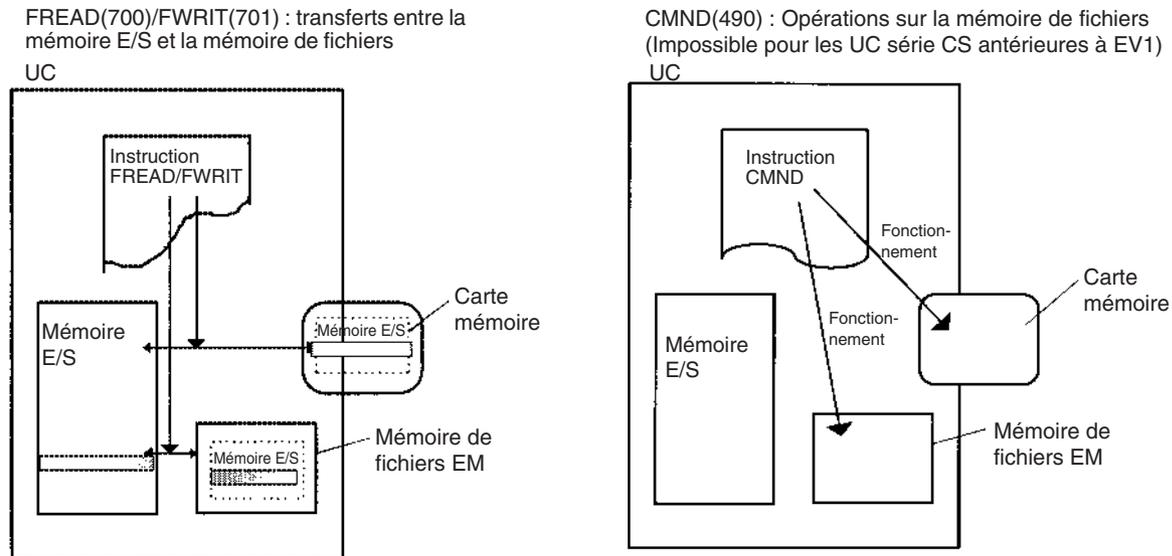
### 5-2-3 FREAD(700), FWGIT(701) et CMND(490)

L'instruction FWGIT(701) (WRITE DATA FILE) permet de créer un fichier de données contenant les données de la mémoire E/S spécifiée sur une carte mémoire ou dans la mémoire de fichiers EM. Elle permet également d'ajouter ou d'écraser des données à partir de n'importe quel point dans les fichiers existants.

L'instruction FREAD(700) (READ DATA FILE) permet de lire des données de mémoire E/S à partir d'un emplacement spécifié dans un fichier de données, sur une carte mémoire ou dans la mémoire de fichiers EM, et de les écrire dans la partie de la mémoire E/S spécifiée. La lecture est possible à partir de n'importe quel point dans le fichier spécifié.

**Remarque** Ces instructions ne transfèrent pas le fichier spécifié, mais plutôt la quantité de données spécifiée, à partir de la position de départ indiquée.

Vous pouvez exécuter l'instruction CMND(490) (DELIVER COMMAND) pour émettre une commande FINS sur l'UC proprement dite afin d'exécuter des opérations sur les fichiers. Les opérations sur les fichiers (formatage, suppression, copie, changement de nom, etc.) peuvent être exécutées sur les fichiers contenus sur la carte mémoire ou dans la mémoire de fichiers EM (non pris en charge par les UC CS1 série CS antérieures à EV1).



**Instructions FREAD(700)/FWRIT(701)**

FREAD(700) et FWRIT(701) transfèrent les fichiers entre la mémoire E/S et la mémoire de fichiers. Toutes les UC CJ peuvent transférer des données binaires (fichiers .IOM) et les UC V1 peuvent également transférer des fichiers ASCII (fichiers .TXT et CSV).

Nom	Mnémonique	Description
READ DATA FILE	FREAD(700)	Lit les données des fichiers de données ou les éléments de données spécifiés dans la mémoire E/S spécifiée.
WRITE DATA FILE	FWRIT(701)	Utilise les données de la zone de mémoire E/S spécifiée pour créer un fichier de données spécifié.

**Transfert de fichiers ASCII (non pris en charge par les UC CS1 série CS antérieures à EV1).**

Les fichiers ASCII peuvent être transférés au même titre que les fichiers binaires ; les troisième et quatrième chiffres de l'opérande du mot de contrôle de l'instruction (C) indiquent le type du fichier de données transféré et le nombre de champs entre les retours chariot.

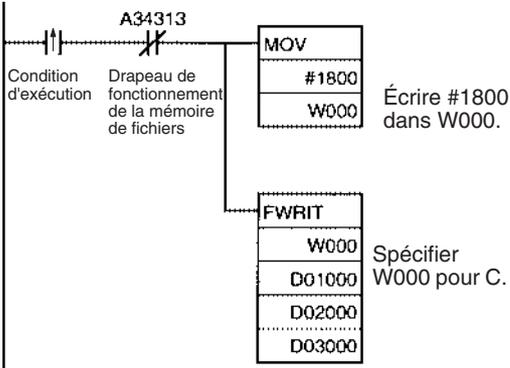
Bits dans C	Paramètres	Limites du périphérique de programmation
12 à 15	Type de données 0 : Binaire (.IOM) 1 : Mots non délimités (.TXT) 2 : Mots doubles non délimités (.TXT) 3 : Mots délimités par des virgules (.CSV) 4 : Mots doubles délimités par des virgules (.CSV) 5 : Mots délimités par des tabulations (.TXT) 6 : Mots doubles délimités par des tabulations (.TXT)	Si vous utilisez CX-Programmer V1.1 ou une version antérieure, seul 0 hex. (fichiers .IOM) peut être spécifié directement. Si vous utilisez CX-Programmer V1.2 ou une version supérieure (ou une console de programmation), les bits du mot de contrôle peuvent prendre une valeur comprise entre 0 et 6 hex.
08 à 11	Retours chariot 0 : Pas de retours 8 : Retour tous les 10 champs 9 : Retour tous les 1 champ A : Retour tous les 2 champs B : Retour tous les 4 champs C : Retour tous les 5 champs D : Retour tous les 16 champs	Si vous utilisez CX-Programmer V1.1 ou une version antérieure (ou une console de programmation), seul 0 hex. (par de retours) peut être spécifié directement. Si vous utilisez CX-Programmer V1.2 ou une version ultérieure, les bits du mot de contrôle peuvent prendre la valeur 0 hex. ou une valeur comprise entre 8 et D hex.

**CX-Programmer V1.1 ou version inférieure :**  
**Configuration indirecte du mot de contrôle**

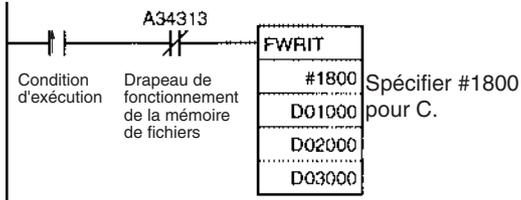
Lorsque vous utilisez CX-Programmer V1.1 ou une version antérieure, les fichiers ASCII ne peuvent pas être transférés avec FREAD(700) et FWRT(701) si vous entrez une constante pour le mot de contrôle afin de spécifier le traitement des types de données et des retours chariot. Seules les données binaires sans retours chariot peuvent être transférées si une constante est utilisée.

Vous pouvez toutefois transférer les fichiers ASCII avec FREAD(700) et FWRT(701) en définissant le mot de contrôle de façon indirecte. Ecrivez le paramètre du mot de contrôle voulu dans un mot et spécifiez ce mot en tant que mot de contrôle dans FREAD(700) ou FWRT(701), comme indiqué dans la partie gauche du schéma suivant.

CX-Programmer Versions V1.1 et antérieures



CX-Programmer Versions V1.2 et ultérieures



**Remarque** L'heure de l'horloge interne de l'UC est utilisée pour dater les fichiers créés dans la mémoire de fichiers avec FWRT(701).

Vous ne pouvez exécuter qu'une seule opération à la fois sur la mémoire de fichiers ; par conséquent, FREAD(700) et FWRT(701) ne doivent pas être exécutées lorsque l'une des opérations suivantes sur la mémoire de fichiers est en cours :

- 1,2,3...**
1. Exécution de FREAD(700) ou FWRT(701)

2. Exécution de CMND(490) pour envoyer une commande FINS à l'UC proprement dite
3. Remplacement du programme complet par des opérations de bit de contrôle de zone auxiliaire
4. Exécution d'une opération de sauvegarde simple

Utilisez le drapeau de fonctionnement de la mémoire fichier (A34313) pour contrôler de façon exclusive les instructions de mémoire de fichiers afin d'empêcher leur exécution pendant qu'une autre opération est en cours sur la mémoire de fichiers.

Lorsque FREAD(700) est en cours d'exécution, le drapeau d'erreur de lecture de fichier (A34310) passe à ON et l'instruction n'est pas exécutée si le fichier spécifié contient le type de données incorrect ou si les données du fichier sont altérées. Pour les fichiers texte ou CSV, le code des caractères doit correspondre à des données hexadécimales et des séparateurs doivent être placés tous les 4 chiffres pour les données des mots et tous les 8 chiffres pour les données des doubles mots. Les données seront lues jusqu'au point où un caractère interdit sera détecté.

**Bits auxiliaires/mots associés**

Nom	Adresse	Opération
Type de carte mémoire	A34300 à A34302	Indique le type de la carte mémoire installée, le cas échéant.
Drapeau d'erreur de formatage de mémoire de fichiers EM	A34306	Passé à ON lorsqu'une erreur de formatage se produit dans la première banque EM affectée à la mémoire de fichiers. Passe à OFF lorsque le formatage s'est terminé normalement.
Drapeau d'erreur du format carte mémoire	A34307	Sur ON lorsque la carte mémoire n'est pas formatée ou lorsqu'une erreur de formatage s'est produite.
Drapeau d'erreur d'écriture de fichier	A34308	Passé à ON lorsqu'une erreur est survenue pendant l'écriture dans le fichier.
Drapeau d'erreur d'écriture de fichier	A34309	A ON lorsque les données n'ont pas pu être écrites, le fichier étant protégé en écriture ou l'espace disponible étant insuffisant.
Drapeau d'erreur lecture de fichier	A34310	A ON lorsqu'un fichier n'a pas pu être lu parce que ses données sont altérées ou qu'il contient un type de données incorrect.
Drapeau d'absence de fichier	A34311	A ON lorsque les données n'ont pas pu être lues parce que le fichier spécifié n'existe pas.
Drapeau de fonctionnement de la mémoire de fichiers	A34313	ON pour chacune des situations suivantes : L'UC traite une commande FINS qu'elle a reçue avec CMND(490). FREAD(700) ou FWRT(701) sont en cours d'exécution. Le programme est remplacé à l'aide d'un bit de contrôle de zone auxiliaire. Une opération de sauvegarde simple est en cours d'exécution.
Drapeau d'accès au fichier	A34314	A ON lorsque le système est en train d'accéder aux données du fichier.
Drapeau de carte mémoire détectée	A34315	A ON lorsque la carte mémoire a été détectée. (non pris en charge par les UC CS1 série CS antérieures à EV1).
Nombre d'éléments à transférer	A346 à A347	Ces mots indiquent le nombre de mots ou de champs restant à transférer (32 bits). Lorsqu'un fichier binaire (.IOM) est transféré, ce nombre est décrétementé chaque fois qu'un mot est lu. Lorsqu'un fichier texte ou CSV est transféré, ce nombre est décrétementé chaque fois qu'un champ est transféré.

**CMND(490)w: DELIVER COMMAND**

CMND(490) permet d'émettre une commande FINS sur l'UC locale pour exécuter des opérations sur la mémoire de fichiers (formatage, suppression de fichiers, etc.). Effectuez les réglages suivants dans les mots de contrôle de CMND(490) lorsque vous émettez une commande FINS de mémoire de fichiers sur l'API locale :

- 1,2,3...**
1. Réglez l'adresse du réseau de destination sur 00 (réseau local) dans C+2.
  2. Réglez l'adresse de la carte de destination sur 00 (UC de l'API) et le nœud de destination sur 00 (dans le nœud local) dans C+3.
  3. Définissez le nombre de tentatives à 0 dans C+4. (En effet, ce paramètre n'est pas valide.)

**Commandes FINS relatives à la mémoire de fichiers**

Pour plus d'informations sur les commandes FINS, reportez-vous à la section 5-2-2 *Commandes FINS*.

**Remarque**

D'autres commandes FINS relatives à la mémoire de fichiers ne figurant pas dans le tableau suivant peuvent être exécutées. Veuillez vous reporter au *Manuel de référence des commandes de communication (W342)* pour de plus amples informations sur les commandes FINS.

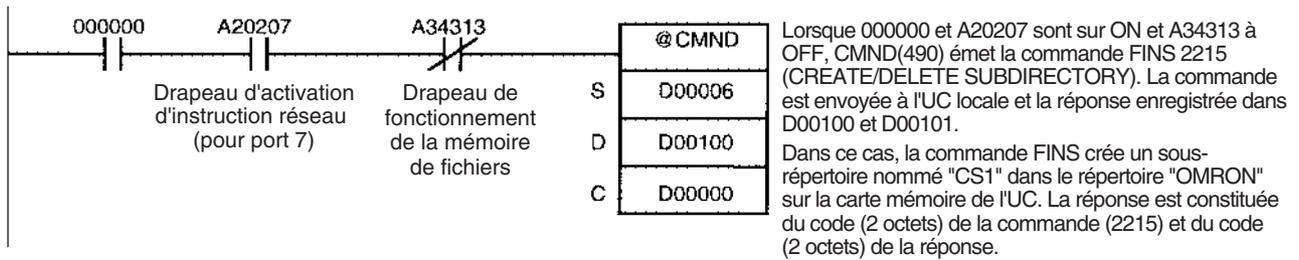
CMND(490) ne peut pas être exécutée sur l'UC locale si une autre instruction CMND(490) est en cours d'exécution sur une autre UC, si FREAD(700) ou FWRT(701) est en cours d'exécution, si le remplacement du programme par une opération de bit de contrôle de zone auxiliaire est en cours ou si une opération de sauvegarde simple est en cours d'exécution. Veillez à inclure le drapeau de fonctionnement de la mémoire de fichiers (A34313) comme une condition normalement fermée pour empêcher l'exécution de CMND(490) pendant qu'une autre opération sur la mémoire de fichiers est en cours.

Si CMND(490) ne peut pas être exécutée pour l'UC locale, le drapeau d'erreur passera à ON.

**Bits auxiliaires/mots associés**

Nom	Adresse	Opération
Drapeau de fonctionnement de la mémoire de fichiers	A34313	ON pour chacune des situations suivantes : <ul style="list-style-type: none"> <li>• L'UC s'est envoyée une commande FINS à elle-même avec CMND(490).</li> <li>• FREAD(700) ou FWRT(701) sont en cours d'exécution.</li> <li>• Le programme est remplacé à l'aide d'un bit de contrôle de zone auxiliaire.</li> <li>• Une opération de sauvegarde simple est en cours d'exécution.</li> </ul>
Drapeau de carte mémoire détectée	A34315	A ON lorsque la carte mémoire a été détectée. (non pris en charge par les UC CS1 série CS antérieures à EV1).

L'exemple suivant montre comment utiliser CMND(490) pour créer un sous-répertoire sur la carte mémoire.



	15	8	7	0		
S:	D00006	2	2	1	5	Code de la commande : 2215 Hex (CREATE/DELETE SUBDIRECTORY)
S+1:	D00007	8	0	0	0	Numéro de disque : 8000 Hex (carte mémoire)
S+2:	D00008	0	0	0	0	Paramètre : 0000 Hex (créer un sous-répertoire.)
S+3:	D00009	4	3	5	3	Nom du sous-répertoire : CS1□□□□□.□□□ (□ : un espace)
S+4:	D00010	3	1	2	0	
S+5:	D00011	2	0	2	0	
S+6:	D00012	2	0	2	0	
S+7:	D00013	2	E	2	0	
S+8:	D00014	2	0	2	0	
S+9:	D00015	0	0	0	6	Longueur du répertoire : 0006 Hex (6 caractères)
S+10:	D00016	5	C	4	F	Chemin du répertoire : \OMRON
S+11:	D00017	4	D	5	2	
S+12:	D00018	4	F	4	E	

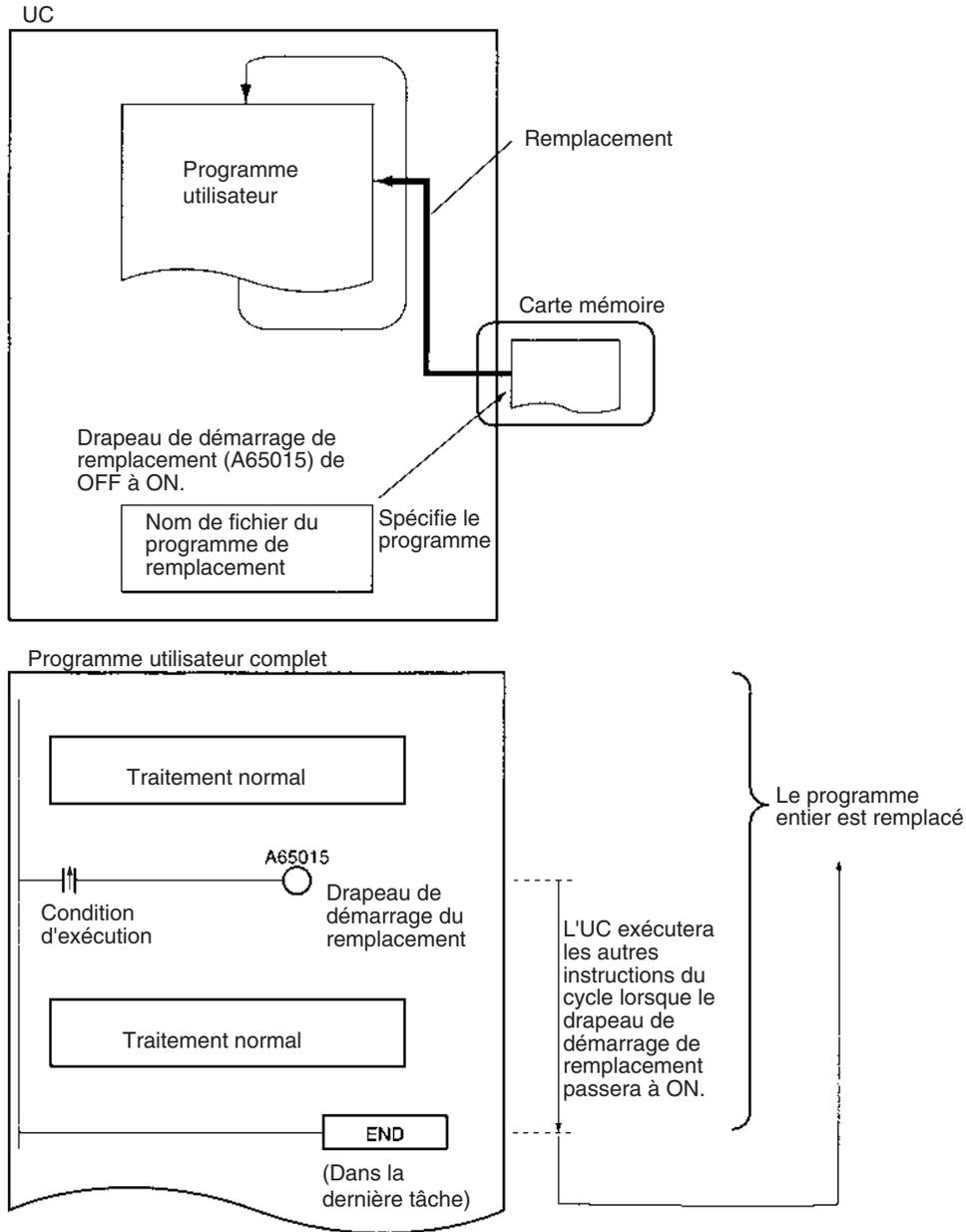
	15	8	7	0		
C:	D00000	0	0	1	A	Nombre d'octets de données de commande : 001A Hex (26 octets)
C+1:	D00001	0	0	0	4	Nombre d'octets de données de réponse : 0004 Hex (4 octets)
C+2:	D00002	0	0	0	0	Adresse de destination : 0000 Hex (réseau local)
C+3:	D00003	0	0	0	0	00 Hex (nœud node) et 00 Hex (UC)
C+4:	D00004	0	7	0	0	Réponse demandée, port de communication 7, 0 nouvelle tentative
C+5:	D00005	0	0	0	0	Temps de réponse : FFFF Hex (6 553,5 s)

**Remarque** Outre les commandes relatives aux opérations sur la mémoire de fichiers figurant dans le tableau précédent, d'autres commandes FINS peuvent être envoyées à l'API locale. Le drapeau de fonctionnement de la mémoire de fichiers doit être utilisé pour éviter l'exécution simultanée de ces autres commandes FINS.

### 5-2-4 Remplacement du programme entier en cours d'opération

(non pris en charge par les UC CS1 série CS antérieures à EV1).

Vous pouvez remplacer le programme entier en cours d'opération (mode RUN ou MONITOR) en mettant le drapeau de démarrage de remplacement (A65015) à ON. Le fichier spécifié sera lu à partir de la carte mémoire et remplacera le programme exécutable à la fin du cycle en cours. Le mot de passe du programme de remplacement (A651) et le nom du fichier de programme (A654 à A657) doivent être enregistrés à l'avance et le fichier de programme spécifié doit exister sur la carte mémoire pour remplacer le programme en cours d'opération.



Il est également possible de remplacer le programme lorsque son exécution est arrêtée (mode PROGRAM) en mettant le drapeau de démarrage de remplacement à ON à partir d'un périphérique de programmation.

**Remarque** Le fichier du programme de remplacement ne peut pas être lu à partir de la mémoire de fichiers EM.

Le drapeau de démarrage de remplacement (A65015) peut être mis à ON à n'importe quel emplacement (adresse de programme) du programme. L'UC exécutera les autres instructions du cycle une fois que le drapeau de démarrage de remplacement sera passé de OFF à ON.

Le programme ne sera pas exécuté pendant le remplacement. Après le remplacement du programme, l'opération sera redémarrée comme si l'UC était passée du mode PROGRAM au mode RUN ou MONITOR.

Le programme sera remplacé à la fin du cycle dans lequel le drapeau de démarrage de remplacement sera passé de OFF à ON, c'est-à-dire après l'exécution de END(001) dans la dernière tâche du programme.

- Remarque**
1. Mettez le bit de maintien IOM (A50012) ON si vous souhaitez conserver l'état des données de la mémoire E/S tout au long du remplacement du programme.  
Mettez le bit de maintien état forcé (A50013) à ON si vous souhaitez conserver l'état des bits définis de force et réinitialisés de force tout au long du remplacement du programme.
  2. Si le bit de maintien IOM (A50012) est à ON avant le remplacement du programme, l'état des bits dans la mémoire E/S sera conservé après le remplacement. Assurez-vous que les charges externes fonctionnent correctement avec les mêmes données de mémoire E/S.  
De même, si le bit de maintien état forcé (A50013) est à ON avant le remplacement du programme, l'état des bits définis de force et réinitialisés de force sera conservé après le remplacement. Assurez-vous que les charges externes fonctionnent correctement avec les mêmes bits définis de force et réinitialisés de force.

**Fichier de remplacement** Le fichier de programme spécifié dans le nom du fichier de programme (A654 à A657) sera lu à partir de la carte mémoire et remplacera le programme existant à la fin du cycle dans lequel le drapeau de démarrage de remplacement (A65015) sera passé de OFF à ON.

Fichier	Nom de fichier et extension	Spécification du nom du fichier de remplacement (*****)
Fichier de programme	*****.OBJ	Ecrivez le nom du fichier du programme de remplacement dans les mots A654 à A657 avant le remplacement du programme.

**Conditions requises pour le remplacement du programme** Les conditions suivantes doivent être réunies pour que vous puissiez remplacer le programme en cours d'opération.

- Le mot de passe du programme (A5A5) a été écrit dans A651.
- Le fichier de programme spécifié dans les mots du nom du fichier de programme (A654 à A657) existe dans le répertoire racine de la carte mémoire.
- La carte mémoire a été détectée par l'UC. (A34315 ON)
- Aucune erreur fatale ne s'est produite.
- Aucune opération sur la mémoire de fichiers n'est en cours d'exécution. (A34313 OFF)
- Aucune écriture de données n'est en cours dans la zone de programme.
- Le droit d'accès est disponible. (Par exemple, aucun transfert de données n'est en cours entre CX-Programmer et l'API.)

**Remarque** Le programme peut être transféré dans n'importe quel mode de fonctionnement.

**Fonctionnement de l'UC pendant le remplacement du programme** L'UC fonctionnera de la façon suivante pendant le remplacement du programme :

- Exécution du programme : arrêté
- Surveillance du temps de cycle : pas de surveillance

**Opérations qui continuent pendant et après le remplacement du programme** Lorsque le bit de maintien IOM (A50012) est à ON, les données des zones de mémoire suivantes sont conservées : zone CIO, zone de travail (W), drapeaux de fin de temporisation (T), registres d'index (IR), registres de données (DR) et numéro de banque EM courant.

**Remarque** Les valeurs actuelles de temporisation sont effacées pendant le remplacement du programme.

Si le bit de maintien IOM est à ON lorsque le programme est transféré, les charges qui étaient produites avant le remplacement du programme continueront d'être produites après le remplacement. Assurez-vous que les charges fonctionnent correctement après le remplacement du programme.

L'état des bits définis et réinitialisés de force sera conservé pendant le remplacement du programme si le bit de maintien état forcé (A50013) est à ON.

Les interruptions seront masquées.

Si la traçabilité des données est effectuée, elle sera interrompue.

Les conditions des instructions (verrouillages, coupures et exécution de programme de bloc) seront initialisées.

Les drapeaux de différenciation seront initialisés, que le bit de maintien IOM soit à ON ou à OFF.

**Opérations après le remplacement du programme**

L'état des tâches cycliques dépend de leurs propriétés de démarrage d'opération. (Elles ont le même état que si l'API passait du mode PROGRAM au mode RUN/MONITOR.)

Le drapeau du premier cycle (A20011) sera à ON pendant un cycle après la reprise de l'exécution du programme. (L'état est le même que si l'API passait du mode PROGRAM au mode RUN/MONITOR.)

**Temps requis pour le remplacement du programme**

Taille du programme entier	Temps de traitement des périphériques défini dans la configuration de l'API	Temps approximatif requis pour le remplacement du programme
60 Kpas	Par défaut (4 % du temps de cycle)	6 s
250 Kpas		25 s

**Bits auxiliaires/mots associés**

Nom	Adresse	Opération
Drapeau de fonctionnement de la mémoire de fichiers	A34313	ON pour chacune des situations suivantes : L'UC s'est envoyé une commande FINS à elle-même avec CMND(490). FREAD(700) ou FWRT(701) sont en cours d'exécution. Le programme est écrasé à l'aide d'un bit de contrôle de zone auxiliaire (A65015). Une opération de sauvegarde simple est en cours d'exécution.
Drapeau de détection de carte mémoire (Non pris en charge par les UC CS1 série CS antérieures à EV1)	A34315	A ON lorsque la carte mémoire a été détectée.
Bit de maintien IOM	A50012	Lorsque ce bit est à ON, le contenu de la mémoire E/S est conservé pendant le remplacement du programme.
Bit de maintien de état forcé	A50013	Lorsque ce bit est à ON, l'état des bits définis et réinitialisés de force est conservé pendant le remplacement du programme.
Code d'achèvement fin de remplacement (Non pris en charge par les UC CS1 série CS antérieures à EV1)	A65000 à A65007	Codes pour un remplacement de programme normal (A65014 OFF) : 01 hex. : le fichier de programme (.OBJ) a remplacé le programme. Codes pour un remplacement de programme incomplet (A65014 ON) : 00 hex. : Une erreur fatale s'est produite. 01 hex. : Une erreur de mémoire s'est produite. 11 hex. : le programme est protégé en écriture. 12 hex. : le mot de passe du programme dans A651 est incorrect. 21 hex. : aucune carte mémoire n'est installée. 22 hex. : le fichier spécifié n'existe pas. 23 hex. : le fichier spécifié est trop grand (erreur de mémoire). 31 hex. : l'une des opérations suivantes était en cours d'exécution : • Opération sur la mémoire de fichiers. • Ecriture du programme. • Modification du mode de fonctionnement.
Code d'erreur de remplacement (Non pris en charge par les UC CS1 série CS antérieures à EV1)	A65014	A ON lorsqu'une erreur est survenue pendant une tentative de remplacement du programme après que A65015 est passé de OFF à ON. Passe à OFF lorsque A65015 repasse de OFF à ON.

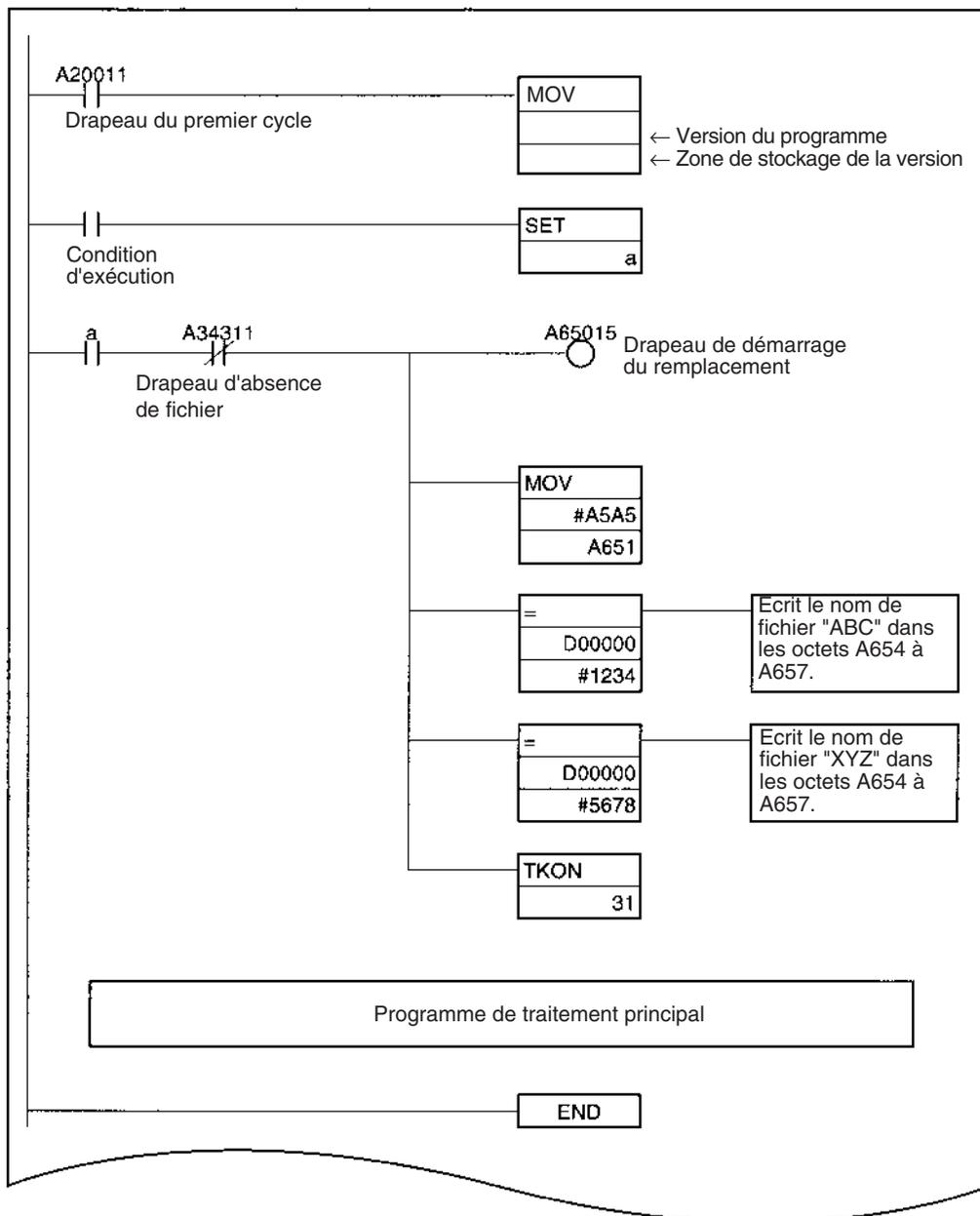
Nom	Adresse	Opération															
Drapeau de démarrage de remplacement (Non pris en charge par les UC CS1 série CS antérieures à EV1)	A65015	Si vous avez activé ce bit en définissant le mot de passe du programme (A651) à A5A5 hex., le remplacement du programme commencera lorsque ce bit passera de OFF à ON. Ne refaites pas passer ce bit de OFF à ON pendant le remplacement du programme. Ce bit passe automatiquement à OFF lorsque le remplacement du programme est terminé (normalement ou avec une erreur) ou que l'alimentation est branchée. L'état de ce bit peut être lu à partir d'un périphérique de programmation, d'un terminal opérateur programmable (IHM) ou d'un ordinateur hôte pour déterminer si le remplacement du programme a été effectué ou non.															
Mot de passe du programme (Non pris en charge par les UC CS1 série CS antérieures à EV1)	A651	Inscrivez le mot de passe dans ce mot pour activer le remplacement du programme. A5A5 hex. : active le drapeau de démarrage de remplacement (A65015). Autre valeur : désactive le drapeau de démarrage de remplacement (A65015). Ce bit passe automatiquement à OFF lorsque le remplacement du programme est terminé (normalement ou avec une erreur) ou que l'alimentation est branchée.															
Nom du fichier de programme (Non pris en charge par les UC CS1 série CS antérieures à EV1)	A654 à A657	Avant de démarrer le remplacement du programme, écrivez le nom du fichier du programme de remplacement dans ces mots en ASCII. Ecrivez simplement les 8 caractères du nom de fichier ; l'extension .OBJ est ajoutée automatiquement. Ecrivez les caractères dans l'ordre à partir de A654 (l'octet le plus significatif en premier). Si le nom du fichier comporte moins de 8 caractères, remplissez les octets restants par des codes de caractère d'espacement (20 Hex). N'incluez aucun caractère NULL ou espace dans le nom du fichier proprement dit. L'exemple suivant présente les données du fichier de programme ABC.OBJ : <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td></td> <td style="text-align: center;">15</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="padding-right: 10px;">A654</td> <td style="border: 1px solid black; padding: 2px 5px;">41</td> <td style="border: 1px solid black; padding: 2px 5px;">42</td> </tr> <tr> <td style="padding-right: 10px;">A655</td> <td style="border: 1px solid black; padding: 2px 5px;">43</td> <td style="border: 1px solid black; padding: 2px 5px;">20</td> </tr> <tr> <td style="padding-right: 10px;">A656</td> <td style="border: 1px solid black; padding: 2px 5px;">20</td> <td style="border: 1px solid black; padding: 2px 5px;">20</td> </tr> <tr> <td style="padding-right: 10px;">A657</td> <td style="border: 1px solid black; padding: 2px 5px;">20</td> <td style="border: 1px solid black; padding: 2px 5px;">20</td> </tr> </table>		15	0	A654	41	42	A655	43	20	A656	20	20	A657	20	20
	15	0															
A654	41	42															
A655	43	20															
A656	20	20															
A657	20	20															

**Exemple de programme 1**

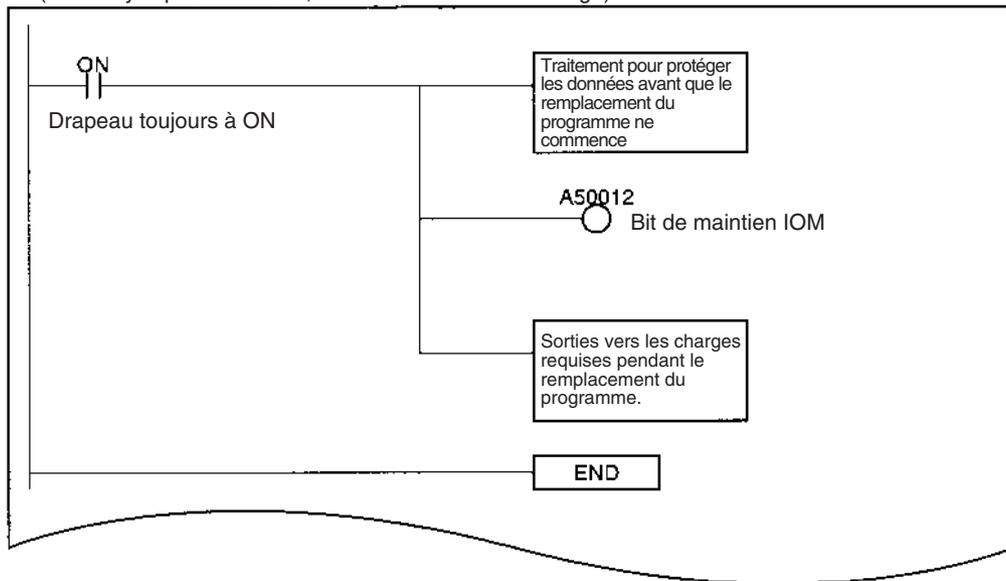
Enregistrez les fichiers de programmes ABC.OBJ et XYZ.OBJ sur la carte mémoire et sélectionnez l'un ou l'autre programme en fonction de la valeur de D00000. Définissez D00000 sur #1234 lorsque vous sélectionnez ABC.OBJ ou sur #5678 lorsque vous sélectionnez XYZ.OBJ.

Démarrez et exécutez une autre tâche pour effectuer le traitement requis avant le remplacement du programme ou le traitement du bit de maintien IOM.

Tâche principale (tâche cyclique numéro 0)



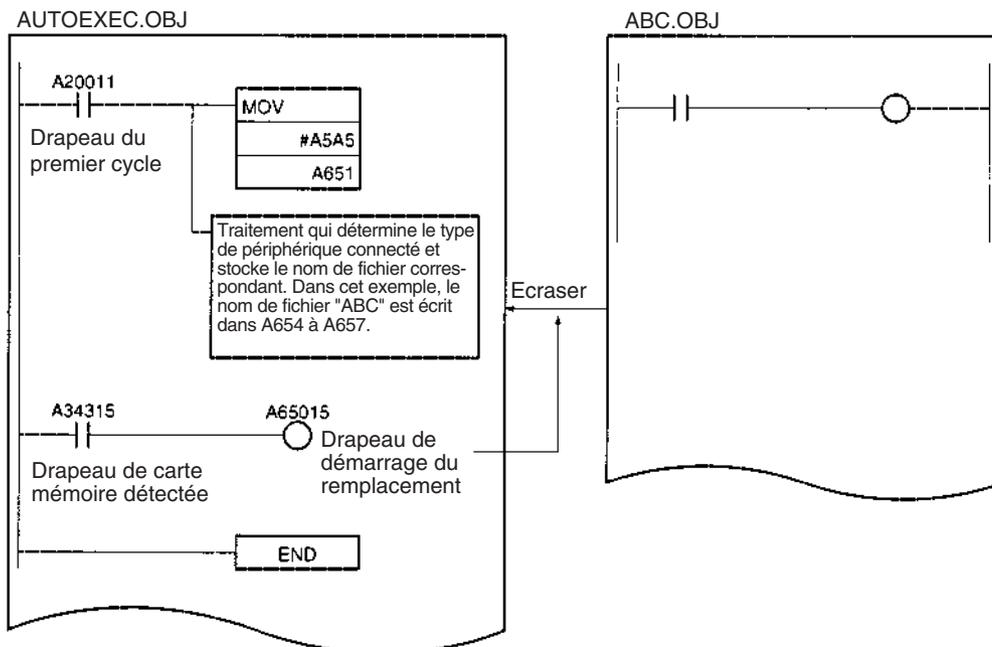
Tâche protégeant les données pendant le remplacement du programme  
(Tâche cyclique numéro 31, état en attente au démarrage)



**Exemple de programme 2**

Enregistrez les fichiers de programmes de plusieurs périphériques et le fichier de programme du transfert automatique au démarrage (AUTOEXEC.OBJ ou REPLACE.OBJ (voir remarque) sur une carte mémoire. Lorsque l'API est mis sous tension, le fichier de transfert automatique au démarrage est lu et ce programme est remplacé ultérieurement par un fichier de programme pour un périphérique différent.

**Remarque** REPLACE.OBJ est pris en charge par les UC série CS/CJ Ver. 2.0 ou supérieure uniquement.



### 5-2-5 Transfert automatique au démarrage

Le transfert automatique au démarrage est utilisé pour lire les données du programme utilisateur, des paramètres et de la mémoire E/S à partir d'une carte mémoire sur l'UC lors de la mise sous tension.

**Remarque**

Les fichiers suivants peuvent être lus automatiquement dans la mémoire de l'UC. Il n'est pas possible d'utiliser cette fonction pour lire la mémoire de fichiers EM.

Le nom du fichier de programme varie selon qu'un fichier de zone de paramètres doit également être transféré ou non.

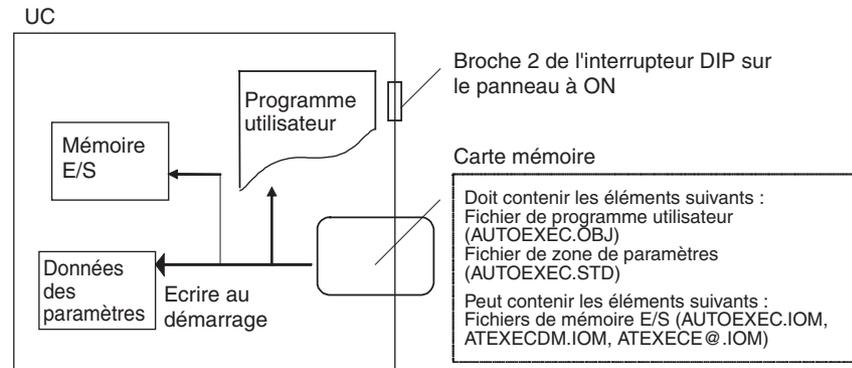
**Transfert d'un fichier de zone de paramètres**

Utilisez les noms de fichiers suivants.

Fichier de programme : AUTOEXEC.OBJ

Fichier de zone de paramètres : AUTOEXEC.STD

Fichiers de données : AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE@.IOM



Fichier	Nom de fichier	Au démarrage	Requis pour le transfert automatique
Fichier de programme	AUTOEXEC.OBJ	Le contenu de ce fichier est automatiquement transféré et remplace tout le programme utilisateur, y compris les attributs des tâches de l'UC.	Requis sur la carte mémoire.
Fichier de zone de paramètres	AUTOEXEC.STD	Le contenu de ce fichier est transféré automatiquement et recouvre toutes les données des paramètres initiaux dans l'UC.	Requis sur la carte mémoire.
Fichier de données	AUTOEXEC.IOM	Mots DM affectés aux cartes d'E/S spéciales, cartes réseau et cartes internes (série CS uniquement) Le contenu de ce fichier est transféré automatiquement dans la zone DM à partir de D20000 lors de la mise sous tension. (Voir remarque 1.)	Non requis sur la carte mémoire.
	ATEXECDM.IOM	Mots DM universels Le contenu de ce fichier est transféré automatiquement dans la zone DM à partir de D00000 lors de la mise sous tension. (Non pris en charge par les UC CS1 série CS antérieures à EV1) (Voir remarque 1.)	
	ATEXECE@.IOM	Mots DM universels Le contenu de ce fichier est transféré automatiquement dans la zone EM en commençant avec E@_00000 lors de la mise sous tension. (non pris en charge par les UC CS1 série CS antérieures à EV1).	

**Remarque**

1. Si les données contenues dans AUTOEXEC.IOM et ATEXECDM.IOM se recouvrent, les données de ATEXECDM.IOM écraseront les données de recouvrement transférées à partir de AUTOEXEC.IOM, ATEXECDM.IOM étant le dernier fichier écrit.
2. Le fichier de programme (AUTOEXEC.OBJ) et le fichier de paramètres (AUTOEXEC.STD) doivent se trouver sur la carte mémoire. En l'absence de ces fichiers, le transfert automatique échouera, une erreur de mémoire se produira et A40115 (drapeau d'erreur de mémoire : erreur fatale)

- passera à ON. (Il n'est pas nécessaire que le fichier de mémoire E/S (AUTOEXEC.IOM) soit présent.)
3. Il est possible de créer les fichiers AUTOEXEC.IOM, ATEXECMD.IOM et ATEXECE□.IOM à partir d'un périphérique de programmation (CX-Programmer ou console de programmation) avec des adresses de début autres que D20000, D00000 et E□\_00000, respectivement. Les données seront de toute façon écrites avec l'adresse de début correcte, mais ne spécifiez pas d'autres adresses de début.
  4. Si la broche 7 de l'interrupteur DIP est sur ON et la broche 8 sur OFF pour utiliser la fonction de sauvegarde simple, celle-ci sera prioritaire, même si la broche 2 est également sur ON. Dans ce cas, les fichiers BACKUP□□ seront transférés vers l'UC, mais le transfert automatique au démarrage ne sera pas exécuté (non pris en charge par les UC CS1 série CS antérieures à EV1).
  5. Le fonction de transfert automatique au démarrage peut être utilisée conjointement avec la fonction de remplacement de programme. Le drapeau de démarrage de remplacement (A65015) peut être mis à ON par le programme qui est transféré automatiquement au démarrage pour le remplacer par un autre programme.
  6. La fonction de transfert automatique au démarrage et le remplacement complet du programme avec les bits de la zone auxiliaire peuvent être utilisés ensemble ; autrement dit, le programme transféré automatiquement vers l'UC au démarrage peut contenir du code pour manipuler les bits de la zone auxiliaire de façon à remplacer le programme par un autre.

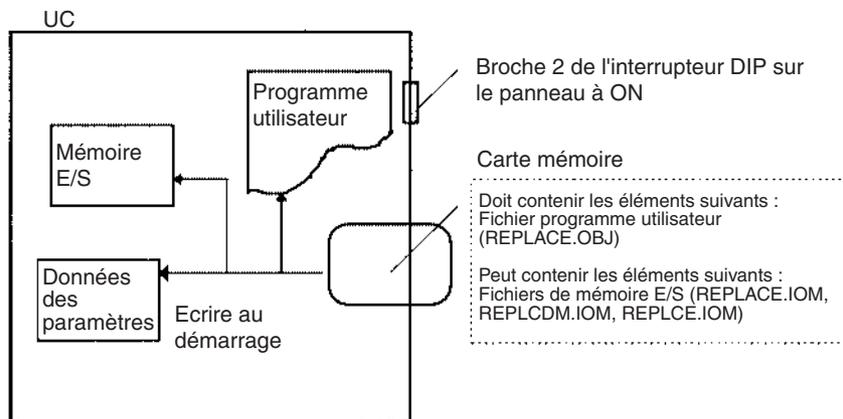
**Transfert sans fichier de zone de paramètres (UC série CS/CJ Ver. 2.0 ou supérieure uniquement)**

Utilisez les noms de fichiers suivants.

Fichier de programme : REPLACE.OBJ

Fichier de zone de paramètres : Non requis et non transféré indépendamment du nom de fichier.

Fichiers de données : REPLACE.IOM, REPLCMD.IOM, REPLCE□.IOM



Fichier	Nom de fichier	Au démarrage	Requis pour le transfert automatique
Fichier de programme	REPLACE.OBJ Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement	Le contenu de ce fichier est transféré automatiquement et recouvre tout le programme utilisateur, y compris les attributs des tâches de l'UC.	Requis sur la carte mémoire.
Fichier de zone de paramètres		Non transféré indépendamment du nom de fichier.	Non obligatoire.

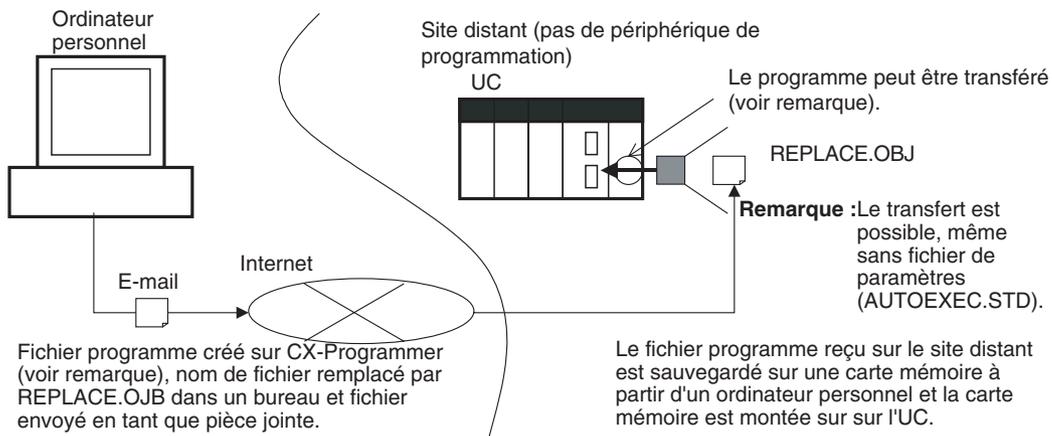
Fichier	Nom de fichier	Au démarrage	Requis pour le transfert automatique
Fichier de données	REPLACE.IOM Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement	Mots DM affectés aux cartes d'E/S spéciales, cartes réseau et cartes internes (série CS uniquement). Le contenu de ce fichier est transféré automatiquement dans la zone DM à partir de D20000 lors de la mise sous tension.	Non requis sur la carte mémoire.
	REPLCDM.IOM Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement	Mots DM universels Le contenu de ce fichier est transféré automatiquement dans la zone DM à partir de D00000 lors de la mise sous tension.	
	REPLCE□.IOM Remarque : UC série CS/CJ Ver. 2.0 ou supérieure uniquement	Mots DM universels Le contenu de ce fichier est transféré automatiquement dans la zone EM à partir de E□_00000 lors de la mise sous tension. □ indique le numéro de la banque.	

- Remarque**
1. Si le nom du fichier de programme est REPLACE.OBJ (UC série CS/CJ Ver. 2.0 ou ultérieure uniquement), le fichier de zone de paramètres ne sera pas transféré, même s'il se trouve sur la carte mémoire et quel que soit son nom.
  2. Lorsque vous créez le fichier REPLACE.IOM, REPLCDM.IOM ou REPLCE□.IOM à partir d'un périphérique de programmation (console de programmation ou CX-Programmer), spécifiez toujours la première adresse appropriée (D20000, D00000 ou E□\_00000). Le contenu du fichier sera toujours transféré à partir de la première adresse appropriée (D20000, D00000 ou E□\_00000), même si un autre mot de début est spécifié, ce qui pourrait entraîner le recouvrement du contenu de cette partie de la zone DM ou de la banque EM par des données incorrectes.
  3. Si la broche 7 de l'interrupteur DIP est sur ON et la broche 8 sur OFF pour utiliser la fonction de sauvegarde simple, celle-ci sera prioritaire, même si la broche 2 est également sur ON. Dans ce cas, les fichiers BACKUP□□ seront transférés vers l'UC, mais le transfert automatique au démarrage ne sera pas exécuté.

**Exemple d'application**

**Transfert sans fichiers sans fichier de zone de paramètres**

Un fichier de programme (.OBJ) peut être créé hors ligne dans un bureau et transféré vers un emplacement distant sans fichier de zone de paramètres (.STD). Il est possible de stocker le fichier de programme sur une carte mémoire sur le site distant sans utiliser de périphérique de programmation et la carte mémoire peut être utilisée pour transférer automatiquement le programme vers l'UC au démarrage.



**Combinaisons de transfert de fichiers prises en charge** Les tableaux suivants indiquent si les fichiers sont transférés automatiquement vers l'UC au démarrage en fonction des fichiers présents sur la carte mémoire.

■ Fichier de programme : AUTOEXEC.OBJ

Fichier de programme	Fichier de zone de paramètres	Fichiers de données	Transféré/non transféré
AUTOEXEC.OBJ	AUTOEXEC.STD	Un ou plusieurs des fichiers suivants : AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Transféré.
		Aucun	
	Aucun	Un ou plusieurs des fichiers suivants : AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Non transféré.
		Aucun	

■ Fichier de programme : REPLACE.OBJ

Fichier de programme	Fichier de zone de paramètres	Fichiers de données	Transféré/non transféré
REPLACE.OBJ	Présent	Un ou plusieurs des fichiers suivants : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Transféré, mais le fichier de zone de paramètres n'est pas transféré.
		Aucun	
	Aucun	Un ou plusieurs des fichiers suivants : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Transféré.
		Aucun	

■ Pas de fichier de programme

Fichier de programme	Fichier de zone de paramètres	Fichiers de données	Transféré/non transféré
Aucun	AUTOEXEC.STD	Un ou plusieurs des fichiers suivants : AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Non transféré.
		Un ou plusieurs des fichiers suivants : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
		Aucun	
	Aucun	Un ou plusieurs des fichiers suivants : AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	
		Un ou plusieurs des fichiers suivants : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
		Aucun	

■ es fichiers AUTOEXEC et REPLACE

Plusieurs fichiers de programmes

Fichiers de programmes		Fichier de zone de paramètres	Fichiers de données	Transféré/non transféré
AUTOEXEC.OBJ	REPLACE.OBJ	AUTOEXEC.STD	Un ou plusieurs des fichiers suivants : AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Non transféré.
			Un ou plusieurs des fichiers suivants : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
			Aucun	
		Aucun	Un ou plusieurs des fichiers suivants : AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	
			Un ou plusieurs des fichiers suivants : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
			Aucun	

**Plusieurs types de fichiers de données**

Fichier de programme	Fichier de zone de paramètres	Fichiers de données		Transféré/non transféré
AUTOEXEC.OBJ	AUTOEXEC.STD	Un ou plusieurs des fichiers suivants : AUTOEXEC.IOM, ATEXECMD.IOM, ATEXECE□.IOM	Un ou plusieurs des fichiers suivants : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Les fichiers de données suivants sont transférés : AUTOEXEC.IOM, ATEXECMD.IOM, ATEXECE□.IOM
	Aucun			Non transféré.
REPLACE.OBJ	Ignoré			Les fichiers de données suivants sont transférés : REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM

**Procédure**

- 1,2,3...**
1. Mettez l'API hors tension.
  2. Placez la broche 2 de l'interrupteur DIP sur ON sur le panneau avant de l'UC. Assurez-vous que les broches 7 et 8 sont sur OFF.

**Remarque** La fonction de sauvegarde simple sera prioritaire sur la fonction de transfert automatique au démarrage ; veillez par conséquent à ce que les broches 7 et 8 soient sur OFF.

3. Préparez une carte mémoire en procédant comme suit :
  - a. Transfert avec un fichier de zone de paramètres  
Insérez une carte mémoire contenant le fichier de programme utilisateur (AUTOEXEC.OBJ), le fichier de zone de paramètres (AUTOEXEC.STD) et/ou les fichiers de mémoire E/S (AUTOEXEC.IOM, ATEXECMD.IOM et ATEXECE□.IOM) créés avec un CX-Programmer. (Le fichier de programme et le fichier de zone de paramètres doivent se trouver sur la carte mémoire. Les fichiers de mémoire E/S sont facultatifs.)
  - b. Transfert sans fichier de zone de paramètres  
Insérez une carte mémoire contenant le fichier de programme utilisateur (REPLACE.OBJ) et/ou les fichiers de mémoire E/S (REPLACE.IOM, REPLCDM.IOM et REPLCE□.IOM) créés avec un CX-Programmer. (Le fichier de programme doit se trouver sur la carte mémoire. Les fichiers de mémoire E/S sont facultatifs.)
4. Mettez l'API sous tension.

**Remarque Echec du transfert automatique au démarrage**

Si le transfert automatique échoue au démarrage, une erreur de mémoire se produira, A40115 passera à ON et l'UC s'arrêtera. Si une erreur se produit, coupez l'alimentation pour l'effacer. (L'erreur ne peut être effacée sans la mise hors tension.)

**Interrupteur DIP sur le panneau avant de l'UC**

Broche(s)	Nom	Paramètre
2	Transfert automatique au démarrage	ON : exécute le transfert automatique au démarrage. OFF : n'exécute pas le transfert automatique au démarrage.
7 et 8	Broches sauvegarde simple	Réglez les deux broches sur OFF.

**Bits auxiliaires/mots associés**

Nom	Adresse	Paramètre
Drapeau d'erreur mémoire (erreur fatale)	A40115	A ON lorsqu'une erreur survient dans la mémoire ou lorsqu'une erreur survient dans le transfert automatique à partir de la carte mémoire quand l'alimentation est mise sous tension (transfert automatique au démarrage). L'UC s'arrête et le voyant ERR/ALM situé à l'avant s'allume. <b>Remarque :</b> A40309 passera à ON si l'erreur s'est produite pendant le transfert automatique au démarrage. (Dans ce cas, l'erreur ne peut pas être effacée.)
Drapeau d'erreur de transfert au démarrage de la carte mémoire	A40309	A ON lorsque le transfert automatique au démarrage a été sélectionné et qu'une erreur se produit pendant le transfert automatique (broche 2 de l'interrupteur DIP sur ON). Une erreur se produit en cas d'erreur de transfert, si le fichier spécifié n'existe pas ou si la carte mémoire n'est pas installée. <b>Remarque :</b> la mise hors tension permet d'effacer l'erreur. (Cette erreur ne peut être effacée lorsque l'alimentation est sous tension.)

**Précautions lorsque vous changez l'état des affectations d'E/S pendant le transfert automatique au démarrage**

L'état des affectations d'E/S dépend des versions de carte des UC source et destination lorsque vous utilisez une seule UC série CJ pour créer les fichiers de paramètres du transfert automatique au démarrage, les enregistrer sur la carte mémoire et les transférer ensuite automatiquement vers une autre UC série CJ au démarrage. Le tableau suivant présente les changements d'état des affectations d'E/S pour les différentes combinaisons de versions de cartes.

UC source		Etat initial des affectations d'E/S	UC destination		
			Version de carte de l'UC vers laquelle les fichiers pour le transfert automatique au démarrage seront envoyés		
			Pré-ver. 2.0	Version de carte 2.0	Version de carte 3.0 ou supérieure
Version de carte de l'UC utilisée afin de créer les fichiers pour le transfert automatique au démarrage	Pré-ver. 2.0	Affectation automatique	Bascule sur spécifié par l'utilisateur	(Idem) Affectation automatique	Bascule sur spécifié par l'utilisateur
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	Bascule sur affectation automatique (Voir remarque 1.)	(Idem) Spécifié par l'utilisateur
	Version de carte 2.0	Affectation automatique	Bascule sur spécifié par l'utilisateur	(Idem) Affectation automatique	Bascule sur opération utilisateur
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur
	Version de carte 3.0 ou supérieure	Affectation automatique	Bascule sur spécifié par l'utilisateur	(Idem) Affectation automatique	(Idem) Affectation automatique
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur

**Remarque**

1. Lorsque les fichiers pour le transfert automatique au démarrage (AUTOEXEC.STD) sont créés et enregistrés sur une carte mémoire via des affectations d'E/S spécifiées par l'utilisateur avec une UC série CJ pré-version 2.0, le système bascule automatiquement sur l'affectation d'E/S automatique au démarrage si les données sont transférées automatiquement depuis la carte mémoire.
2. Lorsque des fichiers pour le transfert automatique au démarrage (AUTOEXEC.STD) sont créés et enregistrés sur la carte mémoire à l'aide d'une UC série CJ avec une version de carte 2.0, l'état des affectations d'E/S bascule automatiquement sur « spécifié par l'utilisateur » si les données sont transférées automatiquement de la carte mémoire vers une UC série CJ avec une version de carte 3.0 ou supérieure.

**Précautions lors de la comparaison des fichiers de paramètres du transfert automatique au démarrage**

La comparaison des données de paramètres entre les fichiers avant le transfert et après le transfert peut générer des erreurs de vérification sur la console de programmation lorsque vous créez des fichiers de paramètres de transfert automatique au démarrage (AUTOEXEC.STD) et exécutez le transfert automatique au démarrage pour des combinaisons d'UC série CJ pré-ver. 2.0, version de carte 2.0 et version de carte 3.0.

UC source		Etat initial des affectations d'E/S	UC destination			
			Version de carte de l'UC à la destination de la sauvegarde/restauration			
			Pré-ver. 2.0	Version de carte 2.0 ou supérieure	Version de carte 3.0 ou supérieure	
Version de carte de l'UC à la source de la sauvegarde	Pré-ver. 2.0	Affectation automatique	Vérification possible	Vérification possible	Erreur de vérification	
		Spécifié par l'utilisateur			Erreur de vérification	
	Version de carte 2.0 ou supérieure	Affectation automatique			Erreur de vérification	
		Spécifié par l'utilisateur			Vérification possible	
	Version de carte 3.0 ou supérieure	Affectation automatique			Erreur de vérification	Vérification possible
		Spécifié par l'utilisateur			Vérification possible	

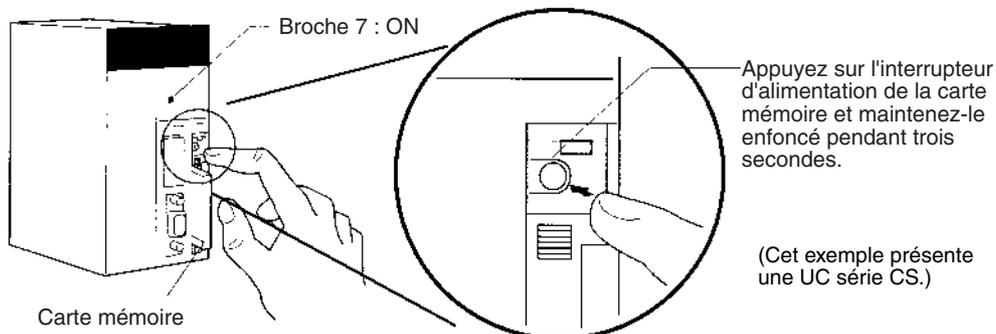
**5-2-6 Fonction de sauvegarde simple**

Cette fonction n'est pas prise en charge par les UC CS1 série CS antérieures à EV1.

**Sauvegarde de données de l'UC sur la carte mémoire**

Pour sauvegarder les données, mettez la broche 7 de l'interrupteur DIP de l'UC sur ON, puis maintenez l'interrupteur d'alimentation de la carte mémoire enfoncé pendant trois secondes. La fonction de sauvegarde crée automatiquement les fichiers de sauvegarde avec des noms de fichier et des extensions fixes, et les écrit sur la carte mémoire. Les fichiers de sauvegarde contiennent le programme, les données de la zone de paramètres et les données de la mémoire E/S, les tables de symboles (voir remarque), les fichiers de commentaires (voir remarque) et les fichiers d'index de programmes (voir remarque). Cette fonction peut être exécutée dans l'importe quel mode de fonctionnement.

**Remarque** Ces données sont prises en charge par les UC série CS/CJ avec une version de carte 3.0 ou supérieure uniquement. Les fichiers de sauvegarde sont créés automatiquement à partir des fichiers de la carte mémoire, de la mémoire des fichiers EM ou de la mémoire des commentaires.



### Restauration des données de la carte mémoire sur l'UC.

Pour restaurer les fichiers de sauvegarde sur l'UC, vérifiez que la broche 7 est sur ON, mettez l'API hors tension, puis remettez-la sous tension. Les fichiers de sauvegarde contenant le programme, les données de zone de paramètres et les données de mémoire E/S sont lues à partir de la carte mémoire sur l'UC.

\*1 Ces données sont prises en charge par les UC série CS/CJ avec une version de carte 3.0 ou supérieure uniquement. Les fichiers de sauvegarde sont lus sur la carte mémoire, dans la mémoire des fichiers EM ou dans la mémoire des commentaires.

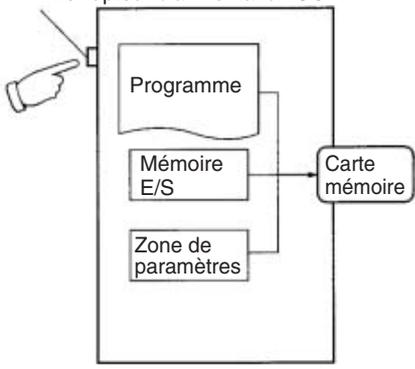
- Remarque**
1. La fonction de sauvegarde sera prioritaire sur la fonction de transfert automatique au démarrage ; les fichiers de sauvegarde seront donc lus sur l'UC à la mise sous tension de l'API, même si la broche 2 de l'interrupteur DIP est sur ON.
  2. Les données ne seront pas lues à partir de la carte mémoire sur l'UC si la broche 1 de l'interrupteur DIP est sur ON (mémoire du programme protégée en écriture).
  3. Lorsque les fichiers de sauvegarde sont lus à partir de la carte mémoire par la fonction, l'état de la mémoire E/S et les bits définis/réinitialisés de force seront effacés, sauf si les paramètres requis sont spécifiés dans la zone auxiliaire et la configuration de l'API.  
 Si le bit de maintien IOM (A50012) est à ON et que la configuration de l'API est définie de façon à conserver l'état du bit de maintien IOM au démarrage lors de l'écriture des fichiers de sauvegarde, l'état des données de la mémoire E/S sera conservé lorsque les données seront lues à partir de la carte mémoire.  
 Si le bit de maintien état forcé (A50013) est à ON et que la configuration de l'API est définie de façon à conserver l'état du bit de maintien de l'état forcé au démarrage lors de l'écriture des fichiers de sauvegarde, l'état des bits définis/réinitialisés de force sera conservé lorsque les données seront lues à partir de la carte mémoire.
  4. Une UC CS1-H, CJ1-H, CJ1M ou CS1D reste en mode PROGRAM après l'exécution de la fonction de sauvegarde simple et ne peut pas passer en mode MONITOR ou RUN tant que l'alimentation n'a pas été redémarrée. Après la sauvegarde, mettez l'UC hors tension, modifiez les paramètres de la broche 7 puis remettez l'UC sous tension.
  5. Les sauvegardes de fichiers peuvent durer de quelques secondes à plusieurs minutes. Pour plus d'informations sur les temps d'exécution, reportez-vous à la page 253.

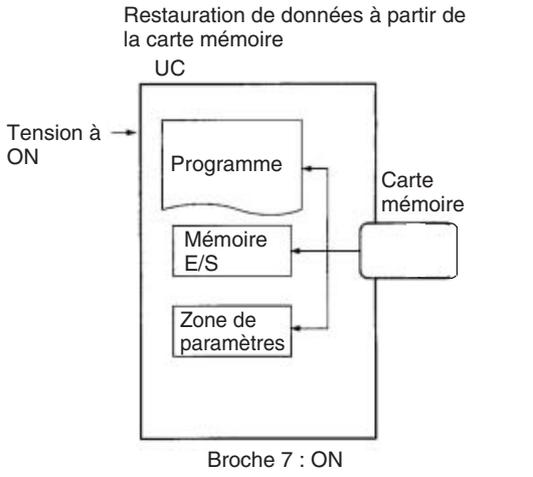
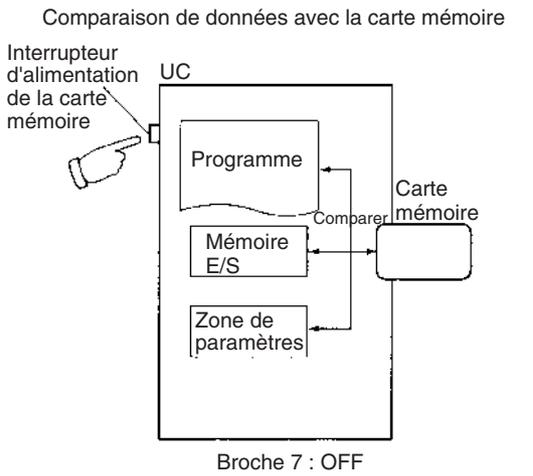
### Comparaison des données sur la carte mémoire et dans l'UC

Pour comparer les fichiers de sauvegarde sur la carte mémoire et les données dans l'UC, mettez la broche 7 sur OFF sur l'interrupteur DIP de l'UC, puis maintenez l'interrupteur d'alimentation de la carte mémoire enfoncé pendant trois secondes. La fonction de sauvegarde comparera le programme, les données de la zone de paramètres, les données de la mémoire E/S, les tables de symboles (voir remarque), les fichiers de commentaires (voir remarque) et les fichiers d'index de programmes (voir remarque) sur la carte mémoire avec les données correspondantes dans l'UC. Cette fonction peut être exécutée dans l'importe quel mode de fonctionnement.

- Remarque** Ces données sont prises en charge par les UC série CS/CJ avec une version de carte 3.0 ou supérieure uniquement.

Le tableau suivant résume les opérations de sauvegarde simple.

Opération de sauvegarde	Etat des broches	Procédure
	Broche 7	
<p>Sauvegarde de données de l'UC sur la carte mémoire</p> <p>Sauvegarde de données sur la carte mémoire</p> <p>Carte mémoire Interrupteur d'alimentation UC</p>  <p>Broche 7 : ON</p>	<p>ON</p>	<p>Appuyez sur l'interrupteur d'alimentation de la carte mémoire et maintenez-le enfoncé pendant trois secondes.</p>

Opération de sauvegarde	Etat des broches	Procédure
	Broche 7	
<p>Restauration des données de la carte mémoire sur l'UC.</p> <p>Restauration de données à partir de la carte mémoire</p>  <p>Broche 7 : ON</p>	<p>ON</p>	<p>Mettez l'API hors tension, puis de nouveau sous tension. (Voir remarque 1.)</p>
<p>Comparaison de données entre la carte mémoire et l'UC</p> <p>Comparaison de données avec la carte mémoire</p>  <p>Broche 7 : OFF</p>	<p>OFF</p>	<p>Appuyez sur l'interrupteur d'alimentation de la carte mémoire et maintenez-le enfoncé pendant trois secondes.</p>

- Remarque**
1. Pour plus d'informations sur les résultats des opérations de lecture, d'écriture et de comparaison, reportez-vous à la section *Vérification des opérations de sauvegarde avec les voyants* à la page 240.
  2. Pour des lignes directrices sur les délais des opérations de sauvegarde de la carte mémoire, reportez-vous à la section 5-3-2 *Procédures de fonctionnement*.

**Fichiers de sauvegarde**

**Fichiers de données**

Nom de fichier et extension	Zone de données et plage d'adresses enregistrée		Sauvegarde de la mémoire E/S sur la carte mémoire (création de fichiers)	Restauration de la carte mémoire vers l'UC	Comparaison de la carte mémoire avec l'UC		Fichiers requis lors de la restauration des données
					CS1/CJ1	CS1-H/CJ1-H	
UC	CS/CJ				CS1/CJ1	CS1-H/CJ1-H	
BACKUP.IOM	DM	D20000 à D32767	Oui	Oui	Oui	---	Requis sur la carte mémoire
BACKUPIO.IOR	CIO	0000 à 6143 (y compris état de bit forcé.)	Oui	--- <sup>4</sup>	Oui	---	Requis sur la carte mémoire
	WR	W0000 à W511 (y compris état de bit forcé.)	Oui	--- <sup>4</sup>	Oui	---	
	RH	H000 à H511	Oui	Oui	Oui	---	
	AR	A000 à A447 A448 à A959	Oui	---	---	---	
			Oui	Oui	Oui	---	
	Temporisation <sup>1</sup>	T0000 à T4095	Oui	Oui <sup>4</sup>	Oui	---	
Compteur	C0000 à C4095	Oui	Oui	Oui	---		
BACKUPDM.IOM	DM	D00000 à D19999	Oui	Oui	Oui	---	Requis sur la carte mémoire
BACKUPE□.IOM <sup>2,3</sup>	EM	E□_00000 à E□_32767	Oui	Oui	Oui	---	Requis sur la carte mémoire (doit correspondre à l'UC)

**Remarque**

1. Les drapeaux de fin et les valeurs actuelles sont sauvegardés.
2. Le □ représente le numéro de banque et le nombre de banques dépend de l'UC utilisée.  
 Lorsque les fichiers BACKUPE□.IOM de la carte mémoire sont restaurés sur l'UC, la lecture des fichiers commence à la banque 0 et se poursuit jusqu'au numéro de banque maximum dans l'UC. Les autres fichiers BACKUPE□.IOM ne seront pas lus si le nombre de banques sauvegardées dépasse le nombre de banques dans l'UC. Inversement, les banques EM restantes dans l'UC resteront inchangées si le nombre de banques sauvegardées est inférieur au nombre de banques dans l'UC.  
 Si un fichier BACKUPE□.IOM est manquant (par exemple : 0, 1, 2, 4, 5, 6), seuls les fichiers consécutifs seront lus. Dans le cas présent, les données seront lues dans les banques 0, 1 et 2 uniquement.
3. Les données de la zone EM seront sauvegardées en tant que données binaires. Les banques EM qui ont été converties dans la mémoire de fichiers seront sauvegardées avec les banques EM non converties.  
 La mémoire de fichiers EM peut être restaurée dans la zone EM d'une autre UC seulement si les fichiers BACKUPE□.IOM sont consécutifs et que le nombre de banques EM sauvegardées correspond au nombre de banques dans l'UC. Si les fichiers BACKUPE□.IOM ne sont pas consécutifs ou que le nombre de banques EM ne correspond pas au nombre de banques dans l'UC, la mémoire de fichiers EM retrouvera sa condition non formatée et les fichiers dans la mémoire de fichiers sont incorrects. (Les banques EM régulières seront lues normalement.)
4. Normalement, le contenu de la zone CIO, de la zone WR, les drapeaux de fin de temporisation, les valeurs actuelles de temporisation et l'état des bits

définis/réinitialisés de force seront effacés lorsque l'API sera mise sous tension et BACKUPIO.IOR lu à partir de la carte mémoire.

Si le bit de maintien IOM (A50012) est à ON et que la configuration de l'API est définie de façon à conserver l'état du bit de maintien IOM au démarrage lors de l'écriture des fichiers de sauvegarde, l'état des données de la mémoire E/S sera conservé lorsque les données seront lues à partir de la carte mémoire.

Si le bit de maintien état forcé (A50013) est à ON et que la configuration de l'API est définie de façon à conserver l'état du bit de maintien de l'état forcé au démarrage lors de l'écriture des fichiers de sauvegarde, l'état des bits définis/réinitialisés de force sera conservé lorsque les données seront lues à partir de la carte mémoire.

**Fichiers de programme**

Nom de fichier et extension	Contenu	Sauvegarde de la mémoire E/S sur la carte mémoire (création de fichiers)	Restauration de la carte mémoire vers l'UC	Comparaison de la carte mémoire avec l'UC	Fichiers requis lors de la restauration des données
<b>UC</b>	<b>CS/CJ</b>				
BACKUP.OBJ	Programme utilisateur complet	Oui	Oui	Oui	Requis sur la carte mémoire

**Fichiers de paramètres**

Nom de fichier et extension	Contenu	Sauvegarde de la mémoire E/S sur la carte mémoire (création de fichiers)	Restauration de la carte mémoire vers l'UC	Comparaison de la carte mémoire avec l'UC	Fichiers requis lors de la restauration des données
<b>UC</b>	<b>CS/CJ</b>				
BACKUP.STD	Configuration de l'API Tables d'E/S enregistrées Tableaux de routage Configuration de carte réseau Etc.	Oui	Oui	Oui	Requis sur la carte mémoire

**Fichiers de sauvegarde d'unité/carte (UC CS1-H, CS1D, CJ1-H, CJ1M ou CS1D uniquement)**

Nom de fichier et extension	Contenu	Sauvegarde de la mémoire E/S sur la carte mémoire (création de fichiers)	Restauration de la carte mémoire vers l'UC	Comparaison de la carte mémoire avec l'UC	Fichiers requis lors de la restauration des données
<b>UC</b>	<b>UC CS1-H, CS1D, CJ1-H, CJ1M ou CS1D uniquement</b>				
BACKUP□□.PRM (où □□ désigne l'adresse d'unité de l'unité/carte sauvegardée)	Les données de sauvegarde de l'unité ou de la carte correspondant à l'adresse d'unité spécifiée (le contenu spécifique dépend de l'unité ou de la carte).	Oui	Oui	Oui	Requis sur la carte mémoire (Voir remarque 2.)

**Remarque**

1. Les adresses d'unité sont les suivantes :  
Cartes réseau : Numéro de carte + 10 hex.  
Cartes d'E/S spéciales : Numéro de carte + 20 hex.  
Carte interne : E1 hex.
2. Aucune erreur ne se produira dans l'UC, même si ce fichier est absent lors du transfert des données de la carte mémoire vers la mémoire E/S, mais une erreur aura lieu sur l'unité ou la carte si les données ne sont pas restaurées. Pour plus d'informations sur les erreurs d'unité ou de carte, reportez-vous au manuel de l'unité ou de la carte.

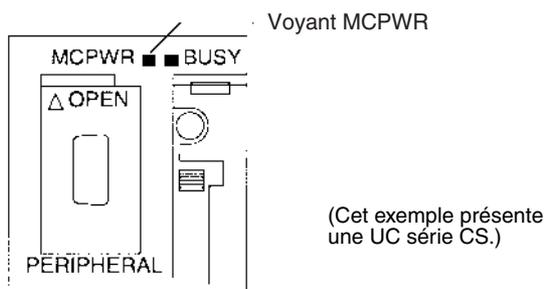
**Tables de symboles, fichiers de commentaires, fichiers d'index de programmes (UC CS1-H/CJ1-H, CJ1M, CS1D avec version de carte 3.0 ou supérieure uniquement)**

Nom de fichier et extension	Sommaire	Sauvegarde de l'UC sur la carte mémoire (création de fichiers)	Restauration de la carte mémoire vers l'UC	Comparaison de la carte mémoire avec l'UC	Fichiers requis lors de la restauration des données
<b>UC CS1-H, CJ1-H, CJ1M ou CS1D CPU avec version de carte 3.0 uniquement</b>					
BKUPSYM.SYM	Fichiers de tables de symboles	Oui (écriture si la mémoire des commentaires de l'UC contient des fichiers)	Oui (lecture dans la mémoire des commentaires de l'UC si la carte mémoire contient des fichiers)	Oui (comparaison avec les fichiers contenus dans la mémoire des commentaires de l'UC)	Ces fichiers ne sont pas absolument indispensables sur la carte mémoire.
BKUPCMT.CMT	Fichiers de commentaires				
BKUPPRG.IDX	Fichiers d'index de programmes				

**Remarque** Les opérations de sauvegarde, de restauration et de comparaison dans le tableau ci-dessus sont exécutées uniquement pour les fichiers contenus dans la mémoire des commentaires.

**Vérification des opérations de sauvegarde avec les voyants**

L'état du voyant d'alimentation de la carte mémoire (MCPWR) indique si une opération de sauvegarde simple s'est déroulée correctement ou non.



Opération de sauvegarde	Déroulement normal (Voir remarque 1).	Erreur survenue	
	Etat de MCPWR	Etat de MCPWR	Erreur
Sauvegarde de données de l'UC sur la carte mémoire	Allumé → Reste allumé pendant que vous appuyez sur l'interrupteur d'alimentation de la carte mémoire. → Clignote une fois. → Allumé pendant l'écriture. → Eteint après l'écriture des données.	Allumé → Reste allumé pendant que vous appuyez sur l'interrupteur d'alimentation de la carte mémoire. → Continue de clignoter. → S'allume lorsque vous appuyez sur l'interrupteur d'alimentation de la carte mémoire.	Aucun fichier ne sera créé avec les erreurs suivantes : Capacité de la carte mémoire insuffisante (Voir remarque 2.) Erreur mémoire dans l'UC Erreur de bus E/S (lors de l'écriture de données sur une unité ou une carte, UC CS1-H, CS1D, ou CJ1-H uniquement)
Restauration des données de la carte mémoire sur l'UC.	Allumé lors de la mise sous tension. → Clignote une fois. → Allumé pendant la lecture. → Eteint après la lecture des données.	Allumé lors de la mise sous tension. → Clignote à cinq reprises. → S'éteint.	Les données ne seront pas lues avec les erreurs suivantes : Le programme sur la carte mémoire dépasse la capacité de l'UC Les fichiers de sauvegarde requis n'existent pas sur la carte mémoire. Le programme ne peut pas être écrit car il est protégé en écriture (La broche 1 de l'interrupteur DIP est sur ON.)
		Allumé lors de la mise sous tension. → Clignote une fois. → Allumé pendant la lecture. → Clignote à trois reprises. → Eteint après la lecture des données.	Attention : les données seront pas lues avec l'erreur suivante. Les fichiers EM et les banques EM de l'UC ne correspondent pas (numéros de banques non consécutifs ou non-concordance du numéro de banque max.)

Opération de sauvegarde	Déroulement normal (Voir remarque 1).	Erreur survenue	
	Etat de MCPWR	Etat de MCPWR	Erreur
Comparaison de données entre la carte mémoire et l'UC	Allumé → Reste allumé pendant que vous appuyez sur l'interrupteur d'alimentation de la carte mémoire. → Clignote une fois. → Allumé pendant la comparaison. → Eteint après la comparaison des données.	Allumé → Reste allumé pendant que vous appuyez sur l'interrupteur d'alimentation de la carte mémoire. → Continue de clignoter. → S'allume lorsque vous appuyez sur l'interrupteur d'alimentation de la carte mémoire.	Les erreurs de comparaison suivantes peuvent survenir (voir remarque 3) : La carte mémoire et l'UC ne correspondent pas. Les fichiers de sauvegarde requis n'existent pas sur la carte mémoire. Les fichiers EM et les banques EM de l'UC ne correspondent pas (numéros de banques non consécutifs ou non-concordance du numéro de banque max.). Erreur mémoire dans l'UC Erreur de bus E/S (lors de la comparaison des données sur une unité ou une carte, UC CS1-H, CS1D, ou CJ1-H uniquement)
Commun à toutes les opérations de sauvegarde.	---	Lecture : Clignote à cinq reprises. → S'éteint. Ecriture ou comparaison : Continue de clignoter. → S'allume lorsque vous appuyez sur l'interrupteur d'alimentation de la carte mémoire.	Erreur d'accès à la carte mémoire (erreur de formatage ou erreur de lecture/écriture)

**Remarque**

1. Une fois l'opération de sauvegarde terminée, l'alimentation de la carte mémoire est coupée lorsque le voyant MCPWR s'éteint. Si vous devez réutiliser la carte mémoire, appuyez sur son interrupteur d'alimentation pour fournir l'alimentation et exécutez l'opération voulue.
2. Lorsque les données sont écrites pour une opération de sauvegarde simple à partir d'une UC CS1-H, CS1D, CJ1-H, CJ1M ou CS1D, les erreurs dues à une capacité insuffisante de la carte mémoire peuvent être contrôlées dans A397 (capacité en écriture de sauvegarde simple). Si A397 contient une valeur autre que 0000 hex. après l'exécution de l'opération d'écriture, la valeur indiquera la capacité requise sur la carte mémoire en kilo-octets.
3. Avec les UC CS1-H, CS1D, CJ1-H, CJ1M ou CS1D, les fichiers de sauvegarde des unités et des cartes sont également comparés.

**Bits auxiliaires/mots associés**

Nom	Adresse	Description
Drapeau de fonctionnement de la mémoire de fichiers	A34313	A ON lorsque l'une des opérations suivantes est exécutée. Sur OFF lorsque l'exécution est terminée. <ul style="list-style-type: none"> <li>Détection de carte mémoire</li> <li>Instruction CMND exécutée pour l'UC locale</li> <li>Instructions FREAD/FWRIT</li> <li>Remplacement de programme via bits de contrôle spéciaux</li> <li>Opération de sauvegarde simple</li> </ul> Il n'est pas possible d'écrire des données ou de vérifier le contenu de la carte mémoire lorsque ce drapeau est à ON.
Banque de démarrage de la mémoire de fichiers EM	A344	Lorsque l'UC commence la lecture à partir de la carte mémoire, elle fait référence à cette valeur. Si le numéro de banque EM maximum des fichiers BACKUPE□.IOM (numéro de banque consécutif maximum en partant de 0) correspond au numéro de banque maximum de l'UC, la zone EM sera formatée en fonction de la valeur de ce mot. Si ces numéros ne coïncident pas, la zone EM reviendra à son état non formaté.
Drapeaux d'activation d'instruction de communications réseau (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement) (Voir remarque.)	A20200 à A20207	<ul style="list-style-type: none"> <li>Passe à OFF lorsque l'écriture ou la comparaison des données de la carte mémoire commence.</li> <li>Passe à ON lorsque l'écriture ou la comparaison des données de la carte mémoire est terminée.</li> </ul> Les données de l'unité et de la carte ne peuvent pas être écrites ou comparées si tous les drapeaux d'activation d'instruction de communications réseau sont à OFF au démarrage des opérations d'écriture ou de comparaison sur la carte mémoire. Toute tentative provoque une erreur.
Code de fin de communications réseau (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement) (Voir remarque.)	A203 à A210	Fournit les résultats des communications avec l'unité ou la carte lors de l'exécution des opérations d'écriture ou de comparaison sur la carte mémoire.
Drapeaux d'erreurs de communications réseau (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement) (Voir remarque.)	A21900 à A21907	<ul style="list-style-type: none"> <li>Passe à ON si une erreur se produit dans les communications avec l'unité ou la carte lors de l'exécution des opérations d'écriture ou de comparaison sur la carte mémoire.</li> <li>Reste à OFF (ou passe à OFF) si aucune erreur ne se produit dans les communications avec l'unité ou la carte lors de l'exécution des opérations d'écriture ou de comparaison sur la carte mémoire.</li> </ul>
Capacité d'écriture de sauvegarde simple (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement)	A397	Fournit la capacité de données en kilo-octets qui serait nécessaire sur la carte mémoire lorsque l'écriture échoue pour un opération de sauvegarde simple, indiquant qu'une erreur d'écriture s'est produite en raison d'une capacité insuffisante. 0001 à FFFF hex. : erreur d'écriture (Indique la capacité de la carte mémoire requise, entre 1 et 65 535 kilo-octets.) (Mis à 0000 hex. lorsqu'une écriture s'effectue correctement.) 0000 hex. : écriture terminée normalement.

**Remarque** Ces drapeaux sont apparentés pour les UC CS1-H, CJ1-H, CJ1M ou CS1D, car l'UC utilisera automatiquement un port de communication disponible lors de l'écriture ou de la comparaison de données pour une carte mémoire.

**Sauvegarde des données d'une carte ou d'une unité**

Cette fonction est prise en charge uniquement par les UC CS1-H, CJ1-H, CJ1M et CS1D.

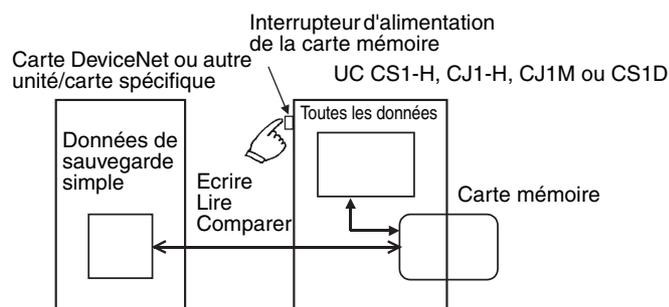
**Introduction**

Les données suivantes sont sauvegardées à partir de l'UC par les UC CS1 et CJ1 pour l'opération de sauvegarde simple : programme utilisateur, zone de paramètres, mémoire E/S entière. Outre les données ci-dessus, les données suivantes sont également sauvegardées pour les UC CS1-H, CJ1-H, CJ1M ou CS1D : données d'unités et de cartes spécifiques montées sur l'API.

**Aperçu**

Lorsque vous utilisez l'opération de sauvegarde simple pour une UC CS1-H, CJ1-H, CJ1M ou CS1D, un fichier de sauvegarde d'unité/de carte contenant

les données d'unités et de cartes spécifiques est écrit sur la carte mémoire. Les données sont sauvegardées séparément pour chaque unité et carte.



**Application**

Cette fonction permet de sauvegarder les données pour l'ensemble de l'API, y compris l'UC, les cartes DeviceNet, les unités/cartes de communications série, etc. Elle peut également être utilisée pour le remplacement des cartes.

**Fichiers de sauvegarde d'unité/de carte**

Les données de chaque unité/carte sont stockées sur la carte mémoire avec les noms de fichiers suivants : BACKUP□□.PRM « □□ » désigne l'adresse d'unité de l'unité ou de la carte en hexadécimal.

**Remarque**

Les adresses d'unité sont les suivantes :  
 Cartes réseau : Numéro de carte + 10 hex.  
 Cartes d'E/S spéciales : Numéro de carte + 20 hex.  
 Carte interne : E1 hex.

Ces fichiers sont également utilisés pour lire sur la carte mémoire ou comparer les données de la carte mémoire.

**Unités et cartes utilisables**

Pour que les données de l'unité et de la carte puissent être sauvegardées, l'unité/la carte doit également prendre en charge la fonction de sauvegarde. Pour plus de détails sur la prise en charge, consultez le manuel d'utilisation de l'unité/la carte.

Unité/carte	Numéros de modèle	Données sauvegardées pour une sauvegarde simple avec une UC CS1-H/CJ1-H	Capacité de données utilisée sur la carte mémoire pour une sauvegarde simple
Cartes DeviceNet	CJ1W-DRM21-V1 CJ1W-DRM21	Paramètres de périphérique (toutes les données dans l'EEPROM de la carte)  (Bien qu'il s'agisse des mêmes données que celles sauvegardées avec la fonction de sauvegarde de carte mémoire prise en charge par la carte ou la Configuration DeviceNet (Ver. 2.0), les fichiers ne sont pas compatibles.)	7 kilo-octets
Cartes de communications série	CS1W-SCU21-V1 CJ1W-SCU41	Données de macro protocole  (y compris les protocoles système standard et les protocoles définis par l'utilisateur de la mémoire Flash sur l'unité ou la carte)	129 kilo-octets
Cartes de communications série	CS1W-SCB21-V1 CS1W-SCB41-V1		129 kilo-octets

Unité/carte	Nombres de modèle	Données sauvegardées pour une sauvegarde simple avec une UC CS1-H/CJ1-H	Capacité de données utilisée sur la carte mémoire pour une sauvegarde simple
Unités de compteur personnalisables	CS1W-HIO01-V1 CS1W-HCP22-V1 CS1W-HCA22-V1 CS1W-HCA12-V1	<ul style="list-style-type: none"> <li>• Programme utilisateur</li> <li>• DM universelle en lecture seule</li> <li>• Zone de paramètres de la fonction de carte</li> <li>• Informations sur les instructions d'extension</li> <li>• Bibliothèque de schémas contact</li> </ul>	64 kilo-octets
Unités de commande de mouvement	CS1W-MCH71	<ul style="list-style-type: none"> <li>• Données de positionnement</li> <li>• Paramètres système</li> <li>• Programmes en langage G</li> </ul>	8 192 kilo-octets
	CS1W-MC221-V1 CS1W-MC421-V1		142 kilo-octets
Unités de commande de position	CS1W-NC113/133/213/233/413/433 Ver. 2.0 ou supérieure CJ1W-NC113/133/213/233/413/433 Ver. 2.0 ou supérieure	<ul style="list-style-type: none"> <li>• Paramètres d'axe</li> <li>• Données de séquence</li> <li>• Données de vitesse</li> <li>• Données du temps d'accélération/décélération</li> <li>• Données de double temporisation</li> <li>• Données de zone</li> </ul>	7 kilo-octets

**Remarque** Les données des unités et des cartes répertoriées ci-dessus seront automatiquement sauvegardées pour l'opération de sauvegarde simple. Il n'existe aucun paramètre disponible pour les inclure ou les exclure. Si vous utilisez une console de programmation, cependant, les opérations sont prises en charge individuellement pour la zone de programme utilisateur, la zone de paramètres et les zones de mémoire E/S. Consultez le *Manuel d'utilisation de la console de programmation (W314)* pour plus de détails.

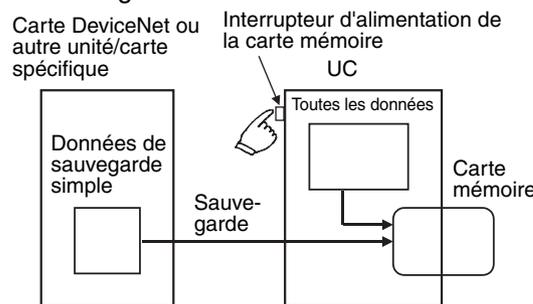
**Procédure**

La procédure associée à l'opération de sauvegarde simple est la même, que les données soient ou non sauvegardées à partir d'unités et de cartes spécifiques (y compris l'écriture, la lecture et la comparaison).

■ **Sauvegarde de données**

- 1,2,3... 1. Mettez la broche 7 de l'interrupteur DIP de l'UC à ON.  
 2. Appuyez sur l'interrupteur d'alimentation de la carte mémoire et maintenez-le enfoncé pendant trois secondes.

Les données de sauvegarde pour les unités et les cartes seront créées dans un fichier et stockées sur la carte mémoire avec les autres données de sauvegarde.

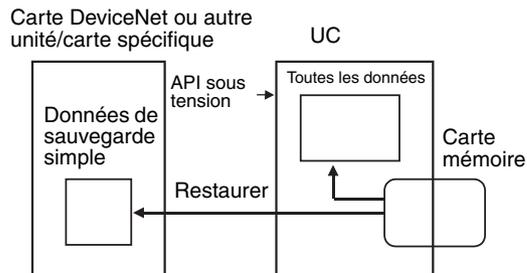


Lorsque vous appuyez sur l'interrupteur d'alimentation, le voyant MCPWR clignote une fois, s'allume pendant l'opération d'écriture, puis s'éteint si l'opération s'est déroulée correctement.

■ **Restauration de données**

- 1,2,3... 1. Mettez la broche 7 de l'interrupteur DIP de l'UC à ON.  
 2. Mettez l'API sous tension. Les fichiers de sauvegarde seront restaurés sur les unités et les cartes.

Les données de sauvegarde des unités et des cartes seront restaurées de la carte mémoire sur les unités et les cartes.

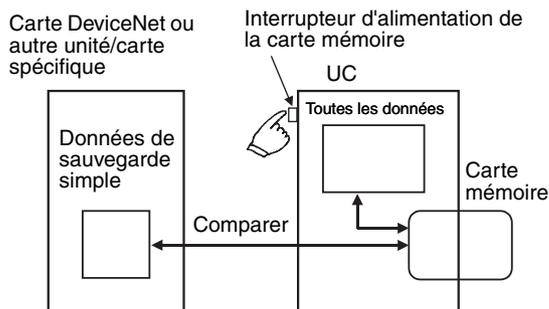


Lorsque le système est sous tension, le voyant MCPWR clignote une fois, s'allume pendant l'opération de lecture, puis s'éteint si l'opération s'est déroulée correctement.

■ **Comparaison de données**

- 1,2,3... 1. Mettez à OFF la broche 7 de l'interrupteur DIP de l'UC.  
 2. Appuyez sur l'interrupteur d'alimentation de la carte mémoire et maintenez-le enfoncé pendant trois secondes.

Les données de sauvegarde sur la carte mémoire seront comparées aux données sur les unités et les cartes.



Lorsque vous appuyez sur l'interrupteur d'alimentation, le voyant MCPWR clignote une fois, s'allume pendant l'opération de comparaison, puis s'éteint si l'opération s'est déroulée correctement et que les données sont les mêmes.

**Remarque** Le délai requis pour une opération de sauvegarde simple est plus long avec le CS1W-SCU21-V1 que sans. La différence est indiquée dans le tableau suivant.

**Délai supplémentaire lorsque le fichier de paramètres de la carte réseau (BACKUP□□.PRM) sur la carte mémoire est de 60 kilo-octets**

Mode de fonctionnement	Délai supplémentaire lors de l'écriture sur une carte mémoire	Délai supplémentaire lors de la vérification d'une carte mémoire	Délai supplémentaire lors de la lecture sur une carte mémoire
PROGRAM	25 s environ	10 s environ	4 s environ
RUN	1 min 30 s environ	30 s environ	4 s environ

**Temps supplémentaire lorsque le fichier de paramètres de la carte réseau (BACKUP□□.PRM) sur la carte mémoire est de 128 kilo-octets**

Mode de fonctionnement	Délai supplémentaire lors de l'écriture sur une carte mémoire	Délai supplémentaire lors de la vérification d'une carte mémoire	Délai supplémentaire lors de la lecture à partir d'une carte mémoire
PROGRAM	40 s environ	14 s environ	8 s environ
RUN	2 min 30 s environ	1 min env.	8 s environ

**Remarque**

1. Avant d'essayer d'exécuter les opérations ci-dessus, assurez-vous que les unités et les cartes fonctionnent correctement. Si ce n'est pas le cas, les opérations d'écriture, de lecture et de comparaison ne seront pas exécutées.
2. Avant d'exécuter une opération de sauvegarde simple pour des unités/cartes spécifiques, assurez-vous que l'UC est en mode PROGRAM ou que l'exécution de l'opération n'aura pas un impact négatif sur les instructions qui utilisent des numéros de ports de communication. Lorsque les données sont sauvegardées à partir d'unités/cartes spécifiques, un port de communication est recherché à partir du port 0 et le premier port disponible est utilisé. Si le numéro de port est le même que celui utilisé par une instruction de communication réseau, cette instruction ne sera pas exécutée avant la fin de l'opération de sauvegarde simple.

**Précautions lorsque vous changez l'état des affectations d'E/S pendant les opérations de sauvegarde/restauration**

L'état des affectations d'E/S dépend des versions de carte des UC source et destination lorsque vous utilisez une seule UC série CJ pour créer les fichiers des paramètres de sauvegarde (BKUP.STD), les enregistrer sur la carte mémoire et les sauvegarder ou les restaurer ensuite sur une autre UC. Le tableau suivant présente les changements d'état des affectations d'E/S pour les différentes combinaisons de versions de cartes.

UC source		Etat initial des affectations d'E/S	UC destination		
			Version de carte de l'UC à la destination de la sauvegarde/restauration		
			Pré-ver. 2.0	Version de carte 2.0	Version de carte 3.0 ou supérieure
Version de carte de l'UC à la source de la sauvegarde	Pré-ver. 2.0	Affectation automatique	(Idem) Affectation automatique		
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur		
	Version de carte 2.0	Affectation automatique	(Idem) Affectation automatique		
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur		
	Version de carte 3.0 ou supérieure	Affectation automatique	(Idem) Affectation automatique		
		Spécifié par l'utilisateur	(Idem) Spécifié par l'utilisateur		

**Précautions en cas de non-correspondance lors de la comparaison des sauvegardes**

Si une UC série CJ avec une version de carte 2.0 ou antérieure est associée à une UC série CJ avec une version de carte version 3.0, des erreurs de vérification peuvent se produire lorsque les données des paramètres sont comparées avec les données de sauvegarde restaurées à partir du fichier de sauvegarde simple qui a été créé.

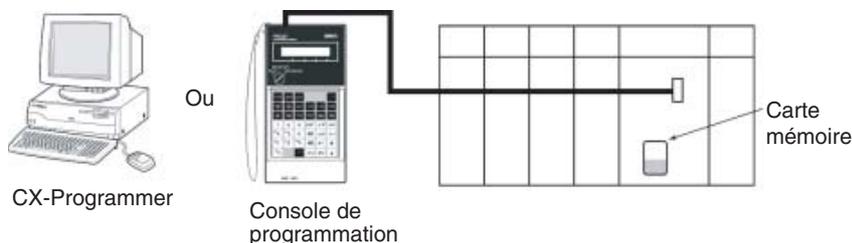
UC source		Etat précédent des affectations d'E/S	UC destination		
			Version de carte de l'UC à la destination de la sauvegarde/restauration		
			Pré-ver. 2.0	Version de carte 2.0	Version de carte 3.0 ou supérieure
Version de carte de l'UC à la source de la sauvegarde	Pré-ver. 2.0	Affectation automatique	Vérification possible	Vérification possible	Erreur de vérification
		Spécifié par l'utilisateur			
	Version de carte 2.0	Affectation automatique	Vérification possible		Vérification possible
		Spécifié par l'utilisateur			
	Version de carte 3.0 ou supérieure	Affectation automatique	Erreur de vérification	Erreur de vérification	Vérification possible
		Spécifié par l'utilisateur	Vérification possible	Vérification possible	

## 5-3 Utilisation de la mémoire de fichiers

### 5-3-1 Initialisation des supports

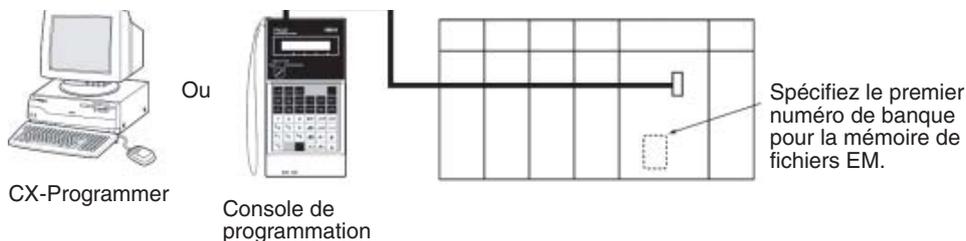
#### Cartes mémoire

- 1,2,3... 1. Utilisez un périphérique de programmation, telle une console de programmation, pour initialiser les cartes mémoire.

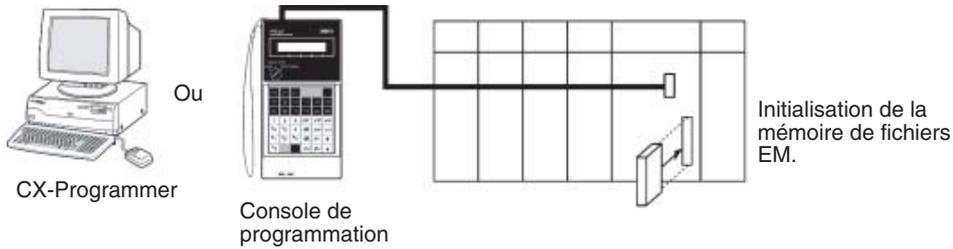


#### Mémoire de fichiers EM

- 1,2,3... 1. Utilisez un périphérique de programmation, telle qu'une console de programmation, et définissez les paramètres de la mémoire de fichiers EM dans la configuration de l'API de manière à activer la mémoire de fichiers EM, puis définissez le numéro de banque spécifié pour la mémoire de fichiers EM sur une valeur comprise entre 0 et C hex.



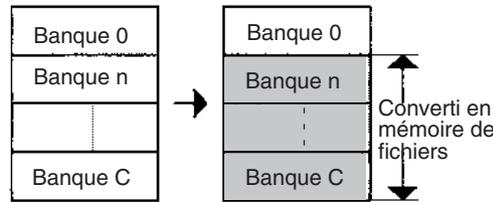
2. Utilisez une commande FINS ou un périphérique de programmation autre qu'une console de programmation pour initialiser la mémoire de fichiers EM.



**Initialisation d'une mémoire de fichiers EM individuelle**

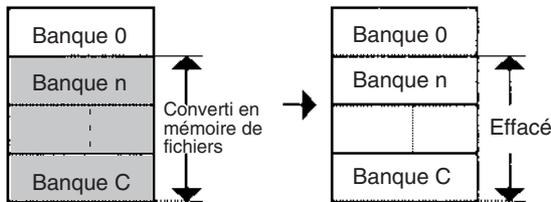
Une banque EM spécifiée peut être convertie d'un EM ordinaire en mémoire de fichiers.

**Remarque** Le numéro de banque maximum pour les UC série CJ est 6.



1. Définissez n dans la configuration de l'API.
2. Utilisez un périphérique de programmation ou une commande FINS pour formater à partir de n.
3. "n" est sauvegardé dans A344.

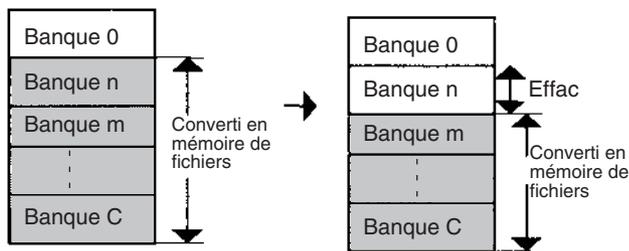
L'EM utilisé pour la mémoire de fichiers peut être restauré à l'état d'EM normal.



1. Mettez la mémoire de fichiers à OFF dans la configuration de l'API.
2. Si un périphérique de programmation ou une commande FINS est utilisé pour le formatage, la mémoire sera remise à 0000 Hex à partir de n.
3. FFFF Hex sera sauvegardé dans A344 pour indiquer qu'il n'existe pas de mémoire de fichiers EM.

**Remarque :** 1. Toutes les données de fichiers présentes seront alors supprimées.  
2. Seules les banques 0 à 6 peuvent être spécifiées pour une UC série CJ.

Le numéro de la première banque peut être modifié.



1. Remplacez n par m dans la configuration de l'API.
2. Utilisez un périphérique de programmation ou une commande FINS pour convertir les banques à partir de m en mémoire de fichiers.

**Remarque :** les banques n à m-1 seront remises à 0000 Hex.

3. m sera sauvegardé dans A344.

**Remarque :** 1. Toutes les données de fichiers présentes seront alors supprimées.

2. Seules les banques 0 à 6 peuvent être spécifiées pour une UC série CJ.

**Configuration de l'API**

Adresse	Nom	Description	Paramètre initial
136	Banque de démarrage de la mémoire de fichiers EM	0000 hex. : aucune 0080 hex. : commence à la banque N° 0 008C hex : banque N° C la zone EM qui commence au numéro de banque spécifié sera convertie en mémoire de fichiers.	0000 hex.

## Relais auxiliaire spécial associé

Nom	Adresse	Description
Banque de démarrage de la mémoire de fichiers EM	A344	Le numéro de banque qui correspond au début de la zone de mémoire de fichiers EM sera enregistrée. Le fichier EM sera converti en mémoire de fichiers du premier numéro de banque au dernier. FFFF hex. indiquera qu'il n'existe aucune mémoire de fichiers EM.

## Lecture/écriture des tableaux de symboles et des commentaires avec CX-Programmer

Procédez comme suit pour transférer les tableaux de symboles ou les commentaires créés sur CX-Programmer vers et à partir d'une carte mémoire ou d'une mémoire de fichiers EM.

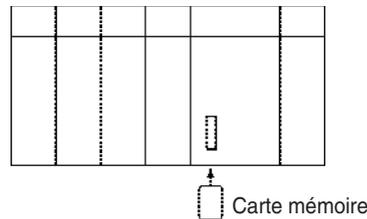
- 1,2,3...**
1. Insérez une carte mémoire formatée dans l'UC ou formatez la mémoire de fichiers EM.
  2. Mettez CX-Programmer en ligne.
  3. Sélectionnez **Transfer**, puis **To PLC** ou **From PLC** dans le menu de l'API.
  4. Sélectionnez **Symbols** ou **Comments** pour les données à transférer.

## 5-3-2 Procédures de fonctionnement

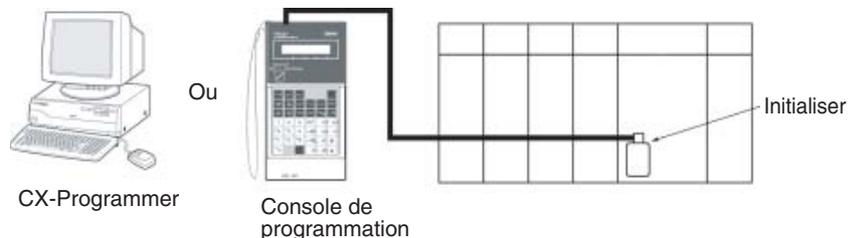
Cartes mémoire

## Utilisation d'un périphérique de programmation

- 1,2,3...**
1. Insérez une carte mémoire dans l'UC.



2. Initialisez la carte mémoire à l'aide d'un périphérique de programmation.



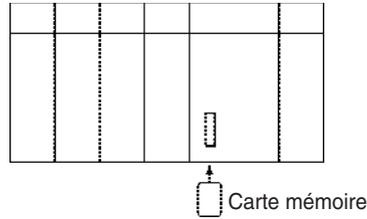
3. Utilisez un périphérique de programmation pour nommer les données de l'UC (programme utilisateur, mémoire E/S, zone de paramètres), puis sauvegardez les données sur la carte mémoire. (Utilisez un périphérique de programmation pour lire le fichier de carte mémoire sur l'UC.)

**Transfert automatique des fichiers au démarrage**

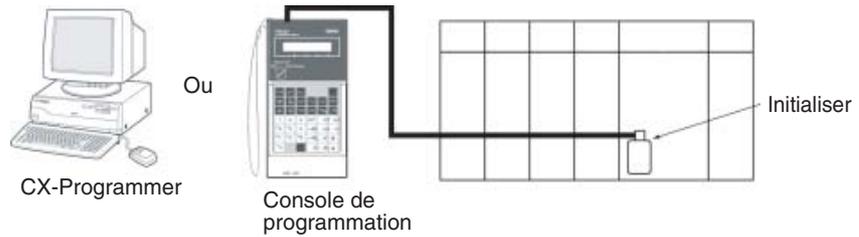
**Procédure pour le transfert d'un fichier de zone de paramètres**

1,2,3...

1. Insérez une carte mémoire initialisée dans l'UC.



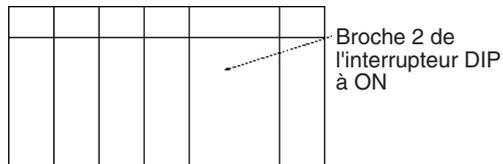
2. Utilisez un périphérique de programmation pour écrire les fichiers de transfert automatique au démarrage sur la carte mémoire. Ces fichiers incluent le fichier de programme (AUTOEXEC.OBJ), le fichier de zone de paramètres (AUTOEXEC.STD) et les fichiers de mémoire E/S (AUTOEXEC.IOM, ATEXECDM.IOM et/ou ATEXECE□.IOM).



**Remarque**

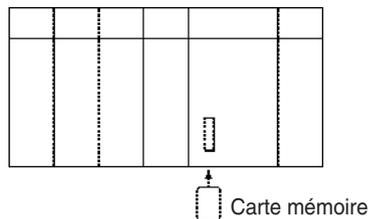
Un fichier de programme utilisateur et de zone de paramètres doit être présent sur la carte mémoire.

3. Mettez l'API hors tension.
4. Placez la broche 2 de l'interrupteur DIP sur ON (transfert automatique au démarrage).



**Remarque** Si la broche 7 est sur ON et la broche 8 sur OFF, la fonction de sauvegarde sera activée et aura la priorité sur la fonction de transfert automatique au démarrage. Mettez les broches 7 et 8 sur OFF pour le transfert automatique au démarrage.

5. Insérez la carte mémoire dans l'UC.



6. Mettez l'API sous tension pour lire le fichier.

**Procédure sans transfert d'un fichier de zone de paramètres**

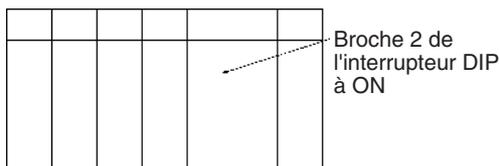
1,2,3...

1. Insérez une carte mémoire initialisée dans l'UC.
2. Utilisez un périphérique de programmation pour écrire les fichiers de transfert automatique au démarrage sur la carte mémoire. Ces fichiers

incluent le fichier de programme (REPLACE.OBJ) et les fichiers de mémoire E/S (REPLACE.IOM, REPLCDM.IOM et/ou REPLCDE□.IOM).

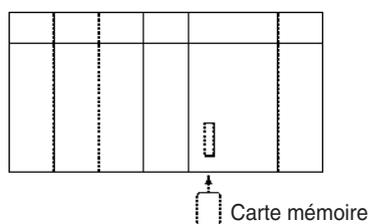
**Remarque** Une zone de paramètres ne sera pas transférée, même s'il en existe une sur la carte mémoire.

3. Mettez l'API hors tension.
4. Placez la broche 2 de l'interrupteur DIP sur ON (transfert automatique au démarrage).



**Remarque** Si la broche 7 est sur ON et la broche 8 sur OFF, la fonction de sauvegarde sera activée et aura la priorité sur la fonction de transfert automatique au démarrage. Mettez les broches 7 et 8 sur OFF pour le transfert automatique au démarrage.

5. Insérez la carte mémoire dans l'UC.



6. Mettez l'API sous tension pour lire le fichier.

#### Utilisation de FREAD(700)/FWRITE(701)/CMND(490)

1,2,3...

1. Insérez une carte mémoire dans l'UC. (Déjà initialisée.)
2. Utilisez FWRITE(701) pour nommer le fichier dans la zone spécifiée de la mémoire E/S, puis sauvegardez le fichier sur la carte mémoire.

**Remarque** Une carte mémoire contenant des fichiers de données TXT ou CSV peut être installée dans l'emplacement de cartes API d'un PC avec un adaptateur de carte mémoire HMC-AP001 et les fichiers de données peuvent être lus dans un tableur avec les fonctions Windows standard (non pris en charge par les UC CS1 série CS antérieures à EV1).

3. Utilisez FREAD(700) pour lire le fichier de la carte mémoire dans la mémoire E/S de l'UC.

Vous pouvez exécuter des opérations sur les fichiers de la carte mémoire en émettant des commandes FINS sur l'UC locale avec CMND(490) (non pris en charge par les UC CS1 série CS antérieures à EV1).

#### Remplacement du programme en cours d'opération

1,2,3...

1. Insérez une carte mémoire dans l'UC. (Déjà initialisée.)
2. Écrivez le mot de passe du programme (A5A5 hex.) dans A651 et le nom du fichier de programme dans les mots A654 à A657.
3. Faites passer le drapeau de démarrage de remplacement (A65015) de OFF à ON.

#### Fonction de sauvegarde simple

Il existe 3 opérations de sauvegarde : sauvegarde de données sur la carte mémoire, restauration de données à partir de la carte mémoire et comparaison de données avec la carte mémoire.

##### Sauvegarde de données de l'UC sur la carte mémoire

1,2,3...

1. Insérez une carte mémoire dans l'UC. (Déjà initialisée.)
2. Mettez la broche 7 de l'interrupteur DIP de l'UC à ON et la broche 8 à OFF.

3. Appuyez sur l'interrupteur d'alimentation de la carte mémoire et maintenez-le enfoncé pendant trois secondes.
4. Vérifiez que le voyant MCPWR clignote une fois, puis s'éteint. (Les autres changements indiquent qu'une erreur s'est produite pendant la sauvegarde des données.)

#### Restauration des données de la carte mémoire sur l'UC.

1,2,3...

1. Insérez la carte mémoire contenant les fichiers de sauvegarde dans l'UC.
2. Mettez la broche 7 de l'interrupteur DIP de l'UC à ON et la broche 8 à OFF.
3. Les fichiers de sauvegarde seront restaurés lorsque l'API sera mis sous tension.
4. Vérifiez que le voyant MCPWR clignote une fois, puis s'éteint. (Les autres changements indiquent qu'une erreur s'est produite pendant la restauration des données.)

#### Comparaison des données sur la carte mémoire et sur l'UC

1,2,3...

1. Insérez la carte mémoire contenant les fichiers de sauvegarde dans l'UC.
2. Mettez les broches 7 et 8 de l'interrupteur DIP de l'UC à OFF.
3. Appuyez sur l'interrupteur d'alimentation de la carte mémoire et maintenez-le enfoncé pendant trois secondes.
4. Les données correspondent si le voyant MCPWR clignote une fois, puis s'éteint.

#### Remarque

Le voyant MCPWR clignote si une erreur survient pendant l'écriture ou la comparaison des données. Ce clignotement s'arrête et le voyant MCPWR s'allume lorsque vous appuyez sur l'interrupteur d'alimentation de la carte mémoire.

Le tableau suivant indique la durée des opérations de sauvegarde avec un programme de 20 Kpas et un temps de cycle de 10 ms en mode RUN :

Mode	Sauvegarde	Restauration	Comparaison
PROGRAM	50 s environ	30 s environ	7 s environ
RUN	5 min env.	2 min env.	7 s environ

Le tableau suivant indique la durée des opérations de sauvegarde avec un programme de 30 Kpas et un temps de cycle de 10 ms en mode RUN :

Mode	Sauvegarde	Restauration	Comparaison
PROGRAM	50 s environ	30 s environ	7 s environ
RUN	5 min 3 s environ	2 min 40 s environ	7 s environ

Le tableau suivant indique la durée des opérations de sauvegarde avec un programme de 250 Kpas et un temps de cycle de 12 ms en mode RUN :

Mode	Sauvegarde	Restauration	Comparaison
PROGRAM	1 min 30 s environ	1 min 30 s environ	20 s environ
RUN	13 min env.	7 min 30 s environ	20 s environ

#### Création de la table des variables et des fichiers de commentaires

1,2,3...

Utilisez la procédure du CX-Programmer suivante pour créer des fichiers de table de variables ou des fichiers de commentaires sur les cartes mémoire ou dans la mémoire de fichiers EM.

1. Insérez une carte mémoire formatée dans l'UC ou formatez la mémoire de fichiers EM.
2. Mettez CX-Programmer en ligne.
3. Sélectionnez **Transfer**, puis **To PLC** ou **From PLC** dans le menu de l'API.
4. Sélectionnez **Symbols** ou **Comments** pour les données à transférer.

**Remarque** Si une carte mémoire est installée dans l'UC, les données ne peuvent être transférées qu'avec la carte mémoire. (Cela ne sera pas possible avec la mémoire de fichiers EM.)

## **Mémoire de fichiers EM**

### **Utilisation d'un périphérique de programmation**

- 1,2,3...**
1. Utilisez la configuration de l'API pour spécifier la première banque EM à convertir en mémoire de fichiers.
  2. Utilisez un périphérique de programmation pour initialiser la mémoire de fichiers EM.
  3. Utilisez un périphérique de programmation pour nommer les données de l'UC (programme utilisateur, mémoire E/S, zone de paramètres), puis sauvegardez les données dans la mémoire de fichiers EM.
  4. Utilisez un périphérique de programmation pour lire le fichier de la mémoire de fichiers EM sur l'UC.

### **Utilisation de FREAD(700)/FWRITE(701)/CMND(490)**

- 1,2,3...**
1. Utilisez la configuration de l'API pour spécifier la première banque EM à convertir en mémoire de fichiers.
  2. Utilisez un périphérique de programmation pour initialiser la mémoire de fichiers EM.
  3. Utilisez FWRITE(701) pour nommer le fichier dans la zone spécifiée de la mémoire E/S, puis sauvegardez le fichier dans la mémoire de fichiers EM.
  4. Utilisez FREAD(700) pour lire le fichier de la mémoire de fichiers EM dans la mémoire E/S de l'UC.

Vous pouvez exécuter des opérations sur la mémoire de fichiers EM en émettant des commandes FINS sur l'UC avec CMND(490).

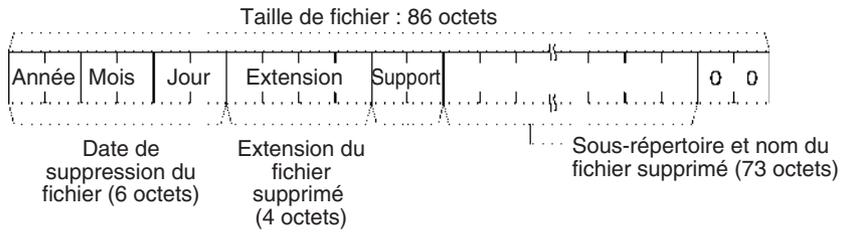
## **5-3-3 Interruptions de l'alimentation pendant l'accès à la mémoire de fichiers**

Si une interruption de l'alimentation se produit pendant que l'UC accède à la mémoire de fichiers (la carte mémoire ou la mémoire de fichiers EM), le contenu de la carte mémoire risque de ne pas être exact. Le remplacement du fichier mis à jour peut ne pas se dérouler correctement et, dans certains cas, la carte mémoire proprement dite peut être endommagée.

Le fichier affecté sera supprimé automatiquement par le système à la prochaine mise sous tension. Le drapeau de notification de suppression de fichier correspondant (A39507 pour la carte mémoire, A39506 pour la mémoire de fichiers EM) passera à ON. Le drapeau passera à OFF lors de la prochaine mise hors tension.

Lorsqu'un fichier sera supprimé, un fichier journal de suppression (DEL\_FILE.IOM) sera créé dans le répertoire racine de la carte mémoire ou de la mémoire de fichiers EM. Vous pouvez lire le fichier journal de suppression avec CX-Programmer ou FREAD(700) pour vérifier les informations suivantes : date de suppression du fichier, type de la mémoire de fichiers (support) existante, sous-répertoire, nom de fichier et extension. Si nécessaire, recréez ou recopiez le fichier supprimé.

Le schéma suivant décrit la structure du fichier journal de suppression.



- Type de support (2 octets)
- 0000 : Carte mémoire
- 0001 : Mémoire de fichiers EM

# CHAPITRE 6

## Fonctions avancées

Ce chapitre fournit des informations détaillées sur les fonctions avancées suivantes : fonctions de traitement à grande vitesse/temps de cycle, fonctions de registres d'index, fonctions de communications série, fonctions de démarrage et de maintenance, fonctions de diagnostic et de débogage, fonctions des périphériques de programmation et les paramètres de temps de réponse des entrées de la carte d'E/S standard.

6-1	Temps de cycle et traitement à grande vitesse . . . . .	257
6-1-1	Temps de cycle minimum . . . . .	257
6-1-2	Temps de cycle maximum (temps de cycle d'horloge) . . . . .	258
6-1-3	Surveillance du temps de cycle . . . . .	258
6-1-4	Entrées à grande vitesse . . . . .	259
6-1-5	Fonctions d'interruption . . . . .	259
6-1-6	Méthodes de mise à jour d'E/S . . . . .	260
6-1-7	Désactivation de la mise à jour cyclique des cartes d'E/S spéciales . . . . .	261
6-1-8	Amélioration de la réponse de mise à jour des données de la carte réseau . . . . .	262
6-1-9	Temps maximal de réponse d'E/S de liaison de données . . . . .	264
6-1-10	Exécution en arrière-plan . . . . .	266
6-1-11	Partage de registres d'index et de données entre des tâches . . . . .	273
6-2	Registres d'index . . . . .	274
6-2-1	Définition des registres d'index . . . . .	274
6-2-2	Utilisation des registres d'index . . . . .	275
6-2-3	Traitement associé aux registres d'index . . . . .	278
6-3	Communications en série . . . . .	284
6-3-1	Communications Host Link . . . . .	286
6-3-2	Communications sans protocole . . . . .	291
6-3-3	Liaison NT (mode 1:N) . . . . .	292
6-3-4	Passerelle série de l'UC . . . . .	293
6-3-5	Liaisons API série (cartes CJ1M uniquement) . . . . .	299
6-4	Modification du mode de mise à jour de la PV de temporisation/compteur . . . . .	305
6-4-1	Présentation . . . . .	305
6-4-2	Caractéristiques fonctionnelles . . . . .	306
6-4-3	Sélection du mode BCD ou du mode binaire et confirmation . . . . .	307
6-4-4	Mnémiques et données en mode BCD ou en mode binaire . . . . .	308
6-4-5	Restrictions . . . . .	309
6-4-6	Instructions et opérandes . . . . .	310
6-5	Utilisation d'une interruption programmée comme temporisation de haute précision (CJ1M uniquement) . . . . .	313
6-5-1	Configuration de l'interruption programmée par pas de 0,1 ms . . . . .	313
6-5-2	Spécification d'un démarrage avec réinitialisation à l'aide de MSKS(690) . . . . .	314
6-5-3	Lecture de la valeur actuelle (PV) de temporisation interne à l'aide de MSKR(692) . . . . .	314
6-6	Paramètres du démarrage et maintenance . . . . .	315
6-6-1	Fonctions de démarrage/d'arrêt à chaud . . . . .	315
6-6-2	Paramètre du mode de démarrage . . . . .	316

6-6-3	Sortie RUN .....	317
6-6-4	Paramètre du retard de la détection de mise hors tension .....	317
6-6-5	Désactivation des interruptions à la mise hors tension .....	317
6-6-6	Fonctions d'horloge .....	318
6-6-7	Protection du programme .....	319
6-6-8	Protection en écriture à partir de commandes FINS envoyées aux UC via des réseaux .....	321
6-6-9	Programmation et surveillance à distance .....	322
6-6-10	Profils des unités .....	322
6-6-11	Mémoire flash .....	323
6-6-12	Paramètres des conditions au démarrage .....	324
6-7	Fonctions de diagnostic .....	326
6-7-1	Journal d'erreurs .....	326
6-7-2	Fonction sortie OFF .....	327
6-7-3	Fonctions d'alarme erreur .....	327
6-7-4	Détection du point d'erreur .....	328
6-7-5	Simulation d'erreurs système .....	330
6-7-6	Désactivation du stockage des erreurs FAL définies par l'utilisateur dans le journal d'erreurs .....	330
6-8	Modes de traitement UC .....	331
6-8-1	Modes de traitement UC .....	331
6-8-2	Mode de traitement parallèle et temps de cycle minimums .....	336
6-8-3	Concurrence de données en traitement parallèle avec accès mémoire asynchrone .....	336
6-9	Mode prioritaire du traitement des périphériques .....	336
6-9-1	Mode prioritaire du traitement des périphériques .....	337
6-9-2	Désactivation temporaire du traitement en mode prioritaire .....	339
6-10	Fonctionnement sans batterie .....	342
6-11	Autres fonctions .....	344
6-11-1	Paramètres des temps de réponse d'E/S .....	344
6-11-2	Affectation de la zone d'E/S .....	345

## 6-1 Temps de cycle et traitement à grande vitesse

Ce chapitre décrit les fonctions suivantes :

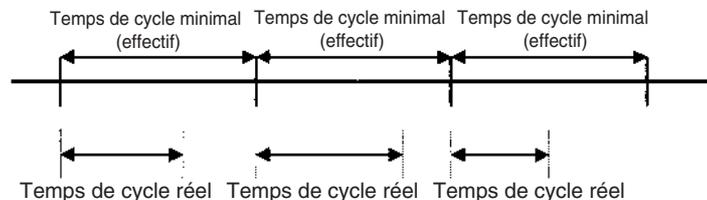
- la fonction de temps de cycle minimum ;
- la fonction de temps de cycle maximum (temps de cycle d'horloge) ;
- la surveillance du temps de cycle ;
- les entrées à réponse rapide ;
- les fonctions d'interruption ;
- les méthodes de mise à jour d'E/S ;
- la désactivation de la mise à jour cyclique des cartes d'E/S spéciales ;
- l'amélioration de la réponse de mise à jour pour les liaisons de données et d'autres données de cartes réseau (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement) ;
- la réduction des fluctuations du temps de cycle grâce à des manipulations de données en arrière-plan (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement).

### 6-1-1 Temps de cycle minimum

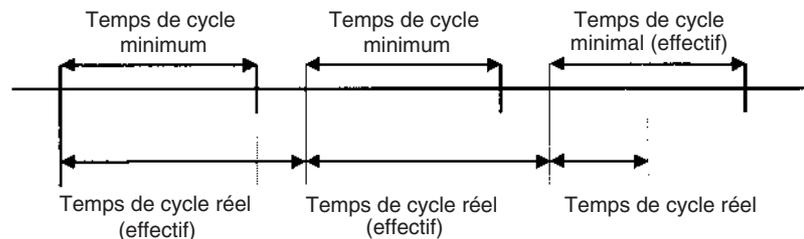
Les API série CS/CJ permettent de configurer un temps de cycle minimum (ou fixe). (Voir remarque). Pour éliminer les variations de temps de réponse d'E/S, répétez le programme avec un temps de cycle fixe.

**Remarque** Le temps de cycle peut également être fixe dans le cas d'une UC CS1D pour système à UC seule, des UC CS1-H, CJ1-H ou CJ1M, ou si vous utilisez un mode de traitement parallèle.

Le temps de cycle minimum (1 à 32 000 ms) est spécifié par pas de 1 ms dans la configuration de l'API.



Si le temps de cycle réel est supérieur au temps de cycle minimum, la fonction de temps de cycle minimum est inopérante et le temps de cycle varie d'un cycle à l'autre.



#### Configuration de l'API

Adresse	Nom	Paramètre	Par défaut
208 Bits : 0 à 15	Temps de cycle minimum	0001 à 7D00 : 1 à 32 000 ms (pas de 1 ms)	0000 (absence de minimum)

### 6-1-2 Temps de cycle maximum (temps de cycle d'horloge)

Si le temps de cycle (voir remarque) dépasse le paramètre de temps de cycle maximum, le drapeau de temps de cycle trop long (A40108) passe à ON et l'API s'arrête de fonctionner.

**Remarque** Dans ce cas, le temps de cycle correspond au temps d'exécution du programme lorsque vous utilisez un mode de traitement parallèle pour les UC CS1-H, CJ1-H ou CJ1M, ou les UC pour systèmes à UC seule.

#### Configuration de l'API

Adresse	Nom	Paramètre	Par défaut
209 Bit : 15	Active le paramètre du temps de cycle d'horloge	0 : valeur par défaut (1 s) (1) bits 0 à 14	0001 (1 s)
209 Bits : 0 à 14	Paramètre du temps de cycle d'horloge (Activé lorsque le bit 15 est configuré à 1).	001 à FA0 : 10 à 40 000 ms (pas de 10 ms)	

#### Drapeaux et mots de la zone auxiliaire

Nom	Adresse	Description
Drapeau de temps de cycle trop long	A40108	Si le temps de cycle dépasse le paramètre du temps de cycle d'horloge, A40108 passe à ON et l'UC s'arrête de fonctionner. Le « temps de cycle » correspond au temps d'exécution du programme lorsque vous utilisez un mode de traitement parallèle pour les UC CS1-H, CJ1-H ou CJ1M, ou les UC pour systèmes à UC seule.

**Remarque** Si le cycle du traitement des périphériques est supérieur à 2,0 s pour les UC CS1-H, CJ1-H ou CJ1M, ou les UC CS1D pour systèmes à UC seule en mode de traitement parallèle, une erreur de dépassement de temps de cycle du traitement des périphériques se produit et l'UC s'arrête de fonctionner. Si cela se produit, A40515 (drapeau de dépassement de temps de cycle du traitement des périphériques) passe à ON.

### 6-1-3 Surveillance du temps de cycle

Le temps de cycle maximum et le temps de cycle actuel sont mémorisés dans la zone auxiliaire après chaque cycle. Dans le cas des UC CS1-H, CJ1-H ou CS1D pour systèmes à UC seule en mode de traitement parallèle, les temps d'exécution du programme sont également mémorisés.

#### Drapeaux et mots de la zone auxiliaire

Nom	Adresse	Description
Temps de cycle maximum (temps d'exécution du programme pour les UC CS1-H, CJ1-H ou CJ1M en mode de traitement parallèle)	A262 et A263	Mémorisé à chaque cycle sous forme de valeur binaire de 32 bits, dans la plage suivante : 0 à 429 496 729,5 ms par pas de 0,1 ms (0 à FFFF FFFF)
Temps de cycle actuel (temps d'exécution du programme pour les UC CS1-H, CJ1-H ou CJ1M en mode de traitement parallèle)	A264 et A265	Mémorisé à chaque cycle sous forme de valeur binaire de 32 bits, dans la plage suivante : 0 à 429 496 729,5 ms par pas de 0,1 ms (0 à FFFF FFFF)

Vous pouvez utiliser un périphérique de programmation (CX-Programmer ou console de programmation) pour lire la moyenne des temps de cycle sur les 8 derniers cycles.

**Réduire le temps de cycle**

Vous pouvez réduire efficacement le temps de cycle des API série CS/CJ en appliquant l'une des méthodes suivantes :

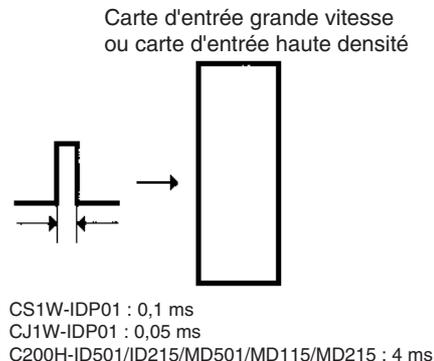
- 1,2,3...**
1. Mettez en attente les tâches qui ne sont pas en cours d'exécution.
  2. Sautez les sections de programme qui ne sont pas en cours d'exécution à l'aide de JMP(004) et JME(005).

Dans le cas des UC CS1-H ou CJ1-H, ou des UC CS1D pour systèmes à UC seule en mode de traitement parallèle, le temps de cycle du traitement des périphériques est stocké en A268 (temps de cycle du traitement des périphériques) après chaque cycle de service.

**6-1-4 Entrées à grande vitesse**

Si vous souhaitez recevoir des impulsions inférieures au temps de cycle, utilisez la carte d'entrée à grande vitesse CS1W-IDP01 ou les entrées à grande vitesse des cartes d'E/S haute densité C200H-ID501/ID215 et C200H-MD501/MD115/MD215.

Les entrées à grande vitesse peuvent recevoir des impulsions d'une largeur (temps ON) de 1 ms ou 4 ms pour les cartes d'entrées haute densité C200H et de 0,1 ms pour la carte d'entrée à grande vitesse CS1W-IDP01.

**6-1-5 Fonctions d'interruption**

Des tâches d'interruption peuvent être exécutées dans les conditions suivantes. Reportez-vous à la section 4-3 *Tâche d'interruption* pour plus de détails.

**Remarque** Les UC CS1D pour systèmes à UC en duplex ne prennent pas en charge les interruptions. Dans le cas des UC CS1D, les tâches d'interruption ne peuvent s'utiliser qu'en tant que tâches cycliques extra, autrement dit, vous ne pouvez utiliser aucun autre type de tâche d'interruption.

**Interruptions E/S (tâches d'interruption 100 à 131)**

Une tâche d'interruption E/S est exécutée lorsque l'entrée correspondante (sur le front montant du signal ou, pour les cartes d'entrées d'interruption, sur les fronts montant ou descendant) est reçue d'une carte d'entrée d'interruption.

**Interruptions programmées (tâches d'interruption 2 à 3)**

Une tâche d'interruption programmée est exécutée à intervalles réguliers.

**Interruption de mise hors tension (tâche d'interruption 1)**

Cette tâche est exécutée lorsque l'alimentation est interrompue.

**Interruptions externes (tâches d'interruption 0 à 255)**

Une tâche d'interruption externe est exécutée lorsqu'une interruption est reçue d'une carte d'E/S spéciales, d'une carte réseau ou d'une carte interne.

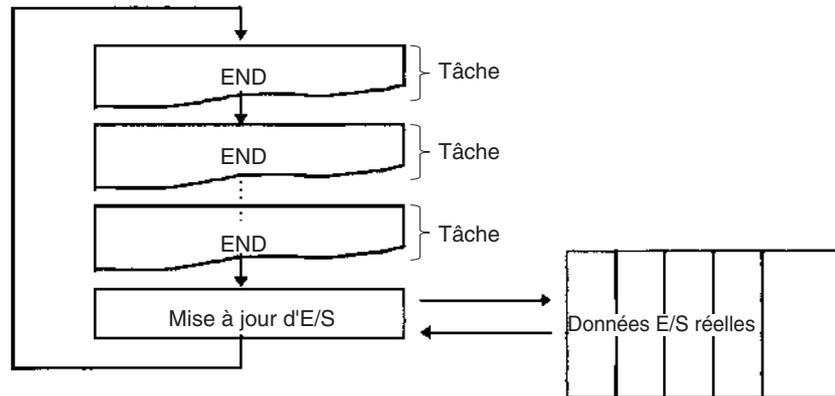
**Remarque** Sur une UC CJ1M, vous pouvez utiliser les entrées intégrées d'interruption et de compteur à grande vitesse pour activer des tâches d'interruption. Reportez-vous au *Manuel d'utilisation des E/S intégrées, série CJ* pour plus de détails.

### 6-1-6 Méthodes de mise à jour d'E/S

Dans le cas des UC série CS/CJ, il existe 3 manières différentes de mettre à jour les données à l'aide de cartes d'E/S standard et de cartes d'E/S spéciales : la mise à jour cyclique, la mise à jour immédiate et l'exécution de l'instruction IORF(097).

#### 1. Mise à jour cyclique

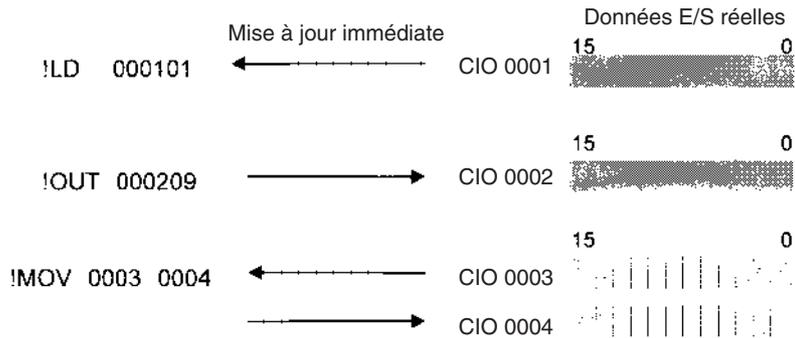
La mise à jour d'E/S s'effectue une fois que toutes les instructions contenues dans les tâches exécutables ont été exécutées. (Vous pouvez définir la configuration de l'API de manière à désactiver la mise à jour cyclique des différentes cartes d'E/S spéciales).



#### 2. Mise à jour immédiate

Lorsqu'une adresse de la zone d'E/S est spécifiée en tant qu'opérande dans la variation par mise à jour immédiate d'une instruction, ces données d'opérande sont mises à jour lors de l'exécution de l'instruction. Les instructions de mise à jour immédiate peuvent mettre à jour des données affectées à des cartes d'E/S standard.

La mise à jour immédiate est également possible pour les E/S intégrées des UC CJ1M.



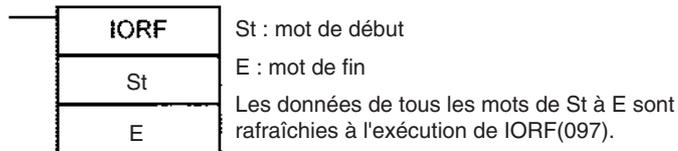
**Remarque**

1. Lorsque l'instruction contient un opérande bit, le mot entier contenant ce bit est mis à jour. Lorsque l'instruction contient un opérande mot, ce mot est mis à jour.
2. Les données d'entrée et sources sont mises à jour juste avant l'exécution de l'instruction. Les données de sortie et de destination sont mises à jour juste après l'exécution de l'instruction.
3. Les temps d'exécution des variations par mise à jour immédiate sont supérieurs à ceux des variations régulières d'instructions, le cycle sera donc plus long. Reportez-vous au chapitre 10-5 Temps d'exécution d'instruction et nombre de pas dans le Manuel d'utilisation pour plus de détails.
4. Les UC CS1D pour systèmes à UC en duplex ne prennent pas en charge la mise à jour immédiate.

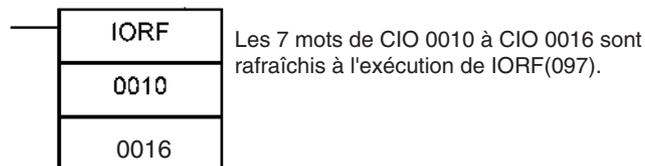
3. Exécution des instructions IORF(097) et DLNK(226)

■ **IORF(097) : MISE A JOUR D'E/S**

L'exécution de l'instruction IORF(097) permet de mettre à jour une plage de mots d'E/S. L'instruction IORF(097) peut mettre à jour des données affectées à des cartes d'E/S standard et spéciales.



Dans l'exemple suivant, l'instruction IORF(097) est utilisée pour mettre à jour 8 mots de données d'E/S.



Lorsqu'une réponse à grande vitesse est requise pour l'entrée et la sortie d'un calcul, utilisez IORF(097) juste avant et juste après l'instruction de calcul.

**Remarque**

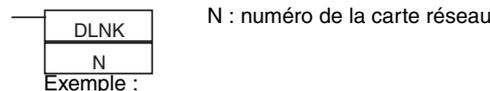
Le temps d'exécution de l'instruction IORF(097) est relativement long et augmente proportionnellement au nombre de mots mis à jour, ce qui peut entraîner un allongement significatif du temps de cycle. Reportez-vous au chapitre 10-5 *Temps d'exécution d'instruction et nombre de pas* dans le *Manuel d'utilisation* pour plus de détails.

■ **DLNK(226) : mise à jour d'E/S de carte réseau (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement)**

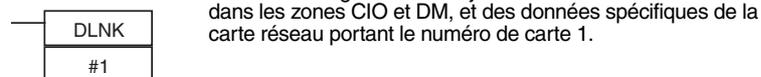
DLNK(226) s'utilise pour mettre à jour des données d'une carte réseau portant le numéro de carte spécifié. Les données mises à jour sont les suivantes :

- mots affectés à la carte dans la zone CIO ;
- mots affectés à la carte dans la zone DM ;
- données spécifiques de la carte (voir remarque) ;

**Remarque** Les données spécifiques d'une carte réseau comprennent des liaisons de données pour la carte Controller Link ou les cartes SYSMAC LINK, ainsi que des E/S déportées pour les cartes DeviceNet.



Exemple :



Exemple d'application : Lorsque le temps de cycle est long, l'intervalle de mise à jour des liaisons de données de Controller Link peut être très long. Vous pouvez raccourcir cet intervalle en exécutant l'instruction DLNK(226) pour que la carte Controller Link augmente la fréquence de mise à jour des liaisons de données.

**6-1-7 Désactivation de la mise à jour cyclique des cartes d'E/S spéciales**

Dix mots de la zone de carte d'E/S spéciales (CIO 2000 à CIO 2959) sont affectés à chaque carte d'E/S spéciales en fonction du numéro de carte configuré sur l'avant de la carte. Les données sont mises à jour entre cette zone et l'UC à chaque cycle de la mise à jour d'E/S. Vous pouvez toutefois

désactiver cette mise à jour cyclique pour les différentes cartes dans la configuration de l'API.

Les trois principaux motifs de désactivation de la mise à jour cyclique sont les suivants :

- 1,2,3...**
1. La mise à jour cyclique des cartes d'E/S spéciales peut être désactivée lorsque le temps de cycle est trop long, en raison du grand nombre de cartes d'E/S spéciales installées.
  2. Si le temps de mise à jour d'E/S est trop court, le traitement interne de la carte risque de ne pas suivre le rythme, le drapeau d'erreur des cartes d'E/S spéciales (A40206) passera à ON et la carte d'E/S spéciales ne fonctionnera pas correctement.  
 Dans ce cas, le temps de cycle peut être allongé en configurant un temps de cycle minimum dans la configuration de l'API ou en désactivant la mise à jour d'E/S cyclique de la carte d'E/S spéciales.
  3. Désactivez toujours la mise à jour cyclique d'une carte d'E/S spéciales si cette dernière est mise à jour au cours d'une tâche d'interruption par l'instruction IORF(097). Si la mise à jour cyclique et la mise à jour par IORF(097) s'effectuent simultanément pour la même carte, une erreur de tâche d'interruption se produit et le drapeau d'erreur de tâche d'interruption (A40213) passe à ON.

Lorsque la mise à jour cyclique est désactivée, les données de la carte d'E/S spéciales peuvent être mises à jour pendant l'exécution du programme à l'aide de l'instruction IORF(097).

**Configuration de l'API**

Les bits de désactivation de la mise à jour cyclique des cartes d'E/S spéciales 0 à 95 correspondent directement aux 96 bits des adresses 226 à 231.

Adresse	Nom	Paramètre	Par défaut
226 bit 0	Bit de désactivation de la mise à jour cyclique de la carte d'E/S spéciales 0	0 : activé 1 : désactivé	0 (activé)
:	:	:	:
231 bit 15	Bit de désactivation de la mise à jour cyclique de la carte d'E/S spéciales 95	0 : activé 1 : désactivé	0 (activé)

**6-1-8 Amélioration de la réponse de mise à jour des données de la carte réseau**

Cette fonction est uniquement prise en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D.

En règle générale, les liaisons de données et les autres données spéciales des cartes réseau sont mises à jour en même temps que les mots des zones CIO et DM affectés aux cartes, pendant la période de mise à jour d'E/S qui suit l'exécution du programme.

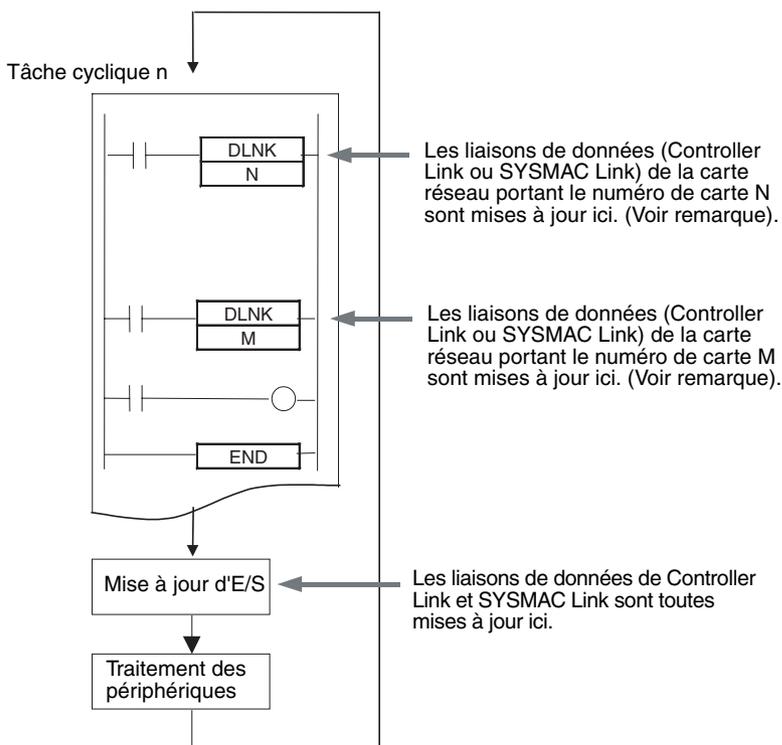
Le tableau suivant présente quelques exemples de données spéciales de cartes réseau.

Cartes	Données spéciales
Cartes Controller Link et cartes SYSMAC LINK	Liaisons de données Controller Link et SYSMAC LINK (y compris les liaisons configurées automatiquement et par l'utilisateur)
Cartes DeviceNet série CS/CJ	Communications d'E/S déportées DeviceNet (y compris les affectations fixes et configurées par l'utilisateur)

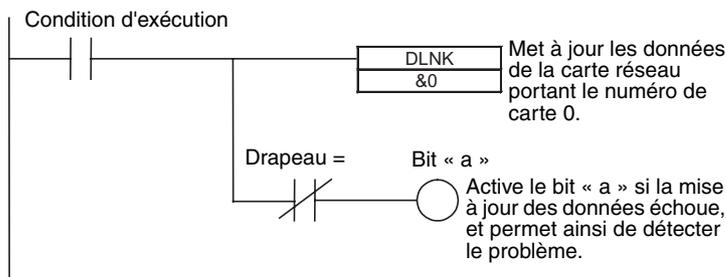
Les fonctions suivantes permettent d'améliorer la réponse de mise à jour des données spéciales des cartes réseau avec les UC CS1-H, CJ1-H, CJ1M ou CS1D.

- Réduction du temps de cycle à l'aide du mode de traitement parallèle ou des instructions à grande vitesse. (Le mode de traitement parallèle n'est pas pris en charge par les UC CS1D pour systèmes d'UC en duplex).
- Exécution de l'instruction DLNK(226) pour mettre à jour des cartes réseau spécifiques en indiquant leurs numéros (DLNK(226) peut s'utiliser plusieurs fois dans le programme).

**Remarque** 1. Les temps de cycle plus longs (exemple : 100 ms) augmentent l'intervalle de mise à jour des liaisons de données. Dans ce cas, vous pouvez utiliser l'instruction DLNK(226), comme illustré dans l'exemple suivant.



**Remarque** Si l'instruction DLNK(226) est exécutée pour une carte réseau qui est en train de mettre à jour des données, les données ne sont pas mises à jour et le drapeau d'égalité passe à OFF. En règle générale, le drapeau d'égalité doit être programmé comme indiqué ci-dessous, pour être sûr que la mise à jour s'est terminée normalement.

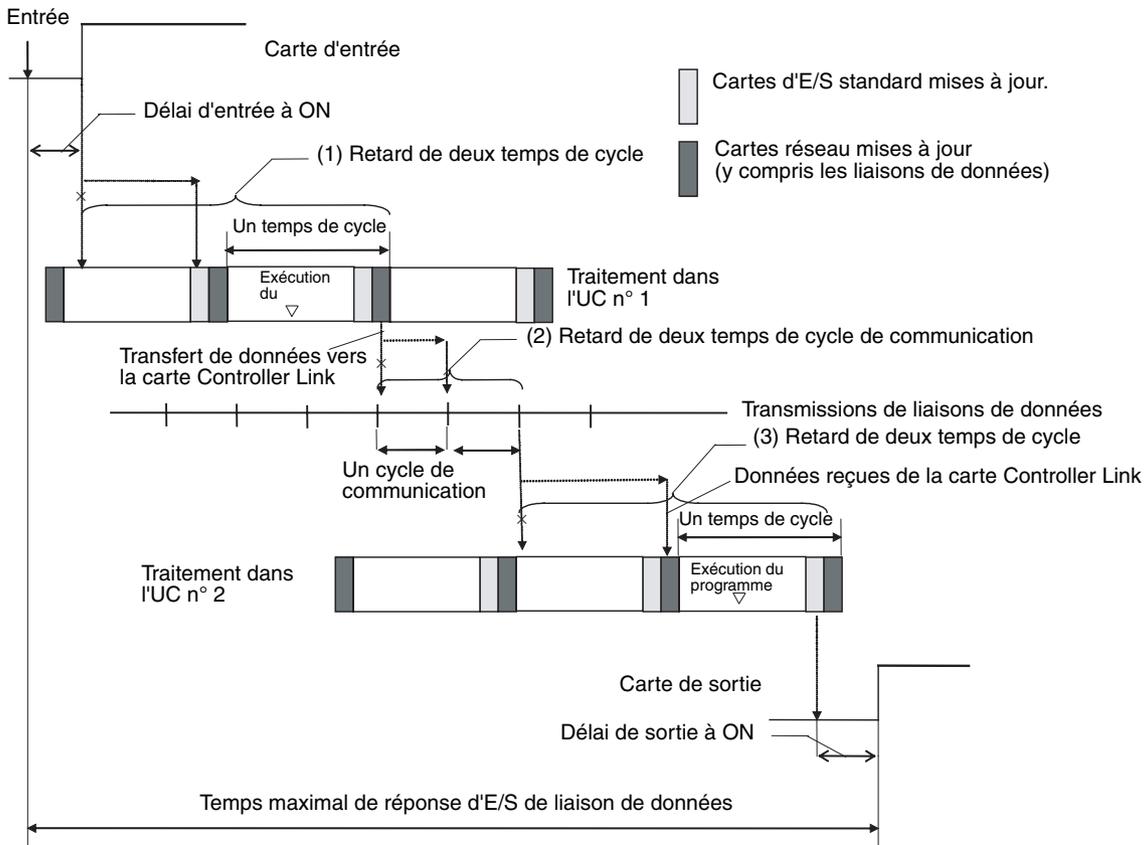


2. L'instruction IORF(097) permet de mettre à jour des données de cartes d'E/S standard et spéciales. L'instruction (226) permet de mettre à jour des cartes réseau (mots des zones CIO et DM, affectés aux cartes et aux données spéciales des cartes).

### 6-1-9 Temps maximal de réponse d'E/S de liaison de données

Traitement normal

Le diagramme suivant représente le flux de données qui génère le temps maximal de réponse d'E/S de liaison de données lorsque l'instruction DLNK(226) n'est pas utilisée.



Dans le diagramme ci-dessus, le traitement est retardé en 3 points, ce qui augmente le temps de réponse d'E/S de liaison de données.

1,2,3...

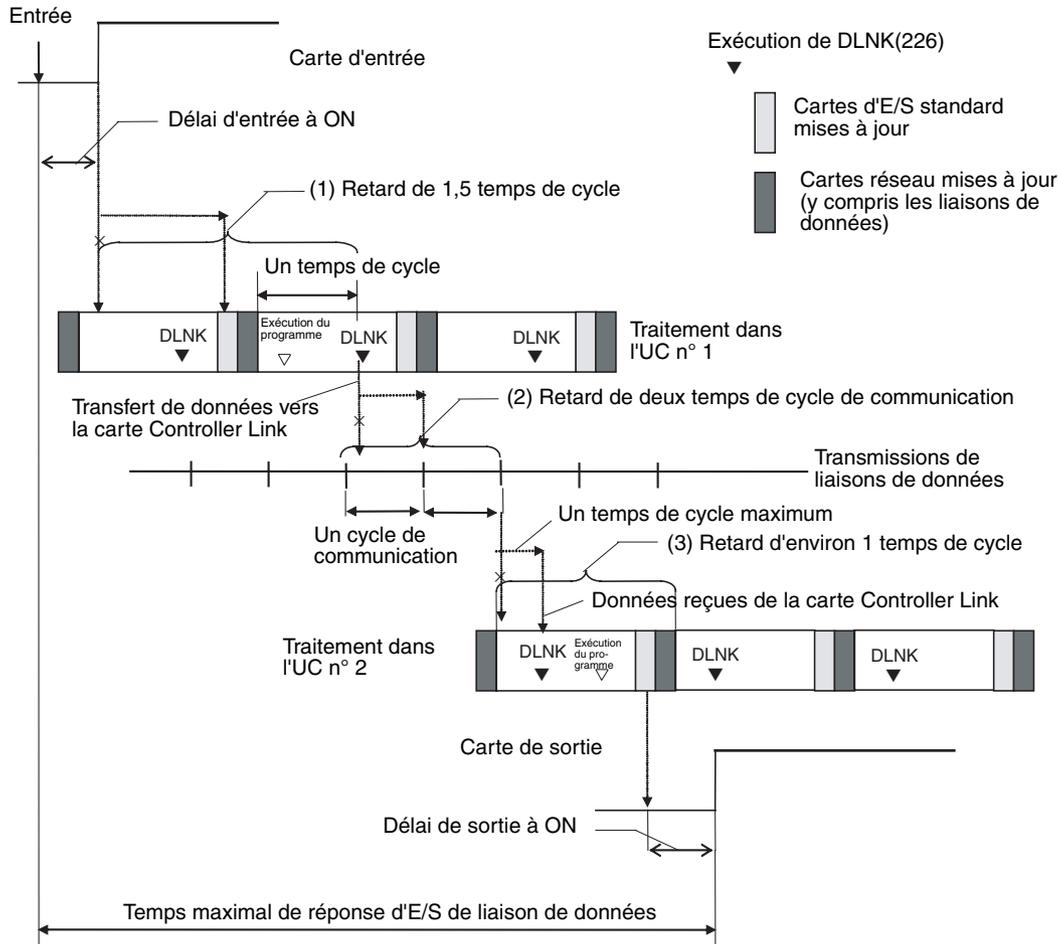
1. L'entrée arrive à l'API (UC n° 1) juste après la mise à jour d'E/S, entraînant un retard d'un cycle avant l'enregistrement de l'entrée dans l'API. Les cartes réseau sont mises à jour après l'exécution du programme, générant un retard total de deux cycles.
2. L'échange de données se produit immédiatement après que l'API a transmis le jeton qui le transforme en nœud d'analyse, générant ainsi un retard de l'ordre d'un temps de cycle de communication (maximum), avant le transfert des données par traitement de liaison de données. Un retard de l'ordre d'un temps de cycle de communication (maximum) se forme également après la réception du jeton, générant un retard total maximum de deux temps de cycle de communication.
3. Les données transférées par traitement de liaison de données arrivent à l'API (UC n° 2) après l'échange de données, de sorte que les données ne seront pas enregistrées dans l'API avant le prochain échange de données, générant ainsi un retard inférieur ou égal à un cycle. Les cartes réseau sont mises à jour après l'exécution du programme, générant un retard total de deux cycles.

L'équation du temps maximal de réponse d'E/S de liaison de données est la suivante :

Délai d'entrée à ON	1,5 ms
Temps de cycle de l'API sur l'UC n° 1 × 2	25 ms × 2
Temps de cycle de communication × 2	10 ms × 2
Temps de cycle de l'API sur l'UC n° 2 × 2	20 ms × 2
Délai de sortie à ON	15 ms
Total (temps de réponse d'E/S de liaison de données)	126,5 ms

**Utilisation de l'instruction DLNK(226)**

Le diagramme suivant représente le flux de données qui génère le temps maximal de réponse d'E/S de liaison de données lorsque l'instruction DLNK(226) est utilisée.



Dans le diagramme ci-dessus, le traitement est retardé en 3 points, ce qui augmente le temps de réponse d'E/S de liaison de données.

**Remarque**

Dans cet exemple, l'instruction DNLK(226) est supposée être placée après d'autres instructions du programme dans les deux UC.

**1,2,3...**

1. L'entrée arrive à l'API (UC n° 1) juste après la mise à jour d'E/S, entraînant un retard d'un cycle avant l'enregistrement de l'entrée dans l'API. Les cartes réseau sont mises à jour pendant l'exécution du programme, ce qui réduit le retard total à environ 1,5 temps de cycle.
2. L'échange de données se produit immédiatement après que l'API a transmis le jeton qui le transforme en nœud d'analyse, générant ainsi un retard de l'ordre d'un temps de cycle de communication (maximum), avant le transfert des données par traitement de liaison de données. Un retard de l'ordre d'un temps de cycle de communication (maximum) se forme également après la réception du jeton, générant un retard total maximum de deux temps de cycle de communication.
3. Les données transférées par traitement de liaison de données arrivent à l'API (UC n° 2) après la mise à jour d'E/S, mais l'instruction DLNK(226) met à jour les données, et les données seront donc enregistrées dans l'API sans générer de retard de l'ordre d'un cycle (maximum). Les cartes d'E/S standard sont mises à jour après l'exécution du programme, générant ainsi un retard total d'environ un cycle.

L'équation du temps maximal de réponse d'E/S de liaison de données est la suivante :

Délai d'entrée à ON	1,5 ms	---
Temps de cycle de l'API sur l'UC n° 1 × 1,5	25 ms × 1,5	Plus rapide de 12,5 ms (25 ms x 0,5)
Temps de cycle de communication × 2	10 ms × 2	---
Temps de cycle de l'API sur l'UC n° 2 × 1	20 ms × 1	Plus rapide de 20 ms (20 ms x 1)
Délai de sortie à ON	15 ms	---
Total (temps de réponse d'E/S de liaison de données)	94 ms	Plus rapide de 32,5 ms (26 % plus rapide)

### 6-1-10 Exécution en arrière-plan

L'exécution en arrière-plan permet de réduire les fluctuations du temps de cycle. L'exécution en arrière-plan n'est prise en charge que par les UC CS1-H, CJ1-H ou CJ1M, ou les UC CS1D pour systèmes à UC seule.

Le traitement des données de tables (par exemple, les recherches de données) et le traitement des chaînes de caractères (par exemple les recherches de chaînes de caractères) nécessitent un certain temps d'exécution et peuvent induire d'importantes fluctuations du temps de cycle, en raison du temps supplémentaire requis pour leur exécution.

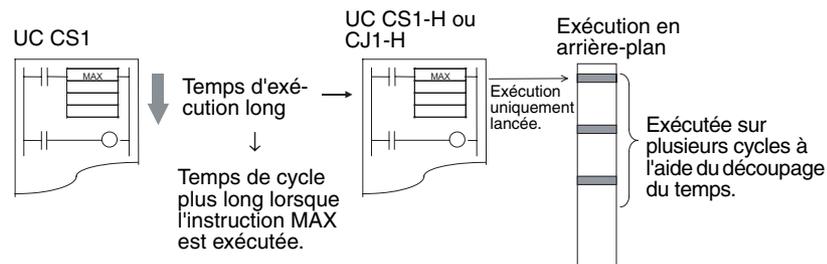
Toutefois, dans le cas des UC CS1-H, CJ1-H ou CJ1M, ou des UC CS1D pour systèmes à UC seule (voir remarque), l'exécution en arrière-plan (découpage du temps) permet d'exécuter les instructions suivantes sur plusieurs cycles afin de mieux contrôler les fluctuations du temps de cycle. La configuration de l'API permet de paramétrer une exécution en arrière-plan pour chaque type d'instruction.

**Remarque**

Les UC CS1D pour systèmes à UC en duplex ne prennent pas en charge l'exécution en arrière-plan.

- Instructions de traitement de données de tables
- Instructions de traitement de chaînes de caractères
- Instructions de décalage de données (ASYNCHRONOUS SHIFT REGISTER uniquement)

Configurer l'exécution en arrière-plan des instructions ci-dessus peut faciliter le contrôle des allongements temporaires du temps de cycle.



**Applications**

Vous pouvez utiliser l'exécution en arrière-plan pour le traitement de grands volumes de données, tel que la compilation ou le traitement de données, requis à certaines heures seulement (exemple : 1 fois par jour), lorsque la réduction de l'effet sur le temps de cycle a plus d'importance que la vitesse du traitement des données.

**Procédure**

1,2,3...

1. Paramétrez la configuration de l'API pour activer l'exécution en arrière-plan des instructions requises.
2. Configurez le numéro du port des communications (numéro de port logique) à utiliser pour l'exécution en arrière-plan, dans la configuration de l'API. Ce numéro de port sera utilisé pour toutes les instructions traitées en arrière-plan.

**Remarque** Un seul port est utilisé pour toutes les exécutions en arrière-plan. Par conséquent, une exécution en arrière-plan ne peut être lancée si l'exécution en arrière-plan d'une autre instruction est déjà en cours. Utilisez le drapeau d'activation du port des communications pour contrôler les instructions spécifiées pour l'exécution en arrière-plan et éviter que plusieurs instructions ne soient exécutées simultanément.

3. Si une instruction pour laquelle vous avez spécifié une exécution en arrière est exécutée, l'exécution ne sera lancée qu'au cours du cycle pendant lequel la condition d'exécution a été satisfaite, et ne se terminera pas sur le même cycle.
4. Lorsque l'exécution en arrière-plan est lancée, le drapeau d'activation du port des communications correspondant à ce port passe à OFF.
5. L'exécution en arrière se poursuit sur plusieurs cycles.
6. Lorsque le traitement est terminé, le drapeau d'activation du port des communications correspondant à ce port passe à ON. Ceci active l'exécution en arrière-plan d'une autre instruction.

**Instructions utilisables**

■ **Instructions de traitement des données de tables**

Instruction	Mnémonique	Code fonction
DATA SEARCH	SRCH	181
SWAP BYTES	SWAP	637
FIND MAXIMUM	MAX	182
FIND MINIMUM	MIN	183
SUM	SUM	184
FRAME CHECKSUM	FCS	180

■ **Instructions de traitement des chaînes de caractères**

Instruction	Mnémonique	Code fonction
MOVE STRING	MOV\$	664
CONCATENATE STRING	+\$	656
GET STRING LEFT	LEFT\$	652
GET STRING RIGHT	RIGHT\$	653
GET STRING MIDDLE	MID\$	654
FIND IN STRING	FIND\$	660
STRING LENGTH	LEN\$	650
REPLACE IN STRING	RPLC\$	661
DELETE STRING	DEL\$	658
EXCHANGE STRING	XCHG\$	665
CLEAR STRING	CLR\$	666
INSERT INTO STRING	INS\$	657

■ **Instructions de décalage de données**

Instruction	Mnémonique	Code fonction
ASYNCHRONOUS SHIFT REGISTER	ASFT	017

**Différences entre les instructions exécutées normalement et celles exécutées en arrière-plan**

Les différences entre l'exécution normale des instructions et leur exécution en arrière-plan sont répertoriées ci-dessous.

■ **Sortie vers les registres d'index (IR)**

Si MAX(182) ou MIN(183) est exécutée pour diriger l'adresse de mémoire d'E/S du mot contenant la valeur minimale ou maximale vers un registre d'index, l'adresse sera dirigée vers A595 et A596 au lieu d'être dirigée vers le registre d'index. Pour stocker l'adresse dans un registre d'index, utilisez une instruction de déplacement de données (exemple : MOVL(498)) pour copier l'adresse dans A595 et A596 vers un registre d'index.

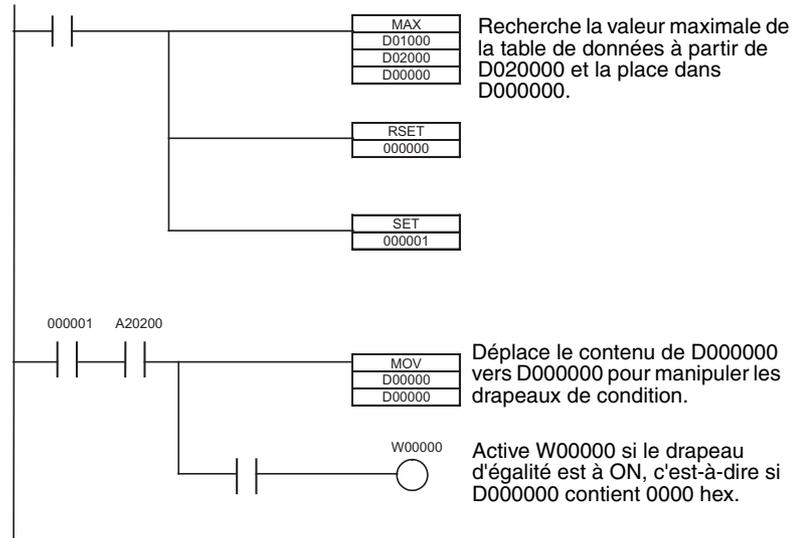
■ **Drapeaux de condition**

Les drapeaux de condition ne sont pas mis à jour après l'exécution d'instructions traitées en arrière-plan. Pour accéder à l'état des drapeaux de condition, exécutez une instruction qui modifie les drapeaux de condition en procédant comme dans l'exemple suivant, puis accédez aux drapeaux de condition.

**Exemple :**

MOV(021) modifie les drapeaux d'égalité et négatif de manière identique à MAX(182), autrement dit, ces deux instructions mettent toutes deux le drapeau d'égalité à ON pour 0 et mettent le drapeau négatif à ON si le bit de poids fort (MSB) est activé. Vous pouvez donc utiliser MOV(021) pour copier les résultats de MAX(182) vers la même adresse afin de manipuler les drapeaux de condition et de pouvoir ainsi accéder à l'état.

Condition d'exécution



■ **Sortie vers le registre d'index IR00**

Si SRCH(181) est exécutée pour diriger l'adresse de mémoire d'E/S du mot contenant la valeur correspondante (le premier mot s'il en existe plusieurs) vers un registre d'index, l'adresse est dirigée vers A595 et A596 au lieu d'être dirigée vers le registre d'index.

■ **Sortie vers des registres de données (DR) avec SRCH(181)**

Si SRCH(181) est exécutée pour diriger les données correspondantes vers un registre de données, les données sont dirigées vers A597 au lieu d'être dirigées vers le registre de données.

■ **Chaînes de caractères correspondantes**

Si SRCH(181) trouve des données correspondantes, elle active A59801 au lieu de mettre le drapeau d'égalité à ON.

■ **Erreurs d'instructions**

Si une erreur d'exécution d'instruction ou une erreur d'accès illégal se produit pendant le traitement en arrière-plan d'une instruction, les drapeaux ER ou AER ne sont pas mis à ON mais, au lieu de cela, A39510 est activée. A39510 reste à ON jusqu'au prochain traitement en arrière-plan d'une instruction.

■ **Sortie vers des registres de données (DR) avec SRCH(181)**

Si MAX(182) ou MIN(183) est exécutée avec un registre de données spécifié comme mot de sortie de la valeur minimale ou maximale, une erreur d'exécution d'instruction se produit et le drapeau ER passe à ON.

**Configuration de l'API**

Mot	Bits	Nom	Paramètre	Par défaut et fréquence de mise à jour
198	15	Exécution en arrière-plan des instructions de la table de données	0 : non exécutée en arrière-plan. 1 : exécutée en arrière-plan.	0 : non exécutée en arrière-plan. Début du fonctionnement
	14	Exécution en arrière-plan des instructions de chaînes de caractères	0 : non exécutée en arrière-plan. 1 : exécutée en arrière-plan.	
	13	Exécution en arrière-plan des instructions de décalage de données	0 : non exécutée en arrière-plan. 1 : exécutée en arrière-plan.	
	00 à 03	Numéro du port des communications pour l'exécution en arrière-plan	0 à 7 hex. : ports des communications 0 à 7 (ports logiques internes)	0 hex. : port 0 Début du fonctionnement

**Drapeaux et mots de la zone auxiliaire**

Nom	Adresse	Description
Drapeaux d'activation du port des communications	A20200 à A20207	<p>Passé à ON lorsqu'une instruction de réseau (SEND, RECV, CMND ou PMCR) peut être exécutée avec le numéro de port correspondant ou lorsque l'exécution en arrière-plan peut s'effectuer avec le numéro de port correspondant (UC CS1D pour systèmes à UC seule, ou UC CS1-H, CJ1-H ou CJ1M uniquement). Les bits de 00 à 07 correspondent aux ports des communications de 0 à 7.</p> <p>Lorsque la sauvegarde simple est utilisée pour effectuer une écriture ou une comparaison sur une carte mémoire d'une UC CS1-H, CJ1-H, CJ1M ou CS1D, un port des communications est automatiquement affecté et le drapeau correspondant passe à ON pendant l'opération et à OFF lorsque l'opération est terminée.</p>
Drapeaux d'erreurs du port des communications	A21900 à A21907	<p>Passé à ON lorsqu'une erreur survient pendant l'exécution d'une instruction réseau (SEND, RECV, CMND ou PMCR). Les bits de 00 à 07 correspondent aux ports des communications de 0 à 7.</p> <p>Lorsque la sauvegarde simple est utilisée pour effectuer une écriture ou une comparaison sur une carte mémoire d'une UC CS1-H, CJ1-H, CJ1M ou CS1D, un port des communications est automatiquement affecté. Le drapeau correspondant passe à ON si une erreur survient et passe à OFF si la sauvegarde simple se termine normalement.</p>

Nom	Adresse	Description
Codes d'exécution du port des communications	A203 à A210	Ces mots contiennent les codes de fin des numéros de ports correspondants lorsque les instructions de réseau (SEND, RECV, CMND ou PMCR) ont été exécutées. Le contenu est effacé lorsque l'exécution en arrière-plan est terminée (pour les UC CS1D pour systèmes à UC seule, ou les UC CS1-H, CJ1-H ou CJ1M uniquement). Les mots de A203 à A210 correspondent aux ports des communications de 0 à 7.  Lorsque la sauvegarde simple est utilisée pour effectuer une écriture ou une comparaison sur une carte mémoire d'une UC CS1-H, CJ1-H, CJ1M ou CS1D, un port des communications est automatiquement affecté et un code de fin est sauvegardé dans le mot correspondant.
Drapeaux d'exécution en arrière-plan ER/AER	A39510	Passé à ON lorsqu'une erreur d'exécution de l'instruction ou une erreur d'accès illégal se produit pendant l'exécution en arrière-plan d'une instruction. Passe à OFF lors de la mise sous tension ou au début du fonctionnement.
Sortie IR00 pour l'exécution en arrière-plan	A595 et A596	Ces mots reçoivent la sortie lorsque vous spécifiez la sortie d'une instruction exécutée en arrière-plan pour un registre d'index. Aucune sortie ne sera effectuée vers IR00.  Plage : 0000 0000 à FFFF FFFF hex. 4 chiffres inférieurs : A595, 4 chiffres supérieurs : A596
Sortie DR00 pour l'exécution en arrière-plan	A597	Ce mot reçoit la sortie lorsque vous spécifiez la sortie d'une instruction exécutée en arrière-plan pour un registre de données. Aucune sortie ne sera effectuée vers DR00.  Plage : 0000 à FFFF hex.
Sortie de drapeau d'égalité pour l'exécution en arrière-plan	A59801	Ce drapeau passe à ON si des données correspondantes sont trouvées pour une instruction SRCH(181) exécutée en arrière-plan.

**Remarque** Les ports des communications (ports logiques internes) de l'UC sont utilisés pour l'exécution en arrière-plan et pour les instructions suivantes :

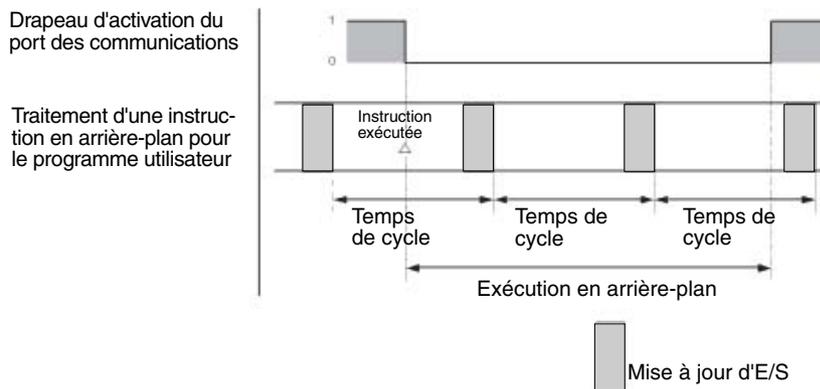
- SEND(090), RECV(098) et CMND(490) (instructions de communication réseau) ;
- PMCR(260) (PROTOCOL MACRO).

Les instructions en arrière-plan et les instructions ci-dessus ne peuvent être exécutées simultanément sur le même port. Utilisez les drapeaux d'activation du port des communications pour vous assurez qu'une seule instruction est exécutée à la fois sur chaque port.

**Remarque** Si vous spécifiez l'exécution en arrière-plan d'une instruction sur un port dont le drapeau d'activation du port des communications est à OFF, le drapeau ER passe à ON et l'instruction en arrière-plan n'est pas exécutée.

#### Drapeaux d'activation du port des communications

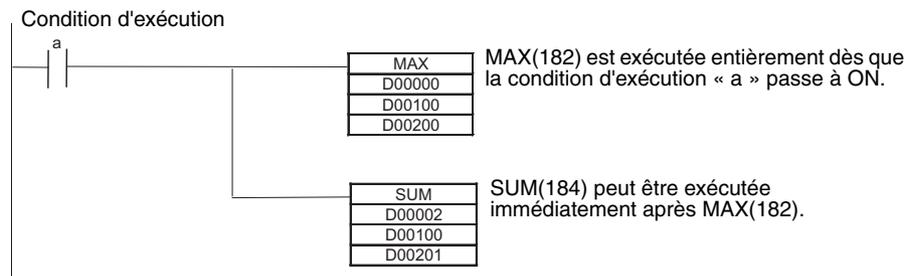
Les drapeaux d'activation du port des communications sont à ON lorsque le port n'est pas utilisé, et à OFF lorsqu'un traitement est en cours d'exécution sur le port.



### Exemple de programmation 1

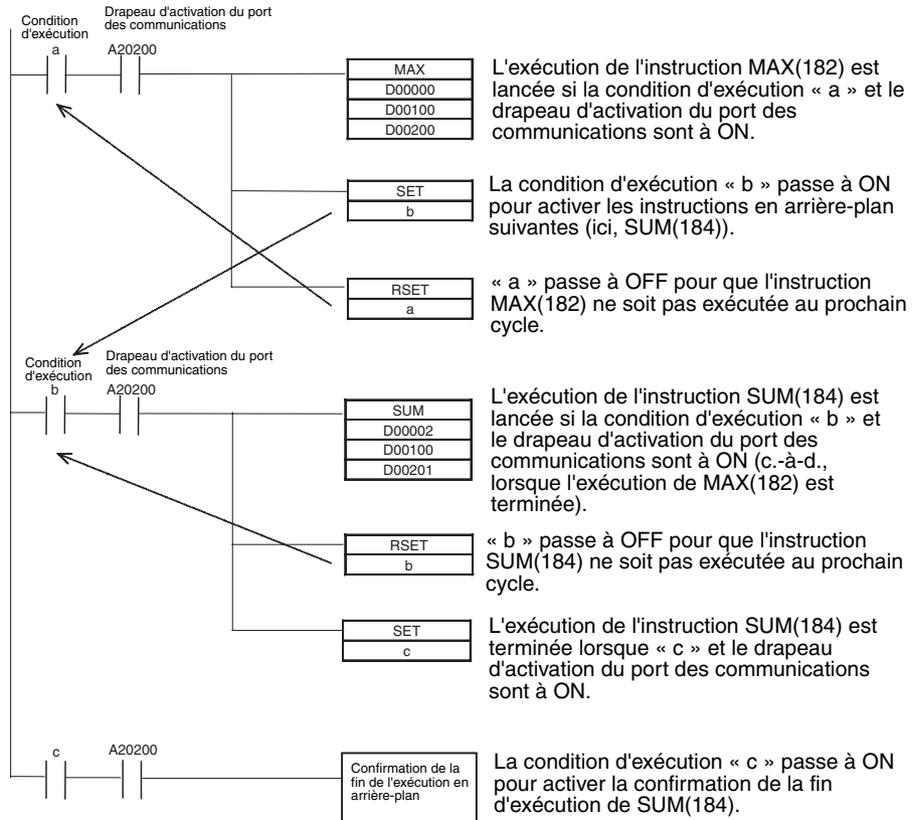
#### ■ Programmation classique sans exécution en arrière-plan

Comme indiqué ci-dessous, le traitement est terminé lorsque l'instruction est exécutée.



#### ■ Programmation avec exécution en arrière-plan

Avec l'exécution en arrière-plan, le programme est modifié de manière que MAX(182) ne soit exécutée que si le drapeau d'activation du port des communications est à ON (c.-à-d., uniquement lorsque le port n'est pas déjà utilisé pour une exécution en arrière-plan ou des communications réseau). Les conditions d'entrée sont également contrôlées à l'aide des instructions SET et RESET pour s'assurer que le traitement s'effectue dans l'ordre correct. (Dans l'exemple suivant, le port des communications 0 est utilisé pour l'exécution en arrière-plan).

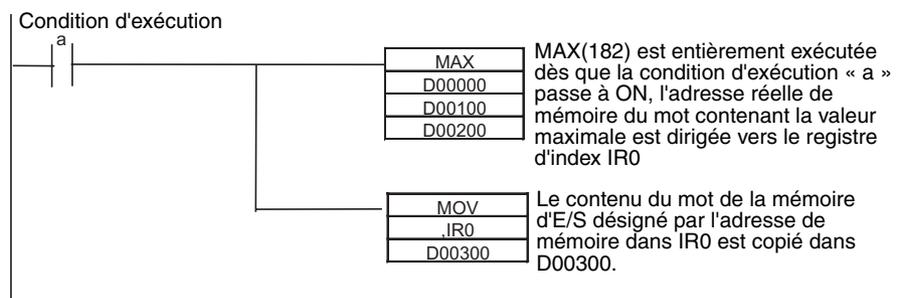


**Exemple de programmation 2**

Ces exemples décrivent l'exécution en arrière-plan lorsque vous avez spécifié la sortie vers un registre d'index, comme cela est possible avec MAX(182), MIN(183) et SRCH(181).

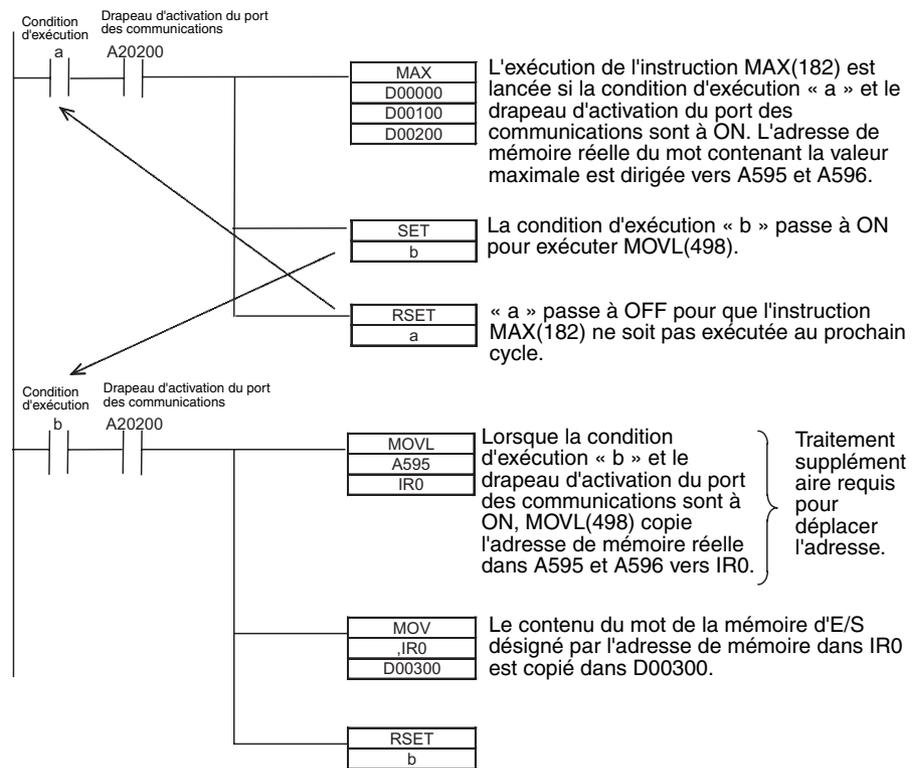
■ **Programmation classique sans exécution en arrière-plan**

Comme indiqué ci-dessous, l'adresse de mémoire réelle du mot contenant la valeur maximale est dirigée vers un registre d'index.



■ **Programmation avec exécution en arrière-plan**

Sans l'exécution en arrière-plan, l'adresse de mémoire réelle du mot contenant la valeur maximale est dirigée vers A595 et A596. MOVL(498) est alors utilisée pour diriger l'adresse de mémoire réelle vers le registre d'index.



### 6-1-11 Partage de registres d'index et de données entre des tâches

Le partage de registres d'index et de données (IR/DR) entre des tâches est uniquement pris en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D. La configuration normale spécifie des registres séparés pour chaque tâche. Vous pouvez confirmer la configuration actuelle dans A09914.

**Remarque**

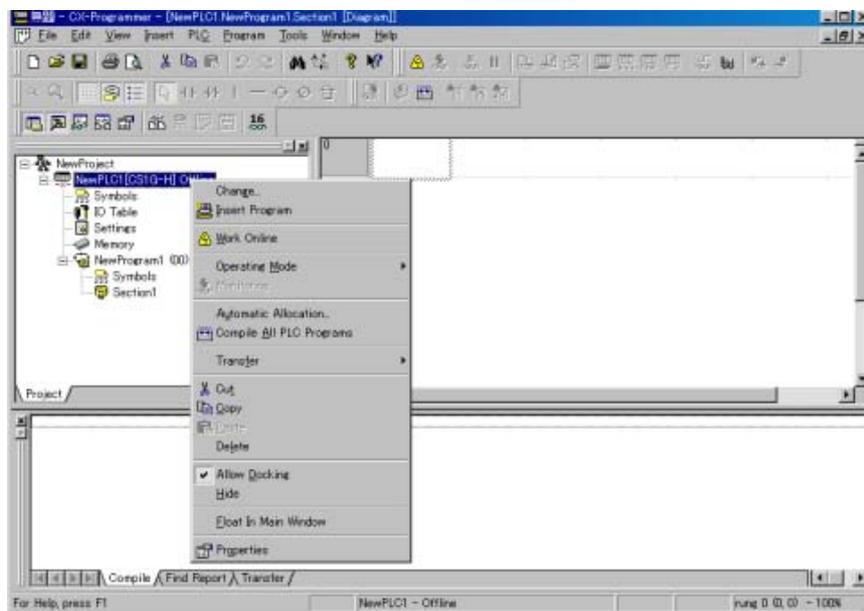
1. Les registres d'index et de données partagés permettent d'éliminer les opérations de stockage et de chargement du contenu des registres entre les tâches, lorsqu'un même contenu est requis dans plusieurs tâches. Consultez le chapitre sur les registres d'index dans le *Manuel d'utilisation, série CS (W339)* ou le *Manuel d'utilisation, série CJ (W393)*, pour plus d'informations sur le stockage et le chargement de contenu d'un registre d'index.
2. Le temps de commutation entre les tâches est un peu plus rapide lorsque les registres d'index et de données sont partagés. Il est recommandé de configurer des registres partagés si les registres ne sont pas en cours d'utilisation ou si des registres séparés ne sont pas absolument nécessaires dans chaque tâche.

**Méthode de configuration**

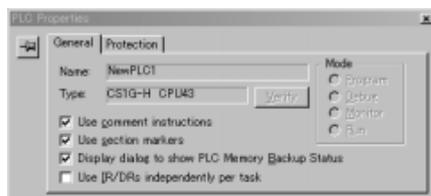
Configurez les registres d'index et de données à l'aide du CX-Programmer. Vous ne pouvez pas réaliser ce paramétrage à partir de la console de programmation.

**1,2,3...**

1. Sélectionnez un API (PLC) dans l'arborescence du projet du CX-Programmer et cliquez avec le bouton droit de la souris.



2. Sélectionnez **Propriétés**. La boîte de dialogue suivante s'affiche.



3. Laissez la coche pour permettre l'utilisation indépendante des registres IR/DR par tâche, si des registres d'index et de données séparés sont nécessaires pour chaque tâche. Retirez la coche pour utiliser des registres d'index et de données pour toutes les tâches.

**Drapeaux et mots de la zone auxiliaire**

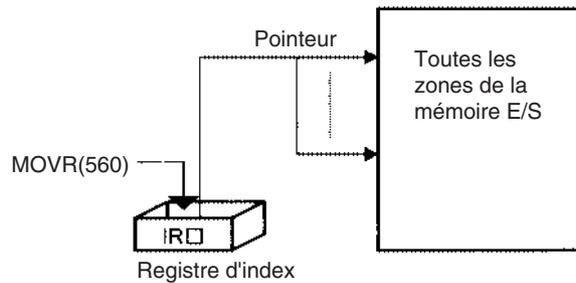
Nom	Adresse	Description
Fonctionnement de l'IR/DR entre tâches	A09914	Indique si des registres d'index et de données sont partagés entre les tâches. 0 : registres séparés pour chaque tâche (par défaut) 1 : registres partagés pour toutes les tâches

## 6-2 Registres d'index

### 6-2-1 Définition des registres d'index

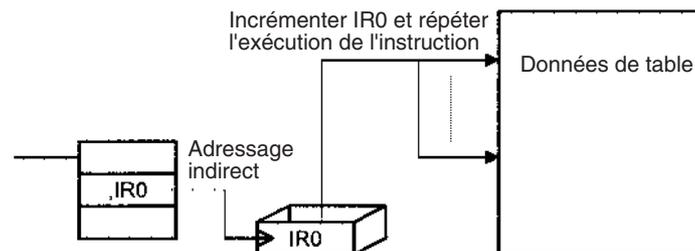
Les registres d'index fonctionnent comme des pointeurs permettant de spécifier des adresses de mémoire API, qui correspondent à des adresses de mémoire absolues dans la mémoire d'E/S. Après le stockage d'une adresse de la mémoire API dans un registre d'index à l'aide de l'instruction MOVR(560) ou MOVW(561), entrez le registre d'index sous forme d'opérande dans d'autres instructions pour adresser indirectement l'adresse de mémoire API stockée.

Les registres d'index ont l'avantage de permettre la spécification d'un bit ou d'un mot quelconque de la mémoire d'E/S, y compris les valeurs actuelles (PV) de temporisation et de compteur.



## 6-2-2 Utilisation des registres d'index

Les registres d'index peuvent se transformer en outil puissant s'ils sont associés à des boucles de type FOR-NEXT. Vous pouvez facilement incrémenter, décrémenter et décaler le contenu des registres d'index, et un petit nombre d'instructions d'une boucle peuvent ainsi traiter des tables de données consécutives avec une grande efficacité.



### Fonctionnement de base

En principe, les registres d'index s'utilisent en procédant de la manière suivante :

- 1,2,3...**
1. Utilisez MOVR(560) pour stocker l'adresse de mémoire API du bit ou du mot désiré dans un registre d'index.
  2. Spécifiez le registre d'index en tant qu'opérande dans presque toutes les instructions pour l'adressage indirect du bit ou du mot désiré.
  3. Décalez ou incrémentez l'adresse de mémoire API d'origine (voir plus bas) pour rediriger le pointeur vers une autre adresse.
  4. Poursuivez en reprenant les étapes 2 et 3 pour exécuter l'instruction sur le nombre d'adresses de votre choix.

### Décalage, incrémentation et décrément d'adresses

Le tableau suivant présente les variantes possibles pour l'adressage indirecte.

Variante	Syntaxe
Adressage indirect	,IR□
Adressage indirect avec décalage constant	Constant ,IR□ (Inclure un + ou un – dans la constante).
Adressage indirect avec décalage DR	DR□,IR□
Adressage indirect avec auto-incrémentation	Incrément de 1 : ,IR□+ Incrément de 2 : ,IR□++
Adressage indirect avec auto-décrément	Décrément de 1 : ,–IR□ Décrément de 2 : ,– –IR□

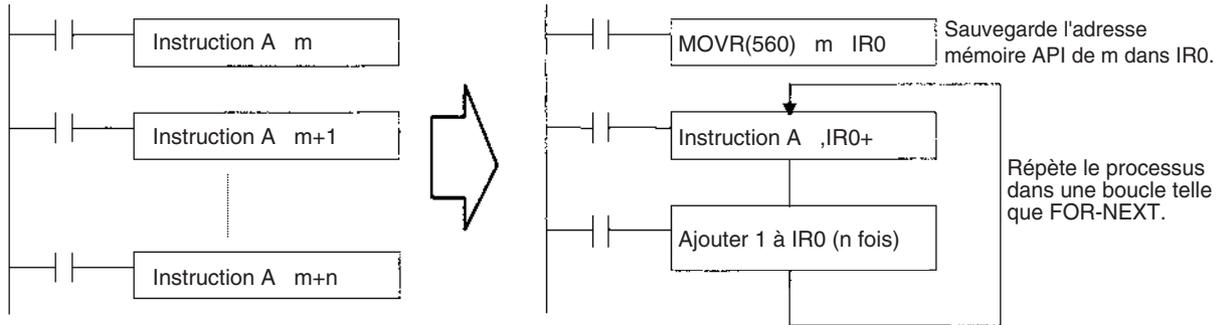
### Instructions d'adressage direct des registres d'index

Les registres d'index peuvent être adressés directement par les instructions suivantes :

DOUBLE SIGNED BINARY ADD WITHOUT CARRY: +L(401), DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY: –L(411), DOUBLE INCREMENT BINARY: ++L(591) et DOUBLE DECREMENT BINARY: – –L(593)

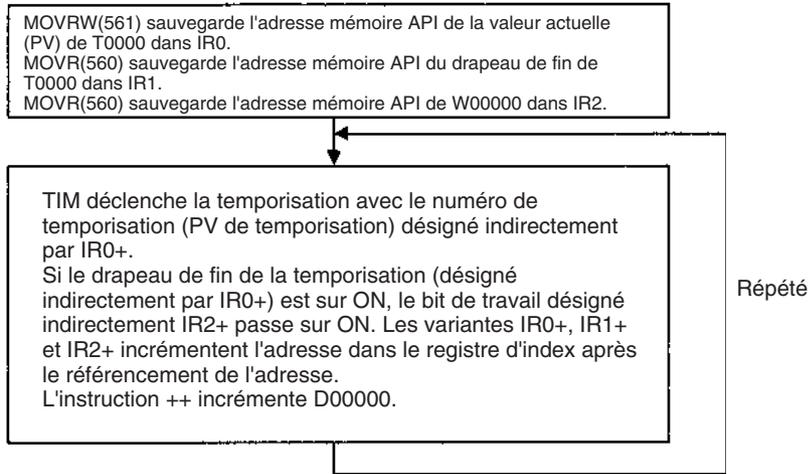
**Exemple 1**

L'exemple suivant décrit comment un registre d'index peut remplacer une longue série d'instructions, dans une boucle de programme. Dans ce cas, l'instruction A est répétée n+1 fois pour effectuer une opération comme, par exemple, lire et comparer les valeurs d'une table.

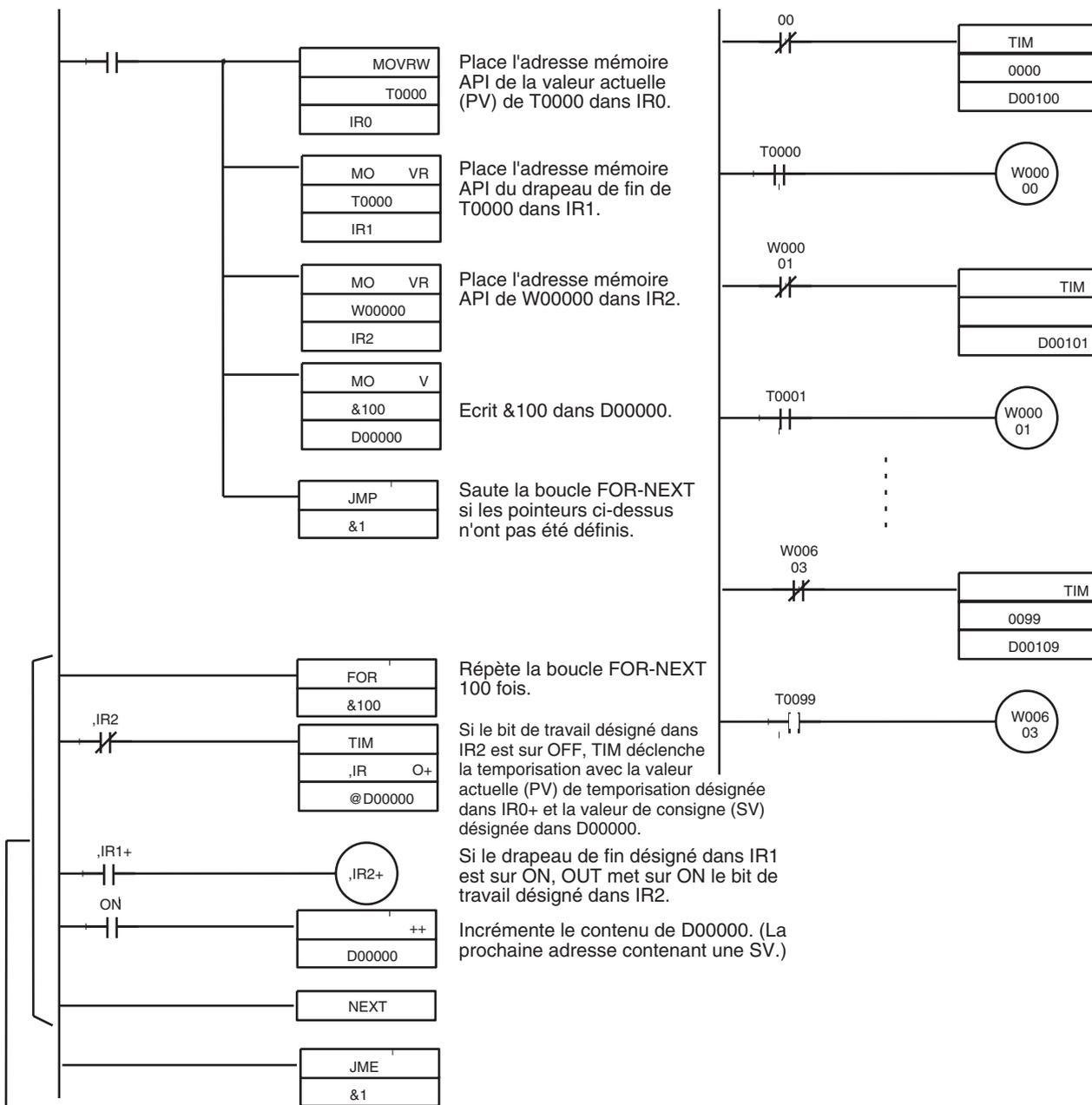


**Exemple 2**

Dans l'exemple suivant, des registres d'index sont utilisés dans une boucle FOR-NEXT pour définir et démarrer 100 temporisations (T0000 à T099) à l'aide des valeurs de consigne contenues dans la plage D00100 à D00109. Le numéro et le drapeau de fin de chaque temporisation sont spécifiés dans des registres d'index et la boucle se répète, tandis que les registres d'index sont incrémentés de 1 à chaque répétition.



Le sous-programme de 11 instructions, à gauche, est équivalent au sous-programme de 200 instructions, à droite.



La boucle FOR-NEXT déclenche les temporisations T0000 à T0099 en répétant la boucle 100 fois, tout en incrémentant le contenu de IR0 (numéro de temporisation/adresse de PV), IR1 (adresse du drapeau de fin), IR2 (adresse du bit de travail) et D00000 (adresse de SV).

**Adressage direct de registres d'index**

Les registres d'index peuvent être adressés directement uniquement dans les instructions répertoriées dans le tableau suivant.

Groupe d'instruction	Nom de l'instruction	Mnémonique	Fonction primaire
Instructions de déplacement de données	MOVE TO REGISTER (déplacer vers un registre)	MOVR(560)	Stocke l'adresse de mémoire API d'un bit ou d'un mot dans un registre d'index.
	MOVE TIMER/COUNTER PV TO REGISTER (déplacer la valeur actuelle de temporisation/compteur vers un registre)	MOVRW(561)	
Instructions de traitement de données de tables	SET RECORD LOCATION (configurer emplacement d'enregistrement)	SETR(635)	Extrait l'adresse de mémoire API stockée dans un registre d'index.
	GET RECORD NUMBER (obtention du numéro d'enregistrement)	GETR(636)	
Instructions de déplacement de données	DOUBLE MOVE (déplacement double)	MOVL(498)	Transferts entre des registres d'index. Utilisés pour les échanges et les comparaisons.
	DOUBLE DATA EXCHANGE (échange de données doubles)	XCGL(562)	
Instructions de comparaison	DOUBLE EQUAL (égalité double)	=L(301)	
	DOUBLE NOT EQUAL (non égalité double)	< >L(306)	
	DOUBLE LESS THAN (double inférieur à)	< L(311)	
	DOUBLE LESS THAN OR EQUAL (double inférieur ou égal à)	< =L(316)	
	DOUBLE GREATER THAN (double supérieur à)	>L(321)	
	DOUBLE GREATER THAN OR EQUAL (double supérieur ou égal à)	> =L(326)	
Instructions d'incrémentatation et de décrémentation	DOUBLE INCREMENT BINARY (binaire d'incrémentatation double)	++L(591)	Modifie l'adresse de mémoire API dans le registre d'index en incrémentant, décrémentation ou décalant son contenu.
	DOUBLE DECREMENT BINARY (binaire de décrémentation double)	--L(593)	
Instructions mathématiques à symboles	DOUBLE SIGNED BINARY ADD WITHOUT CARRY (addition de binaires doubles signés sans retenue)	+L(401)	
	DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY (soustraction de binaires doubles signés sans retenue)	-L(411)	
Instructions spéciales	CONVERT ADDRESS FROM CV	FRMCV(284)	Convertit des adresses réelles de mémoire API entre la série CV et la série CS/CJ. (UC CS1-H, CJ1-H, CJ1M ou CS1D CPU uniquement).
	CONVERT ADDRESS TO CV	TOCV(285)	

**Remarque** Les instructions pour opérantes de longueur double (par exemple, celles qui se terminent par « L ») s'utilisent avec les registres d'index IR0 à IR15, car chaque registre contient deux mots.

**6-2-3 Traitement associé aux registres d'index**

Les instructions de traitement de données de table de l'UC série CS/CJ complètent les fonctions des registres d'index. Ces instructions peuvent se diviser en deux groupes : les instructions de traitement de pile et les instructions de traitement de table.

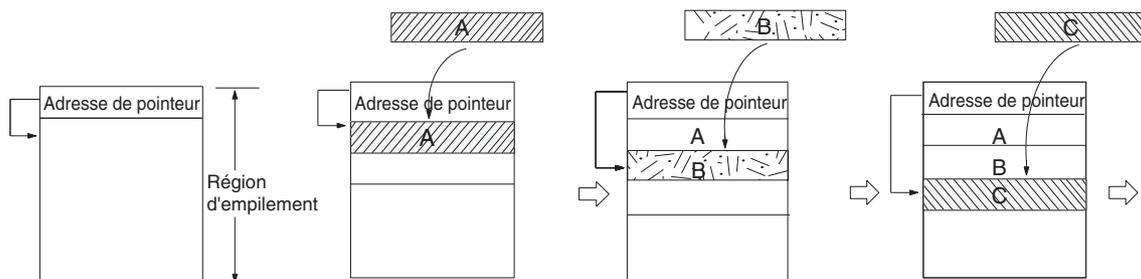
Traitement	Objet	Instructions
Traitement de pile	Exploiter des tables de données FIFO (premier entré, premier sorti) ou LIFO (dernier entré, premier sorti), et lire, écrire, insérer, supprimer ou compter les entrées de données dans les tables de données.	SSET(630), PUSH(632), FIFO(633), LIFO(634) et, pour les UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement, SREAD(639), SWRITE(640), SINS(641), SDEL(642), SNUM(638).

Traitement			Objet	Instructions
Traite- ment de tables	Tables d'enregis- trements d'un mot (Instructions de plage)	Traitement standard	Rechercher des valeurs telles que la checksum, une valeur particulière, la valeur maximale ou la valeur minimale de la plage.	FCS(180), SRCH(181), MAX(182), MIN(183) et SUM(184)
		Traitement spécial	Effectuer de nombreux autres traitements de tables tels que les comparaisons et le tri.	Associer des registres d'index à des instructions telles que SRCH(181), MAX(182), MIN(183), et des instructions de comparaison.
	Tables d'enregistrements de plusieurs mots (Instructions de table d'enregistrements)		Traiter des données dans des enregistrements composés de plusieurs mots.	Associer des registres d'index à des instructions telles que DIM(631), SETR(635), GETR(636), et des instructions de comparaison.

**Traitement de pile**

Les instructions de pile opèrent sur des tables de données spécialement définies, appelées piles. Les données peuvent être extraites d'une pile suivant la méthode du premier entré, premier sorti (FIFO), ou du dernier entré, premier sorti (LIFO).

Vous devez définir une région particulière de la mémoire d'E/S comme pile. Les premiers mots de la pile indiquent la longueur de la pile et contiennent le pointeur d'empilement. Ce dernier est incrémenté à chaque fois que des données sont écrites dans la pile pour indiquer la prochaine adresse de stockage des données.

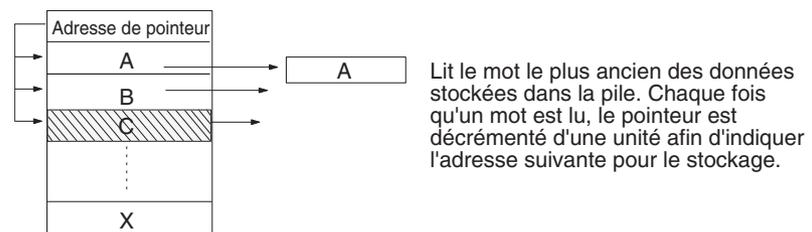


(Le diagramme ci-dessus montre l'état des données du pointeur avant l'ajout des données).

**Remarque** En réalité, les deux premiers mots de la pile contiennent l'adresse de mémoire API du dernier mot de la pile, et le mot suivant contient le pointeur d'empilement.

**Traitement FIFO (premier entré, premier sorti)**

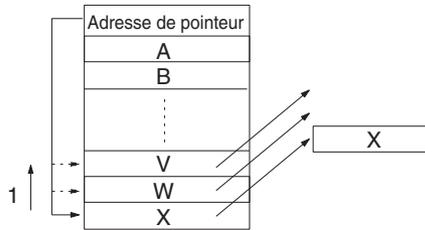
Le diagramme suivant illustre le fonctionnement d'une pile premier entré, premier sorti (FIFO).



Lit le mot le plus ancien des données stockées dans la pile. Chaque fois qu'un mot est lu, le pointeur est décrémenté d'une unité afin d'indiquer l'adresse suivante pour le stockage.

**Traitement LIFO (dernier entré, premier sorti)**

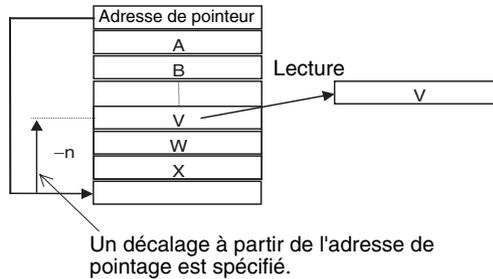
Le diagramme suivant illustre le fonctionnement d'une pile dernier entré, premier sorti (LIFO).



Lit le mot le plus récent des données stockées dans la pile. Chaque fois qu'un mot est lu, le pointeur est décrémenté d'une unité afin d'indiquer l'adresse suivante pour le stockage. Les données à la position qui a été lue restent inchangées.

**Gestion de données de tables spécifiques**

Vous pouvez lire, écrire, insérer ou supprimer différentes entrées dans une table. Le diagramme suivant montre un exemple de lecture.



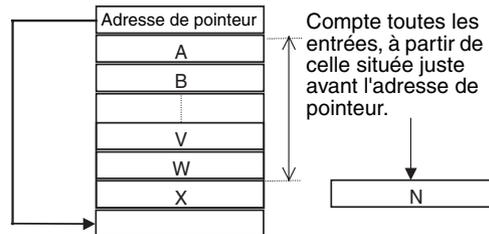
Les données sont lues à partir d'un décalage spécifique de l'adresse de pointage dans la table.

La gestion de données de tables spécifiques s'utilise, par exemple, pour tracer des éléments sur un convoyeur.

Un décalage à partir de l'adresse de pointage est spécifié.

**Comptage des données de tables**

Le diagramme suivant décrit la procédure de comptage des données d'une table.



Compte toutes les entrées, à partir de celle située juste avant l'adresse de pointeur.

Le nombre d'entrées dans la table de données est compté à partir de l'emplacement situé juste avant l'adresse du pointeur jusqu'au début de la table.

Ce traitement s'utilise, par exemple, pour compter le nombre d'éléments sur un convoyeur.

**Instructions d'empilement**

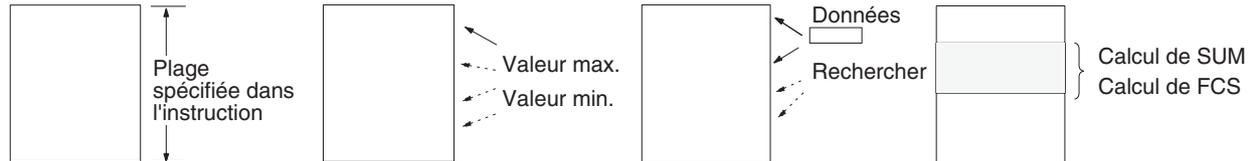
Le tableau suivant répertorie les instructions d'empilement et leurs fonctions. Exemples d'applications caractéristiques des piles : le traitement des informations relatives aux rayonnages pour les systèmes d'entreposage automatisés, le traitement des résultats de test et la gestion des informations relatives aux pièces de fabrication sur un convoyeur.

Instruction	Fonction
SSET(630)	Définit une région d'empilement.
PUSH(632)	Stocke des données dans le prochain mot disponible de la pile.
FIFO(633)	Lit des données de la pile suivant la méthode du premier entré, premier sorti.
LIFO(634)	Lit des données de la pile suivant la méthode du dernier entré, premier sorti.
SREAD(639)	Lit une entrée spécifique de la table (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement).
SWRITE(640)	Ecrit une entrée spécifique dans la table (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement).
SINS(641)	Insère une entrée spécifique dans la table (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement).
SDEL(642)	Supprime une entrée spécifique de la table (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement).
SNUM(638)	Compte le nombre d'entrées dans la table (UC CS1-H, CJ1-H, CJ1M ou CS1D uniquement).

**Traitement de tables (instructions de plage)**

Les instructions de plage opèrent sur une plage de mots, assimilable à une table d'enregistrements composés d'un mot. Ces instructions effectuent des opérations standard comme la recherche de la valeur maximale ou minimale de la plage, la recherche d'une valeur particulière de la plage ou le calcul de la somme ou FCS.

L'adresse de mémoire API du mot résultat (mot contenant la valeur maximale, la valeur minimale, les données de recherche, etc.) est automatiquement stockée dans IRO. Vous pouvez utiliser le registre d'index (IRO) en tant qu'opérande dans des instructions plus récentes, telles que MOV(021), pour lire le contenu du mot et effectuer un autre traitement.



Le tableau suivant répertorie les instructions de plage et leurs fonctions.

Instruction	Fonction	Description
SRCH(181)	Recherche des données (de recherche).	Recherche les données de recherche dans la plage spécifiée et dirige l'adresse de mémoire API du mot contenant cette valeur vers IRO.
MAX(182)	Recherche la valeur maximale.	Recherche la valeur maximale dans la plage spécifiée et dirige l'adresse de mémoire API du mot contenant cette valeur vers IRO.
MIN(183)	Recherche la valeur minimale.	Recherche la valeur minimale dans la plage spécifiée et dirige l'adresse de mémoire API du mot contenant cette valeur vers IRO.
SUM(184)	Calcule la somme.	Calcule la somme des données contenues dans la plage spécifiée.
FCS(180)	Calcule la checksum (somme de contrôle).	Calcule la somme de contrôle du cadre des données contenues dans la plage spécifiée.

Vous pouvez associer les registres d'index à d'autres instructions (par exemple, des instructions de comparaison) dans des boucles FOR-NEXT, pour réaliser des opérations plus compliquées sur des plages de mots.

**Traitement de tables (instructions de table d'enregistrements)**

Les instructions de table d'enregistrements opèrent sur des tables de données spécialement définies contenant des enregistrements de longueur égale. Pour faciliter le traitement, vous pouvez accéder aux enregistrements à l'aide d'un numéro d'enregistrement.

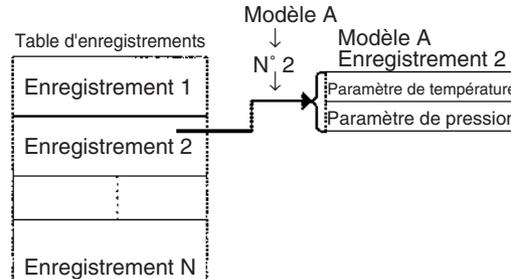
Instruction	Fonction	Description
DIM(631)	Définit une table d'enregistrements.	Déclare la longueur de chaque enregistrement et le nombre d'enregistrements.
SETR(635)	Définit l'emplacement de l'enregistrement.	Ecrit l'emplacement de l'enregistrement spécifié (l'adresse de mémoire API du début de l'enregistrement) dans le registre d'index spécifié.
GETR(636)	Recherche l'emplacement de l'enregistrement.	Renvoie le numéro de l'enregistrement contenant l'adresse de mémoire API dans le registre d'index spécifié.

**Remarque**

Les numéros des enregistrements et les adresses des mots sont liés par les registres d'index. Spécifiez un numéro d'enregistrement dans SETR(635) pour stocker l'adresse de mémoire API du début de cet enregistrement dans un registre d'index. Si vous avez besoin de données de cet enregistrement, ajoutez le décalage requis à ce registre d'index pour accéder à un mot quelconque de l'enregistrement.

Utilisez les instructions de table d'enregistrements avec des registres d'index pour exécuter des opérations du type : lecture/écriture de données d'enregistrements, recherche d'enregistrements, tri de données d'enregistrements, comparaison de données d'enregistrements et calculs avec des données d'enregistrements.

Une application caractéristique des tables d'enregistrements consiste à stocker les données de fabrication de différents modèles d'un produit (par exemple, les paramètres de température et de pression) sous forme d'enregistrements, et à passer d'un modèle à l'autre en modifiant simplement le numéro de l'enregistrement.



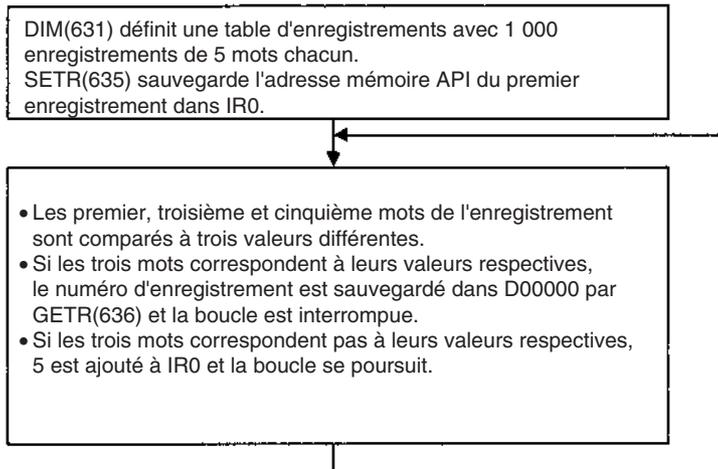
La procédure fondamentale d'utilisation des tables d'enregistrements est la suivante :

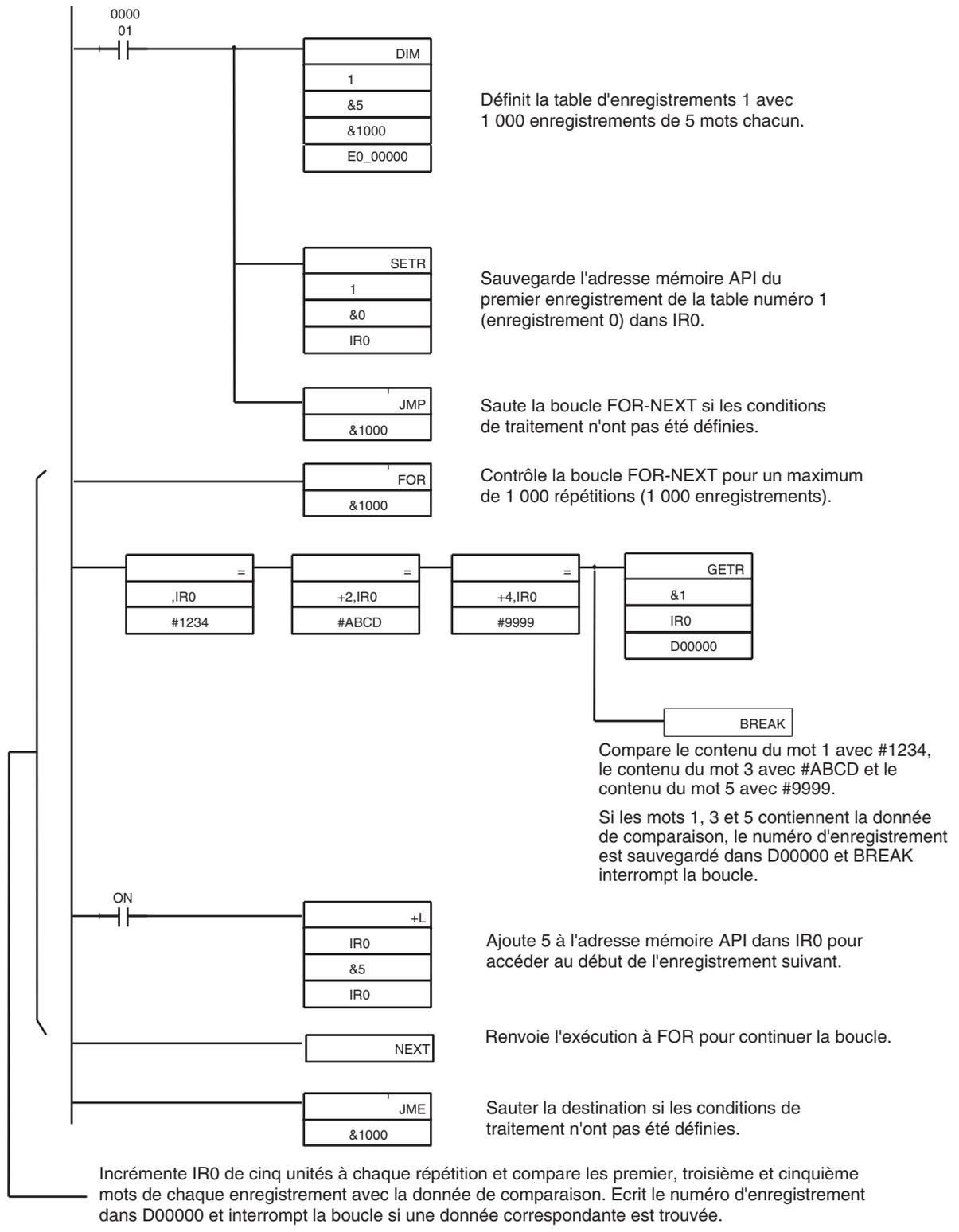
**1,2,3...**

1. Définissez la structure de la table d'enregistrements à l'aide de DIM(631) et configurez l'adresse de mémoire API d'un enregistrement dans un registre d'index à l'aide de SETR(635).
2. Décalez ou incrémentez l'adresse de mémoire API dans le registre d'index pour lire ou comparer des mots dans un enregistrement.
3. Décalez ou incrémentez l'adresse de mémoire API dans le registre d'index pour passer à un autre enregistrement.
4. Répétez les étapes 2 et 3 autant de fois que nécessaire.

### Exemple

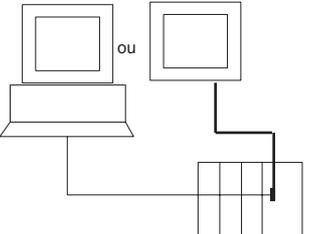
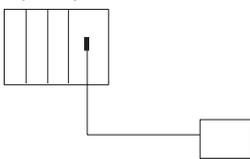
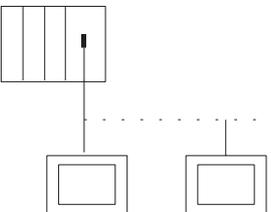
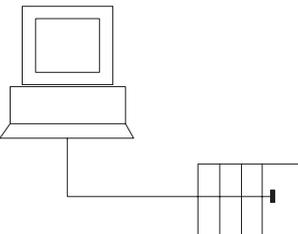
L'exemple suivant illustre l'utilisation des registres d'index et des instructions de table d'enregistrements pour comparer trois valeurs aux mots 1, 3 et 5 de chaque enregistrement. Si une correspondance est trouvée, le numéro de l'enregistrement est stocké dans D00000.





### 6-3 Communications en série

Les UC série CS/CJ prennent en charge les fonctions de communications en série suivantes : les communications Host Link et les communications sans protocole sont décrites en détail plus loin dans ce chapitre.

Protocole	Connexions	Description	Ports	
			Périphérique	RS-232C
Host Link	<p>Ordinateur hôte ou IHM OMRON (terminal opérateur programmable)</p> 	<p>1) Vous pouvez exécuter diverses commandes de contrôle comme la lecture et l'écriture dans la mémoire d'E/S, le changement de mode de fonctionnement et la configuration/réinitialisation forcée en émettant des commandes de liaison à l'ordinateur (Host Link) ou des commandes FINS de l'ordinateur hôte vers l'UC.</p> <p>2) Vous pouvez également émettre des commandes FINS de l'UC vers l'ordinateur, pour envoyer des données ou des informations.</p> <p>Utilisez les communications Host Link pour surveiller des données telles que l'état de fonctionnement, les informations sur les erreurs et les données de qualité dans l'API, ou pour envoyer des données, par exemple des informations de planning de production, vers l'API.</p>	OK	OK
Sans protocole	<p>Périphérique externe standard</p> 	<p>Permet de communiquer avec des périphériques standard connectés au port RS-232C sans format commande-réponse. Au lieu de cela, les instructions TXD(236) et RXD(235) sont exécutées à partir du programme pour transmettre des données depuis le port de transmission ou lire des données sur le port de réception. Vous pouvez spécifier les entêtes et les codes de fin de cadre.</p>	Non autorisé	OK
Liaison NT 1:N ou 1:1	<p>IHM OMRON (terminaux opérateur programmable)</p> 	<p>L'échange de données avec des IHM (terminaux opérateurs programmables) est possible sans utiliser un programme de communications dans l'UC.</p>	OK	OK
Bus périphérique	<p>Périphériques de programmation (pas les consoles de programmation)</p> 	<p>Assure des communications à grande vitesse avec des périphériques de programmation différents des consoles de programmation. (La programmation déportée via modems n'est pas prise en charge).</p>	OK	OK

Protocole	Connexions	Description	Ports	
			Périphérique	RS-232C
Passerelle série (conversion vers CompoWay/F) (version de carte 3.0 ou supérieure uniquement)		Convertit les commandes FINS reçues en commandes CompoWay/F et les transfère sur le chemin de communications série.	OK	OK
Liaisons API série (CJ1M uniquement).		9 UC maximum, comprenant une carte d'analyse et huit cartes interrogées, peuvent partager jusqu'à 10 mots par carte.  Un convertisseur RS-422A peut être connecté au port RS-232C de chaque UC pour communiquer via RS-422A/485, ou deux UC peuvent communiquer via une connexion RS-232C.  Les liaisons API en série peuvent également inclure des IHM sous forme de cartes interrogées via des liaisons NT Link (1:N) associées à des UC CJ1M.	Non autorisé	OK

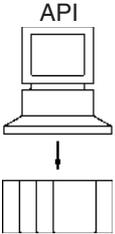
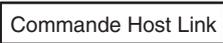
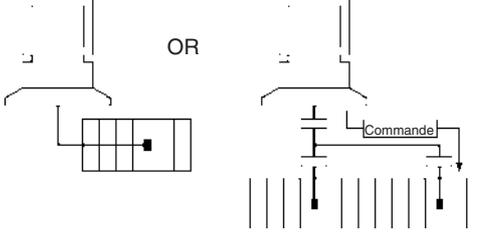
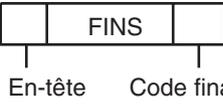
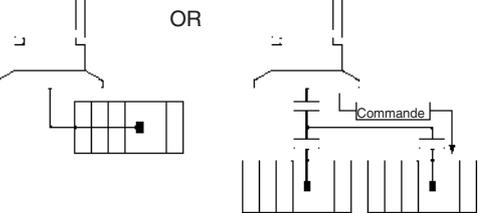
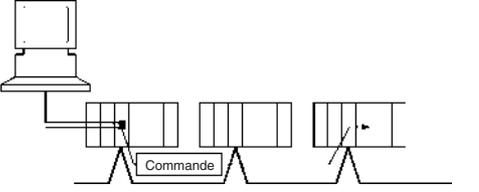
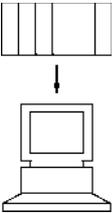
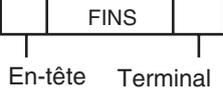
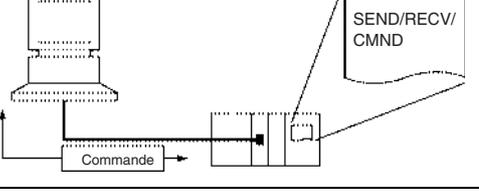
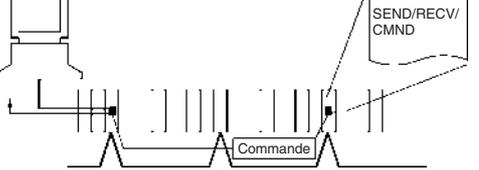
Vous trouverez ici une description des communications Host Link et sans protocole.

**Remarque**

L'adaptateur CJ1W-CIF11 n'est pas isolé et la distance de transmission totale est de 50 mètres maximum. Si la distance de transmission totale est supérieure à 50 mètres, utilisez l'adaptateur isolé NT-AL001 mais jamais le CJ1W-CIF11. Si vous utilisez uniquement l'adaptateur NT-AL001, la distance de transmission totale est de 500 mètres maximum.

### 6-3-1 Communications Host Link

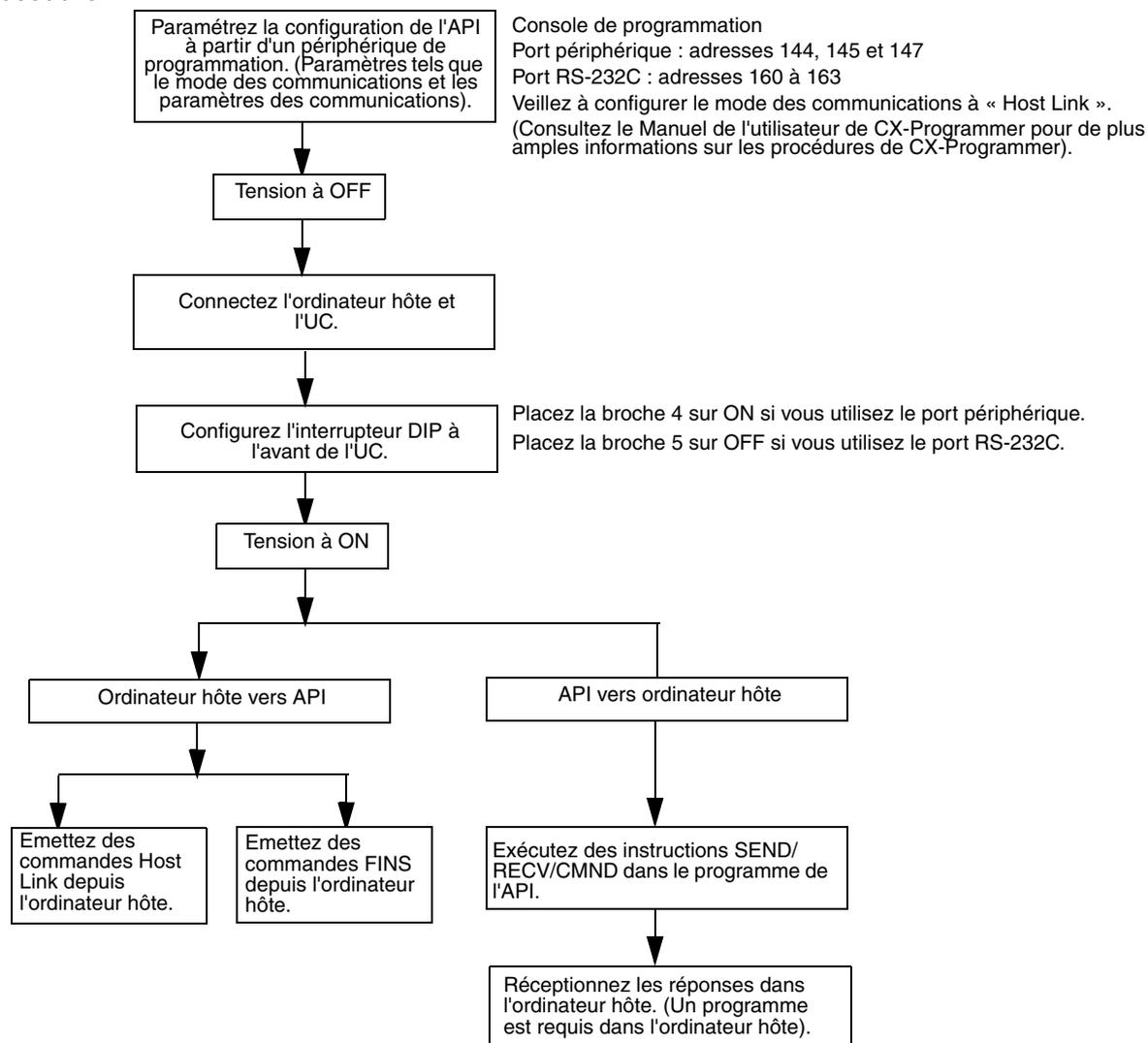
Le tableau suivant présente les fonctions de communication Host Link disponibles dans les API CS/CJ. Sélectionnez la méthode la mieux adaptée à votre application.

Flux de commandes	Type de commande	Méthode de communication	Configuration
Ordinateur hôte 	Commande Host Link 	Créez un cadre dans l'ordinateur hôte et émettez la commande vers l'API. Réceptionnez la réponse depuis l'API. Application : utilisez cette méthode lorsque les communications s'effectuent essentiellement de l'ordinateur hôte vers l'API.	Connexion directe de l'ordinateur dans un système 1:1 ou 1:N. 
	Commande <sup>1</sup> FINS (avec en-tête et code final Host Link)  En-tête    Code final	Créez un cadre dans l'ordinateur hôte et émettez la commande vers l'API. Réceptionnez la réponse depuis l'API. Application : utilisez ces méthodes lorsque les communications s'effectuent essentiellement de l'ordinateur hôte vers des API du réseau.	Connexion directe de l'ordinateur dans un système 1:1 ou 1:N. 
			Communication avec d'autres API sur le réseau à partir de l'ordinateur hôte. (Conversion de Host Link en protocole réseau.) 
API Ordinateur hôte 	Commande <sup>1</sup> FINS (avec en-tête et code final Host Link)  En-tête    Terminal	Emettez un cadre avec les instructions SEND/RCV/CMND de l'UC. Réceptionnez la réponse depuis l'ordinateur hôte. Application : utilisez cette méthode lorsque les communications s'effectuent essentiellement de l'API vers l'ordinateur hôte pour transmettre des informations d'état telles que des informations sur les erreurs.	Connexion directe de l'ordinateur dans un système 1:1. 
			Communication avec l'ordinateur hôte via d'autres API sur le réseau. (Conversion de Host Link en protocole réseau.) 

**Remarque** 1. Vous devez rattacher un en-tête et un code final Host Link de liaison à l'ordinateur hôte, à la commande FINS, avant que celle-ci ne soit transmise depuis l'ordinateur hôte.

- La commande FINS est transmise depuis l'API avec un en-tête et un code final Host Link (rattachés). Vous devez préparer, dans l'ordinateur hôte, un programme qui analyse les commandes FINS et renvoie les réponses appropriées.

**Procédure**



**Commandes Host Link**

Le tableau suivant répertorie les commandes Host Link. Consultez le *Manuel du système des cartes Host Link, série C (W143)* pour plus de détails.

Code en-tête	Nom	Fonction
RR	CIO AREA READ	Lit le contenu du nombre spécifié de mots de la zone CIO, à partir du mot spécifié.
RL	LINK AREA READ	Lit le contenu du nombre spécifié de mots de la zone de liaison, à partir du mot spécifié.
RH	HR AREA READ	Lit le contenu du nombre spécifié de mots de la zone de maintien, à partir du mot spécifié.
RC	PV READ	Lit le contenu du nombre spécifié de valeurs actuelles (PV) de temporisation/compteur, à partir de la temporisation ou du compteur spécifié(e).
RG	T/C STATUS READ	Lit l'état des drapeaux de fin du nombre spécifié de temporisations/compteurs, à partir de la temporisation ou du compteur spécifié(e).
RD	DM AREA READ	Lit le contenu du nombre spécifié de mots de la zone DM, à partir du mot spécifié.
RJ	AR AREA READ	Lit le contenu du nombre spécifié de mots de la zone auxiliaire, à partir du mot spécifié.
RE	EM AREA READ	Lit le contenu du nombre spécifié de mots de la zone EM, à partir du mot spécifié.

Code en-tête	Nom	Fonction
WR	CIO AREA WRITE	Ecrit les données spécifiées (unités de mots uniquement) dans la zone CIO, à partir du mot spécifié.
WL	LINK AREA WRITE	Ecrit les données spécifiées (unités de mots uniquement) dans la zone de liaison, à partir du mot spécifié.
WH	HR AREA WRITE	Ecrit les données spécifiées (unités de mots uniquement) dans la zone de maintien, à partir du mot spécifié.
WC	PV WRITE	Ecrit les valeurs actuelles (PV) du nombre spécifié de temporisations/compteurs, à partir de la temporisation ou du compteur spécifié(e).
WD	DM AREA WRITE	Ecrit les données spécifiées (unités de mots uniquement) dans la zone DM, à partir du mot spécifié.
WJ	AR AREA WRITE	Ecrit les données spécifiées (unités de mots uniquement) dans la zone auxiliaire, à partir du mot spécifié.
WE	EM AREA WRITE	Ecrit les données spécifiées (unités de mots uniquement) dans la zone EM, à partir du mot spécifié.
R#	SV READ 1	Lit la constante BCD à 4 chiffres ou l'adresse mot dans la valeur de consigne de l'instruction de temporisation/compteur spécifiée.
R\$	SV READ 2	Recherche l'instruction de temporisation/compteur spécifiée, à partir de l'adresse de programme spécifiée, et lit la constante à 4 chiffres ou l'adresse mot dans la valeur de consigne.
R%	SV READ 3	Recherche l'instruction de temporisation/compteur spécifiée, à partir de l'adresse de programme spécifiée, et lit la constante BCD à 4 chiffres ou l'adresse mot dans la valeur de consigne.
W#	SV CHANGE 1	Modifie la constante BCD à 4 chiffres ou l'adresse mot dans la valeur de consigne de l'instruction de temporisation/compteur spécifiée.
W\$	SV CHANGE 2	Recherche l'instruction de temporisation/compteur spécifiée, à partir de l'adresse de programme spécifiée, et modifie la constante à 4 chiffres ou l'adresse mot dans la valeur de consigne.
W%	SV CHANGE 3	Recherche l'instruction de temporisation/compteur spécifiée, à partir de l'adresse de programme spécifiée, et modifie la constante à 4 chiffres ou l'adresse mot dans la valeur de consigne.
MS	STATUS READ	Lit l'état de fonctionnement de l'UC (mode de fonctionnement, état de configuration/réinitialisation forcée, état d'erreur fatale).
SC	STATUS CHANGE	Modifie le mode de fonctionnement de l'UC.
MF	ERROR READ	Lit et supprime des erreurs dans l'UC (non-fatales et fatales).
KS	FORCE SET	Met le bit spécifié en configuration forcée.
KR	FORCE RESET	Met le bit spécifié en réinitialisation forcée.
FK	MULTIPLE FORCE SET/RESET	Met les bits spécifiés en configuration/réinitialisation forcée ou remet l'état forcé des bits spécifiés à 0.
KC	FORCE SET/RESET CANCEL	Annule l'état forcé de tous les bits en configuration/réinitialisation forcée.
MM	PLC MODEL READ	Lit le type de modèle de l'API.
TS	TEST	Renvoie, sans le modifier, un bloc de données transmis depuis l'ordinateur hôte.
RP	PROGRAM READ	Lit le contenu de la zone de programme utilisateur de l'UC en langage machine (code objet).
WP	PROGRAM WRITE	Ecrit le programme en langage machine (code objet), transmis depuis l'ordinateur hôte, dans la zone de programme utilisateur de l'UC.
MI	I/O TABLE GENERATE	Crée une table d'E/S enregistrées à l'aide d'une table d'E/S réelles.
QQMR	COMPOUND COMMAND	Enregistre les bits et les mots désirés dans une table.
QQIR	COMPOUND READ	Lit les mots et les bits enregistrés à partir de la mémoire d'E/S.
XZ	ABORT (commande uniquement)	Interrompt la commande Host Link en cours de traitement.

Code en-tête	Nom	Fonction
**	INITIALIZE (commande uniquement)	Initialise la procédure de contrôle de la transmission de tous les API connectés à l'ordinateur hôte.
IC	Commande non définie (réponse uniquement)	Cette réponse est renvoyée si le code en-tête d'une commande n'a pas été reconnu.

**Commandes FINS**

Le tableau suivant répertorie les commandes FINS. Consultez le *Manuel de référence des commandes FINS* pour plus de détails.

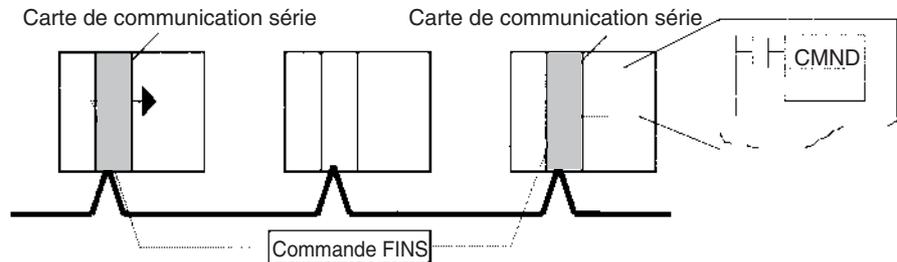
Type	Code de la commande		Nom	Fonction
Accès à la zone de mémoire d'E/S	01	01	MEMORY AREA READ	Lit des données consécutives de la zone de mémoire d'E/S.
	01	02	MEMORY AREA WRITE	Ecrit des données consécutives dans la zone de mémoire d'E/S.
	01	03	MEMORY AREA FILL	Remplit la plage spécifiée de la mémoire d'E/S avec les mêmes données.
	01	04	MULTIPLE MEMORY AREA READ	Lit des données non consécutives de la mémoire d'E/S.
	01	05	MEMORY AREA TRANSFER	Copie et transfère des données consécutives d'une partie de la zone de mémoire d'E/S vers une autre partie.
Accès à la zone de paramètres	02	01	PARAMETER AREA READ	Lit des données consécutives de la zone de paramètres.
	02	02	PARAMETER AREA WRITE	Ecrit des données consécutives dans la zone de paramètres.
	02	03	PARAMETER AREA FILL	Remplit la plage spécifiée de la zone de paramètres avec les mêmes données.
Accès à la zone de programme	03	06	PROGRAM AREA READ	Lit des données de la zone de programme utilisateur.
	03	07	PROGRAM AREA WRITE	Ecrit des données dans la zone de programme utilisateur.
	03	08	PROGRAM AREA CLEAR	Efface la plage spécifiée de la zone de programme utilisateur.
Contrôle de l'exécution	04	01	RUN	Commute l'UC au mode RUN, MONITOR ou DEBUG.
	04	02	SIHM	Commute l'UC au mode PROGRAMME.
Lecture de la configuration	05	01	CONTROLLER DATA READ	Lit des informations relatives à l'UC.
	05	02	CONNECTION DATA READ	Lit les numéros de modèle des cartes spécifiées.
Lecture de l'état	06	01	CONTROLLER STATUS READ	Lit les informations d'état de l'UC.
	06	20	CYCLE TIME READ	Lit les temps de cycle moyens, maximums et minimums.
Accès à l'horloge	07	01	CLOCK READ	Lit l'horloge.
	07	02	CLOCK WRITE	Règle l'horloge.
Accès aux messages	09	20	MESSAGE READ/CLEAR	Lit/efface des messages et des messages FAL(S).
Droit d'accès	0C	01	ACCESS RIGHT ACQUIRE	Acquiert le droit d'accès si aucun autre périphérique ne le détient.
	0C	02	ACCESS RIGHT FORCED ACQUIRE	Acquiert le droit d'accès même si un autre périphérique le détient.
	0C	03	ACCESS RIGHT RELEASE	Cède le droit d'accès quel que soit le périphérique qui le détient.
Accès aux erreurs	21	01	ERROR CLEAR	Efface des erreurs et des messages d'erreur.
	21	02	ERROR LOG READ	Lit le journal d'erreurs.
	21	03	ERROR LOG CLEAR	Remet le pointeur du journal d'erreurs à zéro.

Type	Code de la commande		Nom	Fonction
Mémoire de fichiers	22	01	FILE NAME READ	Lit les informations de fichier de la mémoire de fichiers.
	22	02	SINGLE FILE READ	Lit le volume de données spécifié à partir du point spécifié dans un fichier.
	22	03	SINGLE FILE WRITE	Ecrit le volume de données spécifié à partir du point spécifié dans un fichier.
	22	04	FILE MEMORY FORMAT	Formate la mémoire de fichiers.
	22	05	FILE DELETE	Supprime les fichiers spécifiés de la mémoire de fichiers.
	22	07	FILE COPY	Copie un fichier dans la mémoire de fichiers ou entre deux périphériques de mémoire de fichiers d'un système.
	22	08	FILE NAME CHANGE	Modifie un nom de fichier.
	22	0A	I/O MEMORY AREA FILE TRANSFER	Transfert ou compare des données entre la zone de mémoire d'E/S et la mémoire de fichiers.
	22	0B	PARAMETER AREA FILE TRANSFER	Transfert ou compare des données entre la zone de paramètres et la mémoire de fichiers.
	22	0C	PROGRAM AREA FILE TRANSFER	Transfert ou compare des données entre la zone de programme et la mémoire de fichiers.
	22	15	CREATE/DELETE DIRECTORY	Crée ou supprime un répertoire.
Etat forcé	23	01	FORCED SET/RESET	Force la configuration/réinitialisation ou remet l'état forcé du bit spécifié à 0.
	23	02	FORCED SET/RESET CANCEL	Annule l'état forcé de tous les bits en configuration/réinitialisation forcée.

**Fonctions de communication de messages**

Les commandes FINS répertoriées dans le tableau ci-dessus peuvent également être transmises, via le réseau, depuis d'autres API vers l'UC. Observez les consignes suivantes lorsque vous transmettez des commandes FINS via le réseau.

- Les cartes réseau (par exemple, les cartes Controller Link ou Ethernet) doivent être montées dans l'API local et l'API de destination pour permettre la transmission des commandes FINS.
- Les commandes FINS sont émises à l'aide de l'instruction CMND(490) du programme de l'UC.
- Les commandes FINS peuvent être transmises à travers huit réseaux maximum, dans le cas des UC série CS/CJ version 2.0 ou supérieure, et à travers trois réseaux maximum, dans le cas des autres UC. Il peut s'agir de réseaux de même type ou de types différents.



Consultez le Manuel d'utilisation des cartes réseau pour plus de détails sur les fonctions de communication de messages.

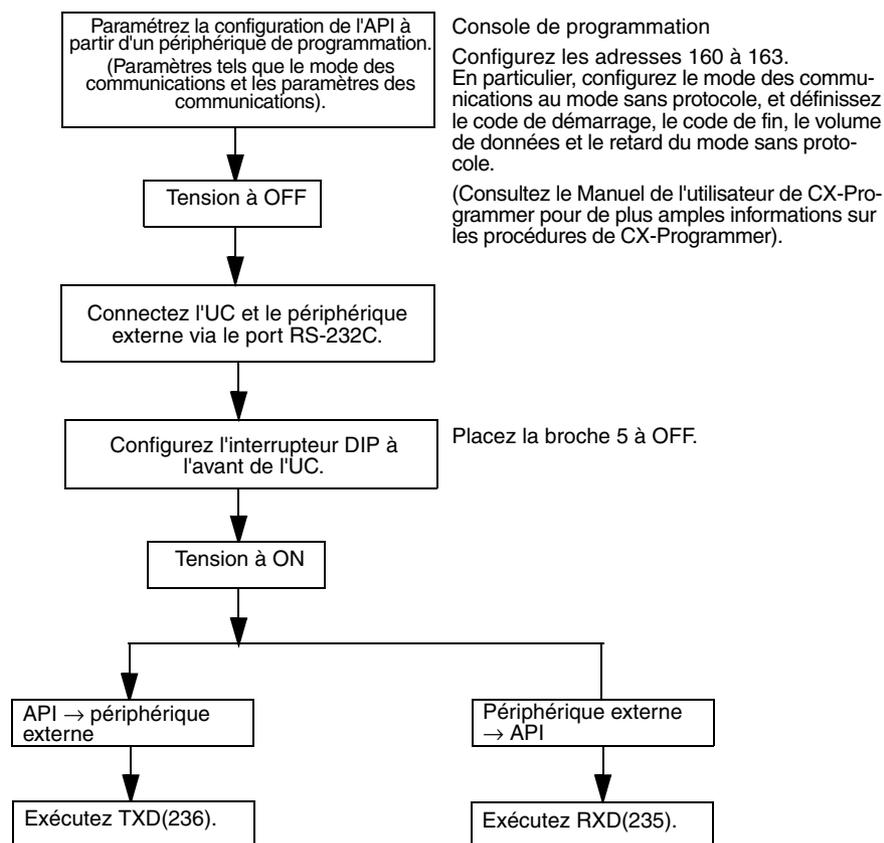
### 6-3-2 Communications sans protocole

Le tableau suivant répertorie les fonctions de communication sans protocole disponibles dans les API CS/CJ.

Sens de transfert	Méthode	Volume max. de données	Format de synchronisation		Autres fonctions
			Code de démarrage	Code de fin	
Transmission de données (API → périphérique externe)	Exécution de TXD(236) dans le programme*	256 octets	Oui : 00 à FF Non : aucune	Oui : 00 à FF ou CR+LF Non : aucun	Temps de retard envoi (retard entre l'exécution de TXD et l'envoi de données depuis le port spécifié) : 0 à 99 990 ms (pas : 10 ms)
Réception de données (Périphérique externe → API)	Exécution de RXD(235) dans le programme	256 octets			---

**Remarque** Vous pouvez spécifier un retard de transmission ou « retard de mode sans protocole » dans la configuration de l'API (adresse 162). Ce paramètre génère un retard de l'ordre de 30 secondes (maximum) entre l'exécution de l'instruction TXD (236) et la transmission des données depuis le port spécifié.

#### Procédure



#### Formats des cadres de messages

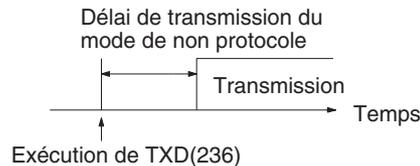
Des données peuvent être placées entre un code de démarrage et un code de fin pour la transmission par TXD(236) et RXD(235) peut recevoir des cadres dans ce même format. Lors d'une transmission à l'aide de TXD(236), seules les données de la mémoire d'E/S sont transmises, et lors de la réception à l'aide de RXD(235), seules les données sont stockées dans la mémoire d'E/S. Vous pouvez transférer jusqu'à 256 octets (y compris les codes de démarrage et de fin) en mode sans protocole.

Le tableau suivant répertorie les formats de messages configurables pour les transmissions et les réceptions en mode sans protocole. Le format est fonction des paramètres du code de démarrage (ST) et du code de fin (ED) dans la configuration de l'API.

Paramètre du code de démarrage	Paramètre du code de fin		
	Non	Oui	CR+LF
Non	données (données : 256 octets max.)	données+ED (données : 255 octets max.)	données+CR+LF (données : 254 octets max.)
Oui	ST+données (données : 255 octets max.)	ST+données+ED (données : 254 octets max.)	ST+données+CR+LF (données : 253 octets max.)

- Si vous utilisez plusieurs codes de démarrage, le premier code de démarrage sera effectif.
- Si vous utilisez plusieurs codes de fin, le premier code de fin sera effectif.

- Remarque**
1. Si les données en cours de transfert contiennent le code de fin, le transfert de données s'arrête à la moitié. Dans ce cas, remplacez le code de fin par CR+LF.
  2. Il existe un paramètre dans la configuration de l'API (adresse 162 : retard du mode sans protocole) qui tempore la transmission de données après l'exécution de TXD(236).



Consultez le *Manuel de référence des instructions pour les automates programmables série CJ (W340)* pour plus de détails sur les instructions TXD(236) et RXD(235).

### 6-3-3 Liaison NT (mode 1:N)

Avec la série CS/CJ, il est possible de communiquer avec les IHM (terminaux opérateurs programmables) à l'aide de liaisons NT (mode 1:N).

- Remarque** Il est impossible de communiquer à l'aide du protocole NT Link en mode 1:1.

Les liaisons NT à grande vitesse sont possibles, en plus des liaisons NT standard précédentes, en utilisant le menu système du terminal opérateur et les paramètres de configuration de l'API répertoriés ci-après (non pris en charge par les UC pré-EV1 CS1 série CS). Les liaisons NT à grande vitesse sont possibles, mais uniquement avec les terminaux opérateurs NT31(C)-V2 ou NT631(C)-V2.

Configuration de l'API

Port des communications	Adresse du paramètre de la console de programmation	Nom	Contenu des paramètres	Valeurs par défaut	Autres conditions
Port périphérique	144 Octets : 8 à 11	Mode des communications en série	02 hex. : NT Link (mode 1:N)	00 hex. : Host Link	Placez la broche 4 de l'interrupteur DIP de l'UC sur ON.
	145 Bits : 0 à 7	Vitesse de transmission	00 à 09 hex. : NT Link standard 0A hex. : NT Link à grande vitesse (voir remarque 1)	00 hex. : NT Link standard	
	150 Bits : 0 à 3	nombre maximum de cartes en mode NT Link	0 à 7 hex.	0 hex. (nombre de cartes max. 0)	
Port RS-232C	160 Bits : 8 à 11	Mode des communications en série	02 hex. : NT Link (mode 1:N)	00 hex. : Host Link	Placez la broche 5 de l'interrupteur DIP de l'UC sur OFF.
	161 Bits : 0 à 7	Vitesse de transmission	00 à 09 hex. : NT Link standard 0A hex. : NT Link à grande vitesse (voir remarque 1)	00 hex. : NT Link standard	
	166 Bits : 0 à 3	nombre maximum de cartes en mode NT Link	0 à 7 hex.	0 hex. (nombre max. de cartes 0)	

**Remarque** Configurez la vitesse de transmission à 115 200 bps lors du réglage des paramètres avec CX-Programmer.

**Menu système du terminal opérateur** Configurez le terminal opérateur en procédant comme suit :

- 1,2,3... 1. Sélectionnez NT Link (1:N) à partir de la méthode Comm. A ou Comm. B dans le menu de l'interrupteur mémoire sous le menu système de la carte du terminal opérateur.
2. Appuyez sur la touche SET pour configurer la vitesse de communication à « grande vitesse ».

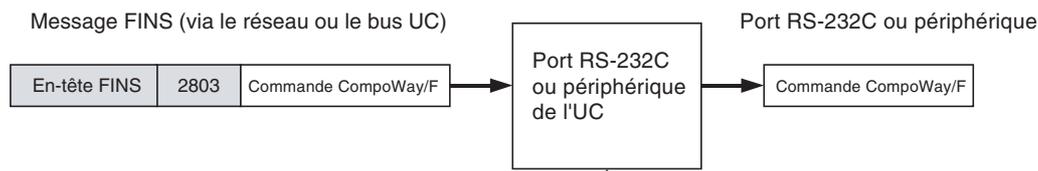
### 6-3-4 Passerelle série de l'UC

#### Présentation de la Passerelle série

Les messages FINS (commandes) qui sont reçus sont automatiquement convertis dans le protocole spécifié, puis envoyés via les communications série. Les réponses sont également converties automatiquement. Lorsque le port RS-232C ou le port périphérique de l'UC est utilisé, les messages FINS peuvent être convertis dans le protocole suivant.

- CompoWay/F

La Passerelle série est activée lorsque le mode de communications série est configuré sur Passerelle série.



La Passerelle série est activée lorsque le port RS-232C ou le port périphérique est configuré sur le mode Passerelle série.

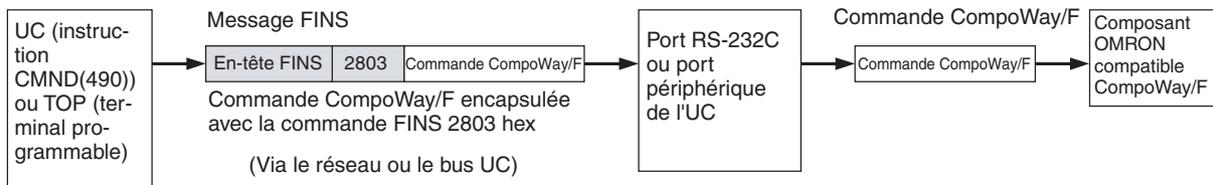
**Caractéristiques techniques de la Passerelle série**

Élément	Détails
Source de la conversion	Commande FINS (reçue via réseau FINS, Host Link FINS, bus périphérique, liaisons NT ou bus UC)
Fonction de conversion	Les commandes FINS reçues sont converties en fonction des valeurs suivantes avant d'être envoyées au port série (port périphérique ou RS-232C) de l'UC. 2803 hex. : supprime l'en-tête FINS et convertit la commande en commande CompoWay/F
Après la conversion	Commandes CompoWay/F
Méthode de communications série	Communications 1:N semi-duplex
Nombre maximal de cartes connectées	31 esclaves
Modes de communications série pris en charge	Mode Passerelle série
Surveillance du délai de réponse	Le délai entre l'envoi d'un message converti dans le protocole CompoWay/F avec la Passerelle série et la réception d'une réponse du périphérique distant est surveillé (activé en mode Passerelle série ou en mode Macro protocole). Par défaut : 5 s ; Plage de réglage : 0,1 à 25,5 s <b>Remarque</b> En cas de dépassement du temps imparti, le code de fin FINS est retourné à la source de la commande FINS (0205 hex : Délai de réponse).
Délai d'envoi	Aucun

**Conversion de FINS en CompoWay/F**

Les composants OMRON connectés en série au port RS-232C ou au port périphérique de l'UC via CompoWay/F sont accessibles à partir de l'API ou du TOP avec des commandes CompoWay/F incluses dans les messages FINS.

- Message FINS envoyé : en-tête FINS + code de commande FINS 2803 hex + commande CompoWay/F
- Message après conversion : Commande CompoWay/F



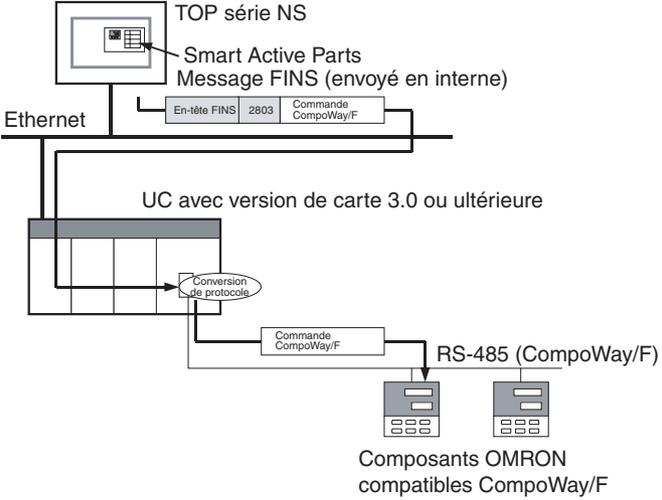
Pour des informations détaillées sur le code de commande FINS 2803 hex, reportez-vous au *Manuel de référence des commandes de communication* (W342).

Composants compatibles CompoWay/F esclave

Composant		Série du modèle
Régulateurs de température	Thermac NEO	E5GN (composants G)
		E5CN
		E5EN
		E5AN
	Thermac R	E5AR
		E5ER
	Régulateurs de température plug-in	E5ZN
Cartes de contrôleur numérique	E5ZM	
Contrôleurs numériques	ES100X	
Temporisation/compteurs	Temporisateurs/compteurs	H8GN (composants G)
Afficheurs numériques	Afficheurs numériques	K3GN (composants G) K3NX
	Compteurs de cellules de charge numériques	K3NV
	Compteurs rotatifs/d'impulsions numériques	K3NR
	Afficheurs numériques à incrémentation	K3NP
	Chronomicromètres numériques	K3NC
	Compteurs de température/processus numériques	K3NH
Capteurs intelligents	Cartes d'interface de communications ZX	ZX-SF11
Positionneurs de came	---	3F88L-160, 3F88L-162
Contrôleurs de sécurité	---	F3SX

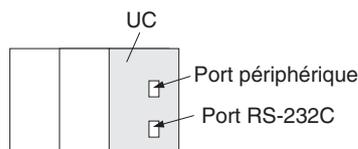
**Modèles de configuration système**

**Exécution de Smart Active Parts avec un TOP série NS (envoi de messages FINS internes)** L'UC convertit les messages FINS dans le protocole CompoWay/F pour demander cette opération.

Accès depuis le TOP sur Ethernet ou la liaison NT série	Détails	Tables de routage pour traiter le chemin de communications série comme un réseau
 <p><b>Remarque</b> Lorsque le TOP série NS est connecté en série à l'API en mode de communications série (liaisons NT 1:N) et qu'il envoie des commandes FINS encapsulées dans les commandes NT Link avec Smart Active Parts, l'UC supprime l'en-tête NT Link, etc. de la commande reçue, le convertit en commande FINS et transfère cette dernière au port série de l'UC. L'UC utilise la Passerelle série pour convertir la commande dans le protocole spécifié. Cette opération permet aux périphériques connectés en série d'accéder au port série de l'UC à partir de Smart Active Parts avec un TOP série NS.</p>	<p>L'accès via les communications série avec CompoWay/F est possible à partir d'un TOP connecté au réseau ; vous devez à cet effet exécuter un Smart Active Part connecté en série, qui envoie automatiquement une commande FINS interne.</p>	<p>En option</p>

**Remarque**

1. L'en-tête FINS contient les informations suivantes.
  - Adresse du réseau de destination déporté (DNA)
    - Avec des tables de routage qui traitent le chemin de communications série comme un réseau : Adresse réseau correspondant au port série dans les tables de routage.
    - Sans tables de routage qui traitent le chemin de communications série comme un réseau : Adresse réseau pour spécifier l'API déporté réel.
  - Adresse de nœud de destination déporté (DA1)
    - Avec des tables de routage qui traitent le chemin de communications série comme un réseau : 00 hex (communications internes de l'API local)
    - Sans tables de routage qui traitent le chemin de communications série comme un réseau : adresse de nœud pour spécifier l'API déporté réel.
  - Adresse d'unité de destination déportée (DA2)  
Adresse d'unité du port série



Port série de l'UC	Adresse d'unité du port série
Port périphérique	FD hex (253 décimal)
Port RS-232C	FC hex (252 décimal)

- Le contenu de la commande CompoWay/F incluse dans le message FINS qui est envoyé se présente de la manière suivante :  
 numéro de nœud + sous-adresse + SID + texte de la commande (ASCII indispensable.)  
 STX, ETX+BCC ne sont pas obligatoires lors de l'envoi du message FINS. Ils sont ajoutés automatiquement pour les communications série.

**Envoi de messages FINS avec CMND(490) dans le schéma contact de l'UC**

L'UC convertit les messages FINS dans le protocole CompoWay/F pour demander cette opération.

**Accès à partir de l'UC (sur le même API)**

Conversion série	Détails	Tables de routage pour traiter le chemin de communications série comme un réseau
<p>UC avec version de carte 3.0 ou supérieure                  Port RS-232C                  Message FINS                  En-tête FINS 2803 Commande CompoWay/F                  Conversion de protocole                  Commande CompoWay/F                  RS-485 (CompoWay/F)                  Composant OMRON compatible CompoWay/F</p>	Les composants OMRON connectés en série au port RS-232C de l'UC avec CompoWay/F sont accessibles à partir d'une UC dans le même API.	En option

**Accès à partir de l'UC (API sur le réseau)**

Message FINS vers série	Détails	Tables de routage pour traiter le chemin de communications série comme un réseau
<p>UC avec version de carte 3.0 ou supérieure                  Réseau (Ethernet, Controller Link, DeviceNet)                  Message FINS                  En-tête FINS 2803 Commande CompoWay/F                  Conversion de protocole                  Commande CompoWay/F                  RS-485 (CompoWay/F)                  Composant OMRON compatible CompoWay/F</p>	Les composants OMRON connectés en série au port RS-232C de l'UC avec CompoWay/F sont accessibles à partir d'une UC dans un API connecté au réseau.	En option

### Trames de communication

#### Trame de commande

Trame avant conversion

En-tête FINS				Commande FINS		CompoWay/F (Voir remarque.)					
Adresse du réseau déporté (DNA)	Adresse de nœud déporté (DA1)	Adresse d'unité déportée (DA2)	Etc.	MRC	SRC						
Adresse allouée au port série ou adresse du réseau local	00 hex. ou adresse de nœud du réseau local	Adresse d'unité du port série		28	03	N° de nœud ( $\times 10^1$ ) ( $\times 10^2$ ) (2 octets ASCII)	Sous- adresse "00" (ASCII 3030 hex) etc.	SID "0" (ASCII 30 hex)	Commande (MRC, SRC) (4 octets ASCII)	Texte (ASCII)	



Trame après conversion

CompoWay/F							
STX (02 hex)	N° de nœud ( $\times 10^1$ ) ( $\times 10^2$ ) (2 octets ASCII)	Sous- adresse "00" (ASCII 3030 hex) etc.	SID "0" (ASCII 0 hex)	Commande (MRC, SRC) (4 octets ASCII)	Texte (ASCII)	ETX (03 hex)	BCC

#### Remarque

Les commandes CompoWay/F utilisent ASCII comme code de transmission. Veillez par conséquent à utiliser ASCII pour la commande CompoWay/F après le code de commande FINS 2803 hex (du numéro de nœud au texte) avec CMND(490) ou une autre instruction.

Exemple : Si la commande CompoWay/F MRC SRC est "01" "02" (où les guillemets (") indiquent des caractères ASCII), 0, 1, 0, 2 doivent être traités comme des caractères ASCII. Par conséquent, fixez "01" à 3031 hex (et non en 01 hex), et "02" à 3032 hex (et non en 02 hex).

En outre, pour écrire la commande CompoWay/F dans la zone de stockage des commandes avec CMND(490) dans l'ordre des trames (sans créer d'octets vides), le composant SID de la commande CompoWay/F exige 1 octet d'ASCII en tant que 30 hex ; les composants suivants (s+3 et après) doivent donc être définis chacun sur un octet.

Trame avant conversion

CompoWay/F								
STX (02 hex)	N° de nœud ( $\times 10^1$ ) ( $\times 10^2$ ) (2 octets ASCII)	Sous- adresse "00" (ASCII 3030 hex), etc.	Code de fin (2 octets ASCII)	Commande (MRC, SRC) (4 octets ASCII)	Réponse (MRES, SRES) (4 octets ASCII)	Texte (ASCII)	ETX (03 hex)	BCC



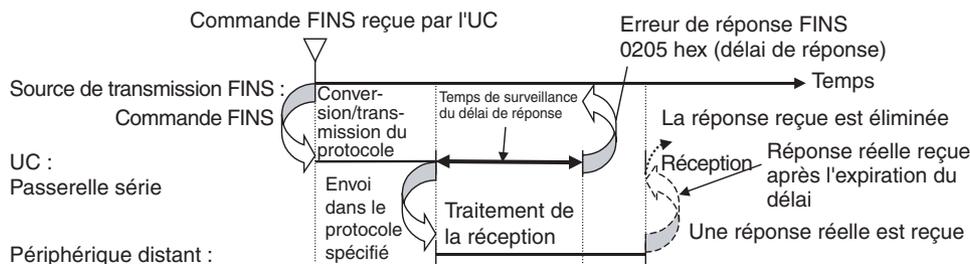
Trame après conversion

En-tête FINS			Commande FINS		Code de fin FINS		CompoWay/F (Voir remarque.)					
Adresse de réseau déporté (DNA)	Adresse de nœud déporté (DA1)	Adresse d'unité déportée (DA2), etc.	MRC	SRC	MRES	SRES						
Adresse allouée au port série	00 hex.	Adresse d'unité du port série	28	03	Spécifié par l'utilisateur	Spécifié par l'utilisateur	N° de nœud ( $\times 10^1$ ) ( $\times 10^2$ ) (2 octets ASCII)	Sous- adresse "00" (ASCII 3030 hex) etc.	Code de fin (2 octets ASCII)	Commande (MRC, SRC) (4 octets ASCII)	Réponse (MRES, SRES) (4 octets ASCII)	Texte (ASCII)

### Surveillance du délai de réponse (mode Passerelle série)

En mode Passerelle série, le délai entre l'envoi du message converti dans le protocole spécifié par la passerelle série et la réception d'une réponse du périphérique distant est surveillé. (Le délai par défaut est de 5 s. La plage de configuration pour une valeur spécifiée par l'utilisateur est comprise entre 0,1 et 25,5 s.)

Si aucune réponse n'est reçue au port série dans le délai spécifié, une erreur réponse FINS est retournée à la source de la commande FINS (code de fin : 0205 hex (délai de réponse)). Si une réponse est reçue après l'expiration du délai, cependant, la réponse reçue pour chaque protocole est supprimée et aucune réponse FINS n'est retournée à la source de la commande FINS.



### 6-3-5 Liaisons API série (cartes CJ1M uniquement)

#### Présentation

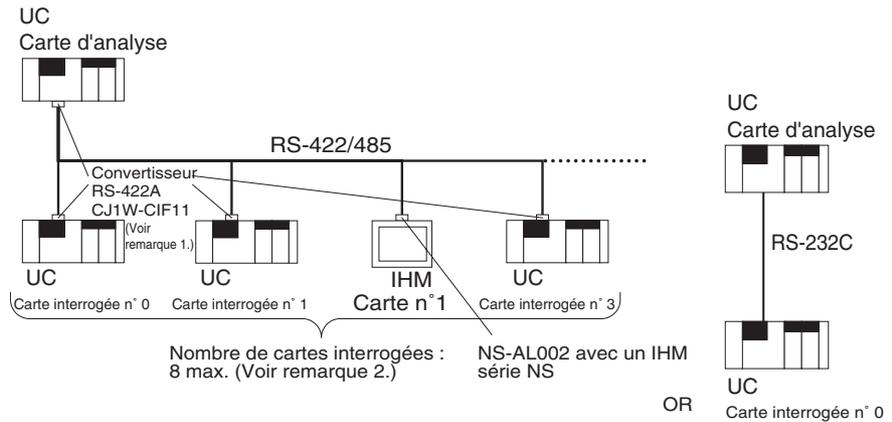
Les liaisons API série sont uniquement prises en charge par les UC CJ1M. Elles permettent d'échanger des données entre les UC CJ1M, via les ports RS-232C intégrés, sans nécessiter de programmation spéciale. Les mots sont affectés en mémoire dans les mots de liaison API série (CIO 3100 à CIO 3199). Vous pouvez utiliser des connexions RS-232C entre les UC, ou des connexions RS-422A/485 en connectant des convertisseurs RS-232C-vers-RS-422A/485 aux ports RS-232C. Des convertisseurs CJ1W-CIF11 RS-422A peuvent être utilisés pour effectuer la conversion entre RS-232C et RS-422A/485.

Un terminal opérateur configuré pour les communications NT Link (1:N) peut également être utilisé sur le même réseau. Le terminal opérateur interrogé utilise le réseau pour communiquer par une liaison NT Link (1:N) avec l'UC d'analyse. Toutefois, lorsqu'un IHM est connecté, les adresses dans les mots de liaison API série correspondant au numéro de carte du IHM ne sont pas définies.

#### Caractéristiques techniques

Caractéristiques techniques	
Mode de connexion	Connexion RS-232C ou RS-422A/485 via le port RS-232C de l'UC.
Zone de données affectée	Mots de liaison API série : CIO 3100 à CIO 3199 (vous pouvez affecter jusqu'à 10 mots pour chaque UC).
Nombre de cartes	9 cartes maximum, comprenant 1 carte d'analyse et 8 cartes interrogées. (Vous pouvez installer un terminal opérateur sur le même réseau dans une liaison NT Link (1:N), mais il doit être compté avec les 8 cartes interrogées).

**Configuration système**



- Remarque**
1. L'adaptateur CJ1W-CIF11 n'est pas isolé et la distance de transmission totale est de 50 mètres maximum. Si la distance de transmission totale est supérieure à 50 mètres, utilisez l'adaptateur isolé NT-AL001 mais jamais le CJ1W-CIF11. Si vous utilisez uniquement l'adaptateur NT-AL001, la distance de transmission totale est de 500 mètres maximum.
  2. Vous pouvez connecter jusqu'à 8 cartes, comprenant le terminal opérateur et les cartes interrogées, à la carte d'analyse, lorsqu'un terminal opérateur configuré pour les communications par liaison API série est installé sur le même réseau.

**Méthodes de mise à jour des données**

Vous pouvez mettre à jour les données à l'aide des deux méthodes suivantes :

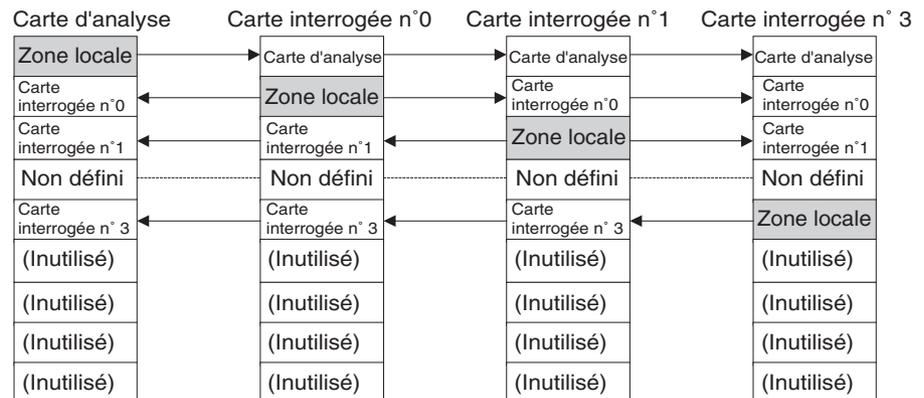
- Méthode de liaison complète
- Méthode de liaison de la carte d'analyse

**Méthode de liaison complète**

Les données provenant de tous les nœuds des liaisons API série sont représentées dans la carte d'analyse et dans les cartes interrogées. (Les seules exceptions sont l'adresse affectée au numéro de carte du terminal opérateur connecté et les adresses des cartes interrogées qui ne sont pas présentes dans le réseau. Ces plages de données sont non définies dans tous les nœuds).

**Exemple : méthode de liaison complète, nombre maximum de cartes : 3.**

Dans le diagramme suivant, la carte interrogée n° 2 est soit un terminal opérateur, soit une carte absente du réseau, de sorte que la zone affectée à la carte interrogée n° 2 est non définie dans tous les nœuds.

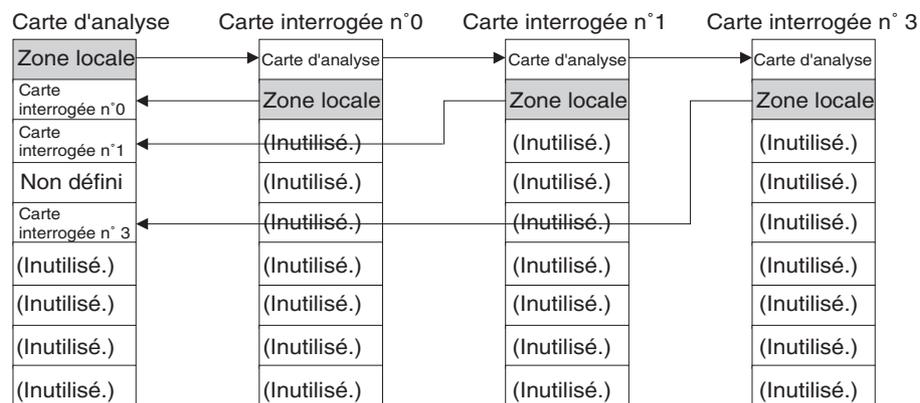


**Méthode de liaison de la carte d'analyse**

Les données de l'ensemble des cartes interrogées dans les liaisons API série sont uniquement représentées dans la carte d'analyse, et chaque carte interrogée représente uniquement les données de la carte d'analyse. L'avantage de la méthode de liaison de la carte d'analyse est que l'adresse affectée aux données des cartes interrogées locales est identique dans chaque carte interrogée, ce qui permet d'accéder aux données à l'aide d'un schéma contact commun. Les zones affectées aux numéros de carte des terminaux opérateurs ou des cartes interrogées absents du réseau sont non définies dans la carte d'analyse seulement.

**Exemple : méthode de liaison de la carte d'analyse, nombre maximum de cartes : 3.**

Dans le diagramme suivant, la carte interrogée n° 2 est soit un terminal opérateur, soit une carte extérieure au réseau. Par conséquent, la zone correspondante dans la carte d'analyse est non définie.



**Mots affectés**

**Méthode de liaison complète**

Adresse		Mots de liaison	1 mot	2 mots	3 mots	jusqu'à	10 mots
CIO 3100	Mots de liaison API série	Carte d'analyse	CIO 3100	CIO 3100 à CIO 3101	CIO 3100 à CIO 3102		CIO 3100 à CIO 3109
		Carte interrogée n° 0	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 1	CIO 3102	CIO 3104 à CIO 3105	CIO 3106 à CIO 3108		CIO 3120 à CIO 3129
		Carte interrogée n° 2	CIO 3103	CIO 3106 à CIO 3107	CIO 3109 à CIO 3111		CIO 3130 à CIO 3139
		Carte interrogée n° 3	CIO 3104	CIO 3108 à CIO 3109	CIO 3112 à CIO 3114		CIO 3140 à CIO 3149
		Carte interrogée n° 4	CIO 3105	CIO 3110 à CIO 3111	CIO 3115 à CIO 3117		CIO 3150 à CIO 3159
		Carte interrogée n° 5	CIO 3106	CIO 3112 à CIO 3113	CIO 3118 à CIO 3120		CIO 3160 à CIO 3169
		Carte interrogée n° 6	CIO 3107	CIO 3114 à CIO 3115	CIO 3121 à CIO 3123		CIO 3170 à CIO 3179
		Carte interrogée n° 7	CIO 3108	CIO 3116 à CIO 3117	CIO 3124 à CIO 3126		CIO 3180 à CIO 3189
		CIO 3199		Non utilisée.	CIO 3109 à CIO 3199	CIO 3118 à CIO 3199	CIO 3127 à CIO 3199

**Méthode de liaison de la carte d'analyse**

Adresse		Mots de liaison	1 mot	2 mots	3 mots	jusqu'à	10 mots
CIO 3100	Mots de liaison API série	Carte d'analyse	CIO 3100	CIO 3100 à CIO 3101	CIO 3100 à CIO 3102		CIO 3100 à CIO 3109
		Carte interrogée n° 0	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 1	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 2	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 3	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 4	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 5	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 6	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		Carte interrogée n° 7	CIO 3101	CIO 3102 à CIO 3103	CIO 3103 à CIO 3105		CIO 3110 à CIO 3119
		CIO 3199		Non utilisée.	CIO 3102 à CIO 3199	CIO 3104 à CIO 3199	CIO 3106 à CIO 3199

**Procédure**

Les liaisons API série fonctionnent d'après les paramètres suivants de la configuration de l'API.

**Paramètres sur la carte d'analyse**

- 1,2,3...**
1. Configure le mode des communications en série du port des communications RS-232C sur les liaisons API série (carte d'analyse).
  2. Configure la méthode de liaison sur la méthode de liaison complète ou sur la méthode de liaison de la carte d'analyse.
  3. Définit le nombre de mots de liaison (jusqu'à 10 mots pour chaque carte).
  4. Définit le nombre maximum de cartes dans les liaisons API série (0 à 7).

**Paramètres sur les cartes interrogées**

- 1,2,3...**
1. Configure le mode des communications en série du port des communications RS-232C sur les liaisons API série (carte interrogée).
  2. Définit le numéro de carte de la carte interrogée de liaison API série.

**Configuration de l'API**

**Paramètres sur la carte d'analyse**

		Adresse API		Valeur de consigne	Par défaut	Fréquence de mise à jour
		Mot	Bit			
Paramètre du port RS-232C	Mode des communications en série	160	11 à 08	8 hex. : carte d'analyse de liaison API série	0 hex	A chaque cycle
	Vitesse de transmission du port	161	07 à 00	00 à 09 hex : standard 0A hex : grande vitesse (voir remarque 2).	00 hex	
	Méthode de liaison	166	15	0 : liaisons complètes 1 : liaisons de cartes d'analyse	0	
	Nombre de mots de liaison		07 à 04	1 à A hex.	0 hex. (voir remarque 1).	
	Nombre maximum de cartes		03 à 00	0 à 7 hex.	0 hex.	

- Remarque**
1. Affecte automatiquement 10 mots (A hex.) lorsque la configuration par défaut 0 hex. est utilisée.
  2. Lorsque CX-Programmer est configuré, spécifiez 115 200 bits/s.

**Paramètres sur la carte interrogée**

		Adresse API		Valeur de consigne	Par défaut	Fréquence de mise à jour
		Mot	Bit			
Paramètres du port RS-232C	Mode des communications en série	160	11 à 08	7 hex : carte interrogée de liaison API série	0 hex.	A chaque cycle
	Vitesse de transmission du port	161	07 à 00	00 à 09 hex. : standard 0A hex. : grande vitesse (voir remarque).	00 hex.	
	Nombre de cartes interrogées	167	03 à 00	0 à 7 hex.	0 hex.	

- Remarque** Lorsque CX-Programmer est configuré, spécifiez 115 200 bits/s.

**Drapeaux de zone auxiliaire associés**

Nom	Adresse	Infos	Lecture/écriture	Fréquence de mise à jour
Drapeau d'erreur des communications du port RS-232C	A39204	<p>Passe à ON lorsqu'une erreur de communication survient sur le port RS-232C.</p> <p>1 : erreur 0 : normal</p>	Lecture	<ul style="list-style-type: none"> <li>Remis à zéro lors de la mise sous tension.</li> <li>Passe à ON lorsqu'une erreur de communication survient sur le port RS-232C.</li> <li>Passe à OFF lorsque le port est redémarré.</li> <li>Désactivé en mode bus périphérique et en mode NT Link.</li> </ul>
Drapeau des communications du port RS-232C avec un IHM (voir remarque).	A39300 à A39307	<p>Lorsque vous utilisez le port RS-232C en mode NT Link, le bit correspondant à la carte qui effectue les communications est à ON. Les bits de 00 à 07 correspondent respectivement aux numéros de carte de 0 à 7.</p> <p>1 : en communication 0 : pas en communication</p>	Lecture	<ul style="list-style-type: none"> <li>Remis à zéro lors de la mise sous tension.</li> <li>Met à ON le bit correspondant au numéro de carte du IHM ou de la carte interrogée qui est en train de communiquer via le port RS-232C en mode NT Link ou en mode de liaison API série.</li> <li>Les bits de 00 à 07 correspondent respectivement aux numéros de carte de 0 à 7.</li> </ul>
Bit de redémarrage du port RS-232C	A52600	<p>Mettez ce bit à ON pour redémarrer le port RS-232C.</p>	Lecture/écriture	<ul style="list-style-type: none"> <li>Remis à zéro lors de la mise sous tension.</li> <li>Passe à ON lors du redémarrage du port RS-232C, (sauf lors de la communication en mode bus périphérique).</li> </ul> <p><b>Remarque</b> : en fonction du système, le bit peut passer automatiquement à OFF lorsque le processus de redémarrage est terminé.</p>
Drapeau d'erreur du port RS-232C	A52800 à A52807	<p>Lorsque qu'une erreur se produit sur le port RS-232C, le code d'erreur correspondant est mémorisé.</p> <p>Bit 00 : non utilisé. Bit 01 : non utilisé. Bit 02 : erreur de parité. Bit 03 : erreur de synchronisation. Bit 04 : erreur d'engorgement. Bit 05 : erreur de dépassement de temps. Bit 06 : non utilisé. Bit 07 : non utilisé.</p>	Lecture/écriture	<ul style="list-style-type: none"> <li>Remis à zéro lors de la mise sous tension.</li> <li>Lorsqu'une erreur se produit sur le port RS-232C, le code d'erreur correspondant est mémorisé.</li> <li>En fonction du système, le drapeau peut être remis à zéro lorsque le port RS-232C est redémarré.</li> <li>Désactivé en mode bus périphérique.</li> <li>En mode NT Link, seul le bit 05 (erreur de dépassement de temps) est activé.</li> </ul> <p>En mode de liaison API série, seuls les bits ci-dessous sont activés. Erreur sur la carte d'analyse : Bit 05 : erreur de dépassement de temps. Bit 04 : erreur d'engorgement. Bit 03 : erreur de synchronisation</p>
Drapeau de modification des paramètres du port RS-232C	A61902	<p>Passe à ON pendant la modification des conditions des communications du port RS-232C.</p> <p>1 : modification. 0 : pas de modification.</p>	Lecture/écriture	<ul style="list-style-type: none"> <li>Remis à zéro lors de la mise sous tension.</li> <li>Passe à ON pendant la modification des conditions des communications du port RS-232C.</li> <li>Passe à ON lorsque l'instruction CHANGE SERIAL PORT SETUP (STUP(237)) est exécutée.</li> <li>Repassé à OFF lorsque les modifications de paramètres sont terminées.</li> </ul>

**Remarque** Comme pour la liaison NT existante (1:N), vous pouvez vérifier l'état (en communication/pas en communication) des IHM de la liaison API série à partir de la carte d'analyse (UC), en lisant le drapeau des communications du port RS-232C avec un IHM (A393 bits 00 à 07 pour les numéros de carte 0 à 7).

## 6-4 Modification du mode de mise à jour de la PV de temporisation/compteur

### 6-4-1 Présentation

Jusqu'à présent, les UC CS1 n'utilisaient les valeurs BCD que pour le mode de mise à jour de la valeur actuelle (PV) de temporisation/compteur. Par conséquent, tous les paramètres de temporisation/compteur étaient entrés sous forme de valeurs BCD. Sur d'autres UC (voir remarques 1 et 2), il est possible d'utiliser le mode BCD ou le mode binaire pour la mise à jour des valeurs actuelles d'instructions de temporisation et de compteur (voir remarque 3).

En mode binaire, le temps de configuration de temporisation/compteur précédent de 0 à 9 999 peut être allongé jusqu'à 0 à 65 535. Les valeurs de consigne de temporisation/compteur peuvent également correspondre à des données binaires calculées à l'aide d'autres instructions. Le mode de mise à jour de la PV de temporisation/compteur peut également être spécifié lorsque la valeur de consigne de temporisation/compteur est spécifiée sous forme d'adresse (spécification indirecte). (La configuration du mode détermine, suivant qu'il s'agit du mode BCD ou binaire, si le contenu du mot adressé est considéré comme une valeur BCD ou binaire).

Il existe toutefois des différences entre les opérandes d'instructions des modes BCD et binaire. Par conséquent, vérifiez et assurez-vous d'avoir bien compris les différences entre ces modes avant de modifier le mode de mise à jour de la PV de temporisation/compteur.

#### Remarque

1. Dans ce cas, les UC différentes des UC CS1 sont les suivantes :
  - UC CS1-H
  - UC CJ1-H
  - UC CJ1M
  - UC CS1D
2. Lorsque la surveillance du mnémonique s'effectue à partir de la console de programmation pour des UC CS1-H, CJ1-H, CJ1M ou CS1D fabriquées avant le 31 mai 2002 (inclus), et que le mode de mise à jour de la PV de temporisation/compteur est configuré sur le mode binaire, le mnémonique du binaire s'affiche sous la forme du mnémonique ou de l'instruction BCD (exemple : TIMX #0000 &16 s'affiche sous la forme TIM #0000 &16), mais les opérations s'effectuent en mode binaire.
3. Vous ne pouvez sélectionner le mode de mise à jour de la PV qu'avec CX-Programmer version 3.0. La sélection des modes n'est pas prise en charge par CX-Programmer version 2.1 ou inférieure, ou par les consoles de programmation.
4. CX-Programmer version 2.1 ou inférieure ne peut pas lire les programmes utilisateur des UC contenant des instructions en mode binaire, mais il peut lire les programmes définis à l'aide d'instructions en mode BCD.

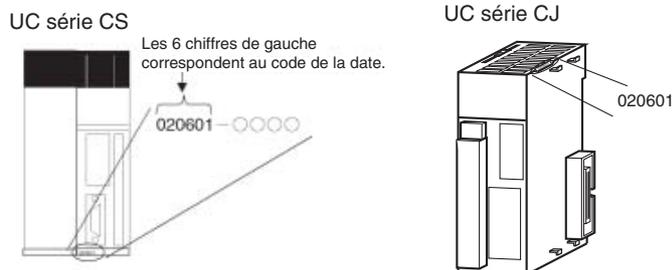
### 6-4-2 Caractéristiques fonctionnelles

	Infos		
Méthode de configuration de la mise à jour de la PV de temporisation/compteur	A configurer à l'aide de CX-Programmer version 3.0 (non prise charge par CX-Programmer version 2.1 ou inférieure). Configurée dans les propriétés de l'API de CX-Programmer version 3.0.		
UC prises en charge	Les UC CS1-H/CJ1-H du lot n° 020601 (fabriqué le 1er juin 2002) ou d'un lot postérieur (voir remarque 1), et les UC CJ1M et CS1D.		
Mode	<b>Mode BCD</b>	<b>Mode binaire</b>	
Mnémonique	Identique aux modèles précédents Exemple : TIM	Ajout d'un X au mnémonique du mode BCD Exemple : TIMX	
Code fonction	Identique aux modèles précédents	Nouveaux codes	
Plage PV/SV	#0000 à #9999	&0 à &65536	#0000 à #FFFF
Affichage de la PV sur le périphérique de programmation (CX-Programmer version 3.0 ou console de programmation)	BCD Exemple : #0100	Exemple : &100	Hexadécimal Exemple : #64

**Remarque** Lorsque la surveillance du mnémonique s'effectue à partir de la console de programmation pour des UC CS1-H/CJ1-H fabriquées avant le 31 mai 2002 (inclus), et que le mode de mise à jour de la PV de temporisation/compteur est configuré sur le mode binaire, le mnémonique du binaire s'affiche sous la forme du mnémonique ou de l'instruction BCD (exemple : TIMX #0000 &16 s'affiche sous la forme TIM #0000 &16), mais les opérations s'effectuent en mode binaire.

#### Vérification du numéro de lot de l'UC

- 1,2,3...**
- Le numéro de lot est imprimé au bas de la face avant (série CS) ou dans le coin droit de la face supérieure de l'UC (série CJ), et se compose des deux derniers chiffres de l'année, du mois et du jour, dans cet ordre, comme illustré ci-dessous.  
Exemple : 020601 (fabriqué le 1er juin 2002).

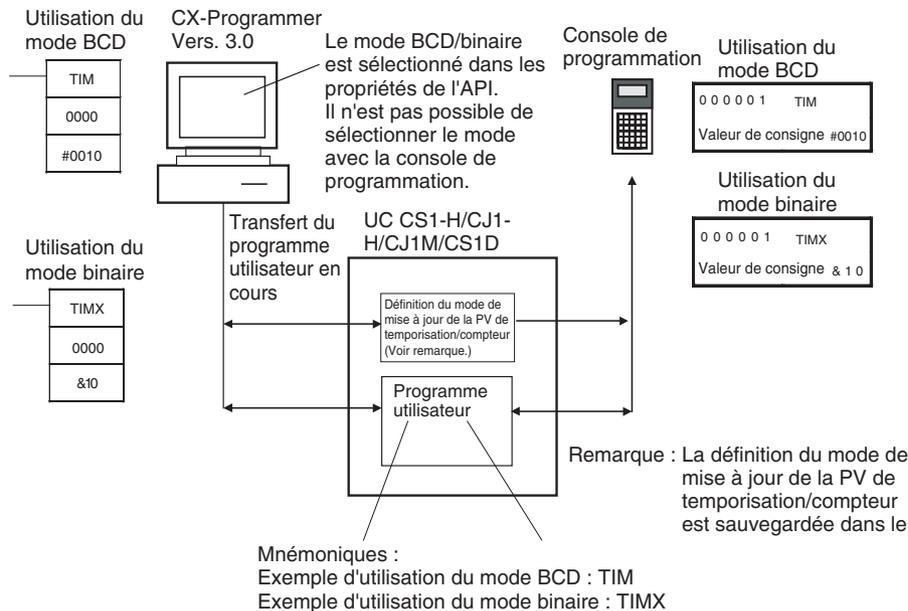


- Vérifiez le mode sélectionné. Pour ce faire, mettez CX-Programmer en ligne, ouvrez la fenêtre des tables d'E/S et sélectionnez **Unit Information - CPU Unit**. Le numéro de lot s'affiche dans le format représenté plus haut, c'est-à-dire composé des deux derniers chiffres de l'année, du mois et du jour, dans cet ordre.

### 6-4-3 Sélection du mode BCD ou du mode binaire et confirmation

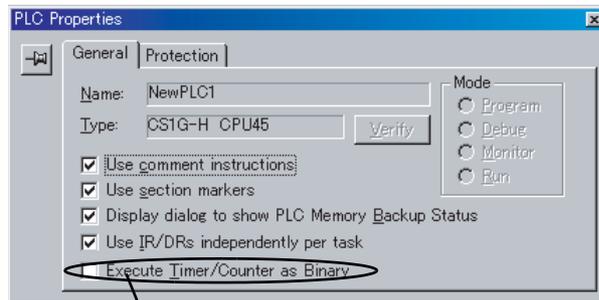
Lors de l'écriture d'un nouveau programme, le mode BCD ou binaire est sélectionné dans les paramètres des propriétés de l'API, dans CX-Programmer version 3.0.

**Remarque** La sélection du mode BCD/binaire est uniquement prise en charge par CX-Programmer version 3.0 ou supérieure. CX-Programmer version 2.1 ou les versions inférieures ne permettent pas la sélection de modes.



### Sélection du mode BCD ou du mode binaire

- 1,2,3... 1. Sélectionnez le nom de l'API, cliquez avec le bouton droit de la souris et sélectionnez **PLC Properties**.



Cochez cette case pour activer la configuration.

2. Cliquez sur l'onglet **General** et cochez la case **Execute Timers/Counters as Binary**.

- Non cochée (par défaut) : mode BCD
- Cochée : mode binaire

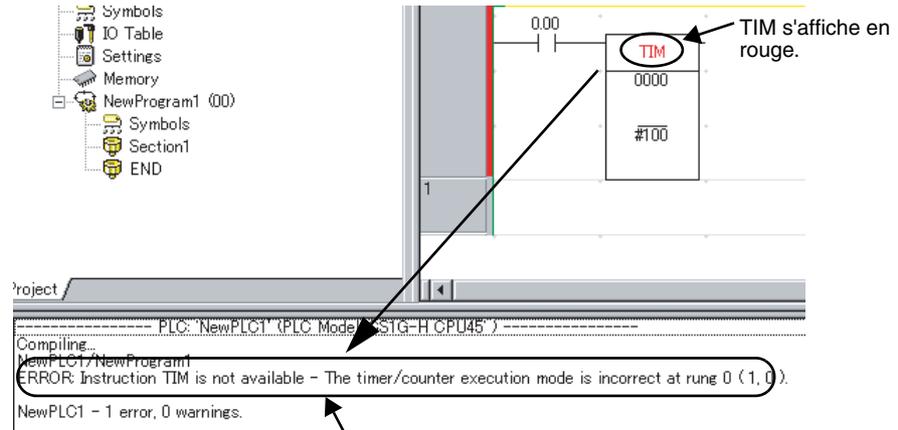
La valeur de consigne du mode de mise à jour de la PV de temporisation/compteur configurée dans les propriétés de l'API sera stockée dans la mémoire utilisateur de l'UC lors du transfert du programme utilisateur, du CX-Programmer vers l'UC.

Lorsque vous modifiez la configuration, la boîte de dialogue ci-dessous s'affiche automatiquement.



Cliquez sur **OK** pour exécuter la vérification du programme. Les résultats de la vérification du programme s'affichent dans la fenêtre de résultats.

Exemple : l'instruction TIM a été utilisée bien que vous soyez passé au mode binaire.



Les résultats de la vérification du programme s'affichent dans la fenêtre de résultats.  
Exemple : le mode de fonctionnement de la temporisation ou du compteur est différent, vous ne pouvez pas utiliser l'instruction TIM.

### Confirmation du mode BCD ou du mode binaire

A09915 dans la zone auxiliaire (drapeau de mise à jour de la PV de temporisation/compteur) permet de vérifier si une UC fonctionne en mode BCD ou en mode binaire.

Nom	Adresse	Infos
Drapeau de mise à jour de la PV de temporisation/compteur	A09915	0 : mode BCD 1 : mode binaire

### 6-4-4 Mnémoniques et données en mode BCD ou en mode binaire

#### Mnémoniques en mode BCD ou en mode binaire

Les mnémoniques en mode binaire sont caractérisés par l'ajout du suffixe X au mnémonique BCD.

Exemple : mnémoniques de l'instruction TIMER

Mode BCD : TIM

Mode binaire : TIMX

**Affichage des données en mode BCD ou en mode binaire**

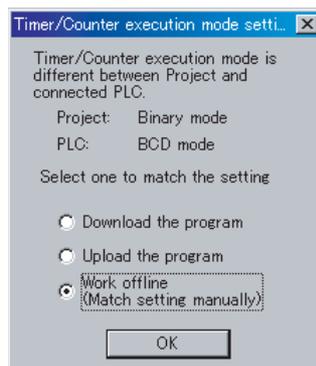
Propriété de l'API	Signification des symboles d'entrée et d'affichage	Plage de configuration	Exemple : temporisation n° : 0000, valeur de consigne : 10 s
Mode BCD	Le symbole # indique la valeur de l'instruction (une valeur BCD lorsque le mode BCD est utilisé)	#0000 à #9999 ou #00000000 à #99999999	
Mode binaire	Le symbole & indique une valeur décimale.	&0 à &65535 ou &0 à &4294967295	
	Le symbole # indique la valeur de l'instruction (une valeur hexadécimale lorsque le mode BCD est utilisé).	#0000 à #FFFF ou #0000 à #FFFFFFFF	

**Remarque**

Lorsque vous utilisez CX-Programmer en mode BCD ou en mode binaire, si vous saisissez la valeur numérique sans inclure le symbole d'entrée/affichage # ou & indiquant qu'il s'agit de la constante, (exemple : TIM 0000 0010), la valeur de consigne de temporisation/compteur est entrée en tant qu'adresse (par exemple, la valeur contenue dans le mot CIO 0010 sera utilisée comme valeur de consigne).

**6-4-5 Restrictions**

- Vous ne pouvez pas utiliser ensemble, sur une même UC, le mode BCD et le mode binaire.
- Lorsque vous utilisez la console de programmation pour créer un nouveau programme utilisateur ou pour effacer la mémoire, le mode de mise à jour de la PV de temporisation/compteur est fixé au mode BCD.
- Lorsque vous mettez l'UC en ligne à l'aide de CX-Programmer version 3.0, la valeur de consigne du mode de mise à jour de la PV de temporisation/compteur, stockée dans la mémoire utilisateur de l'UC, est automatiquement utilisée. Si le paramètre de l'UC est différent du paramètre du projet CX-Programmer, une erreur se produit et la connexion en ligne est impossible à établir. Le message suivant s'affiche.



Modifiez la sélection de manière que le paramètre de l'UC corresponde à celui du projet CX-Programmer ou que le paramètre de propriétés du projet CX-Programmer corresponde à celui de l'UC.

- CX-Programmer version 2.1 ou inférieure ne peut pas lire les programmes utilisateur des UC définis à l'aide du mode binaire, mais il peut lire les programmes définis à l'aide du mode BCD.

- Lorsque vous entrez une instruction de mode de mise à jour de la PV de temporisation/compteur incorrecte, les différences entre les opérations de CX-Programmer et celles d'une console de programmation sont les suivantes :
  - CX-Programmer :  
 Une erreur se produit si vous entrez une instruction correspondant à un mode différent de celui défini comme mode de mise à jour de la PV de temporisation/compteur sous *PLC properties*.  
 Exemple : lorsque l'API du projet est configuré au mode binaire, une erreur se produit si vous entrez TIM comme mnémotique. En mode BCD, une erreur se produit si vous entrez TIMX comme mnémotique.
  - Console de programmation :  
 Lorsque vous entrez un code fonction pour une instruction dans un mode différent du mode de mise à jour de la PV de temporisation/compteur configuré dans l'UC, le mnémotique est automatiquement remplacé par celui correspondant au mode de mise à jour de la PV de temporisation/compteur configuré dans l'UC.

## 6-4-6 Instructions et opérades

### Instructions

Type d'instruction	Nom	Mnémotique	
		Mode BCD	Mode binaire
Instructions de temporisation et de compteur	TIMER (100 ms)	TIM	TIMX(550)
	HIGH-SPEED TIMER (10 ms)	TIMH(015)	TIMHX(551)
	ONE-MS TIMER (1 ms)	TMHH(540)	TMHHX(552)
	ACCUMULATIVE TIMER (100 ms)	TTIM(087)	TTIMX(555)
	LONG TIMER (100 ms)	TIML(542)	TIMLX(553)
	MULTI-OUTPUT TIMER (100 ms)	MTIM(543)	MTIMX(554)
	COUNTER	CNT	CNTX(546)
	REVERSIBLE COUNTER	CNTR(012)	CNTRX(548)
	RESET TIMER/COUNTER	CNR(545)	CNRX(547)
Instructions de programme de bloc	TIMER WAIT (100 ms)	TIMW(813)	TIMWX(816)
	HIGH-SPEED TIMER WAIT (10 ms)	TMHW(815)	TMHWX(817)
	COUNTER WAIT	CNTW(814)	CNTWX(818)

**Instructions et opérandes**

**Instructions de temporisation et de compteur**

**TIMER (100 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	TIM	TIMX(550)
S (valeur de consigne temporisation)	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Configuration du temps (pas : 0,1 s)	0 à 999,9 s	0 à 6 553,5 s

**HIGH-SPEED TIMER (10 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	TIMH(015)	TIMHX(551)
S (valeur de consigne temporisation)	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Configuration du temps (pas : 0,01 s)	0 à 99,99 s	0 à 655,35 s

**ONE-MS TIMER (1 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	TMHH(540)	TMHHX(552)
S (valeur de consigne temporisation)	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Configuration du temps (pas : 0,001 s)	0 à 9,999 s	0 à 65,535 s

**ACCUMULATIVE TIMER (100 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	TTIM(087)	TTIMX(555)
S (valeur de consigne temporisation)	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Configuration du temps (pas : 0,1 s)	0 à 999,9 s	0 à 6 553,5 s

**LONG TIMER (100 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	TIML(542)	TIMLX(553)
S, S+1 (valeurs de consigne temporisation)	#00000000 à #99999999 (BCD)	&0 à &4294967295 (décimal) ou #0000 à #FFFFFFFF (hexadécimal)
Configuration du temps (pas : 0,1 s)	0 à 999,9 s	0 à 6 553,5 s

**MULTI-OUTPUT TIMER (100 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	MTIM(543)	MTIMX(554)
S à S-7 (chaque valeur de consigne)	#0000 à #9999 (BCD)	&0 à &65535 (hexadécimal) ou #0000 à #FFFF (hexadécimal)
Configuration du temps (pas : 0,1 s)	0 à 999,9 s	0 à 6 553,5 s

**COUNTER**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	CNT	CNTX(546)
S (valeur de consigne compteur)	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Paramètre	0 à 9 999 fois	0 à 65 535 fois

**REVERSIBLE COUNTER**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	CNTR(012)	CNTRX(548)
S (valeur de consigne compteur)	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Paramètre	0 à 9 999 fois	0 à 65 535 fois

**RESET TIMER/COUNTER**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	CNR(545)	CNRX(547)

**Instructions de programme de bloc**

**TIMER WAIT (100 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	TIMW(813)	TIMWX(816)
S (valeur de consigne temporisation)	#0000 ou # 9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Configuration du temps (pas : 0,1 s)	0 à 999,9 s	0 à 6 553,5 s

**HIGH-SPEED TIMER WAIT (10 ms)**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	TMHW(815)	TMHGX(817)
S (valeur de consigne temporisation) Pas : 0,01 s	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Configuration du temps (pas : 0,01 s)	0 à 999,9 s	0 à 655.35 s

**COUNTER WAIT**

Nom de l'instruction	Mode BCD	Mode binaire
Mnémonique	CNTW(814)	CNTX(818)
S (valeur de consigne compteur)	#0000 à #9999 (BCD)	&0 à &65535 (décimal) ou #0000 à #FFFF (hexadécimal)
Paramètre	0 à 9 999 fois	0 à 65 535 fois

## 6-5 Utilisation d'une interruption programmée comme temporisation de haute précision (CJ1M uniquement)

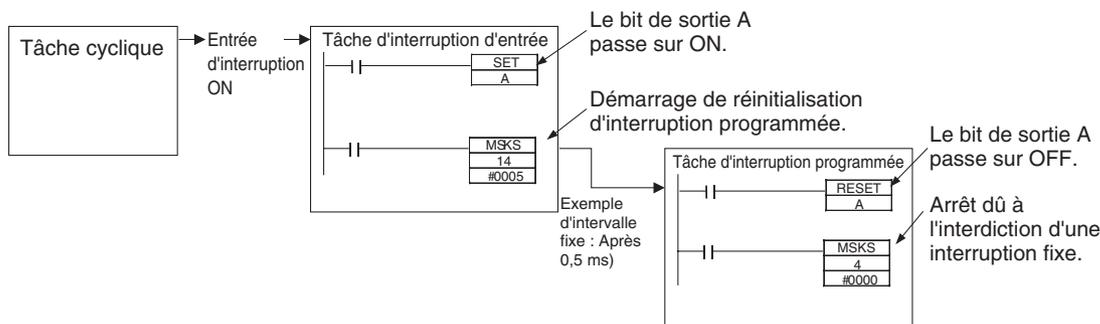
Lorsque vous utilisez une UC CJ1M, les fonctions ci-dessous vous permettent de créer une temporisation de haute précision à l'aide d'une interruption programmée.

- Vous pouvez entrer la temporisation d'interruption programmée par pas de 0,1 ms (temporisation interne de haute précision).
- La réinitialisation (c.-à-d., le redémarrage) est possible à l'aide de l'instruction MSKS(690) (temps fixe jusqu'à la première interruption).
- Les valeurs actuelles (PV) de temporisation interne peuvent être lues à l'aide de l'instruction MSKR(692) (lecture des PV de temporisation interne).

Ces fonctions offrent des applications telles que celle illustrée dans l'exemple suivant. Il s'agit ici d'une temporisation haute précision à 1 impulsion, dans laquelle l'activation du bit d'entrée agit comme un déclencheur, entraînant le passage à ON du bit de sortie, puis son retour à OFF après un intervalle fixe.

Exemple :

- 1,2,3...**
1. La tâche d'interruption d'entrée démarre lorsque le bit d'entrée intégrée passe à ON.
  2. Le bit de sortie A passe à ON dans la tâche d'interruption d'entrée, et l'instruction MSKS(690) est exécutée pour réaliser un démarrage de réinitialisation d'interruption programmée.
  3. Après un intervalle fixe, la tâche d'interruption programmée démarre, le bit de sortie A dans la tâche d'interruption programmée passe à OFF et l'instruction MSKS(690) est exécutée pour interdire toute interruption programmée.



### 6-5-1 Configuration de l'interruption programmée par pas de 0,1 ms

Le temps d'interruption programmée est défini à l'aide du paramètre de temps de l'interruption programmée dans la configuration de l'API, et de l'instruction MSKS(690).

Dans le cas des UC CJ1M, vous pouvez définir le temps d'interruption programmée par pas de 0,1 ms, entre un intervalle minimum de 0,5 ms et un intervalle maximum de 999,9 ms.

#### Configuration de l'API

	Adresse API		Valeur de consigne	Par défaut	Fréquence de mise à jour
	Mot	Bit			
Configuration du pas de réglage de l'interruption programmée	195	00 à 03	0 hex. : pas de 10 ms 1 hex. : pas de 1 ms 2 hex. : pas de 0,1 ms (UC CJ1M uniquement)	0 hex.	Au début du fonctionnement.

### 6-5-2 Spécification d'un démarrage avec réinitialisation à l'aide de MSKS(690)

Lorsque vous utilisez des UC CJ1M et que vous démarrez l'interruption programmée à l'aide de l'instruction MSKS(690), la temporisation interne peut être réinitialisée avant le démarrage de l'interruption (cette opération est appelée démarrage avec réinitialisation).

Cette méthode permet de spécifier le temps jusqu'à la première interruption sans l'aide de l'instruction CLI(691).

Les interruptions programmées sont démarrées à l'aide de l'instruction MSKS(690) pour configurer le temps d'interruption programmée (intervalle entre deux interruptions). Toutefois, après l'exécution de l'instruction MSKS(690), le temps requis avant le démarrage de la première tâche d'interruption programmée (temps de démarrage de la première interruption) n'est fixé que si l'instruction CLI(691) est spécifiée. Les UC CJ1M offrent donc une fonction de démarrage avec réinitialisation de la temporisation interne, permettant de définir le temps jusqu'à la première interruption sans l'aide de l'instruction CLI(691).

#### Opérande de l'instruction MSKS(690) (uniquement si l'interruption programmée est spécifiée)

Opérande	Valeur de consigne
N (identifiant d'interruption)	4 : interruption programmée 0, configuration normale (temporisation interne non réinitialisée).
	5 : interruption programmée 1, configuration normale (temporisation interne non réinitialisée).
	14 : interruption programmée 0, spécifie un démarrage de réinitialisation (UC CJ1M uniquement).
	15 : interruption programmée 1, spécifie un démarrage réinitialisation (UC CJ1M uniquement).

### 6-5-3 Lecture de la valeur actuelle (PV) de temporisation interne à l'aide de MSKR(692)

Les UC CJ1M permettent de lire la PV de la temporisation interne qui mesure le temps d'interruption programmée. Le temps est lu à partir du point de démarrage de l'interruption programmée ou du point d'interruption programmée précédent. La PV de temporisation interne est lue en exécutant l'instruction MSKR(692). L'unité de temps dépend du paramètre de temps de l'interruption programmée dans la configuration de l'API, comme pour le temps d'interruption programmée.

#### Opérandes de l'instruction MSKR(692) (uniquement si l'interruption programmée est spécifiée)

Opérande	Valeur de consigne
N (identifiant d'interruption)	4 : interruption programmée 0, lit le temps d'interruption programmée (valeur de consigne).
	5 : interruption programmée 1, lit le temps d'interruption programmée (valeur de consigne).
	14 : interruption programmée 0, lit la PV de temporisation interne (UC CJ1M uniquement).
	15 : interruption programmée 1, lit la PV de temporisation interne (UC CJ1M uniquement).

## 6-6 Paramètres du démarrage et maintenance

Cette section décrit les fonctions relatives au démarrage et à la maintenance suivantes :

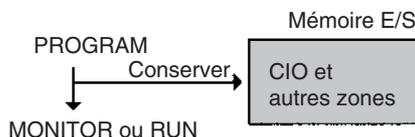
- les fonctions de démarrage/d'arrêt à chaud ;
- le paramètre du mode de démarrage ;
- le paramètre du retard de la détection de mise hors tension ;
- la désactivation des interruptions à la mise hors tension ;
- la sortie RUN ;
- l'horloge ;
- la protection du programme ;
- la surveillance et la programmation déportées ;
- la mémoire flash ;
- la configuration des conditions au démarrage.

### 6-6-1 Fonctions de démarrage/d'arrêt à chaud

#### Changement de mode de fonctionnement

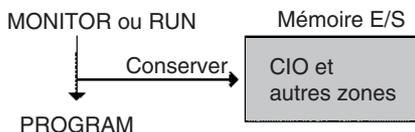
##### Démarrage à chaud

Mettez le bit de maintien IOM (A50012) à ON de manière à conserver toutes les données dans la mémoire d'E/S lorsque vous commutez l'UC du mode PROGRAM au mode RUN ou MONITOR pour lancer l'exécution du programme.



##### Arrêt à chaud

Lorsque le bit de maintien IOM (A50012) est à ON, toutes les données dans la mémoire d'E/S sont également conservées lorsque vous commutez l'UC du mode RUN ou MONITOR au mode PROGRAM pour arrêter l'exécution du programme.



#### Remarque

\*Si le bit de maintien IOM n'est pas à ON, les zones suivantes de la mémoire d'E/S seront effacées pendant les changements de mode (PROGRAM ↔ RUN/MONITOR) : la zone CIO (zone I/O, zone de liaison de données, zone de cartes réseau, zone de cartes d'E/S spéciales, zone de cartes internes, zone de bus SYSMAC BUS, zone de bornes d'E/S, zone DeviceNet (CompoBus/D) et zones d'E/S internes), la zone de travail, les drapeaux de fin de temporisation et les PV de temporisation. (Les zones de cartes internes, de bus SYSMAC BUS et de bornes d'E/S sont uniquement prises en charge par les UC série CS).

#### Drapeaux et mots de la zone auxiliaire

Nom	Adresse	Description
Bit de maintien IOM	A50012	Si ce bit est à ON, la totalité de la mémoire d'E/S est conservée lorsque vous changez de mode de fonctionnement (PROGRAM ↔ RUN/MONITOR).

Si le bit de maintien IOM est à ON, toutes les sorties des cartes de sorties sont maintenues lorsque l'exécution du programme s'arrête. Au redémarrage du programme, les sorties seront à l'état dans lequel elles étaient avant l'arrêt du programme.

(Si le bit de maintien IOM est à OFF, les instructions seront exécutées après la remise à zéro des sorties).

**Mise sous tension de l'API**

Pour conserver toutes les données\* dans la mémoire d'E/S lors de la mise sous tension de l'API (OFF → ON), le bit de maintien IOM doit être à ON et protégé dans la configuration de l'API (adresse 80, état du bit de maintien IOM au démarrage).



**Drapeaux et mots de la zone auxiliaire**

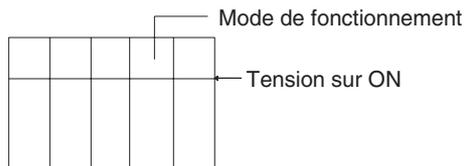
Nom	Adresse	Description
Bit de maintien IOM	A50012	Si ce bit est à ON, la totalité de la mémoire d'E/S est conservée lorsque vous changez de mode de fonctionnement (PROGRAM ↔ RUN/MONITOR).

**Configuration de l'API**

Adresse dans console de programmation	Nom	Paramètre	Par défaut
80 bit 15	Etat du bit de maintien IOM au démarrage	0 : le bit de maintien IOM est remis à 0 lors de la mise sous tension. 1 : le bit de maintien IOM est conservé lors de la mise sous tension.	0 (remis à zéro)

**6-6-2 Paramètre du mode de démarrage**

Vous pouvez configurer le mode de fonctionnement initial de l'UC (à la mise sous tension) dans la configuration de l'API.



**Configuration de l'API**

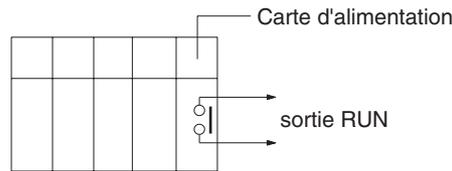
Adresse dans console de programmation	Nom	Signification	Paramètre	Par défaut
81	Mode de démarrage	Spécifie le mode de fonctionnement à utiliser au démarrage.	PRCN : commutateur de mode de la console de programmation PRG : mode PROGRAM MON : mode MONITOR RUN : mode RUN	PRCN : commutateur de mode de la console de programmation

**Remarque**

Si le mode de démarrage est configuré sur PRCN (commutateur de mode de la console de programmation) mais qu'aucune console de programmation n'est connectée, l'UC démarrera en mode RUN. Modifie la valeur par défaut dans la configuration de l'API afin de démarrer en mode MONITOR ou en mode PROGRAM à la mise sous tension. (Les UC CS1 série CS démarrent toutefois en mode PROGRAM dans les mêmes conditions).

### 6-6-3 Sortie RUN

Certaines cartes d'alimentation (les C200HW-PA204R, C200HW-PA209R, CJ1W-PA205R et CS1D-PA207R) sont équipées d'une sortie RUN. Ce point de sortie est à ON (fermé) lorsque l'UC fonctionne en mode RUN ou MONITOR, et à OFF (ouvert) lorsque l'UC est en mode PROGRAM.



Vous pouvez utiliser cette sortie RUN pour créer un circuit de sécurité externe, par exemple, un circuit d'arrêt d'urgence qui empêche l'alimentation externe d'une carte de sortie de délivrer du courant lorsque l'API n'est pas sous tension.

**Remarque** Si vous utilisez une carte d'alimentation sans sortie RUN, vous pouvez créer un sortie équivalente en programmant le drapeau Toujours ON (A1) en tant que condition d'exécution d'un point de sortie à partir d'une carte de sortie.

**Attention** Si l'alimentation externe de la carte de sortie s'allume avant l'alimentation de l'API, la carte de sortie risque de dysfonctionner temporairement à la première mise en marche de l'API. Pour éviter tout dysfonctionnement, ajoutez un circuit externe qui empêche l'alimentation externe de la carte de sortie de s'allumer avant l'alimentation de l'API. Créez un circuit de sécurité similaire à celui décrit plus haut pour garantir que le courant est exclusivement délivré par une alimentation externe lorsque l'API fonctionne en mode RUN ou MONITOR.

### 6-6-4 Paramètre du retard de la détection de mise hors tension

En règle générale, une interruption d'alimentation est détectée dans un délai de 10 à 25 ms (2 à 5 ms pour les alimentations c.c.) lorsque la tension chute en dessous de 85 % de la valeur nominale minimale (80 % pour les alimentations c.c.). Il existe, dans la configuration de l'API, un paramètre (adresse 225 bits 0 à 7, Retard de la détection de mise hors tension) qui permet d'allonger cette durée jusqu'à 10 ms (jusqu'à 2 ms pour les alimentations c.c.).

Lorsque la tâche d'interruption de mise hors tension est activée, elle est exécutée lorsque l'interruption d'alimentation est confirmée, sinon l'UC est réinitialisée et s'arrête de fonctionner.

**Paramètres associés**

Adresse	Nom	Signification	Paramètre	Par défaut
CIO 256, bits 00 à 07	Retard de la détection de mise hors tension	Définir le temps de retard avant la détection d'une interruption d'alimentation.	00 à 0A (hex.) : 0 à 10 ms	00 (hex.) : 0 ms

### 6-6-5 Désactivation des interruptions à la mise hors tension

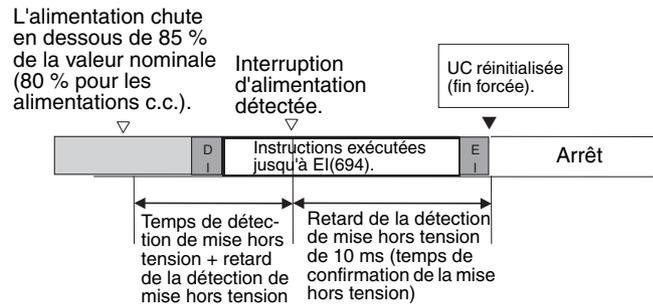
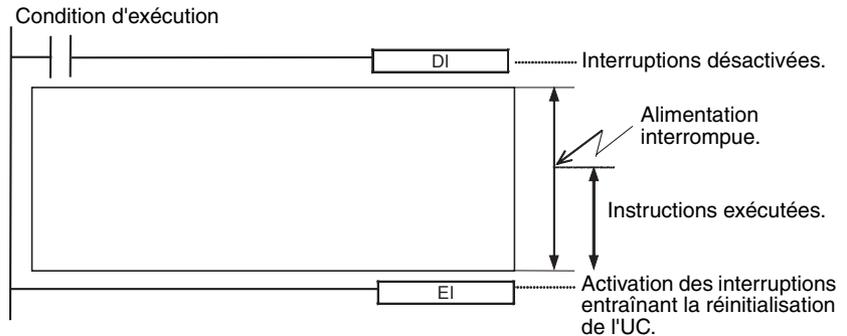
Cette fonction est uniquement prise en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D.

Dans le cas des UC CS1-H, CJ1-H, CJ1M ou CS1D, des zones du programme peuvent être protégées des interruptions à la mise hors tension et être ainsi exécutées avant l'UC, même si l'alimentation est interrompue. Cette opération est réalisée à l'aide des instructions DISABLE INTERRUPTS (DI(693)) et ENABLE INTERRUPTS (EI(694)).

Vous pouvez utiliser cette fonction avec des ensembles d'instructions à exécuter sous forme groupée, par exemple, afin que l'exécution ne démarre pas avec des données stockées intermédiaires à la prochaine mise sous tension.

**Procédure**

- 1,2,3...
1. Attribuez la valeur A5A5 hex. au paramètre Désactiver le paramètre d'interruption à la mise hors tension dans A530, afin d'activer la désactivation des interruptions à la mise hors tension.
  2. Activez la désactivation des interruptions à la mise hors tension dans la configuration de l'API (correspond à la configuration par défaut).
  3. Utilisez DI(693) pour désactiver les interruptions avant la section de programme à protéger, puis utilisez EI(694) pour activer les interruptions après la section. Toutes les instructions comprises entre DI(693) et EI(694) seront achevées avant l'exécution de l'interruption à la mise hors tension, même si l'interruption d'alimentation survient pendant l'exécution des instructions comprises entre DI(693) et EI(694).



**Paramètres associés**

Nom	Adresse	Signification
Désactiver le paramètre d'interruption à la mise hors tension	A530	Active l'utilisation de DI(693) pour désactiver l'interruption à la mise hors tension (sauf pour l'exécution de la tâche d'interruption à la mise hors tension) jusqu'à ce que EI(694) soit exécutée. A5A5 hex. : active l'utilisation de DI(693) pour désactiver l'interruption à la mise hors tension. Autre valeur : désactive l'utilisation de DI(693) pour désactiver l'interruption à la mise hors tension.

**6-6-6 Fonctions d'horloge**

Les API série CS/CJ possèdent les fonctions d'horloge suivantes :

- la surveillance de l'heure à laquelle les interruptions d'alimentation se sont produites ;
- la surveillance de l'heure à laquelle l'API a été mis sous tension ;
- la surveillance du temps total pendant lequel l'API a été sous tension.

**Remarque** Les UC CSI série CS sont livrées sans la batterie de remplacement installée, et l'horloge interne de l'UC indique 00/01/01 00:00:00 ou éventuellement une autre valeur lorsque la batterie est connectée. Pour utiliser les fonctions d'horloge, connectez la batterie, mettez l'UC sous tension et réglez l'heure et la date à l'aide d'un périphérique de programmation (console de programmation ou CX-Programmer) ou de la commande FINS (07 02, CLOCK WRITE). L'horloge interne de l'UC commence à fonctionner une fois que vous l'avez réglée.

#### Drapeaux et mots de la zone auxiliaire

Nom	Adresses	Fonction
Données d'horloge	A35100 à A35107	Seconde : 00 à 59 (BCD)
	A35108 à A35115	Minute : 00 à 59 (BCD)
	A35200 à A35207	Heure : 00 à 23 (BCD)
	A35208 à A35215	Jour du mois : 00 à 31 (BCD)
	A35300 à A35307	Mois : 00 à 12 (BCD)
	A35308 à A35315	Année : 00 à 99 (BCD)
	A35400 à A35407	Jour de la semaine : 00 : dimanche, 01 : lundi, 02 : mardi, 03 : mercredi, 04 : jeudi, 05 : vendredi, 06 : samedi
Heure de démarrage	A510 et A511	Contiens l'heure à laquelle l'alimentation a été mise sous tension.
Heure d'interruption de l'alimentation	A512 et A513	Contiens l'heure à laquelle l'alimentation a été interrompue.
Temps total de l'alimentation sous tension	A523	Contient le temps total (en binaire) pendant lequel l'API a été sous tension, en unités de 10 heures.

#### Instructions associées

Instruction	Nom	Fonction
SEC(065)	HOURS TO SECONDS	Convertit les données de temps au format heures/minutes/secondes en un temps équivalent en secondes uniquement.
HMS(066)	SECONDS TO HOURS	Convertit les données en secondes en un temps équivalent au format heures/minutes/secondes.
CADD(730)	CALENDAR ADD	Ajoute des données de temps aux données de calendrier dans les mots spécifiés.
CSUB(731)	CALENDAR SUBTRACT	Soustrait des données de temps aux données de calendrier dans les mots spécifiés.
DATE(735)	CLOCK ADJUSTMENT	Règle le paramètre de l'horloge interne sur celui des mots sources spécifiés.

### 6-6-7 Protection du programme

Le programme utilisateur de la série CS/CJ peut être protégé en écriture ou entièrement protégé (protection en lecture/écriture).

#### Protection en écriture à l'aide de l'interrupteur DIP

Le programme utilisateur peut être protégé en écriture en plaçant la broche 1 de l'interrupteur DIP de l'UC sur ON. Lorsque cette broche est activée, il est impossible de modifier le programme utilisateur à partir d'un périphérique de programmation (y compris les consoles de programmation). Cette fonction permet d'empêcher l'écrasement involontaire du programme sur le lieu de travail.

La lecture et l'affichage du programme restent possibles lorsque celui-ci est protégé en écriture.

**Remarque** Dans le cas des UC série CS/CJ version 2.0 ou supérieure, vous pouvez utiliser le paramètre d'interrupteur DIP ci-dessus ou, avec CX-Programmer version 4.0 ou supérieure, vous pouvez configurer la protection en lecture/écriture du programme en tant qu'option si vous définissez un mot de passe pour le programme entier ou pour une tâche au moins. Pour plus de détails, reportez-vous à la section *Activation/Désactivation de la création de fichiers programme de mémoire de fichiers* sous *1-4-2 Protection de lecture améliorée grâce aux mots de passe* dans le *Manuel d'utilisation des API série CS* ou le *Manuel d'utilisation des API série CJ*.

#### Protection en lecture/écriture à l'aide de mots de passe

Vous pouvez bloquer l'accès en lecture et en écriture à la zone de programme utilisateur à partir du CX-Programmer. La protection du programme empêche la copie non autorisée du programme et la perte de la propriété intellectuelle. Un mot de passe est défini pour la protection du programme à partir d'un périphérique de programmation et l'accès à la totalité du programme est bloqué.

**Remarque**

1. Si vous oubliez le mot de passe, vous ne pourrez pas transférer le programme contenu dans l'API vers l'ordinateur. Notez le mot de passe et conservez-le en lieu sûr.
2. Si vous oubliez le mot de passe, vous ne pourrez pas transférer de programmes de l'ordinateur vers l'API. Le transfert des programmes de l'ordinateur vers l'API est possible même si la protection par mot de passe n'a pas été levée.

#### Protection par mot de passe

- 1,2,3...**
1. Pour enregistrer un mot de passe en ligne ou hors ligne, procédez de la manière suivante :
    - a. Sélectionnez l'API puis sélectionnez **Propriétés** dans le menu Affichage.
    - b. Sélectionnez **Protection** dans la boîte de dialogue des propriétés de l'API et entrez le mot de passe.
  2. Pour configurer la protection par mot de passe en ligne, procédez de la manière suivante :
    - a. Sélectionnez **PLC, Protection,,** puis **Set**. La boîte de dialogue de configuration de la protection s'affiche.
    - b. Cliquez sur **OK**.

**Remarque** Dans le cas des UC série CS/CJ version 2.0 ou supérieure, vous pouvez configurer la protection en lecture pour le programme entier mais aussi pour des tâches spécifiques. Pour plus de détails, reportez-vous à la section *Protection contre la lecture des différentes tâches grâce aux mots de passe* sous *1-4-2 Protection de lecture améliorée grâce aux mots de passe* dans le *Manuel d'utilisation des API série CS* ou dans le *Manuel d'utilisation des API série CJ*.

#### Confirmation de la date du programme utilisateur

Dans le cas des UC CS1-H, CJ1-H, CJ1M ou CS1D, vous pouvez confirmer les dates de création du programme et des paramètres en vérifiant le contenu de A090 à A097.

Mots de la zone auxiliaire

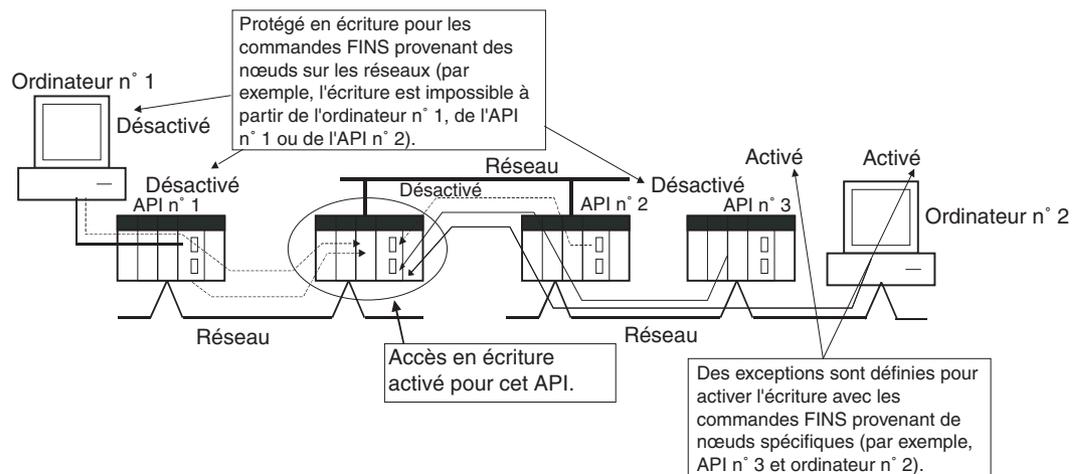
Nom	Adresse	Description	
Date du programme utilisateur	A090 à A093	L'heure et la date de la dernière réécriture du programme utilisateur dans la mémoire sont indiquées en BCD.	
		A09000 à A09007	Secondes (00 à 59 BCD)
		A09008 à A09015	Minutes (00 à 59 BCD)
		A09100 à A09107	Heure (00 à 23 BCD)
		A09108 à A09115	Jour du mois (01 à 31 BCD)
		A09200 à A09207	Mois (01 à 12 BCD)
		A09208 à A09215	Année (00 à 99 BCD)
		A09300 à A09307	Jour (00 à 06 BCD) Jour de la semaine : 00 : dimanche, 01 : lundi, 02 : mardi, 03 : mercredi, 04 : jeudi, 05 : vendredi, 06 : samedi
Date des paramètres	A094 à A097	L'heure et la date de la dernière réécriture des paramètres dans la mémoire sont indiquées en BCD. Le format est identique à celui de la date du programme utilisateur indiqué plus haut.	

### 6-6-8 Protection en écriture à partir de commandes FINS envoyées aux UC via des réseaux

Dans le cas des UC série CS/CJ version 2.0 ou supérieure, vous pouvez configurer une protection pour empêcher l'écriture dans les UC et leur contrôle à l'aide de commandes FINS via des réseaux (c.-à-d., via des connexions différentes des connexions série directes). Cette protection comprend l'écriture à partir d'applications à l'aide de FinsGateway, et à partir de CX-Programmer, CX-Protocol et CX-Process. La lecture est toujours possible dans ce cas.

Si la protection en écriture est configurée, les opérations suivantes sont interdites : le téléchargement du programme utilisateur, de la configuration de l'API, de la mémoire d'E/S ou d'autres données ; le changement de mode de fonctionnement, l'édition en ligne ; et toute autre opération d'écriture ou de contrôle.

Même si la protection en écriture est configurée, vous pouvez définir des nœuds spécifiques à titre exceptionnel, pour permettre l'exécution d'opérations d'écriture ou de contrôle depuis ces nœuds.



**Remarque** Cette fonction empêche les opérations d'écriture ou de contrôle uniquement pour les commandes FINS. Elle n'influe pas sur les opérations d'écriture et de contrôle exécutées par toute autre méthode, par exemple, pour les liaisons de données.

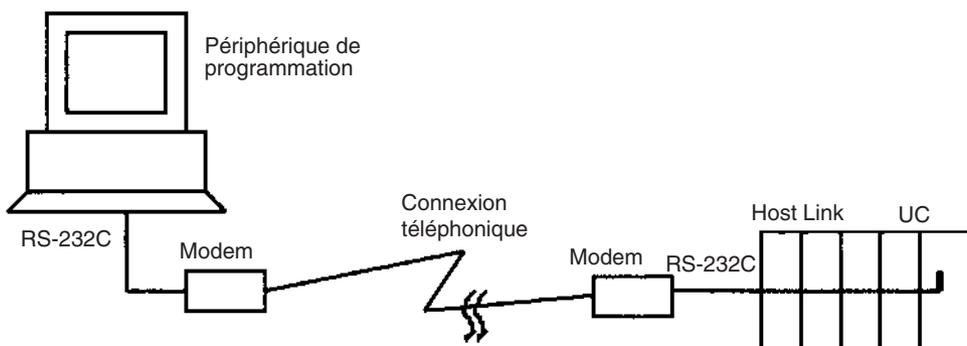
Pour plus de détails, reportez-vous au chapitre 1-4-3 *Protection en écriture à partir de commandes FINS envoyées aux UCs via le réseau* dans le *Manuel d'utilisation des API série CS* ou le *Manuel d'utilisation des API série CJ*.

## 6-6-9 Programmation et surveillance à distance

Les API série CS/CJ peuvent être programmés et surveillés à distance via un modem ou un réseau Controller Link.

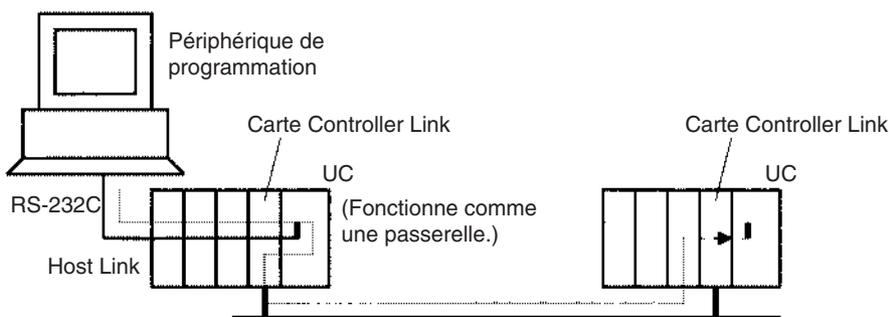
### 1,2,3... 1. Connexions par modem

La fonction Host Link peut fonctionner via un modem qui permet de surveiller par téléphone le fonctionnement d'un API déporté, les transferts de données ou encore, l'édition en ligne d'un programme d'API déporté. Toutes les opérations en ligne du périphérique de programmation sont prises en charge dans ces connexions.



### 2. Connexions par réseau Controller Link

Les API d'un réseau Controller Link ou Ethernet peuvent être programmés et surveillés via la liaison Host Link. Toutes les opérations en ligne du périphérique de programmation sont prises en charge dans ces connexions.



## 6-6-10 Profils des unités

Dans le cas des UC série CS/CJ, les informations que vous pouvez lire à partir du CX-Programmer sont les suivantes :

- les informations de fabrication (numéro de lot, numéro de série, etc.) : facilitent la fourniture d'informations à OMRON lorsque des problèmes surviennent sur des cartes ;
- les informations relatives aux cartes (type, numéro de modèle, position correcte de rack/emplacement) : permettent d'obtenir facilement des informations de montage ;
- le texte défini par l'utilisateur (256 caractères max.) : permet d'enregistrer des informations nécessaires à la maintenance (historique d'inspection des cartes, numéros de ligne de fabrication ou autres informations d'application) dans des cartes mémoire.

## 6-6-11 Mémoire flash

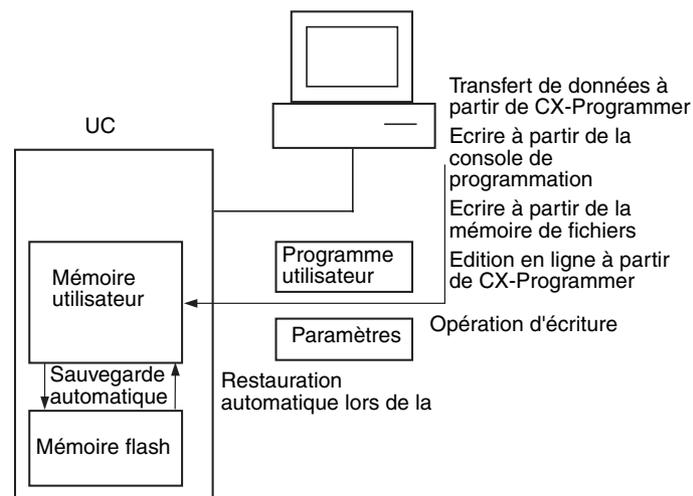
Cette fonction est uniquement prise en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D.

Dans le cas des UC CS1-H, CJ1-H, CJ1M ou CS1D, le programme utilisateur et les paramètres sont sauvegardés automatiquement dans la mémoire flash à chaque fois qu'ils sont écrits ou modifiés dans l'UC.

Les données sauvegardées automatiquement sont les suivantes : programme utilisateur, paramètres (y compris la configuration de l'API, les tables d'E/S enregistrées, les tables de routage et les données de cartes réseau telles que les tables de liaison de données).

Les données sont sauvegardées automatiquement à chaque fois que le programme utilisateur ou des paramètres sont écrits dans l'UC, y compris pour des opérations de transfert de données à partir du CX-Programmer, pour l'écriture de données à partir d'une console de programmation, l'édition en ligne, les transferts de données d'une carte mémoire ou d'une mémoire de fichiers EM, etc.

Les données de programme utilisateur et de paramètres écrites dans la mémoire flash sont transférées automatiquement dans la mémoire utilisateur de l'UC au démarrage.



### Remarque

1. Le voyant BKUP à l'avant de l'UC s'allume lorsque des données sont écrites dans la mémoire flash. Ne mettez pas l'UC hors tension tant que la sauvegarde n'est pas terminée (c.-à-d. tant que le voyant BKUP n'est pas éteint), après un transfert de données depuis un périphérique de programmation ou une mémoire de fichiers, ou après une édition en ligne.
2. Uniquement dans le cas de l'édition en ligne et si l'UC contient une batterie, l'UC redémarre dans l'état précédent (par exemple, avec le voyant BKUP allumé) même si le système est mis hors tension avant la fin de la sauvegarde, bien que le démarrage nécessite jusqu'à 1 minute. Dans ce cas également (et même si l'UC contient une batterie), vérifiez toujours que l'opération de sauvegarde est bien terminée avant de mettre l'alimentation hors tension, si vous prévoyez de ne pas vous servir de l'UC pendant une période prolongée.

Le temps requis pour la sauvegarde de données (durée pendant laquelle le voyant BKUP reste allumé) dépend de la taille du programme utilisateur, comme illustré dans le tableau suivant.

Taille du programme utilisateur	Temps de traitement de la sauvegarde		
	Mode MONITOR		Mode PROGRAM
	Temps de cycle de 0,4 ms (exemple)	Temps de cycle de 10,0 ms (exemple)	
10 Kpas	2 s	8 s	1 s
60 Kpas	11 s	42 s	6 s
250 Kpas	42 s	170 s	22 s

- Remarque**
1. Le voyant BKUP s'allume lorsque l'UC est alimentée en courant.
  2. Suivant le type d'édition en ligne réalisée, la sauvegarde des données peut nécessiter jusqu'à 1 minute.

**⚠ Attention** Les UC CS1-H, CJ1-H, CJ1M et CS1D enregistrent automatiquement le programme utilisateur ainsi que les données des paramètres dans la mémoire flash lorsque ces données sont écrites sur l'UC. Cependant, la mémoire d'E/S (y compris les zones DM, EM et HR) n'est pas sauvegardée dans la mémoire flash. Il est possible de conserver les zones DM, EM et HR en cas d'interruption d'alimentation grâce à une batterie. Si la batterie est défaillante, le contenu de ces zones risque de ne pas être exact après une interruption d'alimentation. Si le contenu des zones DM, EM et HR est utilisé pour contrôler des sorties externes, empêchez l'exécution de sorties inappropriées à chaque fois que le drapeau d'erreur de batterie (A40204) est à ON.

**Remarque** Le CX-Programmer affiche un état de sauvegarde dans une fenêtre d'état de sauvegarde de la mémoire, pendant la sauvegarde de données depuis CX-Programmer, pour des opérations de transfert différentes des transferts de données normaux (**PLC/Transfer**). Pour visualiser cette fenêtre, activez le paramètre d'affichage de la boîte de dialogue de l'état de sauvegarde dans les propriétés de l'API et sélectionnez la fenêtre dans le menu Affichage. Pour les opérations de transfert normales, l'état de sauvegarde s'affiche dans la fenêtre de transfert, après l'état de transfert du programme et d'autres données.

#### Drapeaux de la zone auxiliaire

Nom	Adresse	Signification
Drapeau d'erreur mémoire flash	A40310	Passé à ON lorsque la mémoire flash est défaillante.

## 6-6-12 Paramètres des conditions au démarrage

Cette fonction est uniquement prise en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D.

Certaines unités et cartes internes sont plus longues à démarrer après la mise sous tension, ce qui influe sur le temps de démarrage de l'UC. Vous pouvez paramétrer la configuration de l'API de sorte que l'UC démarre sans avoir besoin d'initialiser ces cartes.

Ce paramètre s'applique aux cartes d'interface bus ITNC-EIS01-CST et ITNC-EIX01-CST Open Network Controller-CS1. (Aucune carte interne n'est actuellement utilisable pour cette application depuis octobre 2001).

Cette fonction est contrôlée en configurant la condition au démarrage et le paramètre de carte interne décrits dans le tableau suivant.

Conditions au démarrage	Configuration de l'API	
	Condition au démarrage (adresse dans la console de programmation 83, bit 15)	Paramètre de la carte interne (adresse dans la console de programmation 84, bit 15)
Démarrer sans attendre toutes les unités et toutes les cartes	1 : activer le fonctionnement sans attendre.	1 : ne pas attendre de cartes internes spécifiques.
Démarrer sans attendre toutes les unités (attendre les cartes)	1 : activer le fonctionnement sans attendre.	0 : attendre toutes les cartes avant de démarrer.
Attendre toutes les unités et toutes les cartes avant de démarrer	0 : toujours attendre toutes les unités et toutes les cartes.	Quelconque

**Remarque** Dans le cas des UC CJ1, CS1-H, CJ1-H, CJ1M ou CS1D, l'UC ne démarrera pas avant que le processus de démarrage soit terminé pour l'ensemble des unités et des cartes.

#### Configuration de l'API

Adresse de la console de programmation		Nom	Paramètre	Par défaut	Fréquence de mise à jour UC
Mot	Bit				
83	15	Condition au démarrage	0 : attendre les cartes. 1 : ne pas attendre.	0 : attendre	Mise en marche
84	15	Paramètre de carte interne	0 : attendre toutes les cartes. 1 : ne pas attendre de cartes spécifiques.	0 : attendre	Mise en marche

#### Condition au démarrage

0 : si le processus de démarrage d'une ou plusieurs cartes ou unités spécifiques n'est pas terminé, l'UC se met en attente en mode MONITOR ou PROGRAM et attend toutes les unités et toutes les cartes.

1 : même si le processus de démarrage d'une ou plusieurs cartes ou unités spécifiques n'est terminé, l'UC poursuit son traitement et démarre en mode MONITOR ou PROGRAM. Toutefois, le fonctionnement des cartes internes dépend aussi du paramètre suivant.

#### Paramètre de carte interne

Ce paramètre ne s'utilise que si la condition au démarrage est définie sur 1 afin d'activer le démarrage sans attendre d'unités ou de cartes spécifiques. Ce paramètre n'est pas pris en compte si la condition au démarrage est définie sur 0.

0 : si le processus de démarrage d'une ou plusieurs cartes spécifiques n'est pas terminé, l'UC se met en attente en mode MONITOR ou PROGRAM, et attend toutes les cartes.

1 : même si le processus de démarrage d'une ou plusieurs cartes spécifiques n'est terminé, l'UC poursuit son traitement et démarre en mode MONITOR ou PROGRAM.

## 6-7 Fonctions de diagnostic

Ce chapitre fournit un bref récapitulatif des fonctions de diagnostic et de débogage suivantes :

- le journal d'erreurs ;
- la fonction sortie OFF ;
- les fonctions d'alarme erreur (FAL(006) and FALS(007)) ;
- la fonction de détection du point d'erreur (FPD(269)).

### 6-7-1 Journal d'erreurs

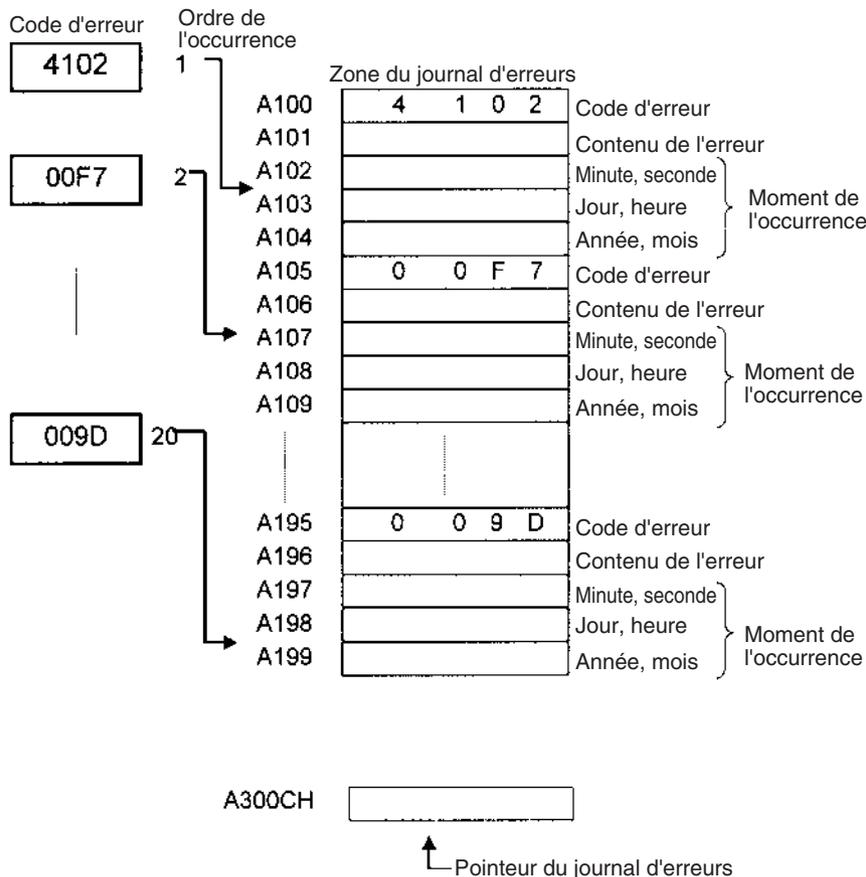
Chaque fois qu'une erreur survient dans un API série CS/CJ, l'UC enregistre les informations relatives aux erreurs dans la zone du journal d'erreurs. Les informations relatives aux erreurs incluent le code d'erreur (stocké dans A400), le contenu des erreurs et l'heure à laquelle l'erreur est survenue. Le journal d'erreurs peut stocker jusqu'à 20 enregistrements.

En plus des erreurs générées par le système, l'API enregistre les erreurs FAL(006) et FALS(007) définies par l'utilisateur, ce qui facilite le suivi de l'état de fonctionnement du système.

Reportez-vous au chapitre sur la résolution des problèmes dans le *Manuel d'utilisation, série CS/CJ* pour plus de détails.

**Remarque** Une erreur définie par l'utilisateur est générée lorsque les instructions FAL(006) ou FALS(007) sont exécutées dans le programme. Les conditions d'exécution de ces instructions constituent les conditions d'erreur définies par l'utilisateur. L'instruction FAL(006) génère une erreur non fatale et l'instruction FALS(007) une erreur fatale qui arrête l'exécution du programme.

Lorsque plus de 20 erreurs se produisent, les données des erreurs les plus anciennes (de A100 à A104) sont supprimées et le nouvel enregistrement est stocké de A195 à A199.



Le nombre d'enregistrements est mémorisé en binaire dans le pointeur du journal d'erreurs (A300). Le pointeur n'est pas incrémenté lorsque plus de 20 erreurs se sont produites.

### 6-7-2 Fonction sortie OFF

En tant que mesure d'urgence lorsqu'une erreur se produit, toutes les sorties des cartes de sorties peuvent être mises à OFF en activant le bit de sortie à OFF (A50015). Le mode de fonctionnement reste au mode RUN ou MONITOR, mais toutes les sorties passent à OFF.

**Remarque** En général (lorsque le bit de maintien IOM = OFF), toutes les sorties des cartes de sorties passent à OFF lorsque le mode de fonctionnement passe du mode RUN/MONITOR au mode PROGRAM. Le bit de sortie à OFF permet de désactiver toutes les sorties sans passer au mode PROGRAM ni arrêter l'exécution du programme.

#### Précautions d'application de DeviceNet

Si vous utilisez la fonction maître avec la carte CS1W-DRM21 ou CJ1W-DRM21, toutes les sorties esclaves passeront à OFF. Si vous utilisez la fonction esclave, toutes les entrées vers le maître seront à OFF. Cependant, si vous utilisez la carte C200HW-DRM21-V1, toutes sorties esclaves ne passeront pas à OFF.

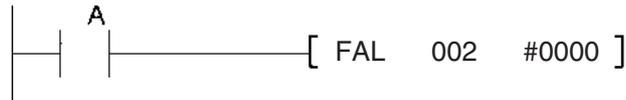
### 6-7-3 Fonctions d'alarme erreur

Les instructions FAL(006) et FALS(007) génèrent des erreurs définies par l'utilisateur. L'instruction FAL(006) génère une erreur non fatale et l'instruction FALS(007) une erreur fatale qui arrête l'exécution du programme.

Lorsque les conditions d'erreur définies par l'utilisateur (conditions d'exécution de FAL(006) ou FALS(007)) sont réunies, l'instruction d'alarme erreur est exécutée et le traitement ci-dessous s'effectue.

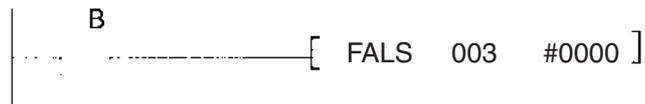
- 1,2,3...**
1. Le drapeau d'erreur FAL (A40215) ou le drapeau d'erreur FALS (A40106) passe à ON.
  2. Le code d'erreur correspondant est écrit sur A400.
  3. Le code d'erreur et l'heure où l'erreur s'est produite sont stockés dans le journal d'erreurs.
  4. Le voyant d'erreur à l'avant de l'UC clignote ou s'allume.
  5. Si FAL(006) a été exécutée, l'UC continue de fonctionner.  
Si FALS(007) a été exécutée, l'UC s'arrête de fonctionner.  
(L'exécution du programme s'arrête).

#### Fonctionnement de l'instruction FAL(006)



Lorsque la condition d'exécution A passe à ON, une erreur avec le numéro FAL 2 est générée, A40215 (drapeau d'erreur FAL) passe à ON et A36002 (drapeau FAL numéro 2) passe à ON. L'exécution du programme se poursuit. Pour effacer des erreurs générées par FAL(006), exécutez l'instruction FAL(006) avec le numéro FAL 00, ou exécutez l'opération de lecture/effacement d'erreur à partir d'un périphérique de programmation (y compris une console de programmation).

#### Fonctionnement de l'instruction FALS(007)



Lorsque la condition d'exécution B passe à ON, une erreur avec le numéro FALS 3 est générée et A40106 (drapeau d'erreur FALS) passe à ON. L'exécution du programme s'arrête.

Pour effacer les erreurs générées par FAL(006), éliminez la cause de l'erreur et exécutez l'opération de lecture/effacement d'erreur à partir d'un périphérique de programmation (y compris une console de programmation).

### 6-7-4 Détection du point d'erreur

FPD(269) effectue la surveillance du temps et le diagnostic logique. La fonction de surveillance du temps génère une erreur non fatale si la sortie de diagnostic ne passe pas à ON dans le temps de surveillance spécifié. La fonction de diagnostic logique indique quelle entrée empêche l'activation de la sortie de diagnostic.

#### Fonction de surveillance du temps

L'exécution de l'instruction FPD(269) démarre la temporisation et met le drapeau de retenue à ON si la sortie de diagnostic ne passe pas à ON dans le temps de surveillance spécifié. Vous pouvez programmer le drapeau de retenue en tant que condition d'exécution d'un bloc de traitement d'erreur. Il est également possible de programmer FPD(269) pour générer une erreur non fatale avec le numéro FAL désiré.

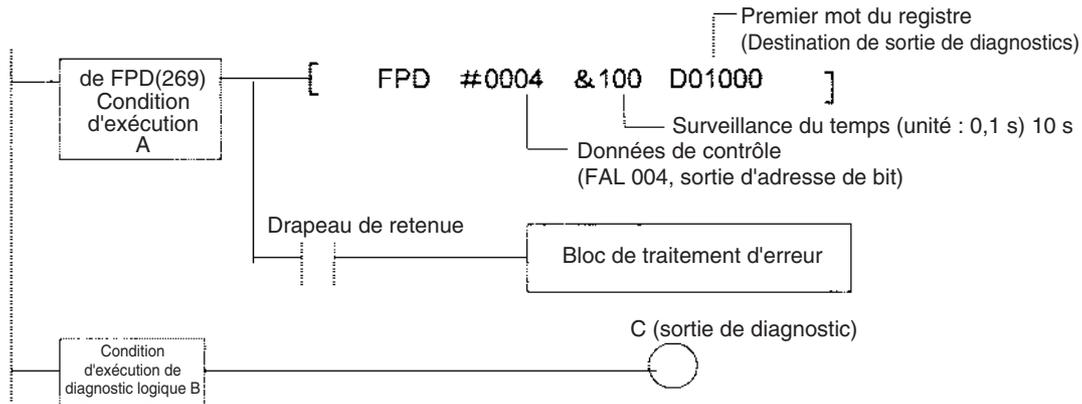
Lorsqu'une erreur FAL est générée, un message prédéfini est enregistré et peut être affiché sur un périphérique de programmation. Vous pouvez configurer FPD(269) pour éditer les résultats de diagnostic logique (l'adresse du bit qui empêche l'activation de la sortie de diagnostic) juste avant le message.

Vous pouvez utiliser la fonction d'apprentissage pour déterminer le temps réel requis pour l'activation de la sortie de diagnostic et configurer le temps de surveillance.

### Fonction de diagnostic logique

FPD(269) détermine quel bit d'entrée empêche l'activation de la sortie de diagnostic, et édite l'adresse de ce bit. Vous pouvez configurer la sortie sur la sortie d'adresse du bit (adresse de mémoire API) ou la sortie de message (ASCII).

- Si vous sélectionnez la sortie d'adresse de bit, l'adresse de mémoire API du bit peut être transférée vers un registre d'index et vous pourrez adresser indirectement le registre d'index lors d'un traitement ultérieur.
- Si vous sélectionnez la sortie de message, l'adresse du bit est enregistrée dans un message ASCII affichable sur un périphérique de programmation.



#### Surveillance du temps :

Surveille si la sortie C passe à ON dans un délai de 10 secondes après l'entrée A. Si ce n'est pas le cas, une erreur est détectée et le drapeau de retenue passe à ON. Le drapeau de retenue exécute le bloc de traitement d'erreur. Une erreur FAL (erreur non fatale) avec le numéro FAL 004 est également générée.

#### Diagnostic logique :

FPD(269) détermine quel bit d'entrée du bloc B empêche l'activation de la sortie C. L'adresse de ce bit est sortie vers D01000 et D01001.

#### Drapeaux et mots de la zone auxiliaire

Nom	Adresse	Fonctionnement
Code d'erreur	A400	Lorsqu'une erreur se produit, son code d'erreur est stocké dans A400.
Drapeau d'erreur FAL	A40215	ON lorsque FAL(006) est exécutée.
Drapeau d'erreur FALS	A40106	ON lorsque FALS(007) est exécutée.
Drapeaux de numéro d'instruction FAL exécutée	A360 à A391	Le drapeau correspondant passe à OFF lorsqu'une erreur FAL(006) ou FALS(007) se produit.
Zone du journal d'erreurs	A100 à A199	La zone de journal d'erreurs contient des informations sur les 20 erreurs les plus récentes.
Pointeur du journal d'erreurs	A300	Lorsqu'une erreur survient, le pointeur du journal d'erreurs est incrémenté de 1 pour indiquer où l'erreur suivante sera enregistrée sous forme de décalage à partir du début de la zone du journal d'erreurs (A100).
Bit de réinitialisation du pointeur du journal d'erreurs	A50014	Mettez ce bit à ON pour réinitialiser le pointeur du journal d'erreur (A300) à 00.
Bit d'apprentissage FPD	A59800	Mettez ce bit à ON si vous voulez que le temps de surveillance soit configuré automatiquement lors de l'exécution de FPD(269).

### 6-7-5 Simulation d'erreurs système

Cette fonction est uniquement prise en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D.

FAL(006) et FALS(007) permettent de créer intentionnellement des erreurs système fatales et non fatales. Vous pouvez utiliser cette fonction pendant le débogage du système, pour tester l'affichage de messages sur les terminaux opérateurs programmables (IHM) ou d'autres interfaces d'opérateurs.

Procédez de la manière suivante :

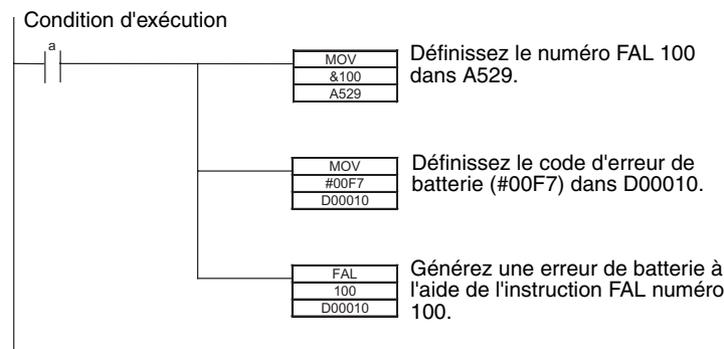
- 1,2,3...**
- Définissez le numéro de FAL/FALS à utiliser pour la simulation dans A529. (A529 est utilisé lors de la simulation d'erreurs pour FAL(006) et FALS(007)).
  - Définissez le numéro FAL/FALS à utiliser pour la simulation en tant que premier opérande de FAL(006) ou FALS(007).
  - Définissez le code d'erreur et l'erreur à simuler en tant que seconde opération (deux mots) de FAL(006) ou FALS(007). Indiquez une erreur non fatale pour FAL(006) et une erreur fatale pour FALS(007).

Pour simuler plusieurs erreurs système, utilisez plusieurs instructions FAL(006) ou FALS(007), comme décrit ci-dessus.

#### Drapeaux et mots de la zone auxiliaire

Nom	Adresse	Fonctionnement
Numéro FAL/FALS pour la simulation des erreurs système	A529	Choisissez un numéro d'instructions FAL/FALS factice pour simuler l'erreur système. 0001 à 01FF hex. : instructions FAL/FALS de numéros 1 à 511. 0000 ou 0200 à FFFF hex. : pas de numéro FAL/FALS pour la simulation des erreurs système.

#### Exemple d'erreur de batterie



**Remarque** Pour supprimer les erreurs systèmes simulées, utilisez les mêmes méthodes que pour les erreurs système réelles. Consultez le *Manuel d'utilisation, série CS* ou le *Manuel d'utilisation, série CJ* pour plus de détails. Vous pouvez supprimer toutes les erreurs système simulées à l'aide des instructions FAL(006) et FALS(007) en mettant l'alimentation hors tension.

### 6-7-6 Désactivation du stockage des erreurs FAL définies par l'utilisateur dans le journal d'erreurs

Cette fonction est uniquement prise en charge par les UC CS1-H, CJ1-H, CJ1M ou CS1D.

La configuration de l'API contient un paramètre qui empêche l'enregistrement des erreurs FAL définies par l'utilisateur, créées à l'aide de FAL(006) et de la surveillance du temps pour FPD(269), dans le journal d'erreurs (A100 à A199).

L'erreur FAL sera générée même si ce paramètre est utilisé, et les informations suivantes s'afficheront également : A40215 (drapeau d'erreur FAL), A360 à A391 (numéros de FAL exécutées) et A400 (code d'erreur).

Vous pouvez utiliser cette fonction lorsque vous avez uniquement besoin d'enregistrer les erreurs FAL système dans le journal d'erreurs, par exemple, lorsque le programme génère un grand nombre d'erreurs définies par l'utilisateur à l'aide de l'instruction FAL(006) et que le journal d'erreurs se remplit trop vite.

**Configuration de l'API**

Adresse dans console de programmation		Nom	Paramètre	Par défaut	Fréquence de mise à jour UC
Mot	Bit				
129	15	Paramètre de stockage des erreurs FAL utilisateur	0 : enregistrer les erreurs FAL définies par l'utilisateur dans le journal d'erreurs. 1 : ne pas enregistrer les erreurs FAL définies par l'utilisateur dans le journal d'erreurs.	0 : enregistrer.	Chaque fois que l'instruction FAL(006) est exécutée (à chaque cycle).

**Remarque** Les éléments suivants sont stockés dans le journal d'erreurs même si vous utilisez le paramètre ci-avant pour empêcher l'enregistrement des erreurs FAL définies par l'utilisateur :

- les erreurs fatales définies par l'utilisateur (FALS(007)) ;
- les erreurs système non fatales ;
- les erreurs système fatales ;
- les erreurs système non fatales simulées par l'utilisateur (FAL(006)) ;
- les erreurs système fatales simulées par l'utilisateur (FALS(007)).

## 6-8 Modes de traitement UC

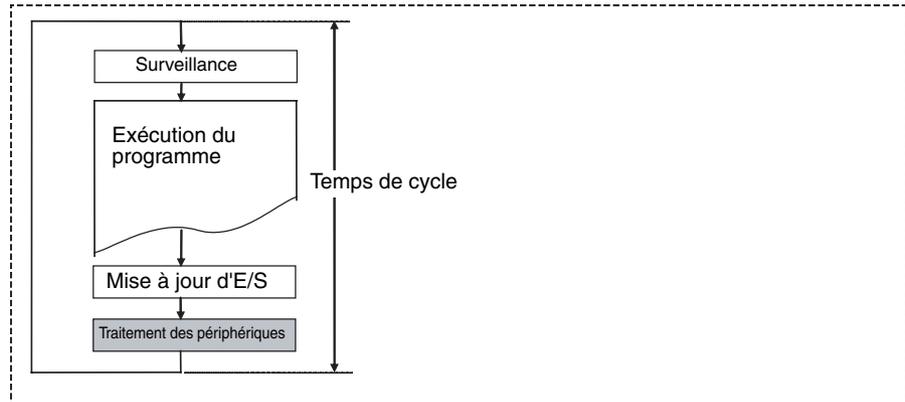
### 6-8-1 Modes de traitement UC

Habituellement, le traitement des périphériques (voir remarque) s'effectue à la fin de chaque cycle (après la mise à jour d'E/S) sur 4 % du cycle ou sur une durée définie par l'utilisateur pour chaque traitement. Par conséquent, il est impossible de traiter les périphériques à une vitesse plus rapide que le temps de cycle, et le temps requis pour le traitement des périphériques s'ajoute au temps de cycle.

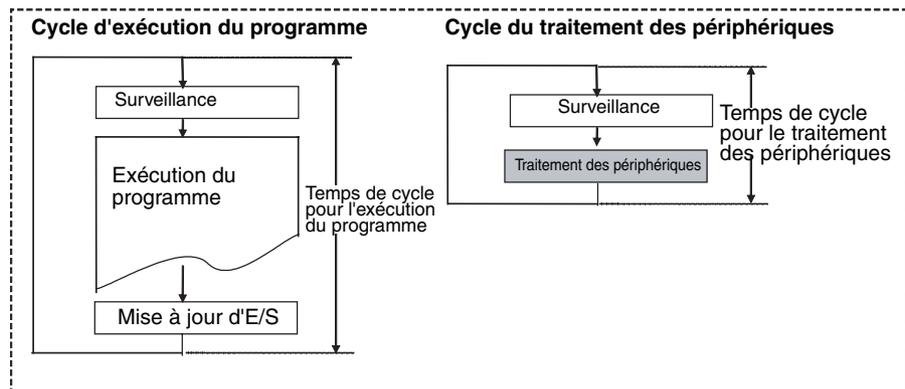
Toutefois, dans le cas des UC CS1D pour systèmes à UC seule, ou des UC CS1-H ou CJ1-H, des modes de traitement parallèle sont pris en charge pour permettre l'exécution de programme en parallèle à l'aide du traitement des périphériques. Ces modes autorisent un traitement des périphériques plus rapide et des temps de cycle plus courts, particulièrement lorsqu'un traitement des périphériques de grande ampleur est requis. (Les UC CS1D pour systèmes à UC en duplex et les UC CJ1M ne prennent pas en charge les modes de traitement parallèle).

**Remarque** Le traitement des périphériques comprend des services non programmés requis par des périphériques externes, tels que le traitement d'événements (exemple : communications de commandes FINS) pour des cartes d'E/S spéciales, des cartes réseau et des cartes internes (série CS uniquement), ainsi que le traitement des ports des communications périphériques et RS-232C (sans inclure les liaisons de données et d'autres mises à jour d'E/S spéciales de cartes réseau).

**Mode normal**



**Modes de traitement parallèle**



**Modes de traitement parallèle**

Il existe deux modes de traitement parallèle différents : le traitement parallèle avec accès à la mémoire synchrone ou le traitement parallèle avec accès à la mémoire asynchrone.

■ **Traitement parallèle avec accès à la mémoire asynchrone**

Dans ce mode, l'accès à la mémoire d'E/S pour le traitement des périphériques n'est pas synchronisé avec l'accès à la mémoire d'E/S pour l'exécution du programme. En d'autres termes, l'intégralité du traitement des périphériques est exécuté en parallèle avec l'exécution du programme, y compris l'accès à la mémoire. Ce mode assure l'exécution la plus rapide (par rapport aux autres modes) pour l'exécution du programme et le traitement des événements, lorsque la charge du traitement des périphériques est importante.

■ **Traitement parallèle avec accès mémoire synchrone**

Dans ce mode, l'accès à la mémoire d'E/S pour le traitement des périphériques n'est pas exécuté en parallèle avec l'exécution du programme, mais plutôt à la suite de l'exécution du programme, exactement comme en mode d'exécution normal, c'est-à-dire après la période de mise à jour d'E/S. Tous les autres traitements périphériques sont exécutés en parallèle avec l'exécution du programme.

Ce mode assure une exécution plus rapide que le mode d'exécution normal, pour l'exécution du programme comme pour le traitement des événements. Le temps de cycle de l'exécution du programme est plus long que celui du traitement parallèle avec accès mémoire asynchrone, du temps de mise à jour d'E/S requis pour le traitement des périphériques.

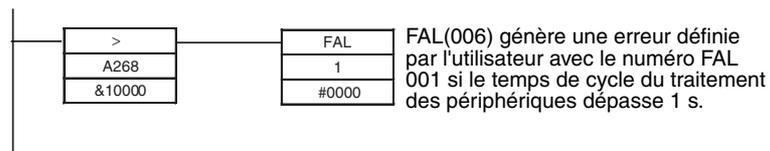
Les temps de cycle et les réponses du traitement des périphériques pour le traitement normal, le traitement parallèle avec accès mémoire asynchrone et le traitement parallèle avec accès mémoire synchrone sont répertoriés dans le tableau suivant. (Ces valeurs s'appliquent à un programme composé d'instructions standard avec un temps de cycle de 10 ms et une carte Ethernet).

Ces valeurs sont fournies à titre de référence seulement et varient en fonction du système.

	Mode normal	Traitement parallèle avec accès à la mémoire asynchrone	Traitement parallèle avec accès mémoire synchrone
Temps de cycle	Défini à 1 arbitrairement.	0,9	0,9
Traitement des périphériques	Défini à 1 arbitrairement.	0,4	1,0

**Remarque**

1. Le traitement des périphériques comprend le traitement d'événements (exemple : communications de commandes FINS) pour des cartes d'E/S spéciales, des cartes réseau et des cartes internes (série CS uniquement), ainsi que le traitement des ports des communications périphériques et RS-232C (sans inclure les liaisons de données et d'autres mises à jour d'E/S spéciales pour des cartes réseau).
2. Les UC CS1 version 1 ou supérieure et les UC CS1-H ou CJ1-H prennent également en charge un mode prioritaire du traitement des périphériques qui effectue le traitement des périphériques sur un cycle fixe pendant l'exécution du programme. Il assure un traitement des périphériques plus rapide que le mode de traitement normal, mais l'exécution du programme est plus lente. Cependant la réponse aux événements n'est pas aussi rapide que pour les modes de traitement parallèle. Par conséquent, vous devez utiliser le traitement parallèle avec accès mémoire asynchrone à chaque fois que la réponse aux événements doit être prioritaire dans le traitement.
3. Lors de l'utilisation du traitement parallèle, des erreurs de dépassement de temps de cycle du traitement des périphériques peuvent survenir dans l'UC, comme décrit dans les paragraphes *a)* et *b)*, ci-dessous. Si cette erreur se produit, le message qui s'affiche sur le périphérique de programmation indique que le temps de cycle est trop long, A40515 (dépassement de temps de cycle du traitement des périphériques) passe à ON et le fonctionnement s'arrête (erreur fatale).
  - a. Si le temps de cycle du traitement des périphériques est supérieur à 2,0 s, une erreur de dépassement de temps de cycle se produit. Vous pouvez surveiller le temps de cycle du traitement des périphériques dans A268 pour détecter les erreurs éventuelles avant qu'elles ne se produisent. Par exemple, une erreur définie par l'utilisateur peut être générée à l'aide de FAL numéro 001, si le temps de cycle du traitement des périphériques est supérieur à 1 s (c.-à-d., si le contenu de A268 dépasse 2710 hexadécimal (10 000 décimal)).



- b. Une erreur de dépassement de temps de cycle du traitement des périphériques peut également survenir si le temps de traitement du cycle d'exécution de l'instruction (c.-à-d., le temps d'exécution de l'instruction) est trop court. Ce temps est mémorisé dans A266 et A267 en mode d'exécution normal. A titre de référence, si le temps d'exécution de l'instruction est inférieur ou égal à 2 ms, une erreur de dépassement de temps de cycle du traitement des périphériques se produit et vous ne pouvez pas utiliser le mode de traitement parallèle. Lors du débogage de sections du programme uniquement (qui peut induire un temps d'exécution de l'instruction très court), utilisez le mode normal pour éviter que cette erreur ne se produise.

La console de programmation doit être déconnectée lorsque des applications utilisateur fonctionnent en mode de traitement parallèle. Du temps de traitement est alloué à la console de programmation pour augmenter la réponse aux touches de la console de programmation, ce qui

augmente le temps de traitement des périphériques et réduit l'efficacité du traitement parallèle.

### Configuration de l'API

Le mode de traitement est spécifié dans la configuration de l'API.

Adresse dans console de programmation		Nom	Paramètre	Par défaut	Fréquence de mise à jour UC
Mot	Bit				
219	08 à 15	Mode de traitement UC	00 hex. : mode normal 01 hex. : traitement parallèle avec accès mémoire synchrone 02 hex. : traitement parallèle avec accès mémoire asynchrone 05 à FF hex. : temps d'exécution du programme par tranche de temps pour le mode prioritaire du traitement des périphériques (5 à 255 ms par incréments de 1 ms)  Les paramètres de 03 et 04 hex. ne sont pas définis (non autorisés) et génèrent des erreurs de configuration de l'API (non fatales).	00 hex. : mode normal	Début du fonctionnement

### Drapeaux et mots de la zone auxiliaire

Nom	Adresse	Fonctionnement
Dépassement du temps de cycle du traitement des périphériques	A40515	Passé à ON lorsque le temps de cycle du traitement des périphériques dépasse 2 s. Le fonctionnement s'arrête.
Temps de cycle du traitement des périphériques	A268	Contient le temps de cycle du traitement des périphériques lorsque l'un des modes de traitement parallèle (accès mémoire synchrone ou asynchrone) est utilisé et que l'API est en mode RUN ou MONITOR. Le temps est indiqué en binaire entre 0,0 et 2000,0 (par incréments de 0,1 ms).
Temps d'exécution des instructions (total de tous les découpages du temps en intervalles d'exécution du programme et tous les découpages du temps en intervalles du traitement des périphériques).	A266 et A267	En mode normal, seul le temps d'exécution des instructions est inclus. Le temps est mémorisé sous forme de valeur binaire de 32 bits. 00000000 à FFFFFFFF hex. (pas : 0,1 ms) (0 à 429 496 729,5 ms) A266 : mot de poids inférieur A267 : mot de poids supérieur

### Traitement parallèle avec accès mémoire asynchrone

#### Exécutions du programme

Surveillance		Vérification du bus d'E/S et autres traitements 0,3 ms
Temps d'exécution des instructions		Temps d'exécution total de toutes les instructions
Calculs du temps de cycle minimum		Temps de traitement pour un temps de cycle d'exécution du programme minimum
Traitement cyclique	Mise à jour d'E/S	Temps de mise à jour d'E/S pour chaque carte multiplié par le nombre de cartes.
	Mise à jour d'E/S spéciales des cartes réseau	Temps de mise à jour d'E/S spéciales pour chaque carte multiplié par le nombre de cartes.
Traitement des périphériques	Accès au fichier	Temps de traitement des périphériques défini dans la configuration de l'API (par défaut : 4 % du temps de cycle).

**Traitement des périphériques**

Surveillance	Vérification de la batterie, vérification de la mémoire du programme utilisateur, etc. 0,2 ms	
Traitement des périphériques	Traitement d'événements pour les cartes d'E/S spéciales	Inclut le traitement d'événements pour accéder à la mémoire d'E/S. (Voir remarque). 1 ms max. pour chaque service.
	Traitement d'événements pour les cartes réseau	
	Traitement du port périphérique	
	Traitement du port RS-232C	
	Traitement d'événements pour les cartes internes (série CS uniquement)	
	Traitement d'événements pour les ports de communications (ports logiques internes) en cours d'utilisation (y compris l'exécution en arrière-plan).	

**Remarque** Le traitement d'événements pour accéder à la mémoire d'E/S comprend : 1) Le traitement de toutes les commandes FIN reçues qui accèdent à la mémoire d'E/S (commandes de lecture/écriture dans la mémoire d'E/S avec des codes communs commençant par 01 hex. ou commandes de configuration/réinitialisation forcée avec des codes communs commençant par 23 hex.) et 2) Le traitement de toutes les commandes en mode C reçues qui accèdent à la mémoire d'E/S (exceptées les liaisons NT utilisant le port périphérique ou RS-232C).

**Traitement parallèle avec accès mémoire synchrone**

**Exécutions du programme**

Surveillance	Vérification du bus d'E/S et autres traitements 0,3 ms	
Temps d'exécution des instructions	Temps d'exécution total de toutes les instructions	
Calculs du temps de cycle minimum	Temps de traitement pour un temps de cycle d'exécution du programme minimum	
Traitement cyclique	Mise à jour d'E/S	Temps de mise à jour d'E/S pour chaque carte multiplié par le nombre de cartes.
	Mise à jour d'E/S spéciales des cartes réseau	Temps de mise à jour d'E/S spéciales pour chaque carte multiplié par le nombre de cartes.
Traitement des périphériques	Accès au fichier	Temps de traitement des périphériques défini dans la configuration de l'API (par défaut : 4 % du temps de cycle).
	Traitement d'événements nécessitant l'accès à la mémoire d'E/S. (Voir remarque).	

**Traitement des périphériques**

Surveillance	Vérification de la batterie, vérification de la mémoire du programme utilisateur, etc. 0,2 ms	
Traitement des périphériques	Traitement d'événements pour les cartes d'E/S spéciales	Sauf pour le traitement d'événements pour accéder à la mémoire d'E/S. (Voir remarque). 1 ms max. pour chaque service.
	Traitement d'événements pour les cartes réseau	
	Traitement du port périphérique	
	Traitement du port RS-232C	
	Traitement d'événements pour les cartes internes (série CS uniquement)	
	Traitement d'événements pour les ports des communications (ports logiques internes) en cours d'utilisation (y compris l'exécution en arrière-plan).	

**Remarque** Le traitement d'événements pour accéder à la mémoire d'E/S comprend : 1) Le traitement de toutes les commandes FIN reçues qui accèdent à la mémoire d'E/S

(commandes de lecture/écriture dans la mémoire d'E/S avec des codes communs commençant par 01 hex. ou commandes de configuration/réinitialisation forcée avec des codes communs commençant par 23 hex.) et 2) Le traitement de toutes les commandes en mode C reçues qui accèdent à la mémoire d'E/S (exceptées les liaisons NT utilisant le port périphérique ou RS-232C).

### 6-8-2 Mode de traitement parallèle et temps de cycle minimums

Si un temps de cycle minimum est spécifié lorsque vous utilisez un mode de traitement parallèle, une pause est insérée après l'exécution du programme jusqu'à ce que le temps de cycle minimum soit atteint, mais le traitement des périphériques continue.

### 6-8-3 Concurrence de données en traitement parallèle avec accès mémoire asynchrone

Lors de l'utilisation du traitement parallèle avec accès mémoire asynchrone, les données ne doivent pas être concurrentes dans les cas suivants

- Lorsque plusieurs mots sont lus dans la mémoire d'E/S à l'aide d'une commande de communication, les données contenues dans les mots ne peuvent pas être concurrentes.
- Si une instruction lit plusieurs de la mémoire d'E/S et qu'un traitement des périphériques est exécuté pendant l'exécution des instructions, les données contenues dans les mots ne peuvent pas être concurrentes.
- Si le même mot de la mémoire d'E/S est lu par plusieurs instructions à différents emplacements du programme et que le traitement des périphériques est exécuté entre l'exécution des instructions, les données contenues dans le mot ne peuvent pas être concurrentes.

Pour garantir la concurrence des données, si nécessaire, procédez de la manière suivante :

1. Utilisez le traitement parallèle avec accès mémoire synchrone.
2. Utilisez l'instruction IOSP(287) pour désactiver le traitement des périphériques aux endroits requis du programme, puis réactivez le traitement des périphériques à l'aide de l'instruction IORS(288).

## 6-9 Mode prioritaire du traitement des périphériques

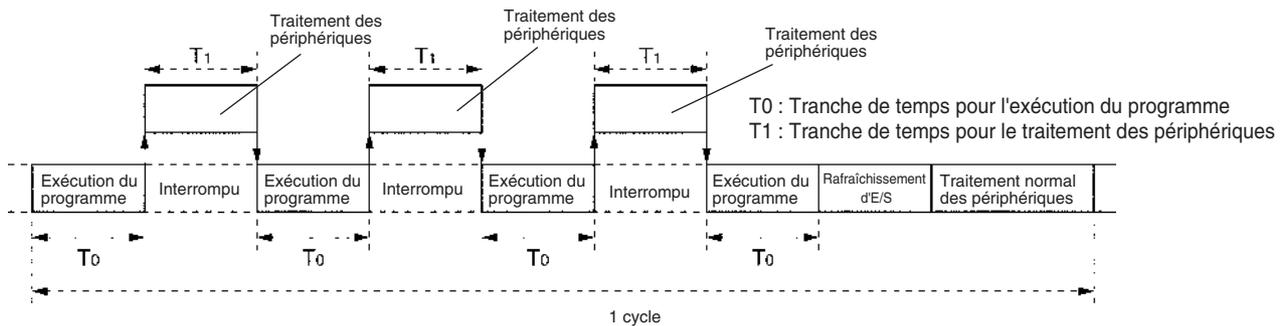
Normalement, le traitement des périphériques du port RS-232C, du port périphérique, de la carte interne (série CS uniquement), des cartes réseau et des cartes d'E/S spéciales n'est pris en charge qu'une seule fois, à la fin du cycle, après la mise à jour d'E/S. Le temps alloué à chaque service est égal à 4 % de temps de cycle ou à un temps défini par l'utilisateur. Il existe cependant un mode qui active le traitement périodique en cours de cycle. Ce mode, appelé mode prioritaire du traitement des périphériques, est paramétré dans la configuration de l'API.

#### Remarque

Vous pouvez utiliser le mode prioritaire du traitement des périphériques avec les UC de la série CJ ou CS, mais les UC CS1 série CS doivent porter le numéro de lot 001201□□□□ ou un numéro postérieur (date de fabrication au 1er décembre 2000 ou postérieure). (Le mode prioritaire du traitement des périphériques n'est pas pris en charge par les UC CS1D pour systèmes à CPU en duplex).

### 6-9-1 Mode prioritaire du traitement des périphériques

Si le mode prioritaire du traitement des périphériques est configuré, l'exécution du programme s'interrompt au moment spécifié, le traitement spécifié s'effectue, puis l'exécution du programme reprend. Ce processus se répète pendant toute l'exécution du programme. Un traitement des périphériques normal s'effectue également après la période de mise à jour d'E/S.



Par conséquent, le mode prioritaire du traitement des périphériques permet d'exécuter un traitement périodique sur des ports ou des cartes spécifiés, parallèlement au traitement des périphériques normal. Ceci autorise l'utilisation d'applications pour lesquelles le traitement des périphériques doit être prioritaire sur l'exécution du programme, par exemple des applications de contrôle de processus qui requièrent une réponse rapide pour la surveillance de l'ordinateur hôte.

- Il est possible de spécifier jusqu'à cinq cartes ou ports pour le traitement prioritaire. Les cartes réseau et les cartes d'E/S spéciales CS/CJ sont spécifiées à l'aide du numéro de carte.
- Une seule carte ou un seul port est exécuté pendant chaque tranche de temps du traitement des périphériques. Si le traitement se termine avant l'expiration du temps spécifié, l'exécution du programme reprend immédiatement et la carte ou le port suivant(e) n'est pris en charge qu'à partir de la prochaine tranche de temps du traitement des périphériques. Il est toutefois possible de traiter plusieurs fois la même carte ou le même port pendant le cycle.
- Les cartes/ports sont traité(e)s dans l'ordre de leur détection par l'UC.

#### Remarque

1. Bien que les instructions ci-dessous utilisent les ports des communications, elles ne seront exécutées qu'une seule fois pendant le cycle d'exécution, même si le mode prioritaire du traitement des périphériques est utilisé :  
RXD(235) (RECEIVE)  
TXD(236) (TRANSMIT)
2. Si plusieurs mots sont lus via une commande de communication, il est impossible de garantir la concurrence des données lues lors de l'utilisation du mode prioritaire du traitement des périphériques.
3. En mode prioritaire du traitement des périphériques, l'UC peut dépasser le temps de cycle maximum. Le temps de cycle maximum est défini dans la configuration de l'API en tant que paramètre du temps de cycle d'horloge. Si le temps de cycle dépasse la valeur du paramètre du temps de cycle d'horloge, le drapeau de temps de cycle trop long (A40108) passe à ON et l'API s'arrête de fonctionner. Si le mode prioritaire du traitement des périphériques est utilisé, surveillez le temps de cycle actuel dans A264 et A265 et réglez le temps de cycle d'horloge (adresse : +209) comme requis. (La plage de configuration va de 10 à 40 000 ms par incréments de 10 ms, et la configuration par défaut est 1 s).

**Paramètres de configuration de l'API**

Pour utiliser le mode prioritaire du traitement des périphériques, vous devez configurer les paramètres suivants dans configuration de l'API.

- Temps découpé en intervalles pour l'exécution du programme : 5 à 255 ms par incréments de 1 ms.
- Temps découpé en intervalles pour le traitement des périphériques : 0,1 à 25,5 ms par incréments de 0,1 ms.
- Cartes et/ou ports pour le traitement prioritaire : carte réseau (par n° de carte), carte d'E/S spéciales CS/CJ (par n° de carte), carte interne (série CS uniquement), port RS-232C, port périphérique.

Adresse dans la console de programmation		Paramètre	Par défaut	Fonction	Application de la nouvelle configuration
Mot	Bit(s)				
219	08 à 15	00 05 à FF (hex.)	00	00 : désactive le traitement en mode prioritaire. 05 à FF : tranche de temps pour l'exécution des instructions (5 à 255 ms par incréments de 1 ms).	Prend effet au début du fonctionnement. (Ne peut pas être modifié au cours du fonctionnement).
	00 à 07	00 à FF (hex.)	00	00 : désactive le traitement en mode prioritaire. 01 à FF : tranche de temps pour l'exécution du traitement des périphériques (0,1 à 25,5 ms par incréments de 0,1 ms).	
220	08 à 15	00 10 à 1F 20 à 2F E1 FC FD (hex.)	00	00 : désactive le traitement en mode prioritaire.	
	00 à 07		00	10 à 1F : numéro de carte des cartes réseau + 10 (hex.). 20 à 7F : numéro de carte des cartes d'E/S spéciales série CS/CJ + 20 (hex.).	
221	08 à 15		00	E1 : carte interne.	
	00 à 07		00	FC : port RS-232C.	
222	08 à 15	00	FD : port périphérique.		

- Le tableau ci-dessous décrit le fonctionnement et les erreurs en fonction des paramètres dans la configuration de l'API.
- Vous ne pouvez pas réaliser ce paramétrage à partir du CX-Programmer pour les UC CS1 ou CJ1. Vous pouvez réaliser ce paramétrage à partir du CX-Programmer version 2.1 ou supérieure pour les UC CS1-H ou CJ1-H.

Conditions			Fonctionnement de l'UC	Erreurs de configuration de l'API
Tranche de temps pour le traitement des périphériques	Tranche de temps pour l'exécution des instructions	Cartes et ports spécifié(e)s		
01 à FF : (0,1 à 25,5 ms)	05 à FF : (5 à 255 ms)	Paramètres tous corrects	Mode prioritaire du traitement des périphériques	Aucune
		Paramètres à 00 et corrects		
		Paramètres corrects mais redondants		
		Plusieurs paramètres non autorisés	Mode prioritaire du traitement des périphériques pour des éléments correctement paramétrés	Générées
		Paramètres tous à 00	Fonctionnement normal	Générées
		Paramètres à 00 et non autorisés		
		Paramètres tous non autorisés		
00	00	---	Fonctionnement normal	Aucune
N'importe quelle autre valeur			Fonctionnement normal	Générées

**Remarque** Si une erreur est détectée dans la configuration de l'API, A40210 passe à ON et une erreur non fatale se produit.

**Informations de la zone auxiliaire**

Si vous configurez des temps découpés en intervalles pour l'exécution du programme et le traitement des périphériques, le total de tous les découpages du temps en intervalles d'exécution du programme et de tous les découpages du temps en intervalles du traitement des périphériques est stocké dans A266 et A267. Ces informations peuvent servir de référence pour effectuer les réglages appropriés sur les découpages du temps.

Lorsque le mode prioritaire du traitement des périphériques n'est pas utilisé, le temps d'exécution du programme est mémorisé. Vous pouvez utiliser cette valeur pour déterminer des paramètres de découpages du temps appropriés.

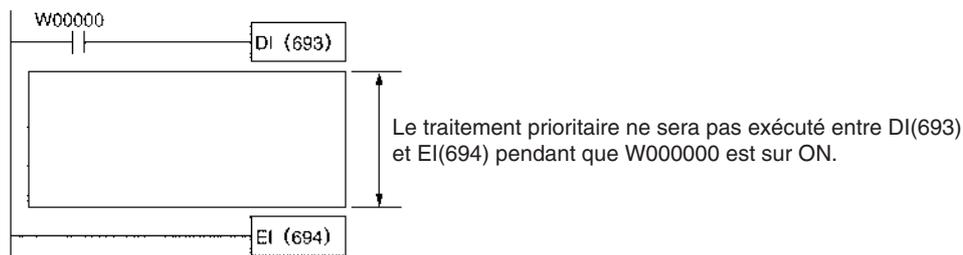
Mots	Table des matières	Signification	Mise à jour			
A266 et A267	00000000 à FFFFFFFF hex. (0 à 4294967295 décimal)	<p>Total de tous les découpages du temps en intervalles d'exécution du programme et tous 0,0 à 429 496 729,5 ms (incrément de 0,1 ms)</p> <table border="1"> <tr> <td>A267 (Octets les plus significatifs)</td> <td>A266 (Octets les moins significatifs)</td> <td>La valeur est sauvegardée sous la forme d'une valeur binaire de 32 bits (8 chiffres hexadécimaux)</td> </tr> </table>	A267 (Octets les plus significatifs)	A266 (Octets les moins significatifs)	La valeur est sauvegardée sous la forme d'une valeur binaire de 32 bits (8 chiffres hexadécimaux)	Le contenu est mis à jour à chaque cycle et effacé au début du fonctionnement.
A267 (Octets les plus significatifs)	A266 (Octets les moins significatifs)	La valeur est sauvegardée sous la forme d'une valeur binaire de 32 bits (8 chiffres hexadécimaux)				

**6-9-2 Désactivation temporaire du traitement en mode prioritaire**

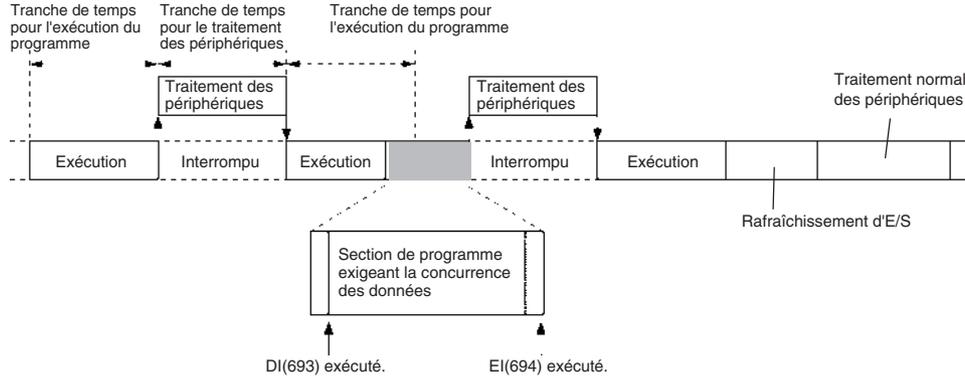
La concurrence des données n'est pas garantie dans les cas suivants, si le mode prioritaire du traitement des périphériques est utilisé.

- Lorsque plusieurs mots sont lus à partir d'un périphérique à l'aide d'une commande de communication. Les données risquent d'être lues pendant différentes tranches de temps du traitement des périphériques, entraînant ainsi une non concurrence des données.
- Lorsque le programme contient des instructions à temps d'exécution longs, par exemple, lors du transfert de grandes quantités de données de mémoire d'E/S. L'opération de transfert risque de s'interrompre pour le traitement des périphériques, entraînant ainsi une non concurrence des données. Cela se vérifie lorsque des mots en train d'être écrits par le programme sont lus depuis un périphérique avant que l'écriture ne soit terminée, ou lorsque des mots en train d'être lus par le programme sont écrits depuis un périphérique avant que la lecture ne soit terminée.
- Lorsque deux instructions accèdent aux mêmes mots dans la mémoire. Si ces mots sont écrits à partir d'un périphérique entre les intervalles d'exécution des deux instructions, ces deux instructions liront différentes valeurs dans la mémoire.

Lorsque la concurrence des données doit être garantie, vous pouvez utiliser les instructions DISABLE INTERRUPTS et ENABLE INTERRUPTS (DI(693) et EI(694)) avec les UC CS1 ou CJ1, pour empêcher le traitement prioritaire pendant des sections obligatoires du programme, comme illustré dans l'exemple suivant. Dans le cas des UC CS1D pour systèmes à UC seule et des UC CS1-H, CJ1-H ou CJ1M, vous pouvez utiliser les instructions DISABLE PERIPHERAL SERVICING et ENABLE PERIPHERAL SERVICING (IOSP(287) et IORS(288)).



**Fonctionnement**



**Remarque**

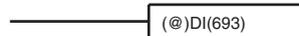
1. DI(693) et IOSP(287) désactivent les interruptions pour le traitement prioritaire, mais également toutes les autres interruptions, y compris les interruptions d'E/S, programmées et externes. Toutes les interruptions générées sont exécutées après l'exécution de la tâche cyclique (après l'exécution de END(001)), à moins que l'instruction CLI(691) ne soit exécutée d'abord pour supprimer les interruptions.
2. La désactivation des interruptions à l'aide de DI(693) ou de IOSP(287) est effective jusqu'à l'exécution de EI(694) ou IORS(288), de END(001), ou jusqu'à ce que l'API s'arrête de fonctionner. Il est donc impossible de créer des sections de programme qui dépassent la fin d'une tâche ou d'un cycle. Utilisez DI(693) et EI(694), ou IOSP(287) et IORS(288), dans chaque tâche cyclique, pour désactiver des interruptions dans plusieurs cycles ou tâches si nécessaire.

**UC CS1 et CJ1**

**DI(693)**

L'exécution de l'instruction DI(693) désactive toutes les interruptions (à l'exception des interruptions pour la tâche d'interruption d'alimentation), y compris les interruptions pour le traitement prioritaire, les interruptions d'E/S, les interruptions programmées ou les interruptions externes. Si DI(693) est exécutée alors que les interruptions sont déjà désactivées, celles-ci restent désactivées.

**Symbole**



**Zones de programme utilisables**

Zone	Application possible
Zones de programmation de bloc	Oui
Zones de programmation de pas	Oui
Sous-programmes	Oui
Tâches d'interruption	Non

**Drapeaux de condition**

Drapeau	Etiquette	Fonctionnement
Drapeau d'erreur	ER	Passé à ON si DI(693) est exécutée dans une tâche d'interruption, sinon à OFF.

**EI(694)**

L'exécution de l'instruction EI(694) active toutes les interruptions (à l'exception des interruptions pour la tâche d'interruption d'alimentation), y compris les interruptions pour le traitement prioritaire, les interruptions d'E/S, les interruptions programmées ou les interruptions externes. Si EI(694) est exécutée alors que les interruptions sont déjà activées, celles-ci restent activées.

**Symbole**



**Zones de programme utilisables**

Zone	Application possible
Zones de programmation de bloc	Oui
Zones de programmation de pas	Oui
Sous-programmes	Oui
Tâches d'interruption	Non

**Drapeaux de condition**

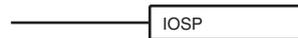
Drapeau	Etiquette	Fonctionnement
Drapeau d'erreur	ER	Passé à ON si EI(694) est exécutée dans une tâche d'interruption.

**UC CS1D pour systèmes à UC seule et UC CS1-H, CJ1-H et CJ1M**

**IOSP(287)**

L'exécution de IOSP(287) désactive le traitement des périphériques. Si IOSP(287) est exécutée alors que le traitement des périphériques est déjà désactivé, celui-ci reste désactivé.

**Symbole**



**Zones de programme utilisables**

Zone	Application possible
Zones de programmation de bloc	Oui
Zones de programmation de pas	Oui
Sous-programmes	Oui
Tâches d'interruption	Non

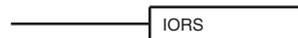
**Drapeaux de condition**

Drapeau	Etiquette	Fonctionnement
Drapeau d'erreur	ER	Passé à ON si IOSP(287) est exécutée dans une tâche d'interruption, sinon à OFF.

**IORS(288)**

L'exécution de IORS(288) active le traitement des périphériques désactivé à l'aide de IOSP(287). Si IORS(288) est exécutée alors que le traitement des périphériques est déjà activé, celui-ci reste activé.

**Symbole**



**Zones de programme utilisables**

Zone	Application (possible ?)
Zones de programmation de bloc	Oui
Zones de programmation de pas	Oui
Sous-programmes	Oui
Tâches d'interruption	Non

**Drapeaux de condition**

Drapeau	Etiquette	Fonctionnement
Drapeau d'erreur	ER	Passé à ON si IORS(288) est exécutée dans une tâche d'interruption.

## 6-10 Fonctionnement sans batterie

Les API série CS et CJ peuvent fonctionner sans batterie (ou avec une batterie vide). La procédure utilisée pour le fonctionnement sans batterie dépend des éléments suivants :

- UC ;
- maintien ou pas de la mémoire d'E/S (par exemple : la zone CIO) ;
- initialisation ou pas des zones DM et EM au démarrage ;
- initialisation ou pas des zones DM et EM à partir du programme utilisateur.

Le tableau suivant récapitule les différences mentionnées ci-dessus.

UC	Pas de maintien de la mémoire d'E/S.		Maintien de la mémoire d'E/S.
	Pas d'initialisation des zones DM et EM au démarrage.	Initialisation des zones DM et EM au démarrage.	
		A partir du programme utilisateur	
CS1-H, CJ1-H, CJ1M ou CS1D	Utilisez le fonctionnement normal (avec la mémoire flash) ou une carte mémoire.	Utilisez le transfert automatique à partir d'une carte mémoire au démarrage. (Placez la broche 2 de l'interrupteur DIP sur ON).	Impossible quelle que soit la méthode. Vous devez installer une batterie.
CS1 ou CJ1	Utilisez le transfert automatique à partir d'une carte mémoire au démarrage. (Placez la broche 2 de l'interrupteur DIP sur ON).		

### Remarque

1. Lorsque vous utilisez le fonctionnement sans batterie, désactivez la détection de tension de batterie faible dans la configuration de l'API, quelle que soit la méthode de fonctionnement sans batterie utilisée.
2. Si aucune batterie n'est connectée ou si la batterie est vide, le fonctionnement de l'UC est soumis aux restrictions suivantes. Cela est vrai quelle que soit l'UC utilisée.
  - L'état du bit de sortie à OFF (A50015) n'est pas fiable. Lorsque le bit de sortie à OFF est activé, toutes les sorties des cartes de sorties passent à OFF.  
Incluez les instructions suivantes au schéma de contact pour éviter que toutes les sorties des cartes de sorties ne passent à OFF, lors de la mise sous tension.

Drapeau du premier cycle  
(A20011)



- Le contenu de la mémoire d'E/S (y compris les zones HR, DM et EM) risque de ne pas être maintenu correctement. Par conséquent, paramétrez la configuration de l'API de manière que le drapeau de maintien de la mémoire d'E/S (A50012) et le drapeau de maintien de l'état forcé (A50013) ne soient pas maintenus lors de la mise sous tension.
- La fonction d'horloge ne peut pas être utilisée. Les données d'horloge de A351 à A354 et l'heure de démarrage dans A510 et A511 ne sont pas fiables. Les dates de fichiers figurant sur les fichiers écrits dans la carte mémoire à partir de l'UC ne sont pas fiables non plus.
- Les données suivantes sont toutes remises à zéro au démarrage : l'heure de mise sous tension (A523), l'heure de l'interruption d'alimentation (A512 et A513) et le nombre d'interruptions d'alimentation (A514).
- La zone de journal d'erreurs de A100 à A199 n'est pas maintenue.
- La banque EM actuelle est toujours à 0 au démarrage.
- La mémoire de fichiers EM ne contient plus aucun fichier au démarrage et les fonctions de la mémoire de fichiers ne peuvent pas être utilisées. Pour utiliser la mémoire de fichiers EM, vous devez la réinitialiser dans la configuration de l'API et la reformater.

**UC CS1-H, CJ1-H, CJ1M ou CS1D**

Les UC CS1-H, CJ1-H, CJ1M ou CS1D peuvent fonctionner sans batterie, en mode de fonctionnement normal. Les données du programme utilisateur et des paramètres sont sauvegardées automatiquement dans la mémoire flash de l'UC et restaurées automatiquement depuis la mémoire flash, au démarrage. Dans ce cas, la mémoire d'E/S n'est pas maintenue et les zones DM et EM doivent être initialisées à partir du programme utilisateur.

Le fonctionnement sans batterie est également possible avec les UC CS1-H, CJ1-H, CJ1M ou CS1D, en transférant automatiquement les données depuis une carte mémoire au démarrage, comme c'est le cas pour les UC CS1. (Avec une carte mémoire, vous pouvez inclure les données des zones DM et EM).

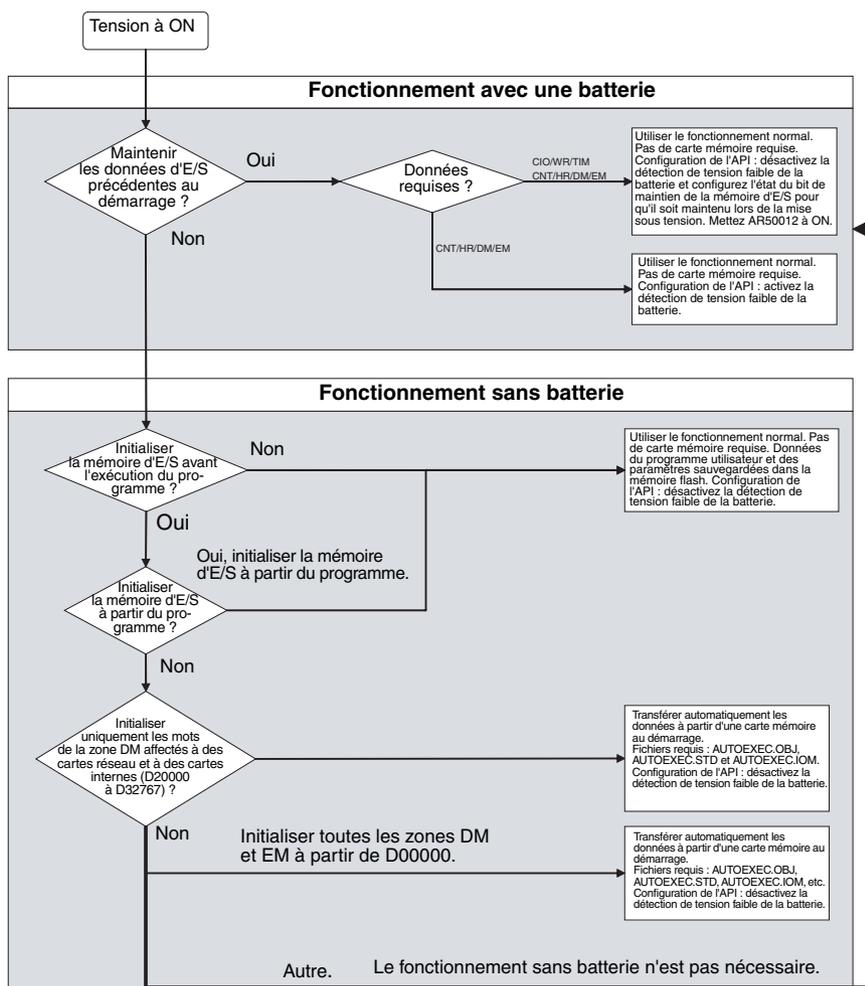
**UC CS1 et CJ1**

Le fonctionnement sans batterie est possible avec les UC CS1 et CJ1, en transférant automatiquement les données depuis une carte mémoire au démarrage. Dans ce cas, la mémoire d'E/S n'est pas maintenue. (Avec une carte mémoire, vous pouvez inclure les données des zones DM et EM).

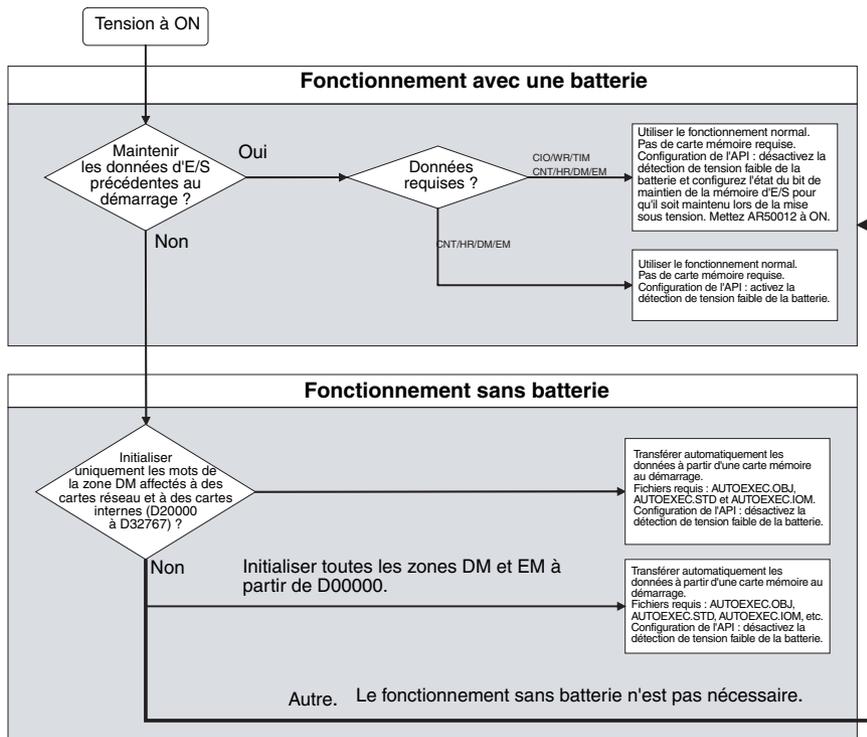
**Procédure**

Les organigrammes suivants décrivent les procédures pour les deux types d'UC.

**UC CS1-H, CJ1-H, CJ1M ou CS1D**



UC CS1 et CJ1

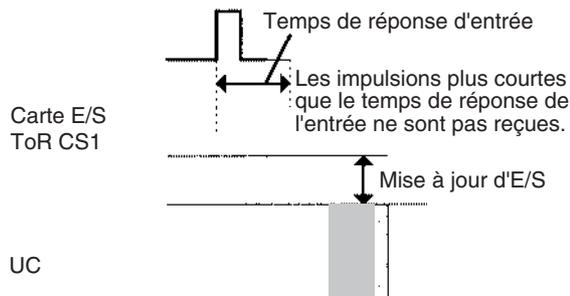
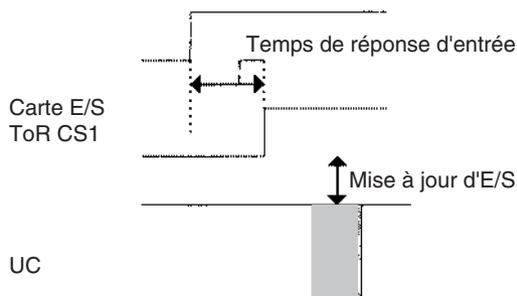


## 6-11 Autres fonctions

### 6-11-1 Paramètres des temps de réponse d'E/S

Les temps de réponse des entrées peuvent être configurés par rack et par numéro d'emplacement pour les cartes d'E/S standard CS/CJ. L'augmentation du temps de réponse des entrées réduit les effets de vibration et de parasites. La réduction du temps de réponse des entrées (en conservant une largeur d'impulsion supérieure au temps de cycle) permet la réception d'impulsions d'entrée plus courtes.

**Remarque** Dans le cas des UC de la série CS, les impulsions plus courtes que le temps de cycle peuvent être traitées à l'aide des entrées à grande vitesse disponibles sur certaines cartes d'E/S haute densité C200H ou à l'aide d'une carte d'entrées à grande vitesse. Voir la section 6-1-4 Entrées à grande vitesse pour plus d'informations.



**Configuration de l'API**

Vous pouvez configurer les temps de réponse des entrées pour les 80 emplacements d'un API CS/CJ (de l'emplacement 0 du rack 0 à l'emplacement 9 du rack 7) dans les 80 octets des adresses 10 à 49.

Adresse dans console de programmation	Nom	Paramètre (hex.)	Par défaut (hex.)
10 Bits 0 à 7	Temps de réponse des entrées des cartes d'E/S standard CS/CJ, pour l'emplacement 0 du rack 0	00 : 8 ms 10 : 0 ms 11 : 0,5 ms 12 : 1 ms 13 : 2 ms 14 : 4 ms 15 : 8 ms 16 : 16 ms 17 : 32 ms	00 (8 ms)
:	:	:	:
49 Bits 8 à 15	Temps de réponse des entrées des cartes d'E/S standard CS/CJ, pour l'emplacement 7 du rack 9	Comme ci-dessus.	00 (8 ms)

**6-11-2 Affectation de la zone d'E/S**

Vous pouvez utiliser un périphérique de programmation pour configurer le premier mot de l'affectation d'E/S dans les racks d'extension (racks d'extension CS/CJ et racks d'extension d'E/S C200H). Cette fonction permet de définir une zone d'affectation d'E/S de chaque rack fixe dans la plage CIO 0000 à CIO 0999. (Les premiers mots sont affectés par numéro de rack).



## CHAPITRE 7

# Transfert du programme, essais de fonctionnement et débogage

Ce chapitre décrit les processus utilisés pour transférer le programme vers l'UC, ainsi que les fonctions disponibles pour tester et déboguer le programme.

7-1	Transfert du programme . . . . .	348
7-2	Essais de fonctionnement et débogage . . . . .	348
7-2-1	Configuration et réinitialisation forcées . . . . .	348
7-2-2	Surveillance différenciée . . . . .	349
7-2-3	Edition en ligne. . . . .	350
7-2-4	Traçabilité des données. . . . .	353

## 7-1 Transfert du programme

Un périphérique de programmation permet de transférer les programmes, la configuration de l'API, les données de mémoire d'E/S et les commentaires d'E/S vers l'UC, lorsque celle-ci est en mode PROGRAM.

### Procédure de transfert du programme pour CX-Programmer

- 1,2,3...
1. Sélectionnez **PLC, Transfer**, puis **To PLC**. La boîte de dialogue Download Options s'affiche.
  2. Sélectionnez les éléments à transférer parmi la liste suivante : programmes (Programs), paramètres (configuration de l'API) (Settings [PLC Setup]), table d'E/S (I/O table), symboles (Symbols), commentaires (Comments) et index de programme (Program index).

**Remarque** Les éléments table d'E/S et commentaires ne sont sélectionnables que s'ils existent sur la carte mémoire de l'UC.

3. Cliquez sur **OK**.

Vous pouvez transférer le programme à l'aide de l'une des méthodes suivantes :

- Transfert automatique à la mise sous tension

Le fichier AUTOEXEC.OBJ de la carte mémoire sera lu sur l'UC (la broche 2 de l'interrupteur DIP doit être à ON) à la mise sous tension.

- Remplacement du programme pendant le fonctionnement

Vous pouvez remplacer le fichier de programme existant par le fichier de programme spécifié dans la zone auxiliaire en activant le bit de démarrage du remplacement dans la zone auxiliaire (A65015), à partir du programme, pendant le fonctionnement de l'UC. Voir la section *CHAPITRE 5 Fonctions de mémoire de fichiers* pour plus d'informations.

**Remarque** Si vous utilisez CX-Programmer version 4.0 ou supérieure avec des UC série CS/CJ version 2.0 ou supérieure, vous pouvez télécharger les programmes de tâches individuellement. Pour plus de détails, consultez le chapitre 1-4-1 *Téléchargement et chargement des différentes tâches* dans le *Manuel d'utilisation des API, série CS* ou dans le *Manuel d'utilisation des API de la série CJ*.

## 7-2 Essais de fonctionnement et débogage

### 7-2-1 Configuration et réinitialisation forcées

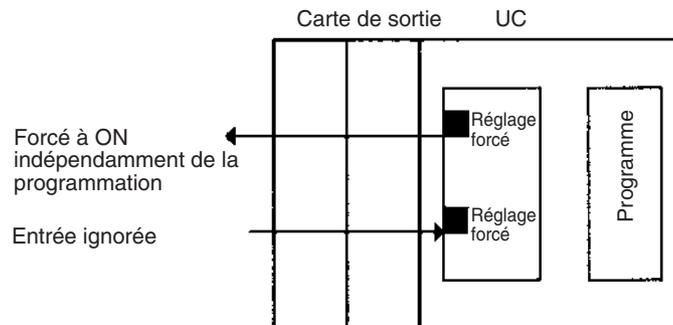
Un périphérique de programmation peut forcer la configuration (ON) ou la réinitialisation (OFF) de bits spécifiés (zone CIO, zone auxiliaire, zone HR et drapeaux de fin de temporisation/compteur). L'état forcé l'emportera sur l'état résultant du programme ou de la mise à jour d'E/S. Cet état ne peut être remplacé par des instructions, et sera mémorisé indépendamment de l'état du programme ou des entrées externes, jusqu'à ce qu'il soit remis à zéro à partir d'un périphérique de programmation.

Les opérations de configuration/réinitialisation forcée s'utilisent pour forcer l'entrée et la sortie lors d'un essai de fonctionnement ou pour forcer certaines conditions lors d'un débogage.

Vous pouvez exécuter des opérations de configuration/réinitialisation forcée en mode MONITOR ou en mode PROGRAM, mais pas en mode RUN.

**Remarque** Activez (ON) simultanément le bit de maintien de l'état forcé (A50013) et le bit de maintien IOM (A50012) pour que l'état des bits en configuration ou réinitialisation forcée soit conservé lorsque vous changez de mode de fonctionnement.

Activez (ON) le bit de maintien de l'état forcé (A50013) et le bit de maintien IOM (A50012), et réglez le paramètre de configuration de l'API « Bit de maintien de l'état forcé au démarrage » de manière à conserver l'état du bit de maintien de l'état forcé et donc à maintenir l'état des bits de configuration ou réinitialisation forcée lors de la mise hors tension.



La configuration et la réinitialisation forcées sont possibles dans les zones suivantes :

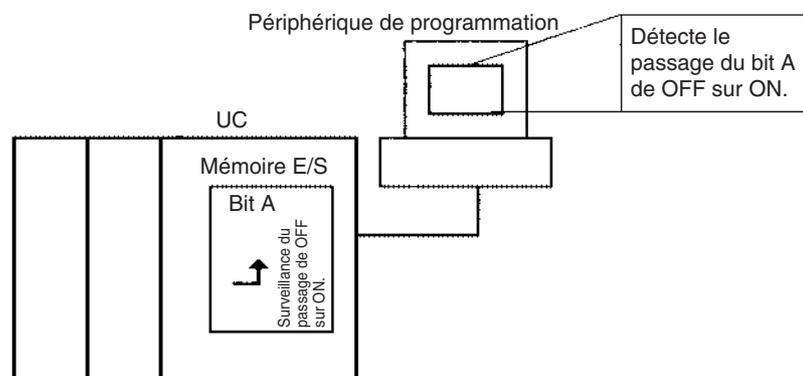
CIO (bits d'E/S, bits de liaison de données, bits de cartes réseaux, bits de cartes d'E/S spéciales, bits de cartes internes, bits SYSMAC BUS, bits de cartes d'E/S optiques, bits de travail), zone de travail (WR), drapeaux de fin de temporisation, zone HR, drapeaux de fin de compteur. (Les zones de cartes internes, SYSMAC BUS et de bornes d'E/S sont prises en charge par les UC série CS uniquement).

### Fonctionnement du périphérique de programmation

- Sélectionnez des bits pour la configuration/réinitialisation forcée.
- Sélectionnez la configuration forcée ou la réinitialisation forcée.
- Remettez l'état forcé à 0 (inclut la remise à zéro simultanée de tous les états forcés).

## 7-2-2 Surveillance différenciée

Lorsque l'UC détecte qu'un bit configuré par un périphérique de programmation est passé de OFF à ON ou de ON à OFF, les résultats sont indiqués par le drapeau de surveillance différenciée terminée (A50809). Le drapeau passe à ON lorsque les conditions définies pour la surveillance différenciée sont satisfaites. Un périphérique de programmation peut surveiller et afficher ces résultats à l'écran.



### Fonctionnement du périphérique de programmation pour CX-Programmer

- 1,2,3... 1. Cliquez avec le bouton droit sur le bit de la surveillance différenciée.
2. Cliquez sur **Differential Monitor** dans le menu de l'API. La boîte de dialogue de la surveillance différenciée s'affiche.
3. Cliquez sur **Rising** ou **Falling**.
4. Cliquez sur le bouton **Start**. L'avertisseur sonore émet un son lorsque le changement spécifié est détecté et le compte est alors incrémenté.
5. Cliquez sur le bouton **Stop**. La surveillance différenciée s'arrête.

**Bits/mots auxiliaires associés**

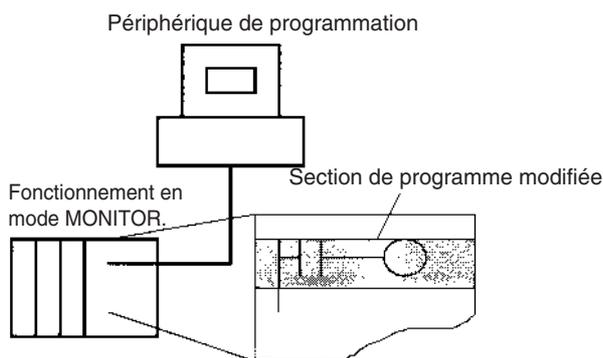
Nom	Adresse	Description
Drapeau de surveillance différenciée terminée	A50809	<p>Passé à ON lorsque la condition de surveillance différenciée a été satisfaite pendant la surveillance différenciée.</p> <p>Remarque : le drapeau est remis à 0 au démarrage de la surveillance différenciée.</p>

**7-2-3 Edition en ligne**

La fonction d'édition en ligne sert à ajouter ou à modifier une partie d'un programme dans une UC, directement à partir des périphériques de programmation, lorsque l'UC est en mode MONITOR ou PROGRAM. Les ajouts et les modifications s'effectuent une instruction à la fois à partir de la console de programmation, et une ou plusieurs sections de programme à la fois à partir de CX-Programmer. Par conséquent, la fonction permet d'apporter des modifications mineures au programme sans arrêter l'UC.

Vous pouvez procéder à l'édition en ligne simultanée à partir de plusieurs ordinateurs exécutant CX-Programmer et à partir d'une console de programmation dans la mesure où les tâches modifiées sont bien distinctes.

## Edition en ligne



Le temps de cycle passera d'un à plusieurs temps de cycle si le programme de l'UC est modifié en ligne en mode MONITOR.

Le temps de cycle des UC CS1-H, CJ1-H, CJ1M et CS1D s'allongera également pour permettre la sauvegarde des données dans la mémoire flash après l'édition en ligne. Le voyant BKUP sera allumé durant cette période. La progression de la sauvegarde s'affiche sur CX-Programmer. Les augmentations par cycle sont répertoriées dans le tableau suivant.

UC	Augmentation du temps de cycle	
	Edition en ligne	Sauvegarde en mémoire flash
UC CS1 pré-EV1	90 ms max.	Non prise en charge.
UC CS1 EV1 ou supérieure	12 ms max.	
UC CS1-H		4 % ou temps de cycle
UC CS1D		
UC CS1		Non prise en charge.
UC CJ1-H		4 % ou temps de cycle
UC CJ1M		

Dans le cas d'une UC CS1-H, CJ1-H, CJ1M ou CS1D, il existe une limite au nombre de modifications que vous pouvez apporter successivement. Le nombre réel dépend du type de modification apportée, toutefois les indications fournies ci-après peuvent être prises pour références.

CJ1M-CPU□□ :	40 modifications
CS1G-CPU□□H/CJ1G-CPU□□H :	160 modifications
CS1H-CPU□□H/CJ1H-CPU□□H/ CS1D-CPU□□H/CS1D-CPU□□S :	400 modifications

Si vous dépassez la limite, un message s'affichera sur CX-Programmer ou la console de programmation et vous ne pourrez poursuivre la modification que lorsque l'UC aura terminé la sauvegarde des données.

### Taille de la tâche et allongement du temps de cycle

Le rapport entre la taille de la tâche en cours d'édition et l'allongement du temps de cycle est le suivant :

Dans le cas d'une UC CS1, CS1-H, CS1D, CJ1 ou CJ1M version 1 ou supérieure, la taille de la tâche (programme) en cours de modification n'a pratiquement aucune incidence sur l'allongement du temps de cycle induit par l'édition en ligne.

Dans le cas d'une UC CS1 antérieure à EV1, la taille de la tâche en cours de modification déterminera la durée d'interruption d'un programme en vue de l'édition en ligne. Si vous divisez le programme en tâches plus petites, le temps de cycle s'allongera moins avec la fonction Edition en ligne qu'avec les anciens modèles d'API.

### Précautions

Si vous réécrivez un programme à l'aide de la fonction Edition en ligne en mode MONITOR, le temps de cycle sera plus long que d'habitude. Par conséquent, assurez-vous qu'il ne dépasse pas le temps de surveillance du cycle défini dans la configuration de l'API. S'il dépasse le temps de surveillance, une erreur de dépassement de temps de cycle se produit et l'UC s'arrête. Pour redémarrer l'UC, sélectionnez d'abord le mode PROGRAM avant de passer au mode RUN ou MONITOR.

### Remarque

Si la tâche modifiée en ligne contient un programme de bloc, les données d'exécution précédentes, par exemple l'état En attente (WAIT) ou Pause, seront effacées par l'édition en ligne, et l'exécution suivante commencera au début.

### Edition en ligne à partir de CX-Programmer

- 1,2,3...**
1. Affichez la section de programme à modifier.
  2. Sélectionnez les instructions à modifier.
  3. Sélectionnez **Program, Online Edit**, puis **Begin**.
  4. Modifiez les instructions.
  5. Sélectionnez **Program, Online Edit**, puis **Send Changes**. Les instructions sont vérifiées et, si elles ne contiennent aucun erreur, elles sont transférées vers l'UC. Les instructions contenues dans l'UC sont réécrites et l'augmentation du temps de cycle se produit à ce stade.

### Attention

Avant de poursuivre l'édition en ligne, vérifiez que l'allongement du temps de cycle n'influera pas sur le fonctionnement. Si le temps de cycle est trop long, certains signaux d'entrée risquent de ne pas être traités.

### Désactivation temporaire de l'édition en ligne

Il est possible de désactiver l'édition en ligne d'un cycle afin d'assurer les caractéristiques de réponse nécessaires au contrôle de la machine pendant ce cycle. L'édition en ligne à partir du périphérique de programmation sera désactivée pour un cycle et toutes les requêtes d'édition en ligne reçues pendant ce cycle seront conservées jusqu'au cycle suivant.

Pour désactiver l'édition en ligne, mettez le bit de désactivation de l'édition en ligne (A52709) à ON et définissez la Validation du bit de désactivation de l'édition en ligne (A52700 à A52707) sur 5A. Lorsque ces paramètres sont configurés de cette manière et qu'une requête d'édition en ligne est reçue, l'édition en ligne est mise en attente et le drapeau d'attente d'édition en ligne (A20110) passe à ON.

Lorsque le bit de désactivation de l'édition en ligne (A52709) est mis à OFF, l'édition en ligne est exécutée, le drapeau de traitement d'édition en ligne (A20111) passe à ON et le drapeau d'attente d'édition en ligne (A20110) passe à OFF. Une fenêtre indiquant la fin de l'installation s'affiche à l'écran.

Vous pouvez également désactiver temporairement l'édition en ligne en mettant le bit de désactivation de l'édition en ligne (A52709) à ON pendant l'édition en ligne. Dans ce cas aussi, le drapeau d'attente d'édition en ligne (A20110) passera à ON.

Si une seconde requête d'édition en ligne est réceptionnée alors que la première est en attente, la seconde requête ne sera pas enregistrée et une erreur se produira.

Vous pouvez également désactiver l'édition en ligne pour éviter qu'elle ne se produise accidentellement. Comme décrit plus haut, pour désactiver l'édition en ligne, mettez le bit de désactivation de l'édition en ligne (A52709) à ON et définissez la Validation du drapeau de désactivation de l'édition en ligne (A52700 à A52707) sur 5A.

### **Activation de l'édition en ligne à partir d'un périphérique de programmation**

Lorsqu'il est impossible d'activer l'édition en ligne à partir du programme, vous pouvez l'activer à partir du CX-Programmer.

**1,2,3...**

#### 1. Edition en ligne à l'aide d'une console de programmation

Si l'édition en ligne est exécutée à partir d'une console de programmation et qu'il est impossible de remettre à zéro l'état En attente de l'édition en ligne, la console de programmation sera verrouillée et vous ne pourrez effectuer aucune opération à l'aide de celle-ci.

Dans ce cas, branchez CX-Programmer sur un autre port série et mettez le bit de désactivation de l'édition en ligne (A52709) à 0. L'édition en ligne sera traitée et vous pourrez à nouveau effectuer des opérations à l'aide de la console de programmation.

#### 2. Edition en ligne à l'aide du CX-Programmer

Si les opérations se poursuivent alors que l'édition en ligne est à l'état En attente, CX-Programmer risque de se déconnecter (hors ligne). Si cela se produit, rétablissez la connexion entre l'ordinateur et l'API, et mettez le bit de désactivation de l'édition en ligne (A52709) à OFF.

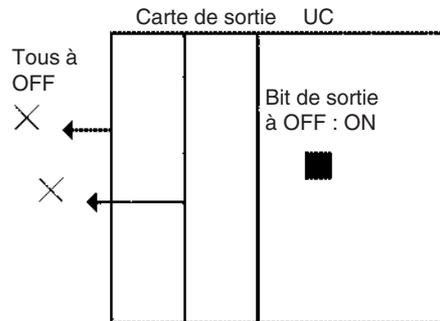
### **Bits/mots auxiliaires associés**

Nom	Adresse	Description
Validation du bit de désactivation de l'édition en ligne	A52700 à A52707	Valide le bit de désactivation de l'édition en ligne (A52709). Autre que 5A : Bit de désactivation de l'édition en ligne non valide 5A : Bit de désactivation de l'édition en ligne valide
Bit de désactivation de l'édition en ligne	A52709	Pour désactiver l'édition en ligne, mettez ce bit à ON et définissez la Validation du bit de désactivation de l'édition en ligne (A52700 à A52707) sur 5A.
Drapeau d'attente d'édition en ligne	A20110	A ON lorsqu'un traitement d'édition en ligne est en attente en raison de la désactivation de l'édition en ligne.
Drapeau de traitement d'édition en ligne	A20111	A ON lorsqu'un traitement d'édition en ligne est en cours d'exécution.

### **Désactivation des sorties**

Si le bit de sortie OFF (A50015) est activé par le biais de l'instruction OUT ou à partir d'un périphérique de programmation, toutes les sorties de toutes les cartes de sorties seront désactivées (ceci s'applique également aux sorties intégrées universelles ou aux sorties d'impulsions des UC), et le voyant INH à l'avant de l'UC s'allumera.

L'état du bit de sortie à OFF est conservé même après une coupure d'alimentation.



### 7-2-4 Traçabilité des données

La fonction de traçage des données échantillonne des données spécifiées de la mémoire d'E/S à l'aide de l'une des méthodes de temporisation ci-dessous, et stocke les données échantillonnées dans la mémoire de tracé, où elles pourront être lues et vérifiées ultérieurement, à partir d'un périphérique de programmation.

- Temps d'échantillonnage spécifié (10 à 2 550 ms par pas de 10 ms).
- Un échantillon par cycle.
- Lors de l'exécution de l'instruction TRACE MEMORY SAMPLING (TRSM).

Vous pouvez spécifier jusqu'à 31 bits et 6 mots dans la mémoire d'E/S, pour l'échantillonnage. La capacité de la mémoire de tracé est de 4 000 mots.

#### Procédure de base

1,2,3...

1. L'échantillonnage démarre lorsque les paramètres ont été configurés à partir du CX-Programmer et que la commande de lancement du traçage a été exécutée.
2. Les données échantillonnées (après l'étape 1 ci-dessus) sont tracées lorsque la condition de déclenchement du tracé est satisfaite, et les données présentes immédiatement après le retard (voir remarque 1) sont stockées dans la mémoire de tracé.
3. Les données de la mémoire de tracé sont échantillonnées, et le tracé est terminé.

#### Remarque

Valeur de retard : spécifie de combien de périodes (d'échantillonnage) l'échantillonnage doit être décalé dans la mémoire de tracé, à partir de l'instant où le bit de démarrage du tracé (A50814) est activé. Les plages de configuration sont indiquées dans le tableau suivant.

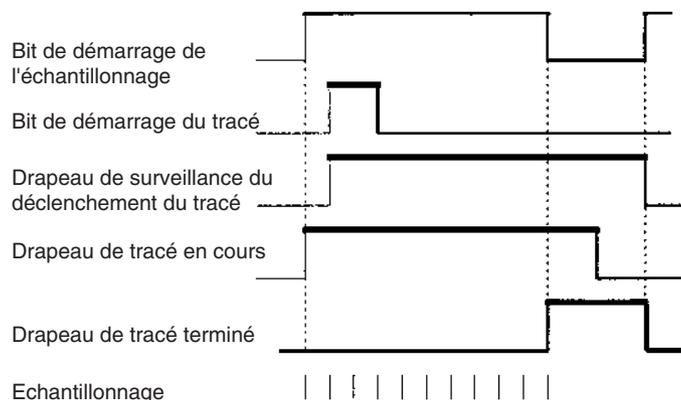
Nombre de mots échantillonnés	Plage de configuration
0	– 1 999 à 2 000
1	– 1 332 à 1 333
2	– 999 à 1 000
3	– 799 à 8 000
4	– 665 à 666
5	– 570 à 571
6	– 500 à 666

Retard positif : mémorisation de données différées du retard défini.

Retard négatif : mémorisation de données antérieures en fonction du retard défini.

**Exemple :** l'échantillonnage à 10 ms avec un temps de retard de – 30 ms donne – 30 x 10 = 300 ms. Par conséquent, les données présentes 300 ms avant le déclenchement seront mémorisées.

**Remarque** Utilisez un périphérique de programmation pour activer le bit de démarrage de l'échantillonnage (A50815). N'activez jamais ce bit à partir du programme utilisateur.



Vous pouvez exécuter les tracés suivants :

#### Tracé de données programmé

Un tracé de données programmé réalise un échantillonnage des données à intervalles fixes. Les temps d'échantillonnage spécifiés vont de 10 à 2 550 ms par pas de 10 ms. N'utilisez jamais l'instruction TRSM dans le programme utilisateur et assurez-vous que la période d'échantillonnage définie est supérieure à 0.

#### Tracé de données sur un cycle

Un tracé de données sur un cycle réalise un échantillonnage des données de mise à jour d'E/S à la fin des tâches du cycle complet. N'utilisez jamais l'instruction TRSM dans le programme utilisateur et assurez-vous que la période d'échantillonnage définie est supérieure à 0.

#### Tracé de données via TRSM

Un échantillon est prélevé une fois, lorsque l'instruction TRACE MEMORY SAMPLING (TRSM) est exécutée. Si plusieurs instructions TRSM sont utilisées dans le programme, un échantillon est prélevé à chaque exécution de l'instruction TRSM, après que la condition de déclenchement du tracé a été satisfaite.

### Procédure de traçage des données

Pour exécuter un tracé, procédez comme suit :

- 1,2,3...**
1. Utilisez CX-Programmer pour configurer les paramètres de tracé de données (exécutez *PLC/Data Trace* et configurez sous *Execute/Set*) : l'adresse des données échantillonnées, la période d'échantillonnage, le temps de retard et les conditions de déclenchement.
  2. Utilisez CX-Programmer pour démarrer l'échantillonnage ou activez le bit de démarrage de l'échantillonnage (A50815).
  3. Appliquez la condition de déclenchement du tracé.
  4. Terminez le tracé des données.
  5. Utilisez CX-Programmer pour lire les données de tracé.
    - a) Sélectionnez **Data Trace** dans le menu de l'API.
    - b) Sélectionnez **Select** dans le menu d'exécution.
    - c) Sélectionnez **Execute** dans le menu d'exécution.
    - d) Sélectionnez **Read** dans le menu d'exécution.

## Bits/mots auxiliaires associés

Nom	Adresse	Description
Bit de démarrage de l'échantillonnage	A50815	Utilisez un périphérique de programmation pour activer ce bit de démarrage de l'échantillonnage. Ce bit doit être activé à partir d'un périphérique de programmation. N'activez ou ne désactivez jamais ce bit à partir du programme utilisateur. Remarque : le bit sera remis à zéro lorsque le tracé de données sera terminé.
Bit de démarrage du tracé	A50814	Lorsque ce bit est activé, le déclenchement du tracé est surveillé et les données échantillonnées sont stockées dans la mémoire de tracé, si la condition de déclenchement est satisfaite. Les tracés activés par ce bit sont les suivants : 1) Tracé programmé (traçage à intervalles fixes de 10 à 2 550 ms) 2) Tracé à l'aide d'une instruction TRSM (traçage lorsque la TRSM est exécutée) 3) Tracé sur un cycle (traçage à la fin de l'exécution de toutes les tâches cycliques)
Drapeau de surveillance du déclenchement du tracé	A50811	Ce drapeau passe à ON lorsque la condition de déclenchement du tracé a été satisfaite après l'activation du bit de démarrage du tracé. Ce drapeau passe à OFF lorsque vous redémarrez l'échantillonnage en activant le bit de démarrage de l'échantillonnage.
Drapeau de tracé en cours	A50813	Ce drapeau passe à ON lorsque l'échantillonnage est démarré par un bit de démarrage de l'échantillonnage, et repasse à OFF lorsque le tracé est terminé.
Drapeau de tracé terminé	A50812	Ce drapeau passe à ON si la mémoire de tracé est pleine après que la condition de déclenchement du tracé a été satisfaite pendant un tracé et il repasse à OFF au démarrage de l'échantillonnage suivant.



# Annexe A

## Tableaux de comparaison des API : API série CJ, série CS, C200HG/HE/HX, CQM1H, CVM1 et série CV

### Comparaison des fonctions

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Fonctions de base	Capacité	2 560 points	5 120 points	1 184 points	6 144 points	512 points	
	Nombre de points E/S	250 Kpas Un pas correspond à un mot. Consultez la fin du chapitre 10-5 <i>Temps d'exécution d'instruction et nombre de pas dans le Manuel d'utilisation pour plus de détails.</i>	250 Kpas Un pas correspond à un mot. Consultez la fin du chapitre 10-5 <i>Temps d'exécution d'instruction et nombre de pas dans le Manuel d'utilisation pour plus de détails.</i>	2 Kmots (63,2 Kmots pour -Z)	62 Kmots	15,2 Kmots	
	Capacité de programme	32 Kmots	32 Kmots	6 Kmots	24 Kmots	6 Kmots	
	Mémoire de données maximale	160 mots (2 560 bits)	320 mots (5 120 bits)	40 mots (640 bits)	128 mots (2 048 bits)	32 mots (512 bits)	
	Bits d'E/S	2 644 mots (42 304 bits) + WR : 512 mots (8 192 bits) = 3 156 mots (50 496 bits)	2 644 mots (42 304 bits) + WR : 512 mots (8 192 bits) = 3 156 mots (50 496 bits)	408 mots (6 528 bits)	168 mots (2 688 bits) +400 mots (6 400 bits)	158 mots (2 528 bits)	
	Bits de travail	512 mots (8 192 bits)	512 mots (8 192 bits)	100 mots (1 600 bits)	300 mots (4 800 bits) Max. : 1 400 mots (2 400 bits)	100 mots (1 600 bits)	
	Bits de maintien	32 Kmots x 13 banques	32 Kmots x 13 banques	6 Kmots x 3 banques (6 Kmots x 16 banques pour -Z)	32 Kmots x 8 banques (en option)	6 Kmots	
	Mémoire de données étendue maximale	4 096 chacun	4 096 chacun	Temporisations/ compteurs combinés : 512	1 024 points	Temporisations/ compteurs combinés : 512	
Vitesse de traitement	Nombre max. de temporisations/compteurs	Instructions de base (LD)	CJ1 : 0,08µs min. CJ1-H : 0,02µs min. CJ1M : 0,1µs min.	CS1 : 0,04µs min. CS1-H : 0,02µs min.	0,104µs min.	0,125µs min.	0,375µs min.
	Instructions spéciales (MOV)	CJ1 : 0,25µs min. CJ1-H : 0,18µs min. CJ1M : 0,3µs min.	CS1 : 0,25µs min. CS1-H : 0,18µs min.	0,417µs min.	4,3µs min.	17,7 µs	
	Temps de dépassement du système	CJ1 : 0,5 ms min. CJ1-H : 0,3 ms min. en mode normal, 0,2 ms en mode de traitement parallèle CJ1M : 0,5 ms min.	CS1 : 0,5 ms min. CS1-H : 0,3 ms min. en mode normal, 0,2 ms en mode de traitement parallèle	0,7 ms	0,5 ms	0,7 ms	
	Retard lors de l'édition en ligne (écriture)	CJ1 : Env. 12 ms CJ1-H : Env. 11 ms pour CPU4□ et 8 ms pour CPU6 CJ1M : Env. 14 ms	CS1 : Env. 12 ms CS1-H : Env. 11 ms pour CPU4□ et 8 ms pour CPU6	80 ms (160 ms pour -Z)	500 ms	Généralement 250 ms	

Élément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Structure	Montage avec vis	Non	Oui	Oui	Oui	Non
	Montage sur rail DIN	Oui	Oui	Oui	Non	Oui
	Rack arrière	Non	Oui	Oui	Oui	Non
	Taille (H x P, mm)	90 x 65	130 x 123	130 x 118	250 x 100	110 x 107
Nombre de cartes/racks	Cartes d'E/S	40 cartes	89 cartes (y compris les racks esclaves)	10 ou 16 cartes	64 cartes (8 racks x 8 cartes)	16 cartes
	Cartes réseau	16 cartes	16 cartes	Aucune	16 cartes	Aucune
	Racks d'extension E/S	3 racks	7 racks	3 racks	7 racks	1 rack
Fonction de tâche		Oui	Oui	Non	Non	Non
Mode de traitement de l'UC (exécution du programme et traitement du périphérique)	Mode normal	Oui	Oui	---	---	---
	Mode prioritaire du traitement des périphériques	Oui	Oui	---	---	---
	Traitement parallèle avec accès mémoire synchrone	CJ1 : Non CS1-H : Oui CJ1M : Non	CS1 : Non CS1-H : Oui	Non	Non	Non
	Traitement parallèle avec accès à la mémoire asynchrone	CS1 : Non CJ1-H : Oui CJ1M : Non	CS1 : Non CS1-H : Oui	Non	Non	Non
Format de mise à jour des E/S	Mise à jour cyclique	Oui	Oui	Oui	Oui	Oui
	Mise à jour programmée	Non	Non	Non	Oui	Non
	Mise à jour au zéro de tension	Non	Non	Non	Oui	Non
	Mise à jour immédiate	Oui	Oui	Non	Oui	Non
	Mise à jour immédiate à l'aide de l'instruction IORF	Oui	Oui	Oui	Oui	Oui
Fonction d'horloge		Oui	Oui	Oui	Oui	Oui (cassette Mémoire obligatoire)
Sortie RUN		Oui (en fonction de la carte d'alimentation)	Oui (en fonction de la carte d'alimentation)	Oui (en fonction de la carte d'alimentation)	Oui	Non
Mode de démarrage (paramètre par défaut de la configuration de l'API lorsqu'aucune console de programmation n'est connectée)		Mode RUN	CS1 : mode PROGRAM CS1-H : mode RUN	Mode RUN	Mode RUN	Mode PROGRAM
Désactivation du traitement des interruptions d'alimentation		CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
Fonctionnement sans batterie		CJ1 : Carte mémoire CJ1-H : Carte mémoire ou mémoire flash CJ1M : Carte mémoire ou mémoire flash	CS1 : Carte mémoire CS1-H : Carte mémoire ou mémoire flash	Carte mémoire	Carte mémoire	Cassette Mémoire
Sauvegarde automatique dans la mémoire flash		CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
Poursuite du redémarrage		Non	Non	Non	Oui	Non

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Mémoire externe	Moyen	Carte mémoire (Flash ROM)	Carte mémoire (Flash ROM)	Cassette Mémoire (EEPROM, EPROM)	Carte mémoire (RAM, EEPROM, EPROM)	Cassette Mémoire (ROM, EEPROM, EPROM)
	Capacité	48 Mo	48 Mo	4 à 32 Kmots (4 à 64 Kmots pour -Z)	32 à 512 Kmots (RAM : 64 à 512 Ko, EEPROM : 64 à 128 Ko, EPROM : 0,5 à 1 Mo)	4 à 16 Kmots
	Table des matières	Programmes, mémoire E/S, paramètres	Programmes, mémoire E/S, paramètres	Programmes, mémoire E/S, paramètres	Programmes, mémoire E/S, paramètres	Programmes, DM en lecture seule, paramètres
	Méthode de lecture/écriture	Périphérique de programmation, programme utilisateur (instructions de la mémoire de fichiers) ou Host Link	Périphérique de programmation, programme utilisateur (instructions de la mémoire de fichiers) ou Host Link	Mise à ON du bit SR	Périphérique de programmation, programme utilisateur (instructions de la mémoire de fichiers), Host Link ou graveur de carte mémoire	Mise à ON du bit AR
	Format de fichier	Binaire	Binaire	Binaire	Binaire	Binaire
	Mémoire de données étendue gérée comme des fichiers	Oui (sauf pour les UC CJ1M)	Oui	Non	Non	Non
	Programmes transférés automatiquement au démarrage	Oui	Oui	Oui	Oui	Oui
Carte interne	Non	Carte de communications série	Carte de communications	Non	Carte de communications	
Ports série intégrés	Oui (RS-232C x 1)	Oui (RS-232C x 1)	Oui (RS-232C x 1)	Oui (RS-232C ou RS-422 x 1)	Oui (RS-232C x 1)	

Élément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Communications série	Port périphérique	Bus périphérique	Oui	Oui	Oui	Oui	
		Host Link (SYS-MAC WAY)	Oui	Oui	Oui	Non (Possible avec connexion vers l'interface périphérique)	Oui
		Passerelle série (conversion vers CompoWay/F)	Oui (UC avec version de carte 3.0 ou supérieure)	Oui (UC avec version de carte 3.0 ou supérieure)	Non	Non	Non
		Sans protocole	Non	Non	Oui	Non	Oui
		NT Link	Oui	Oui	Non	Non	Non
	Port RS-232C intégré dans l'UC	Bus périphérique	Oui	Oui	Oui	Non	Non
		Host Link (SYS-MAC WAY)	Oui	Oui	Oui	Oui	Oui
		Passerelle série (conversion vers CompoWay/F)	Oui (UC avec version de carte 3.0 ou supérieure)	Oui (UC avec version de carte 3.0 ou supérieure)	Non	Non	Non
		Sans protocole	Oui	Oui	Oui	Non	Oui
		NT Link	Oui (1:N)	Oui (1:N)	Oui	Non	Oui (1:1)
		Connexions inter-API	Oui (CJ1M uniquement)	Non	Non	Non	Non
	RS-232C ou RS-422/RS-485 sur la carte de communications	Bus périphérique	Non	Non	Oui	Non	Non
		Host Link (SYS-MAC WAY)	Non	Oui Les commandes WG, MP et CR ne sont pas prises en charge.	Oui La commande CR n'est pas prise en charge.	Oui Les commandes WG et MP ne sont pas prises en charge.	Oui La commande CR n'est pas prise en charge.
		Passerelle série (conversion vers CompoWay/F, Modbus-RTU, Modbus-ASCII ou Host Link FINS)	Non	Oui (Unité/carte de communications série avec version de carte 1.2 ou supérieure)	Non	Non	Non
		Sans protocole	Non	Oui (Unité/carte de communications série avec version de carte 1.2 ou supérieure)	Oui	Non	Oui
		NT Link	Non	Oui	Oui	Non	Oui (1:1 et 1:N)
		Macro protocole	Non	Oui	Oui	Non	Oui
Maître CompoWay/F		Non	Oui (avec macro protocole)	Oui (avec macro protocole)	Non	Oui (avec macro protocole)	

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Interruptions	Interruptions d'E/S	Oui (max. 2 cartes d'entrées d'interruption : 32 points, plus 4 points pour les E/S intégrées sur les UC CJ1M) (les UC CJ1 ne prennent pas en charge les interruptions d'E/S.)	Oui (max. 4 ou 2 cartes d'entrées d'interruption : 32 points)	Oui (max. 2 cartes d'entrées d'interruption : 16 points)	Oui (max. 4 cartes d'entrées d'interruption : 32 points)	Oui (4, intégrées à la carte réseau)
	Interruptions programmées	Oui	Oui	Oui	Oui	Oui
	Interruptions de la temporisation à une impulsion	Non	Non	Non	Non	Oui
	Interruptions d'entrées en mode compteur	Oui (UC CJ1M uniquement)	Non	Non	Non	Oui
	Interruptions compteur à grande vitesse	Oui (UC CJ1M uniquement)	Non	Non	Non	Oui
	Interruptions externes	Oui (Les UC CJ1 ne prennent pas en charge les interruptions externes.)	Oui	Non	Non	Non
	A partir de la carte de communications	Non	Oui	Oui	Non	Non
	Interruption de mise sous tension	Non	Non	Non	Oui	Non
	Interruption de mise hors tension	Oui	Oui	Non	Oui	Non
Temps de réponse d'interruption	0,17 ms E/S intégrées sur les UC CJ1M : 0,12 ms	Carte d'E/S spéciales C200H : 1 ms E/S série CJ : 0,1 ms	1 ms	---	Env. 0,1 ms	
Zone Configuration de l'API		Pas d'adresse utilisateur (paramétrage possible uniquement à partir d'un périphérique de programmation, y compris une console de programmation)	Pas d'adresse utilisateur (paramétrage possible uniquement à partir d'un périphérique de programmation, y compris une console de programmation)	Affectation de zone DM fixe : DM 6600 à DM 6655, DM 6550 à DM 6559. Paramétrage possible à partir d'une console de programmation.	Pas d'adresse utilisateur (paramétrage possible uniquement à partir d'un périphérique de programmation, y compris en partie à partir d'une console de programmation)	Affectation de zone DM fixe : DM 6600 à DM 6655. Paramétrage possible à partir d'une console de programmation.

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Paramètres initiaux	E/S	Temps de réponse des entrées des cartes d'E/S	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Non	Non	Défini dans la configuration de l'API
		Premières adresses de rack	Défini dans la table d'E/S à partir d'un périphérique de programmation (mais l'ordre des numéros de rack est fixe).	Défini dans la table d'E/S à partir d'un périphérique de programmation (mais l'ordre des numéros de rack est fixe).	Non	Défini dans la configuration de l'API (possibilité de définir les numéros des racks.)	Non
		Première adresse des cartes d'E/S optiques SYSMAC BUS par maître	Non	Non	Non	Défini dans la configuration de l'API	Non
		Fonctionnement de l'erreur de vérification d'E/S	Non	Non	Non	Défini dans la configuration de l'API	Non
Mémoire		Protection de la mémoire utilisateur	Défini sur l'interrupteur DIP	Défini sur l'interrupteur DIP	Défini sur l'interrupteur DIP	Déterminé par la valeur de l'interrupteur clé	Défini sur l'interrupteur DIP
		Zones avec maintien	Non	Non	Non	Défini dans la configuration de l'API	Non
		Maintien des mots E/S en cas d'erreur fatale (excepté les coupures d'alimentation)	Non	Non	Non	Défini dans la configuration de l'API	Non
		Mémoire sauvegardée grâce au bit de maintien IOM lors de la mise sous tension de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API
		Mémoire sauvegardée grâce au bit de maintien de l'état forcé lors de la mise sous tension de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API
		Surveillance de l'état de l'interrupteur DIP	Oui	Oui	Oui	Non	Oui
Instructions		Paramétrage des données DM indirectes en données BCD ou binaires	Entrée directe possible	Entrée directe possible	Non	Défini dans la configuration de l'API	Non
		Utilisation répétée de l'instruction JMP(0)	Utilisation répétée déjà possible	Utilisation répétée déjà possible	Non	Défini dans la configuration de l'API	Non
		Fonctionnement en cas d'erreurs d'instruction (continue ou s'arrête)	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Non	Non	Non
		Exécution en arrière plan	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
Mémoire de fichier		Transfert automatique au démarrage	Déterminé par le réglage de l'interrupteur DIP (lu automatiquement à partir de la carte mémoire)	Déterminé par le réglage de l'interrupteur DIP (lu automatiquement à partir de la cassette Mémoire)	Déterminé par le réglage de l'interrupteur DIP (lu automatiquement à partir de la carte mémoire)	Déterminé dans la configuration de l'API ou par le réglage de l'interrupteur DIP (lu automatiquement à partir de la carte mémoire)	Déterminé par le réglage de l'interrupteur DIP (lu automatiquement à partir de la carte mémoire)
		Conversion en fichier EM	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Non	Non	Non
Interruptions		Réponse suite à l'interruption	Non	Non	Défini dans la configuration de l'API (C200H/ Réponse rapide)	Non	Non
		Détection d'erreur	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Non	Non
		Maintien des interruptions d'E/S pendant l'exécution du programme d'interruption d'E/S	Non	Non	Non	Défini dans la configuration de l'API	Non
		Activation/désactivation de l'interruption de l'alimentation	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Non	Défini dans la configuration de l'API	Non
		Paramétrage de l'intervalle d'interruption programmée	Défini dans la configuration de l'API (10 ms, 1 ms) (également 0,1 ms pour l'UC CJ1M uniquement)	Défini dans la configuration de l'API (10 ms, 1 ms)	Défini dans la configuration de l'API	Défini dans la configuration de l'API (10 ms, 1 ms, 0,5 ms)	Non

		Élément	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Configurations initiales (suite)	Alimentation	Redémarrage du bit de continuation	Non	Non	Non	Défini dans la configuration de l'API	Non
		Mode de démarrage	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API
		Paramètres des conditions au démarrage	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
		Trace au démarrage	Non	Non	Non	Défini dans la configuration de l'API	Non
		Détection de tension faible de la batterie	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API
		Temps d'interruption d'alimentation momentanée	Non	Non	Non	Défini dans la configuration de l'API	Non
		Temps de retard de la détection de la mise hors tension	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API (Temps pendant lequel le fonctionnement se poursuivra après la détection de la mise hors tension)	Non	Non
		Interruption d'alimentation momentanée considérée comme erreur fatale/non-fatale	Non	Non	Non	Défini dans la configuration de l'API	Non
Cycles	Mise à jour d'E/S	Non	Non	Défini dans la configuration de l'API (cartes d'E/S spéciales uniquement)	Défini dans la configuration de l'API	Non	
	Temps de cycle constant	Défini dans la configuration de l'API (1 à 32 000 ms)	Défini dans la configuration de l'API (1 à 32 000 ms)	Défini dans la configuration de l'API (1 à 9 999 ms)	Défini dans la configuration de l'API (1 à 32 000 ms)	Défini dans la configuration de l'API (1 à 9 999 ms)	
	Contrôle du temps de cycle	Défini dans la configuration de l'API (10 à 40 000 ms) (paramètre initial : 1 000 ms fixes)	Défini dans la configuration de l'API (10 à 40 000 ms) (paramètre initial : 1 000 ms fixes)	Défini dans la configuration de l'API (0 à 99 ms) Carte : 1 s, 10 ms, 100 ms (paramètre initial : 120 ms fixes)	Défini dans la configuration de l'API (10 à 40 000 ms) (paramètre initial : 1 000 ms fixes)	Défini dans la configuration de l'API (0 à 99 ms) Carte : 1 s, 10 ms, 100 ms (paramètre initial : 120 ms fixes)	
	Désactivation de la détection du dépassement de cycle	Non	Non	Défini dans la configuration de l'API	Non	Défini dans la configuration de l'API	
	Exécution asynchrone des instructions et périphérique de service	Non	Non	Non	Défini dans la configuration de l'API	Non	
Communications série	Paramètres de communications du port RS-232C	Réglage de l'interrupteur DIP sur la détection automatique ou la configuration de l'API	Réglage de l'interrupteur DIP sur la détection automatique ou la configuration de l'API	Réglage de l'interrupteur DIP sur les valeurs par défaut ou la configuration de l'API	Réglage de l'interrupteur DIP sur les valeurs par défaut ou la configuration de l'API	Réglage de l'interrupteur DIP sur les valeurs par défaut ou la configuration de l'API	
	Paramètres de communications du port périphérique	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Configuration de l'API	Défini sur l'interrupteur DIP.	Défini dans la configuration de l'API	
	Paramétrage des communications de la carte de communications	Non	Non	Configuration de l'API	Non	Configuration de l'API	
Mode de traitement UC	Modes de traitement parallèle	CJ1 : Non CJ1-H : Oui CJ1M : Non	CS1 : Non CS1-H : Oui	Non	Non	Non	
	Mode prioritaire du traitement des périphériques	Oui	Oui	Non	Non	Non	
Autres périphériques de service	Temps de traitement	Défini dans la configuration de l'API (Temps fixe de traitement des périphériques)	Défini dans la configuration de l'API (Temps fixe de traitement des périphériques)	Défini dans la configuration de l'API (Port RS-232C intégré, carte de communications, port périphérique)	Non	Défini dans la configuration de l'API (Port RS-232C intégré, carte de communications, port périphérique)	
	Mesure de l'intervalle de traitement de la carte réseau	Non	Non	Non	Défini dans la configuration de l'API	Non	
	Arrêt de la mise à jour cyclique des cartes d'E/S spéciales	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Défini dans la configuration de l'API	Non	Non	
	Application de liaison de la carte réseau	Non	Non	Non	Défini dans la configuration de l'API	Non	

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Configurations initiales (suite)	Console de programmation	Défini sur l'interrupteur DIP	CS1 : défini sur l'interrupteur DIP CS1-H : défini à partir de la console de programmation	Défini sur l'interrupteur DIP	Non	Défini sur l'interrupteur DIP	
	Erreurs	Zone du journal d'erreurs	Non (fixe)	Non (fixe)	Non (fixe : DM 6001 à DM6030)	Défini dans la configuration de l'API	Non (fixe : DM 6569 à DM 6599)
		Non-consignation des erreurs FAL dues à l'utilisateur dans le journal d'erreurs.	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
	Fonctionnement	UC mise en attente	Non	Non	Non	Défini dans la configuration de l'API	Non
Zone auxiliaire	Drapeaux de condition	Drapeau ER, CY, <, >, =, Toujours ON/OFF, etc.	Entrée à l'aide des symboles, ex., ER.	Entrée à l'aide des symboles, ex., ER.	Oui	Oui	Oui
		Impulsions de temporisation	Entrée à l'aide des symboles, ex., 0,1 s.	Entrée à l'aide des symboles, ex., 0,1 s.	Oui	Oui	Oui
	Service	Bit de désactivation du traitement de l'UC	Non	Non	Non	Oui	Non
		Codes des périphériques connectés	Non	Non	Non	Oui	Non
		Temps de cycle du traitement des périphériques	Non	Non	Non	Oui	Non
		Intervalle de traitement de la carte réseau	Non	Non	Non	Oui	Non
		Activation/désactivation des périphériques connectés à l'UC	Non	Non	Non	Oui	Non
		Bit de désactivation du service Host Link/NT Link	Non	Non	Non	Oui	Non
		Bit de désactivation du périphérique de service	Non	Non	Non	Oui	Non
		Bit de désactivation de la mise à jour programmée	Non	Non	Non	Oui	Non
		Zone de surveillance universelle de la carte interne	Non	Oui	Oui	Oui	Non
	Dépassement du cycle	Oui	Oui	Oui	Oui	Oui	Oui
	Tâches	Drapeau de première tâche	Oui	Oui	Non (seulement premier drapeau de scannage)	Non (seulement premier drapeau de scannage)	Non (seulement premier drapeau de scannage)
	Débogage	Drapeau d'édition en ligne désactivée	Oui	Oui	Oui (AR)	Non	Non
		Drapeau d'édition en ligne en attente	Oui	Oui	Oui (AR)	Non	Non
		Bit de sortie OFF	Oui	Oui	Oui	Oui	Oui
		Bit de maintien de l'état forcé	Oui	Oui	Oui	Oui	Oui
	Mémoire de fichier	Drapeau d'instruction de la mémoire de fichiers	Oui	Oui	Non	Oui	Non
		Drapeau d'erreur de format de la mémoire du fichier EM	Oui (sauf pour les UC CJ1M)	Oui	Non	Non	Non
		Banque de démarrage du format du fichier EM	Oui (sauf pour les UC CJ1M)	Oui	Non	Non	Non
	Mémoire	Drapeaux de l'état de l'interrupteur DIP	Oui (broche 6)	Oui (broche 6)	Oui (AR, broche 6 uniquement)	Non	Oui (AR, broche 6)
		Bit de maintien IOM	Oui	Oui	Oui	Oui	Oui
	Interruptions	Temps de traitement max. de l'action/sous-programme	Oui	Oui	Oui	Non	Non
		Drapeau d'erreur des tâches d'interruption	Oui	Oui	Oui	Non	Non

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H		
Zone auxiliaire, suite	Erreurs	Zone/pointeur de la sauvegarde du journal d'erreurs	Oui	Oui	Non	Oui	Non	
		Codes d'erreur	Oui	Oui	Oui	Oui	Oui	
	Paramètres initiaux	Initialisation de la configuration de l'API	Non	Non	Oui	Non	Oui	
	Communications	Drapeaux de niveau de fonctionnement de la liaison API	Oui (bit de la zone auxiliaire de liaison API)	Oui (bit de la zone auxiliaire de liaison API)	Oui (AR)	Non	Non	
		Alimentation	Drapeau d'interruption d'alimentation	Non	Non	Non	Oui	Non
			Heure de l'interruption d'alimentation	Non	Non	Non	Oui	Non
			Heure de la mise sous tension	Oui	Oui	Non	Oui	Non
			Heure de l'interruption d'alimentation (y compris la mise hors tension)	Oui	Oui	Non	Oui	Oui
Nombre d'interruptions momentanées d'alimentation	Oui (Nombre d'interruptions d'alimentation)	Oui (Nombre d'interruptions d'alimentation)	Oui (Nombre d'interruptions d'alimentation)	Oui	Oui (Nombre d'interruptions d'alimentation)			
Temps total de l'alimentation sous tension	Oui	Oui	Non	Non	Non			
Méthodes d'affectation	Format		L'affectation est basée sur le nombre de mots nécessaires à la connexion de la carte.	L'affectation est basée sur le nombre de mots nécessaires pour la carte et les emplacements vides sont ignorés.	Affectation de mots fixe : un mot est affecté automatiquement à chaque carte	L'affectation est basée sur le nombre de mots nécessaires pour la carte et les emplacements vides sont passés.	L'affectation est basée sur le nombre de mots nécessaires à la connexion de la carte.	
	Affectation de cartes d'E/S haute densité groupe 2		Aucune	Comme pour les E/S	Zone d'affectation groupe 2 dans la zone IR (emplacement déterminé par l'interrupteur du panneau avant)	Aucune	Aucune	
	Méthode de réservation des mots		Changement de la table d'E/S à partir de CX-Programmer	Changement de la table d'E/S à partir de CX-Programmer	Création de la table d'E/S avec des emplacements vides ou changement de la table d'E/S à partir de CX-Programmer.	Carte d'E/S factice ou changement de la table d'E/S à l'aide de CX-Programmer.	Affectation automatique au démarrage	
	Affectation d'une carte d'E/S spéciales	Zone CIO	Affectation dans une zone de cartes d'E/S spéciales en fonction du nombre de cartes. 10 mots par carte pour un total de 96 cartes.	Affectation dans une zone de cartes d'E/S spéciales en fonction du nombre de cartes. 10 mots par carte pour un total de 96 cartes.	Affectation dans une zone de cartes d'E/S spéciales (zone IR) en fonction du nombre de cartes. 10 mots par carte pour un total de 16 cartes.	Idem que pour les cartes d'E/S spéciales ; 2 ou 4 mots affectés dans une zone d'E/S (différent pour chaque carte)	Idem que pour les cartes d'E/S spéciales ; 1, 2 ou 4 mots affectés dans une zone d'E/S (différent pour chaque carte)	
		Zone DM	Affectation entre D20000 et D29599 en fonction du nombre de cartes, 100 mots par carte pour un total de 96 cartes.	Affectation entre D20000 et D29599 en fonction du nombre de cartes, 100 mots par carte pour un total de 96 cartes.	Affectation entre DM 1000 et DM 1999, et entre DM 2000 et DM 2599, 100 mots par carte pour un total de 16 cartes.	Aucune	Aucune	
	Carte réseau/affectation des cartes réseau	Zone CIO	Affectation dans une zone de cartes réseau en fonction du nombre de cartes. 25 mots par carte pour un total de 16 cartes.	Affectation dans une zone de cartes réseau en fonction du nombre de cartes. 25 mots par carte pour un total de 16 cartes.	Aucune	Affectation dans une zone de cartes réseau en fonction du nombre de cartes. 25 mots par carte pour un total de 16 cartes.	Aucune	
		Zone DM	Affectation entre D30000 et D39599 en fonction du nombre de cartes. 100 mots par carte pour un total de 16 cartes.	Affectation entre D30000 et D39599 en fonction du nombre de cartes. 100 mots par carte pour un total de 16 cartes.	Aucune	Affectation entre D02000 et D03599 en fonction du nombre de cartes. 100 mots par carte pour un total de 16 cartes.	Aucune	

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Mémoire E/S	Zone CIO	Oui	Oui	Oui	Oui	Oui	
	Zone WR	Oui	Oui	Non	Non	Non	
	Zone de relais temporaires	Oui	Oui	Oui	Oui	Oui	
	Zone auxiliaire	Oui	Oui	Oui	Oui	Oui	
	Zone SR	Non	Non	Oui	Non	Oui	
	Zone de liaison	Oui (Zone de liaison de données)	Oui (Zone de liaison de données)	Oui (Zone de liaison de données)	Non	Oui	
	Zone des cartes d'E/S spéciales C200H	Oui	Oui	Oui (Zone CIO)	Non	Non	
	Zone d'E/S intégrées	Oui (UC CJ1M avec E/S intégrées uniquement.)	Non	Non	Non	Non	
	Zone de liaison API série	Oui (UC CJ1M uniquement)	Non	Non	Non	Non	
	Zone DM	Oui	Oui	Oui	Oui	Oui	
	Zone mémoire de données étendue (EM)	Oui (les adresses contenant le numéro de banque peuvent être choisies) (pas de prise en charge pour l'UC CJ1M)	Oui (les adresses contenant le numéro de banque peuvent être choisies)	Oui (les adresses peuvent être choisies pour -Z, mais pas les banques)	Oui (l'adresse contenant la banque ne peut pas être choisie ; la banque doit être modifiée. Carte EM obligatoire.)	Oui (pas de banque)	
	Zone temporisation/compteur	Oui	Oui	Oui	Oui	Oui	
	Registres d'index	Oui	Oui	Non	Oui	Non	
	Registres de données	Oui	Oui	Non	Oui	Non	
	Zones de configuration/réinitialisation forcée	Zone CIO	Oui	Oui	Oui	Oui	Aucune
		Zone WR	Oui	Oui	Non	Non	Oui
		Zone de maintien	Oui	Oui	Oui	Non	Non
		Zone auxiliaire	Non	Non	Oui	Non	Oui
		Zone SR	Non	Non	Non	Non	Non
Zone de liaison		Non	Non	Oui	Non	Non	
Temporisation/compteurs		Oui (Drapeau)	Oui (Drapeau)	Oui (Drapeau)	Oui (Drapeau)	Oui (Drapeau)	
Zone DM	Non	Non	Non	Non	Non		
Zone EM	Non	Non	Non	Non	Non		
Adresses indirectes/variations d'instructions	Différenciation vers le haut (exécutée une seule fois)	Oui (spécifié par @)	Oui (spécifié par @)	Oui (spécifié par @)	Oui (spécifié par ↑)	Oui (spécifié par @)	
	Différenciation vers le bas (exécutée une seule fois)	Oui (spécifié par %)	Oui (spécifié par %)	Non (remplacé par l'instruction DIFD)	Oui (spécifié par ↓)	Non (obtenu avec DIFD)	
	Mise à jour immédiate	Oui (spécifié par !)	Oui (spécifié par !)	Non (remplacé par l'instruction IORF)	Oui (spécifié par !)	Non (obtenu avec IORF)	
	Adresse indirecte de DM/EM	Mode BCD	Oui (0000 à 9999) Utilisation de l'astérisque.	Oui (0000 à 9999) Utilisation de l'astérisque.	Oui (0 à 9999)	Oui (0 à 9999)	Oui (0000 à 9999) Utilisation de l'astérisque.
		Mode binaire	Oui (0000 à 32 767) Utilisation de @. 0000 à 7FFF hex. : de 0000 à 31 767 8000 à FFFF hex. : 00000 à 32 767 dans la banque suivante	Oui (0000 à 32 767) Utilisation de @. 0000 à 7FFF hex. : de 0000 à 31 767 8000 à FFFF hex. : 00000 à 32 767 dans la banque suivante	Non	Oui, mais uniquement pour l'adresse indirecte à l'aide des adresses mémoire de l'API.	Non
Méthodes d'affectation	Configuration du premier mot d'un rack	Oui (pour toutes les UC)	Oui (pour toutes les UC)	Non	Non	Non	
	Configuration du premier mot d'un emplacement	UC pré-ver. 2.0 : voir remarque. UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : voir remarque. UC ver. 2.0 ou supérieure : Oui	Non	Non	Non	

Elément		Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Connexions en ligne par les réseaux sans création de tables d'E/S		Avec affectation automatique d'E/S au démarrage : Oui (pour toutes les UC) Pour le fonctionnement manuel UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Oui, mais pour Controller Link uniquement
Utilisation de différents niveaux de réseaux		UC pré-ver. 2.0 : 3 niveaux UC ver. 2.0 ou supérieure : 8 niveaux	UC pré-ver. 2.0 : 3 niveaux UC ver. 2.0 ou supérieure : 8 niveaux	Non	Oui, pour 3 niveaux	Non
Connexions en ligne aux API via les IHM série NS		UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
Protection des UC contre l'écriture avec des commandes FINS envoyées par réseau		UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
Téléchargement de tâches individuelles		UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
Protection en lecture grâce aux mots de passe	Tout le programme utilisateur	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	Tâches spécifiques	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	Activation/désactivation de la création de fichiers programmes de la mémoire de fichiers	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	Protection du programme en écriture	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
Transfert automatique à la mise sous tension sans fichier de paramètres (.STD)		UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non

**Remarque** Pour les UC fabriquées après le 1 juin 2002 (numéros de lot 020601□□□□ ou postérieur), il est possible de définir jusqu'à huit adresses d'emplacement.

## Comparaison des instructions

	Élément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions d'entrée de séquence	LOAD/AND/OR	LD/AND/OR	Oui	Oui	Oui	Oui	Oui
	AND LOAD/OR LOAD	AND LD/OR LD	Oui	Oui	Oui	Oui	Oui
	NOT	NOT	Oui	Oui	Oui	Oui	Non
	CONDITION ON	UP	Oui	Oui	Non	Oui (*1)	Non
	CONDITION OFF	DOWN	Oui	Oui	Non	Oui (*1)	Non
	BIT TEST	TST/TSTN	Oui (emplacement du bit donné en binaire : 0000 à 000F hex..)	Oui (emplacement du bit donné en binaire : 0000 à 000F hex..)	Oui (emplacement du bit donné en BCD.) (*2)	Oui (emplacement du bit donné en BCD.) (*1)	Non
Instructions de sortie de séquence	OUTPUT	OUT	Oui	Oui	Oui	Oui	Oui
	TR	TR	Oui	Oui	Oui	Oui	Oui
	KEEP	KEEP	Oui	Oui	Oui	Oui	Oui
	DIFFERENTIATE UP/DOWN	DIFU/DIFD	Oui (LD↑, AND↑, OR↑) (LD↓, AND↓, OR↓)	Oui (LD↑, AND↑, OR↑) (LD↓, AND↓, OR↓)	Oui (DIFU/DIFD)	Oui (LD↑, AND↑, OR↑) (LD↓, AND↓, OR↓)	Oui (DIFU/DIFD)
	SET et RESET	SET/RSET	Oui	Oui	Oui	Oui	Oui
	MULTIPLE BIT SET/RESET	SETA/RSTA	Oui (bit de début et nombre de bits spécifiés en binaire.)	Oui (bit de début et nombre de bits spécifiés en binaire.)	Non	(*1) (Bit de début et nombre de bits spécifiés en données BCD.)	Non
	SINGLE BIT SET/RESET	SET/RSTB	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
Instructions de contrôle de séquence	SINGLE BIT OUTPUT	OUTB	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
	END/NO OPERATION	END/NOP	Oui	Oui	Oui	Oui	Oui
	INTERLOCK/INTERLOCK CLEAR	IL/ILC	Oui	Oui	Oui	Oui	Oui
	Plusieurs instructions de verrouillage	MILH/MILR/MILC	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	JUMP/JUMP END	JMP/JME	Oui (numéro du saut donné en BCD : 0 à 1023)	Oui (numéro du saut donné en BCD : 0 à 1023)	Oui (numéro du saut donné en BCD : 0 à 99.)	Oui (numéro du saut donné en BCD : 0 à 999.)	Oui (numéro du saut donné en BCD : 0 à 99.)
	CONDITIONAL JUMP	CJP/CJPN	Oui (numéro du saut donné en BCD : 0 à 1023.)	Oui (numéro du saut donné en BCD : 0 à 1023.)	Non	Oui (numéro du saut donné en BCD : 0 à 999.) (*1)	Non
	MULTIPLE JUMP/JUMP END	JMP0/JME0	Oui	Oui	Non	Non (mais il est possible, dans la configuration de l'API, d'autoriser plusieurs sauts avec le numéro de saut 0)	Non
	FOR/NEXT LOOPS	FOR/NEXT	Oui	Oui	Non	Non	Non
BREAK LOOP	BREAK	Oui	Oui	Non	Non	Non	

Elément		Mnémor- nique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instruc- tions de temporisa- tion et de compteur	TIMER	TIM (BCD)	Oui	Oui	Oui	Oui	Oui
		TIMX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	HIGH-SPEED TIMER	TIMH (BCD)	Oui	Oui	Oui	Oui	Oui
		TIMHX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	ONE-MS TIMER	TMHH (BCD)	Oui	Oui	Non	Non	Non
		TMHHX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	ACCUMULATIVE TIMER	TTIM (BCD)	Oui	Oui	Oui	Oui	Oui
		TTIMX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	LONG TIMER	TIML (BCD)	Oui	Oui	Non	Oui	Non
		TIMLX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	MULTI-OUTPUT TIMER	MTIM (BCD)	Oui	Oui	Non	Oui	Non
		MTIMX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	COUNTER	CNT (BCD)	Oui	Oui	Oui	Oui	Oui
		CNTX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	REVERSIBLE COUNTER	CNTR (BCD)	Oui	Oui	Oui	Oui	Oui
		CNTRX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non
	RESET TIMER/ COUNTER	CNR (BCD)	Oui (seulement réinitialisation de la temporisation ou du compteur.)	Oui (seulement réinitialisation de la temporisation ou du compteur.)	Non	Oui (remet égale- ment à zéro la plage spécifiée dans la zone CIO.)	Non
		CNRX (binaire)	Oui (*4)	Oui (*4)	Non	Non	Non

Élément		Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions de comparaison	Comparaison des symboles	=, <, etc.	Oui (tous pris en charge pour les instructions LD, OR et AND)	Oui (tous pris en charge pour les instructions LD, OR et AND)	Oui (*2) (pris en charge pour AND uniquement)	Oui (*1) (pris en charge pour AND uniquement)	Non
	Instructions de comparaison des données	=DT, <DT, etc.	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	COMPARE/ DOUBLE COMPARE	CMP/ CMPL	Oui	Oui	Oui	Oui (*3)	Oui
	SIGNED BINARY COMPARE/ DOUBLE SIGNED BINARY COMPARE	CPS/ CPSL	Oui	Oui	Oui	Oui (*1)	Oui
	BLOCK COMPARE	BCMP	Oui	Oui	Oui	Oui	Oui
	EXTENDED BLOCK COMPARE	BCMP2	Oui (UC CJ1M et UC CJ1-H ver. 2.0 uniquement)	Oui (UC CJ1-H/CS1D ver. 2.0 uniquement)	Non	Non	Non
	TABLE COMPARE	TCMP	Oui	Oui	Oui	Oui	Oui
	MULTIPLE COMPARE	MCMP	Oui	Oui	Oui	Oui	Oui
	EQUALS	EQU	Non	Non	Non	Oui	Non
AREA RANGE COMPARE	ZCP/ ZCPL	CJ1 : Non (obtenu grâce aux instructions de comparaison) CJ1-H : Oui CJ1M : Oui	CS1 : Non (obtenu grâce aux instructions de comparaison) CS1-H : Oui	Oui	Non	Non (obtenu grâce aux instructions de comparaison)	

	Elément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions de déplacement de données	MOVE	MOV	Oui	Oui	Oui	Oui	Oui
	DOUBLE MOVE (déplacement double)	MOVL	Oui	Oui	Non	Oui	Non
	MOVE NOT	MVN	Oui	Oui	Oui	Oui	Oui
	DOUBLE MOVE (déplacement double)	MVNL	Oui	Oui	Non	Oui	Non
	DATA EXCHANGE	XCHG	Oui	Oui	Oui	Oui	Oui
	DOUBLE DATA EXCHANGE (échange de données doubles)	XCGL	Oui	Oui	Non	Oui	Non
	MOVE QUICK	MOVQ	Non	Non	Non	Oui	Non
	BLOCK TRANSFER	XFER	Oui (nombre de mots à transférer exprimé en binaire : 0 à 65 535.)	Oui (nombre de mots à transférer exprimé en binaire : 0 à 65 535.)	Oui (nombre de mots à transférer exprimé en BCD : 0 à 6144.)	Oui (nombre de mots à transférer exprimé en BCD : 0 à 9999.)	Oui (nombre de mots à transférer exprimé en BCD : 0 à 9999.)
	BLOCK SET	BSET	Oui	Oui	Oui	Oui	Oui
	MOVE BIT	MOVb	Oui (emplacements du bit source et du bit de destination exprimés en binaire)	Oui (emplacements du bit source et du bit de destination exprimés en binaire)	Oui (emplacements du bit source et du bit de destination exprimés en BCD)	Oui (emplacements du bit source et du bit de destination exprimés en BCD)	Oui (emplacements du bit source et du bit de destination exprimés en BCD)
	MULTIPLE BIT TRANSFER	XFRb	Oui	Oui	Oui	Oui (*1)	Oui
	MOVE DIGIT	MOVD	Oui	Oui	Oui	Oui	Oui
	SINGLE WORD DISTRIBUTE	DIST	Oui (la fonction d'empilement est possible avec une autre instruction. Valeur de décalage exprimée en binaire : 0 à 65 535.)	Oui (la fonction d'empilement est possible avec une autre instruction. Valeur de décalage exprimée en binaire : 0 à 65 535.)	Oui (la fonction d'empilement est possible. Valeur de décalage exprimée en BCD : 0 à 8999.)	Oui (la fonction d'empilement est possible avec une autre instruction. Valeur de décalage exprimée en BCD : 0 à 9999.)	Oui (la fonction d'empilement est possible. Valeur de décalage exprimée en BCD : 0 à 8999.)
	DATA COLLECT	COLL	Oui (la fonction d'empilement est possible avec une autre instruction. Valeur de décalage exprimée en binaire : 0 à 65535.)	Oui (la fonction d'empilement est possible avec une autre instruction. Valeur de décalage exprimée en binaire : 0 à 65535.)	Oui (la fonction d'empilement est possible. Valeur de décalage exprimée en BCD : 0 à 7999.)	Oui (la fonction d'empilement est possible avec une autre instruction. Valeur de décalage exprimée en BCD : 0 à 9999.)	Oui (la fonction d'empilement est possible. Valeur de décalage exprimée en BCD : 0 à 7999.)
	EM BLOCK TRANSFER BETWEEN BANKS	BXFR	Non (Possible jusqu'à 65 535 mots en effectuant directement l'adressage de la zone EM avec XFER)	Non (Possible jusqu'à 65 535 mots en effectuant directement l'adressage de la zone EM avec XFER)	Non	Oui (*1)	Non
	EM BLOCK TRANSFER	XFR2	Non	Non	Oui	Non	Non
	EM BANK TRANSFER	BXF2	Non	Non	Oui	Non	Non
	MOVE TO REGISTER (déplacer pour enregistrer)	MOVR	Oui (aucune adresse spécifiée pour DM/EM indirectes)	Oui (aucune adresse spécifiée pour DM/EM indirectes)	Non	Oui (adresse spécifiée pour EM/DM indirectes)	Non
	MOVE TIMER/COUNTER PV TO REGISTER (déplacer la valeur actuelle de temporisation/compteur pour enregistrer)	MOVRW	Oui	Oui	Non	Non (possible pour les drapeaux de fin uniquement avec MOVR)	Non

Élément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Instructions de décalage de données	SHIFT REGISTER	SFT	Oui	Oui	Oui	Oui	Oui
	REVERSIBLE SHIFT REGISTER	SFTR	Oui	Oui	Oui	Oui	Oui
	ASYNCHRONOUS SHIFT REGISTER	ASFT	Oui	Oui	Oui	Oui	Oui
	WORD SHIFT	WSFT	Oui (idem que CV : 3 opérandes)	Oui (idem que CV : 3 opérandes)	Oui	Oui	Oui
	ARITHMETIC SHIFT LEFT/ ARITHMETIC SHIFT RIGHT	ASL/ASR	Oui	Oui	Oui	Oui	Oui
	ROTATE LEFT/ ROTATE RIGHT	ROL/ROR	Oui	Oui	Oui	Oui	Oui
	ONE DIGIT SHIFT LEFT/ONE DIGIT SHIFT RIGHT	SLD/SRD	Oui	Oui	Oui	Oui	Oui
	SHIFT N-BIT DATA LEFT/SHIFT N-BIT DATA RIGHT	NSFR/ NSFL	Oui (données de décalage et bit de début spécifiés en binaire.)	Oui (données de décalage et bit de début spécifiés en binaire.)	Non	Oui (données de décalage et bit de début spécifiés en BCD.) (*1)	Non
	SHIFT N-BITS LEFT/SHIFT N-BITS RIGHT/DOUBLE SHIFT N-BITS LEFT/DOUBLE SHIFT NITS RIGHT	NASL/ NASR, NSLL/ NSRL	Oui (nombre de bits à décaler exprimé en binaire.)	Oui (nombre de bits à décaler exprimé en binaire.)	Non	Oui (nombre de bits à décaler exprimé en BCD.) (*1)	Non
	DOUBLE SHIFT LEFT/DOUBLE SHIFT RIGHT	ASLL/ ASRL	Oui	Oui	Non	Oui	Non
	DOUBLE ROTATE LEFT/DOUBLE ROTATE RIGHT	ROLL/ RORL	Oui	Oui	Non	Oui	Non
	ROTATE LEFT WITHOUT CARRY/ ROTATE RIGHT WITHOUT CARRY/ DOUBLE ROTATE LEFT WITHOUT CARRY/DOUBLE ROTATE RIGHT WITHOUT CARRY	RLNC/ RRNC, RLNL/ RRNL	Oui	Oui	Non	Oui (*1)	Non
Instructions d'incrément et de décrémentation	INCREMENT BCD/ DECREMENT BCD	++B/--B (INC/ DEC)	Oui (++B/--B)	Oui (++B/--B)	Oui (INC/DEC)	Oui (INC/DEC)	Oui (INC/DEC)
	DOUBLE INCREMENT BCD/DOUBLE DECREMENT BCD	++BL/--BL (INCL/ DECL)	Oui (++BL/--BL)	Oui (++BL/--BL)	Non	Oui (INCL/DECL)	Non
	INCREMENT BINARY/ DECREMENT BINARY	++/-- (INCB/ DECB)	Oui (CY passe à ON pour les retenues et les emprunts) (++)/--)	Oui (CY passe à ON pour les retenues et les emprunts) (++)/--)	Non	Oui	Non
	DOUBLE INCREMENT BINARY/ DOUBLE DECREMENT BINARY	++L/--L (INBL/ DCBL)	Oui (CY passe à ON pour les retenues et les emprunts) (++)L/--L)	Oui (CY passe à ON pour les retenues et les emprunts) (++)L/--L)	Non	Oui	Non
Instructions mathématiques			Oui	Oui	Oui	Oui	Oui

Élément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Instructions de conversion	BCD-TO-BINARY/ DOUBLE BCD-TO- DOUBLE BINARY	BIN/BINL	Oui	Oui	Oui	Oui	Oui
	BINARY-TO-BCD/ DOUBLE BINARY- TO-DOUBLE BCD	BCD/ BCDL	Oui	Oui	Oui	Oui	Oui
	2'S COMPLE- MENT/ DOUBLE 2'S COMPLEMENT	NEG/ NEGL	Oui (idem que CV mais UP ne passe pas à ON pour 8000 hex. à la source)	Oui (idem que CV mais UP ne passe pas à ON pour 8000 hex. à la source)	Oui	Oui	Oui
	16-BIT TO 32-BIT SIGNED BINARY	SIGN	Oui	Oui	Non	Oui	Non
	DATA DECODER	MLPX	Oui	Oui	Oui	Oui	Oui
	DATA ENCODER	DMPX	Oui (idem que CVM1-V2 : Peut spécifier le bit le plus à droite sur ON.)	Oui (idem que CVM1-V2 : Peut spécifier le bit le plus à droite sur ON.)	Oui (bit le plus à gauche sur ON uniquement.)	Oui (CVM1-V2 : Peut spécifier le bit le plus à droite sur ON.)	Oui (bit le plus à gauche sur ON uniquement.)
	ASCII CONVERT	ASC	Oui	Oui	Oui	Oui	Oui
	ASCII TO HEX	HEX	Oui	Oui	Oui	Oui (*1)	Oui
	COLUMN TO LINE/ LINE TO COLUMN	LINE/ COLM	Oui (emplace- ment du bit spéci- fié en binaire.)	Oui (emplace- ment du bit spéci- fié en binaire.)	Oui (emplace- ment du bit donné en BCD)	Oui (emplace- ment du bit donné en BCD)	Oui (emplace- ment du bit donné en BCD)
	SIGNED BCD-TO- BINARY/DOUBLE SIGNED BCD-TO- BINARY	BINS/ BISL	Oui	Oui	Non	Oui (*1)	Non
	SIGNED BINARY- TO-BCD/DOUBLE SIGNED BINARY- TO-BCD	BCDS/ BDSL	Oui	Oui	Non	Oui (*1)	Non
GRAY CODE CONVERSION	GRY	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui (Egalement pris en charge pour le numéro de lot 030201 ou sui- vants)	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui (Egalement pris en charge pour le numéro de lot 030201 ou sui- vants)	Non	Non	Non	
Instructions logi- ques	LOGICAL AND/ LOGICAL OR/ EXCLUSIVE OR/ EXCLUSIVE NOR	ANDW, ORW, XORW, XNRW	Oui	Oui	Oui	Oui	Oui
	DOUBLE LOGICAL AND/DOUBLE LOGICAL OR/DOU- BLE EXCLUSIVE OR/DOUBLE EXCLUSIVE NOR	ANDL, ORWL, XORL, XNRL	Oui	Oui	Non	Oui	Non
	COMPLEMENT/ DOUBLE COMPLEMENT	COM/ COML	Oui	Oui	Oui (COM uni- quement)	Oui	Oui (COM uni- quement)
Instruc- tions mathéma- tiques spéciales	BCD SQUARE ROOT	ROOT	Oui	Oui	Oui	Oui	Oui
	BINARY ROOT	ROTB	Oui	Oui	Non	Oui (*1)	Non
	ARITHMETIC PROCESS	APR	Oui	Oui	Oui	Oui	Oui
	FLOATING POINT DIVIDE	FDIV	Oui	Oui	Oui	Oui	Non
	BIT COUNTER	BCNT	Oui (nombre de mots à compter et résultats du comptage expr- més en binaire : 0 à FFFF hex.)	Oui (nombre de mots à compter et résultats du comptage expr- més en binaire : 0 à FFFF hex.)	Oui (nombre de mots à compter et résultats du comptage expr- més en BCD : 1 à 6656)	Oui (nombre de mots à compter et résultats du comptage expr- més en BCD : 0 à 9999, mais erreur pour 0)	Oui (nombre de mots à compter et résultats du comptage expr- més en BCD : 1 à 6656)

Élément		Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions mathématiques à virgule flottante	FLOATING TO 16-BIT/32-BIT BIN, 16-BIT/32-BIT BIN TO FLOATING	FIX/FIXL, FLT/FTL	Oui	Oui	Non	Oui (*1)	Oui
	FLOATING-POINT ADD/FLOATING-POINT SUBTRACT/FLOATING-POINT MULTIPLY/FLOATING-POINT DIVIDE	+F, -F, *F, /F	Oui	Oui	Non	Oui (*1)	Oui
	DEGREES TO RADIANS/RADIANS TO DEGREES	RAD, DEG	Oui	Oui	Non	Oui (*1)	Oui
	SINE/COSINE/TANGENT/ARC SINE/ARC TANGENT	SIN, COS, TAN, ASIN, ACOS, ATAN	Oui	Oui	Non	Oui (*1)	Oui
	SQUARE ROOT	SQRT	Oui	Oui	Non	Oui (*1)	Oui
	EXPONENT	EXP	Oui	Oui	Non	Oui (*1)	Oui
	LOGARITHM	LOG	Oui	Oui	Non	Oui (*1)	Oui
	EXPONENTIAL POWER	PWR	Oui	Oui	Non	Non	Non
	Comparaison à virgule décimale flottante	Exemples : =F, <>F	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
	A virgule décimale flottante en chaîne de texte	FSTR, FVAL	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
Instructions de calcul et de conversion à virgule flottante à double précision	Idem que les instructions de calcul et de conversion à virgule flottante à simple précision, ci-dessus	Exemple : FIXD	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non

	Élément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions de traitement de données de tableaux	SET STACK	SSET	Oui (quatre mots des informations sur le contrôle des piles. Nombre de mots spécifié en binaire : 5 à 65535)	Oui (quatre mots des informations sur le contrôle des piles. Nombre de mots spécifié en binaire : 5 à 65535)	Non	Oui (quatre mots des informations sur le contrôle des piles. Nombre de mots spécifié en BCD : 3 à 9999)	Non
	PUSH ONTO STACK:	PUSH	Oui	Oui	Non	Oui	Non
	FIRST IN FIRST OUT	FIFO	Oui	Oui	Non	Oui	Non
	LAST IN FIRST OUT	LIFO	Oui	Oui	Non	Oui	Non
	FIND MAXIMUM/ FIND MINIMUM	MAX, MIN	Oui (deux mots dans le champ des données de contrôle. Longueur de la table exprimée en binaire : 1 à FFFF)	Oui (deux mots dans le champ des données de contrôle. Longueur de la table exprimée en binaire : 1 à FFFF)	Oui (un mot dans le champ des données de contrôle. Longueur de la table exprimée en BCD : 1 à 999)	Oui (un mot dans le champ des données de contrôle. Longueur de la table exprimée en BCD : 1 à 999)	Oui (un mot dans le champ des données de contrôle. Longueur de la table exprimée en BCD : 1 à 999)
	DATA SEARCH	SRCH	Oui (longueur de la table exprimée en binaire : 1 à FFFF. Adresse mémoire de l'API envoyée vers IR0. Le nombre de correspondances peut être envoyé vers DR0)	Oui (longueur de la table exprimée en binaire : 1 à FFFF. Adresse mémoire de l'API envoyée vers IR0. Le nombre de correspondances peut être envoyé vers DR0)	Oui (longueur de la table exprimée en BCD : 1 à 6556. Adresse mémoire de l'API envoyée vers C+1. Le nombre de correspondances ne peut être envoyé vers DR0)	Oui (longueur de la table exprimée en BCD : 1 à 9999. Adresse mémoire de l'API envoyée vers IR0. Le nombre de correspondances ne peut être envoyé vers DR0)	Oui (longueur de la table exprimée en BCD : 1 à 6556. Adresse mémoire de l'API envoyée vers C+1. Le nombre de correspondances ne peut être envoyé vers DR0)
	FRAME CHECKSUM	FCS	Oui	Oui	Oui	Non	Oui
	SUM	SUM	Oui (idem que C200HX/HG/HE : Somme possible pour les octets et les mots.)	Oui (idem que C200HX/HG/HE : Somme possible pour les octets et les mots.)	Oui (somme possible pour les octets et les mots.)	Oui (somme possible pour les mots uniquement.)	Oui (somme possible pour les octets et les mots.)
	SWAP BYTES	SWAP	Oui (peut être utilisé pour les communications de données et les autres applications.)	Oui (peut être utilisé pour les communications de données et les autres applications.)	Non	Non	Non
	DIMENSION RECORD TABLE:	DIM	Oui	Oui	Non	Non	Non
	SET RECORD LOCATION	SETR	Oui	Oui	Non	Non	Non
	GET RECORD LOCATION	GETR	Oui	Oui	Non	Non	Non

	Élément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions de contrôle de données	SCALING	SCL	Oui	Oui	Oui	Non	Oui
	SCALING 2	SCL2	Oui	Oui	Non	Non	Oui
	SCALING 3	SCL3	Oui	Oui	Non	Non	Oui
	PID CONTROL	PID	Oui (La sortie peut passer de 0% à 50% lorsque PV = SV. PID et période d'échantillonnage spécifiés en binaire.)	Oui (La sortie peut passer de 0% à 50% lorsque PV = SV. PID et période d'échantillonnage spécifiés en binaire.)	Oui (PID et période d'échantillonnage spécifiés en BCD.)	Oui (PID et période d'échantillonnage spécifiés en BCD) (*1)	Oui (PID et période d'échantillonnage spécifiés en BCD.)
	PID CONTROL WITH AUTOTUNIG	PIDAT	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
	LIMIT CONTROL	LMT	Oui	Oui	Non	Oui (*1)	Non
	DEAD BAND CONTROL	BAND	Oui	Oui	Non	Oui (*1)	Non
	DEAD ZONE CONTROL	ZONE	Oui	Oui	Non	Oui (*1)	Non
	TIME-PROPORTIONAL OUTPUT	TPO	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
AVERAGE	AVG	Oui (nombre d'analyses spécifié en binaire)	Oui (nombre d'analyses spécifié en binaire)	Oui (nombre d'analyses spécifié en BCD)	Non	Oui (nombre d'analyses spécifié en BCD)	
Instructions de sous-programme	SUBROUTINE CALL/SUBROUTINE ENTRY/SUBROUTINE RETURN	SBS, SBN, RET	Oui (numéro du sous-programme donné en BCD : 0 à 1023)	Oui (numéro du sous-programme donné en BCD : 0 à 1023)	Oui (numéro du sous-programme donné en BCD : 0 à 255)	Oui (numéro du sous-programme donné en BCD : 0 à 999)	Oui (numéro du sous-programme donné en BCD : 0 à 255)
	MACRO	MCRO	Oui (numéro du sous-programme donné en BCD : 0 à 1023)	Oui (numéro du sous-programme donné en BCD : 0 à 1023)	Oui (numéro du sous-programme donné en BCD : 0 à 255)	Oui (numéro du sous-programme donné en BCD : 0 à 999) (*1)	Oui (numéro du sous-programme donné en BCD : 0 à 255)
	Instructions de sous-programmes globaux	GSBS, GSBN, RET	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non
Instructions de contrôle de l'interruption	SET INTERRUPT MASK	MSKS	Oui	Oui	Non (traitement de toutes les interruptions réalisés avec INT)	Oui	Non (traitement de toutes les interruptions réalisés avec INT)
	CLEAR INTERRUPT	CLI	Oui	Oui	Non	Oui	Non
	READ INTERRUPT MASK:	MSKR	Oui	Oui	Non	Oui	Non
	DISABLE INTERRUPTS	DI	Oui	Oui	Non	Non	Non
	ENABLE INTERRUPTS	EI	Oui	Oui	Non	Non	Non
	ENABLE TIMER	STIM	Non	Non	Non	Non	Oui

Elément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Instructions du compteur à grande vitesse et de la sortie d'impulsions	MODE CONTROL	INI	Oui (*5)	Non	Non	Non	Oui
	PRESENT VALUE READ	PRV	Oui (*5)	Non	Non	Non	Oui
	COUNTER FREQUENCY CONVERT	PRV2	UC CJ1M ver. 2.0 ou supérieure : Oui (*5) CJ1-H (toutes les UC) : Non	Non	Non	Non	Non
	SET COMPARISON TABLE	CTBL	Oui (*5)	Non	Non	Non	Oui
	SET PULSES	PULS	Oui (*5)	Non	Non	Non	Oui
	SET FREQUENCY	SPED	Oui (*5)	Non	Non	Non	Oui
	ACCELERATION CONTROL	ACC	Oui (*5)	Non	Non	Non	Oui
	POSITION CONTROL	PLS2	Oui (*5)	Non	Non	Non	Oui
	ORIGIN SEARCH	ORG	Oui (*5)	Non	Non	Non	Non
PWM OUTPUT	PWM	Oui (*5)	Non	Non	Non	Oui	
Instructions de pas	STEP DEFINE et STEP START	STEP/SNXT	Oui	Oui	Oui	Oui	Oui
Instructions des cartes d'E/S	I/O REFRESH	IORF	Oui	Oui (utilisé pour les cartes d'E/S haute densité groupe 2 ainsi que pour les cartes d'E/S spéciales. Comprend les fonctionnalités de GROUP-2 HIGH-DENSITY I/O REFRESH (MPRF))	Oui (utilisé pour les cartes d'E/S haute densité groupe 2 C200H ainsi que pour les cartes d'E/S spéciales.	Oui	Oui
	7-SEGMENT DECODER	SDEC	Oui	Oui	Oui	Oui	Oui
	DIGITAL SWITCH INPUT	DSW	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Oui	Non	Oui
	TEN KEY INPUT	TKY	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Oui	Non	Oui
	HEXADECIMAL KEY INPUT	HKY	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Oui	Non	Oui
	MATRIX INPUT	MTR	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Oui	Non	Non
	7-SEGMENT DISPLAY OUTPUT	7SEG	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Oui	Non	Oui
	GROUP-2 HIGH-DENSITY I/O REFRESH	MPRF	Non	Non	Oui	Non	Non
	TEN KEY INPUT	TKY	Non	Non	Oui	Non	Oui
	HEXADECIMAL KEY INPUT	HKY	Non	Non	Oui	Non	Oui
	DIGITAL SWITCH INPUT	DSW	Non	Non	Oui	Non	Oui
	MATRIX INPUT	MTR	Non	Non	Oui	Non	Non
	7-SEGMENT DISPLAY OUTPUT	7SEG	Non	Non	Oui	Non	Oui

Élément		Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions des cartes d'E/S spéciales	SPECIAL I/O UNIT READ et SPECIAL I/O UNIT WRITE (I/O READ et I/O WRITE)	IORD/IOWR (READ/WRIT)	IORD/IOWR (jusqu'à 96 cartes. Ne sera plus utilisé pour envoyer des commandes FINS.	IORD/IOWR (jusqu'à 96 cartes. Ne sera plus utilisé pour envoyer des commandes FINS.	IORD/IOWR	READ/WRIT	Non
	I/O READ 2 et I/O WRITE 2	RD2/WR2	Non	Non	Non	Oui (*1)	Non
Instructions de traitement des chaînes de caractères	MOV STRING	MOV\$	Oui	Oui	Non	Non	Non
	CONCATENATE STRING	+\$	Oui	Oui	Non	Non	Non
	GET STRING LEFT	LEFT\$	Oui	Oui	Non	Non	Non
	GET STRING RIGHT	RGHT\$	Oui	Oui	Non	Non	Non
	GET STRING MIDDLE	MID\$	Oui	Oui	Non	Non	Non
	FIND IN STRING	FIND\$	Oui	Oui	Non	Non	Non
	STRING LENGTH	LEN\$	Oui	Oui	Non	Non	Non
	REPLACE IN STRING	RPLC\$	Oui	Oui	Non	Non	Non
	DELETE STRING	DEL\$	Oui	Oui	Non	Non	Non
	EXCHANGE STRING	XCHG\$	Oui	Oui	Non	Non	Non
	CLEAR STRING :	CLR\$	Oui	Oui	Non	Non	Non
	INSERT INTO STRING	INS\$	Oui	Oui	Non	Non	Non

Élément		Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions de communications série	RECEIVE	RXD	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour le port RS-232C de l'UC. Ne peut être utilisé pour la carte de communications série, ni pour le port périphérique de l'UC)	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour le port RS-232C de l'UC. Ne peut être utilisé pour la carte interne, la carte de communications série, ni pour le port périphérique de l'UC)	Oui (nombre d'octets stockés spécifié en BCD) (Utilisé pour le port périphérique, le port RS-232C ou la carte de communications de l'UC.)	Non	Oui (nombre d'octets stockés spécifié en BCD) (Utilisé pour le port périphérique, le port RS-232C ou la carte de communications de l'UC.)
	RECEIVE VIA SERIAL COMMUNICATIONS UNIT	RXDU	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour la carte de communications série avec la version de carte 1.2 ou supérieure.)	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour la carte de communications série avec la version de carte 1.2 ou supérieure.)	Non	Non	Non
	TRANSMIT	TXD	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour le port RS-232C de l'UC. Ne peut être utilisé pour la carte de communications série, ni pour le port périphérique de l'UC) (Les communications non demandées ne sont pas possibles grâce à la commande Host Link EX)	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour le port RS-232C de l'UC. Ne peut être utilisé pour la carte interne, la carte de communications série, ni pour le port périphérique de l'UC) (Les communications non demandées ne sont pas possibles grâce à la commande Host Link EX)	Oui (nombre d'octets stockés spécifié en BCD) (Utilisé pour le port périphérique, le port RS-232C ou la carte de communications de l'UC.) (Communications non demandées impossibles grâce à la commande Host Link EX)	Non	Oui (nombre d'octets stockés spécifié en BCD) (Utilisé pour le port périphérique, le port RS-232C ou la carte de communications de l'UC.) (Communications non demandées impossibles grâce à la commande Host Link EX)
	TRANSMIT VIA SERIAL COMMUNICATIONS UNIT	TXDU	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour la carte de communications série avec la version de carte 1.2 ou supérieure.)	Oui (nombre d'octets stockés spécifié en binaire) (Utilisé uniquement pour la carte de communications série avec la version de carte 1.2 ou supérieure.)	Non	Non	Non
	CHANGE SERIAL PORT SETUP	STUP	Oui (10 mots définis) Peut servir pour la carte de communications série.	Oui (10 mots définis) Peut servir pour la carte de communications série.	Oui (5 mots définis)	Non	Oui (5 mots définis)
	PROTOCOL MACRO	PMCR	Oui (numéro de séquence spécifié en binaire. Quatre opérandes. Peut spécifier l'adresse de la carte de destination et le numéro du port série.)	Oui (numéro de séquence spécifié en binaire. Quatre opérandes. Peut spécifier l'adresse de la carte de destination et le numéro du port série.)	Oui (numéro de séquence spécifié en BCD. Trois opérandes.)	Non	Oui (numéro de séquence spécifié en BCD. Trois opérandes.)
	PCMCIA CARD MACRO	CMCR	Non	Non	Oui	Non	Non

Élément	Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H	
Instructions réseaux	NETWORK SEND/ NETWORK RECEIVE	SEND/ RECV	Oui (peut être utilisé pour l'ordinateur hôte via les connexions Host Link. Ne peut être utilisé pour la carte de communications série, ni pour le port RS-232C de l'UC)	Oui (peut être utilisé pour l'ordinateur hôte via les connexions Host Link. Ne peut être utilisé pour la carte de communications série, ni pour le port RS-232C de l'UC, ni pour la carte interne.)	Oui (peut être utilisé pour l'ordinateur hôte via les connexions Host Link.)	Oui (peut être utilisé pour l'ordinateur hôte via les connexions Host Link.)	
	DELIVER COMMAND	CMND	Oui (peut être utilisé pour l'ordinateur hôte via les connexions Host Link. Ne peut être utilisé pour la carte de communications série, ni pour le port RS-232C de l'UC)	Oui (peut être utilisé pour l'ordinateur hôte via les connexions Host Link. Ne peut être utilisé pour la carte de communications série, ni pour le port RS-232C de l'UC, ni pour la carte interne.)	Non	Oui (peut être utilisé pour l'ordinateur hôte via les connexions Host Link.)	
	EXPLICIT MESSAGE SEND	EXPLT	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	EXPLICIT GET ATTRIBUTE	EGATR	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	EXPLICIT SET ATTRIBUTE	ESATR	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	EXPLICIT WORD READ	ECHRD	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	EXPLICIT WORD WRITE	ECHWR	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	UC pré-ver. 2.0 : Non UC ver. 2.0 ou supérieure : Oui	Non	Non	Non
	Instructions de mémoire de fichiers	READ DATA FILE/ WRITE DATA FILE	FREAD/ FWRIT	Oui	Oui	Non	Oui (FILR/FILW)
READ PROGRAM FILE		FILP	Non	Non	Non	Oui	Non
CHANGE STEP PROGRAM		FLSP	Non	Non	Non	Oui	Non
Instructions d'affichage	DISPLAY MESSAGE	MSG	Oui (messages se terminant par NUL)	Oui (messages se terminant par NUL)	Oui (messages se terminant par CR)	Oui (messages se terminant par CR)	Oui (messages se terminant par CR)
	DISPLAY LONG MESSAGE	LMSG	Non	Non	Oui (messages se terminant par CR)	Non	Non
	I/O DISPLAY	IODP	Non	Non	Non	Oui	Non
	TERMINAL MODE	TERM	Non	Non	Oui	Non	Non
Instructions de temporisation	CALENDAR ADD	CADD	Oui	Oui	Non	Oui	Non
	CALENDAR SUBTRACT	CSUB	Oui	Oui	Non	Oui	Non
	HOURS TO SECONDS	SEC	Oui	Oui	Oui	Oui	Oui
	SECONDS TO HOURS	HMS	Oui	Oui	Oui	Oui	Oui
	CLOCK ADJUSTMENT	DATE	Oui	Oui	Non	Oui (*1)	Non
Instructions de débogage	TRACE MEMORY SAMPLING	TRSM	Oui	Oui	Oui	Oui	Oui
	MARK TRACE	MARK	Non	Non	Non	Oui (numéro du marqueur donné en BCD)	Non

Elément		Mnémonique	Série CJ	Série CS	C200HX/HG/HE	Série CVM1/CV	CQM1H
Instructions de diagnostic d'erreur	FAILURE ALARM/ SEVERE FAILURE ALARM	FAL/FALS	Oui (messages se terminant par NUL, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en binaire.)	Oui (messages se terminant par NUL, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en binaire.)	Oui (messages se terminant par CR, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en BCD.)	Oui (messages se terminant par CR, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en BCD.) (*1)	Oui (messages se terminant par CR, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en BCD.)
	FAILURE POINT DETECTION	FPD	Oui (messages se terminant par NUL, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en binaire.)	Oui (messages se terminant par NUL, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en binaire.)	Oui (messages se terminant par CR, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en BCD.)	Oui (messages se terminant par CR, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en BCD.) (*1)	Oui (messages se terminant par CR, chaînes de texte sauvegardées dans l'ordre suivant : de l'octet le plus à gauche à l'octet le plus à droite, puis du mot le plus à droite au mot le plus à gauche. Numéro FAL spécifié en BCD.)
Autres instructions	SET CARRY/ CLEAR CARRY	STC/CLC	Oui	Oui	Oui	Oui	Oui
	LOAD FLAGS/SAVE FLAGS	CCL, CCS	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Oui	Non
	EXTEND MAXI- MUM CYCLE TIME	WDT	Oui	Oui	Oui	Oui (*1)	Oui
	CYCLE TIME	SCAN	Non	Non	Oui	Non	Non
	LOAD REGISTER/ SAVE REGISTER	REGL, REGS	Non	Non	Non	Oui	Non
	SELECT EM BANK :	EMBC	Oui	Oui	Oui	Oui	Non
	EXPANSION DM READ	XDMR	Non	Non	Oui	Non	Non
	INDIRECT EM ADDRESSING	IEMS	Non	Non	Oui	Non	Non
	ENABLE ACCESS/ DISABLE ACCESS	IOSP, IORS	Non	CS1 : Non CS1-H : Oui	Non	Oui	Non
Instructions de conversion des adresses CV-CS	FRMCV TOCV	CJ1 : Non CJ1-H : Oui CJ1M : Oui	CS1 : Non CS1-H : Oui	Non	Non	Non	
Instructions de programmation de bloc	BPRG/ BEND, IF/ ELSE/ IEND, WAIT, EXIT, LOOP/ LEND, BPPS/ BPRS, TIMW, CNTW, TMHW	Oui	Oui	Non	Oui (*1)	Non	
Instructions de contrôle des tâches	TASK ON/TASK OFF	TKON/ TKOF	Oui	Oui	Non	Non	Non

**Remarque** \*1 : Pris en charge par CVM1 (V2) uniquement.

\*2 : Pris en charge par les modèles d'UC □□-Z uniquement.

\*3 : Poursuite sur la même exploitation de programme prise en charge par CV1M version 2,

\*4 : Excepté pour les UC CSA et CJ1.

\*5 : UC CJ1M avec E/S intégrées uniquement. Certaines opérantes sont différentes de celles utilisées par CQM1H.



# Annexe B

## Modifications par rapport aux systèmes Host

### Link précédents

Les systèmes Host Link créés à l'aide des cartes de communications série, série CS/CJ (série CS uniquement) ne sont pas identiques aux systèmes Host Link créés à l'aide des cartes Host Link et des UC des autres séries d'API. Ce chapitre décrit ces différences.

### Ports RS-232C

N'oubliez pas de tenir compte des différences suivantes lorsque vous passez d'un système Host Link existant à un système utilisant un port RS-232C sur une UC série CS/CJ ou des cartes de communications série (série CS uniquement) (CS1H/G-CPU□□ port RS-232C, ports CS1W-SCU21, ports CS1W-SCB21, CS1W-SCB41 port 1 ou CJ1W-SCU41 port 2).

Anciens produits	Numéro de modèle	Modifications nécessaires pour le produit de la série CS/CJ	
		Câblage	Autres
Cartes Host Link série C	3G2A5-LK201-E C500-LK203 3G2A6-LK201-E	Le connecteur est passé d'un connecteur à 25 broches à un connecteur à 9 broches.  Les produits de la série CS/CJ ne prennent pas en charge les signaux ST1, ST2 et RT ; il n'est donc pas nécessaire de les câbler.	<b>Il est nécessaire d'apporter les modifications suivantes pour les systèmes synchronisés avec ST1, ST2 et RT.</b>  Les transferts synchronisés ne sont plus possibles.  Les transmissions bidirectionnelles simultanées seront possibles avec le produit de la série CS/CJ mais il faudra changer soit le programme de communication de l'ordinateur hôte, soit le matériel, soit les deux.  <b>Il est nécessaire d'apporter les modifications suivantes pour les systèmes qui ne sont pas synchronisés avec ST1, ST2 et RT.</b>  Il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres longueurs de texte dans les trames ou d'autres caractéristiques de commandes CS/CJ. (voir remarque)
	C200H-LK201	Le connecteur est passé d'un connecteur à 25 broches à un connecteur à 9 broches.	Il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres longueurs de texte dans les trames ou d'autres caractéristiques de commandes CS/CJ. (voir remarque)
UC série C	SRM1 CPM1 CPM1A CQM1-CPU□□-E C200HS-CPU□□-E C200HX/HG/HE-CPU□□-E C200HW-COM□□-E	Aucun changement au niveau du câblage.	Il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres caractéristiques de commandes CS/CJ.

Anciens produits	Numéro de modèle	Modifications nécessaires pour le produit de la série CS/CJ	
		Câblage	Autres
UC série CVM1 ou CV	CVM1/CV-CPU□□-E	Aucun changement au niveau du câblage.	Il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres caractéristiques de commandes CS/CJ.
Carte Host Link série CVM1 ou CV	CV500-LK201	Port 1 : Le connecteur est passé d'un connecteur à 25 broches à un connecteur à 9 broches. Port 2 défini pour RS-232C : Le signal SG est passé de 7 à 9 broches.	<b>Les modifications suivantes doivent être apportées pour les transmissions bidirectionnelles à l'alternat qui utilisent CD.</b> Vérifiez l'absence de problèmes de temporisation dans le système lors de l'utilisation de SEND, RECV ou CMND pour lancer des communications à partir de l'API ou lors de l'envoi de commandes à partir de l'ordinateur hôte. Si nécessaire, passez aux transmissions bidirectionnelles simultanées. <b>Les modifications suivantes doivent être apportées pour les transmissions bidirectionnelles simultanées qui n'utilisent pas CD.</b> Bidirectionnel à l'alternat : il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres caractéristiques de commandes CS/CJ.

**Remarque** Le nombre de mots pouvant être lus et écrits par trame (c-à-d, les longueurs de texte) avec les commandes en mode C est différent pour les cartes Host Link série C et les cartes de communications série de la série CS/CJ. Il se peut qu'un programme installé sur l'ordinateur hôte précédemment utilisé pour les cartes Host Link série C ne fonctionnent pas correctement si vous les utilisez sur des API série CS/CJ. Vérifiez le programme de l'ordinateur hôte avant de l'utiliser et apportez les corrections nécessaires pour pouvoir gérer différentes longueurs de texte du cadre. Consultez le *Manuel de référence des commandes de communications, série CS/CJ (W342)* pour plus de détails.

## Ports RS-422A/485

N'oubliez pas de tenir compte des différences suivantes lorsque vous passez d'un système Host Link existant à un système utilisant un port RS-422A/485 sur une carte de communications série de la série CS (CS1W-SCB41 port 2) ou série CJ (CJ1W-SCU41 port 1).

Anciens produits	Numéro de modèle	Modifications nécessaires pour le produit de la série CS/CJ	
		Câblage	Autres
Cartes Host Link série C	3G2A5-LK201-E C200H-LK202 3G2A6-LK202-E	Les broches des câbles ont été changées de la manière suivante : SDA : 9 broches à 1 broche SDB : 5 broches à 2 broches RDA : 6 broches à 6 broches RDB : 1 broche à 8 broches SG : 3 broches à Non connecté FG : 7 broches à la broche Boîtier du connecteur	Il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres longueurs de texte dans les trames ou d'autres caractéristiques de commandes CS/CJ. (voir remarque)
Carte de communications C200HX/HG/HE	C200HW-COM□□-E	Aucun changement au niveau du câblage.	Il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres caractéristiques de commandes CS/CJ.

Anciens produits	Numéro de modèle	Modifications nécessaires pour le produit de la série CS/CJ	
		Câblage	Autres
UC série CVM1 ou CV	CVM1/CV-CPU□□-E	Aucun changement au niveau du câblage.	Il est possible de continuer à utiliser les programmes de l'ordinateur hôte sans les modifier tant que les paramètres de communication (par exemple, la vitesse de transmission) ne sont pas modifiés. Cependant, il se peut qu'il faille modifier les programmes afin qu'ils acceptent d'autres caractéristiques de commandes CS/CJ.
Carte Host Link série CVM1 ou CV	CV500-LK201		

**Remarque** Le nombre de mots pouvant être lus et écrits par trame (c-à-d, les longueurs de texte) avec les commandes en mode C est différent pour les cartes Host Link série C et les cartes de communications série des séries CS/CJ. Il se peut qu'un programme installé sur l'ordinateur hôte précédemment utilisé pour les cartes Host Link série C ne fonctionnent pas correctement si vous les utilisez sur des API série CS/CJ. Vérifiez le programme de l'ordinateur hôte avant de l'utiliser et apportez les corrections nécessaires pour pouvoir gérer différentes longueurs de texte du cadre. Consultez le *Manuel de référence des commandes de communication, série CS/CJ (W342)* pour plus de détails.



# Index

## A

adressage  
adresses indirectes, 26–27  
adresses mémoire, 24  
opérandes, 25  
registres d'index, 274  
*Voir aussi* registres d'index

affectation d'E/S  
configuration du premier mot, 345

alarmes  
alarmes programmées par l'utilisateur, 327

alarmes d'erreur, 327

anciens produits  
comparaison, 384

API  
comparaison, 357

API C200HX/HG/HE  
comparaison, 357

API série CV  
comparaison, 357

applications  
mémoire de fichiers, 209  
précautions, xxii

arrêt à chaud, 315

## B

batterie  
compartiment, 2  
installation, 2

Bit de maintien IOM, 316

bit de sortie OFF, 352

boucle FOR-NEXT, 60

boucles  
boucles FOR/NEXT, 60

## C

capacité du programme, 42

caractères ASCII, 29

carte d'analyse  
paramètres, 303

carte de communications C200HX/HG/HE  
modifications des caractéristiques de communication, 384

cartes d'E/S standard  
instructions des cartes d'E/S standard, 132  
temps de réponse d'entrée, 344

cartes de communications C200H, 384

cartes Host Link  
modifications des caractéristiques de communication, 384

cartes Host Link série C  
modifications des caractéristiques de

communications, 383

cartes interrogées  
paramètres, 303

Cartes mémoire, 7, 193  
précautions, 193

cartes mémoire  
initialisation, 247

cartes série C  
modifications des caractéristiques de communication, 384

chaînes de texte  
instructions de traitement des chaînes de texte, 150  
opérandes, 28

commandes FINS  
liste, 289  
mémoire de fichiers, 215

Commandes Host Link, 287

communication  
messages, 290

communications  
instructions de communications série, 135  
sans protocole, 291  
*Voir aussi* Communications en série

communications en série  
fonctions, 284

communications Host Link, 286

communications sans protocole, 291

comparaison  
anciens produits, 384

compteurs  
mode de mise à jour, 305

conditions d'exécution  
tâches, 164  
variations, 34

conditions d'instruction  
description, 22

configuration  
*Voir aussi* installation

Configuration de l'API, 7

configuration forcée des bits  
débogage, 348

Consoles de programmation  
mémoire de fichiers, 213

constantes  
opérandes, 28

CX-Programmer, 20  
mémoire de fichiers, 213

cyclique, mise à jour, 260

## D

date  
réglage de l'horloge, 5

date des paramètres, 321

date du programme utilisateur, 321

dates  
  programme et paramètres, 320

débogage, 326, 348  
  instructions de débogage, 141  
  instructions de diagnostics d'erreurs, 142

démarrage  
  démarrage/arrêt à chaud, 315  
  transfert de fichiers automatique, 198, 228

démarrage à chaud, 315

détection du point d'erreur, 328

DeviceNet  
  précautions, 327

diagnostic, 326

directives communautaires, xxvi

données à virgule flottante  
  instructions à virgule flottante double précision, 115  
  instructions mathématiques à virgule flottante, 111

données BCD, 30

données binaires non signées, 30

données binaires signées, 30

données de tables  
  traitement, 281

Drapeau d'égalité, 59

drapeau d'erreur de tâche, 173

drapeau d'exécution de la tâche initiale, 172

drapeau de retenue, 59

drapeau inférieur à, 59

drapeau négatif, 59

drapeau supérieur à, 59

drapeaux, 22  
  drapeaux de condition, 55

drapeaux de condition, 55  
  utilisation dans les tâches, 170

drapeaux de tâches, 171

## **E**

édition en ligne, 350

effacement de la mémoire, 4

entrées à grande vitesse, 259

environnement d'utilisation  
  précautions, xxii

erreurs  
  détection du point d'erreur, 328  
  entrée de programmes, 64  
  erreur d'accès, 66  
  erreur d'instruction illégale, 66  
  erreur de dépassement de la MU, 66  
  erreur de traitement de l'instruction, 66  
  erreurs de programmation, 68  
  erreurs générées par l'utilisateur, 327  
  fatales, 68  
  journal d'erreurs, 326  
  erreurs de programme, 68

essais de fonctionnement, 348

état en attente  
  description, 16

état exécutable  
  description, 16

## **F**

fichiers de données, 209

fichiers de programmes, 210

flux d'alimentation  
  description, 21

fonction sortie OFF, 327

fonctionnement  
  débogage, 348  
  essais de fonctionnement, 348  
  fonctionnement de base, 160  
  UC, 1

formats de données, 30

## **H**

heure  
  réglage de l'horloge, 5

horloge, 318  
  instructions de temporisation, 140  
  réglage de l'horloge, 5

## **I**

immédiate, mise à jour, 260

initialisation  
  cartes mémoire, 247  
  mémoire de fichiers EM, 247  
  mémoire E/S, 10

installation  
  configuration initiale, 2, 5  
  précautions, xxii

instructions  
  boucles, 60  
  conditions d'exécution, 34  
  conditions d'instruction, 22  
  contrôle des tâches, 167  
  différenciation à l'entrée, 35  
  emplacements des programmes, 23  
  en boucle, 22  
  instructions d'affichage, 140  
  instructions d'entrée des séquences, 72  
  instructions d'entrée et de sortie, 21, 23  
  instructions d'incrémementation, 96  
  instructions de base, 21  
  instructions de commande des séquences, 77  
  instructions de communications série, 135  
  instructions de comparaison, 85  
  instructions de compteur, 81  
  instructions de contrôle des données, 123

- instructions de contrôle des tâches, 153
- instructions de conversion, 102
- instructions de débogage, 141
- instructions de décalage de données, 92
- instructions de décrémentation, 96
- instructions de déplacement de données, 89, 154–155
- instructions de diagnostics d'erreurs, 142
- instructions de mémoire de fichiers, 139
- instructions de pas, 132
- instructions de programmation de bloc, 144
- instructions de sortie des séquences, 74
- instructions de sous-programme, 127
- instructions de temporisation, 81, 140
- instructions de traitement d'interruption, 128
- instructions de traitement de données de tableaux, 115, 119
- instructions de traitement des chaînes de texte, 150
- instructions des cartes d'E/S standard, 132
- instructions différenciées, 37
- instructions logiques, 108
- instructions mathématiques à symboles, 97
- instructions mathématiques à virgule flottante, 111
- instructions mathématiques spéciales, 110
- instructions réseaux, 136
- mémoire de fichiers, 217
- opérandes, 22
- programmes de bloc, 63
- registres d'index, 278
- restrictions dans les tâches, 170
- temporisation, 37
- variations, 34
- instructions de plage, 281
- instructions de table d'enregistrements, 281
- instructions différenciées, 37
- instructions différenciées par le bas, 35
- instructions différenciées par le haut, 35
- interruptions, 259
  - désactivation, 188
  - priorité des tâches d'interruption, 184
  - Voir aussi* interruptions externes
- interruptions d'alimentation
  - désactivation, 317
- interruptions d'E/S
  - tâches, 163, 177–180
- interruptions de mise hors tension
  - tâches, 162, 177–183
- interruptions externes
  - tâches, 163, 178–180, 183
- interruptions programmées
  - tâches, 162, 177, 180–181
  - utilisation comme temporisation, 313
- IORF(097), mise à jour, 261

## J

- journal d'erreurs, 326

## L

- Liaisons API série, 299
- liaisons API série, 300
  - configuration de l'API, 303
  - drapeaux associés, 304
  - mots affectés, 302

## M

- mathématiques
  - instructions mathématiques à symboles, 97
  - instructions mathématiques à virgule flottante, 111
  - instructions mathématiques spéciales, 110
- mémoire
  - diagramme des blocs de la mémoire de l'UC, 7
  - effacement, 4
  - Voir aussi* mémoire de fichiers
  - Voir aussi* mémoire E/S
  - Voir aussi* mémoire utilisateur
- mémoire de fichiers, 191
  - accès aux répertoires, 202
  - applications, 209, 247
  - fichiers de paramètres, 210
  - fichiers de programmes, 210
  - fonctions, 191
  - instructions de mémoire de fichiers, 139, 217
  - noms et types de fichiers, 197
  - traitement des fichiers, 211
- Mémoire de fichiers EM, 193
  - Voir aussi* mémoire de fichiers
- mémoire de fichiers EM
  - initialisation, 247
  - opérations, 253
- Mémoire E/S, 7
- mémoire E/S
  - adressage, 24
  - initialisation, 10
  - tâches, 169
- mémoire flash, 323
- messages, 290
- méthode de liaison complète, 300
- méthode de liaison de la carte d'analyse, 300
- mise à jour
  - IORF(097), 41, 187, 261
  - mise à jour cyclique, 39, 260
  - mise à jour d'E/S, 39, 260
  - mise à jour immédiate, 34, 39, 260
- mise à jour cyclique, 39
- mise à jour d'E/S, 39
- mise à jour de données, 300

mise à jour immédiate, 34, 39  
mise à jour IORF(097), 41  
  tâches d'interruption, 187  
mnémonique, 43  
  insertion, 47  
mode de démarrage, 316  
mode de mise à jour, 305  
  temporisations et compteurs, 305  
mode MONITOR  
  description, 9  
mode prioritaire du traitement des périphériques, 336  
Mode PROGRAM  
  description, 9  
mode PROGRAM  
  description, 9  
mode RUN  
  description, 9  
modes de fonctionnement  
  description, 8  
  mode de démarrage, 11

## N

noms de fichiers, 197

## O

opérandes  
  chaînes de texte, 28  
  constantes, 28  
  description, 22  
  spécification, 25

## P

paramètres  
  paramètres du démarrage, 315  
  *Voir aussi* paramètres des interrupteurs  
périphériques de programmation  
  fonctionnement pour les tâches, 189  
  mémoire de fichiers, 211  
ports RS-232C  
  modifications par rapport aux produits précédents, 383  
ports RS-422A/485  
  modifications par rapport aux produits précédents, 384  
précautions, xix  
  applications, xxii  
  environnement d'utilisation, xxii  
  générales, xx  
  mise à jour E/S, 9  
  programmation, 55  
  sécurité, xx  
  tâches d'interruption, 186  
précautions en matière de sécurité, xx  
programmation, 19

capacité du programme, 42  
conception de tâches, 175  
concepts de base, 42  
emplacements des instructions, 23  
erreurs, 64  
exemples, 50  
flux d'alimentation, 21  
mnémonique, 43  
précautions, 55  
programmation à distance, 322  
programmation pas à pas, 60  
  restrictions, 62  
programmes de bloc, 22, 60  
  restrictions, 63  
programmes et tâches, 12, 20  
protection du programme, 319  
restrictions, 45  
structure du programme, 12, 15, 43  
tâches et programmes, 159  
transfert du programme, 348  
vérification des programmes, 64  
  *Voir aussi* programmes de bloc  
programmation pas à pas, 60  
programme  
  *Voir aussi* programmation  
programme utilisateur, 6–7  
  *Voir aussi* programmation  
programmes de bloc, 22, 60, 63  
  instructions de programmation de bloc, 144  
  rapport avec les tâches, 176  
protection en écriture, 319  
protection en lecture/écriture, 320

## R

registres d'index, 27, 274  
réinitialisation forcée des bits  
  débogage, 348  
répertoires, 202  
réseaux  
  instructions réseaux, 136  
retard de la détection de mise hors tension, 317

## S

sauts, 38, 60  
sauvegarde de données, 323  
sortie RUN, 317  
sorties  
  désactivation, 327, 352  
sous-programmes, 60  
structure du programme, 43  
surveillance  
  surveillance à distance, 322  
  surveillance différenciée, 349

**T**

tâches, 12, 157  
  avantages, 158  
  caractéristiques, 158  
  conception, 175  
  conditions d'exécution, 164  
  création de tâches, 189  
  description, 14  
  drapeaux, 171  
  état, 16  
  exécution, 168  
  exemples, 173  
  instructions de contrôle des tâches, 153  
  introduction, 162  
  limites, 170  
  numéros de tâche, 169  
  rapport avec les programmes de bloc, 176  
  tâches cycliques, 159, 162  
  Tâches d'interruption, 178  
  tâches d'interruption, 159, 162  
  temporisations, 170  
  temps d'exécution, 18  
  utilisation des drapeaux de condition, 170  
  *Voir aussi* tâches cycliques  
  *Voir aussi* tâches d'interruption

tâches cycliques, 159, 162  
  état, 165  
  état Désactivé (INI), 165  
  état READY, 165  
  état RUN, 165  
  état WAIT, 165

Tâches d'interruption, 177

tâches d'interruption, 159, 162–188  
  drapeaux et mots associés, 185  
  précautions, 186  
  priorité, 184

temporisations, 305  
  création à l'aide d'interruptions programmées, 313

temps de cycle  
  paramètre, 258  
  surveillance, 258  
  temps d'exécution des tâches, 18  
  temps de cycle minimum, 257

temps de cycle (fixe) minimum, 257

temps de cycle maximum, 258

temps de réponse d'E/S  
  cartes d'E/S standard série CS/CJ, 344

traçage de données, 353

traitement de pile, 279

traitement des périphériques  
  traitement prioritaire, 336

transfert automatique au démarrage, 198, 228

transfert du programme, 348

types de fichiers, 197

**U**

UC  
  capacités, 42  
  fonctionnement, 1  
  fonctionnement de base, 160  
  structure interne, 6

UC CVM1  
  modifications des caractéristiques de communication, 384–385

UC série CV  
  modifications des caractéristiques de communication, 384–385

unités  
  profils, 322

**V**

verrouillage, 22, 38, 60  
virgule décimale flottante, 31

**Z**

Zone de paramètres, 7  
zone de paramètres  
  fichiers, 210  
zones de données  
  adressage, 24



## Historique des révisions

Un code de révision du manuel apparaît sous la forme d'un suffixe à côté du numéro de catalogue, sur la couverture du manuel.

Cat. No. W394-FR1-07



Le tableau suivant présente les modifications apportées au manuel au cours des différentes révisions. Les numéros de page se rapportent à la version précédente.

Code de révision	Date	Contenu de la révision
01	Avril 2001	Produit d'origine
02	Octobre 2001	Ajout d'informations sur les UC série CS et CJ à grande vitesse (CS1G/H-CPU□□H et CJ1G/H-CPU□□H) dans tout le manuel.
03	Juillet 2002	Informations sur les UC CJ1M ajoutées dans tout le manuel. Un automate programmable est maintenant uniquement désigné par le terme d'API. Les autres changements sont : <b>Pages xvi et xviii</b> : Ajout de Attention. <b>Page xix</b> : Modification de l'élément 2 dans le bas de la page. <b>Page 28</b> : Modification de la description de la chaîne de texte. <b>Page 167</b> : Modification de l'exemple de programmation. <b>Pages 168, 169, 265 et 266</b> : Ajout d'informations sur l'alimentation c.c. <b>Page 179</b> : Ajout des précautions relatives aux cartes mémoire. <b>Page 229</b> : Modification de l'illustration. <b>Page 262</b> : Ajout d'informations sur la méthode de mise à jour des temporisations/compteurs. <b>Page 273</b> : Ajout des précautions relatives à DeviceNet. <b>Page 301</b> : Correction de la vitesse de traitement des cartes. <b>Page 304</b> : Correction du temps de réponse d'interruption. <b>Page 320</b> : Modification de la prise en charge par CJ1 de IOSP/IORS.
04	Septembre 2002	Informations sur les UC CJ1D ajoutées dans tout le manuel. Les autres changements sont : <b>Page xv</b> : Ajout du paragraphe Attention relatif à la fonction de sauvegarde. <b>Page xvi</b> : Modification du premier Attention et correction de l'Attention relatif aux types d'UC en mode de fonctionnement au démarrage. <b>Page xviii</b> : Ajout d'un Attention relatif au port RS-232C, vers le milieu de la page. <b>Page 6</b> : Ajout d'informations sur les versions de CX-Programmer. <b>Page 184</b> : Correction dans le tableau des informations relatives à la carte mémoire. <b>Page 274</b> : Modification de la plage acceptable pour le nombre de carte maximale, vers le haut de la page. <b>Page 294</b> : Modification du mot Remarque en Attention et réécriture du paragraphe. <b>Page 303</b> : Réécriture de la remarque 3. <b>Page 304</b> : Ajout dans le tableau du milieu.
05	Avril 2003	<b>Page 44</b> : Réécriture du premier concept de base du schéma contact. <b>Page 45</b> : Modification des informations de la seconde restriction. <b>Page 46</b> : Modification des informations de la seconde restriction. Suppression de la sixième restriction. <b>Page 54</b> : Modification des informations relatives aux équations logiques auxquelles il faut faire attention ou qu'il faut réécrire. <b>Pages 157 et 158</b> : Modifications dans le tableau des différences entre les tâches cycliques normales et extra. <b>Page 220</b> : Ajout de la Remarque 5. <b>Page 226</b> : Modifications des informations concernant les cartes prises en charge. <b>Page 233</b> : Modifications des informations concernant les interruptions d'alimentation lors de l'accès à la mémoire de fichiers. <b>Page 264</b> : Ajout d'une remarque. <b>Page 273</b> : Ajout d'une remarque. <b>Page 276</b> : Modifications des informations dans les tableaux. Ajout d'une remarque en-dessous du premier tableau et modification de la remarque sous le deuxième tableau. <b>Page 310</b> : Modification des informations sous le premier tableau concernant CX-Programmer. <b>Page 314</b> : Ajout d'informations dans la deuxième remarque. <b>Page 326</b> : Ajout d'informations dans l'étape 1 de la procédure d'enregistrement des données.

## Historique des révisions

Code de révision	Date	Contenu de la révision
06	Décembre 2003	<p>Ajout d'informations sur les nouvelles fonctions prises en charge par les nouvelles versions des UC (trop nombreuses à énumérer).</p> <p><b>Pages xi à xx</b> : Mise à jour des informations PLP</p> <p><b>Page 72</b> : Ajout de remarques au-dessus du tableau et ajout des instructions AND NOT et OR NOT.</p> <p><b>Pages 160, 201, 202, 228, 293 et 320</b> : Ajout de remarques.</p> <p><b>Page 189</b> : Suppression d'informations sur CX-Programmer version 1.0 ou supérieure.</p> <p><b>Page 191</b> : Ajout d'informations au-dessus et dans le tableau et ajout d'un nouveau tableau.</p> <p><b>Pages 197, 199, 216 à 218, 230 et 231</b> : Ajout d'informations sur le remplacement sans fichier de zone de paramètres.</p> <p><b>Page 226</b> : Agrandissement du tableau.</p> <p><b>Page 294</b> : Ajout de la section 6-6-8.</p> <p><b>Pages 337 à 339, 342, 344, 345 et 347</b> : Ajout de lignes dans le tableau.</p> <p><b>Page 339</b> : Ajout d'informations sur BCMP2.</p>
07	Juillet 2004	<p>Des modifications ont été apportées dans le manuel aux informations sur les nouvelles fonctions prises en charge par la mise à niveau de la version de carte 2.0 vers la version 3.0 des UC série CS/CJ, notamment les modifications suivantes :</p> <p><b>Page 8</b> : Graphique modifié.</p> <p><b>Page 32</b> : Valeurs modifiées dans les colonnes décimal et hexadécimal pour les binaires signés.</p> <p><b>Page 137</b> : Informations ajoutées à l'explication des instructions TXD(236) et RXD(235). Informations sur les nouvelles instructions ajoutées dans les nouvelles sections 3-23 et 3-24.</p> <p><b>Page 190</b> : Informations modifiées et ajoutées dans la présentation et le graphique. Modification des informations dans le tableau et ajout de nouvelles informations.</p> <p><b>Pages 192 et 193</b> : Informations réorganisées et modifiées dans 5-1-2.</p> <p><b>Page 194</b> : Tableaux développés.</p> <p><b>Page 198</b> : Tableau développé, suppression et ajout de remarques, ajout d'informations sur les fichiers système du CX-Programmer.</p> <p><b>Page 202</b> : « numériques ou caractères » remplacé par « caractères » pour le format des cellules.</p> <p><b>Page 204</b> : Tableau modifié et ajout d'une remarque.</p> <p><b>Page 207</b> : Ajout d'informations sur les fichiers de tables de symboles et les fichiers de commentaires.</p> <p><b>Page 228</b> : Ajout d'informations, y compris des tableaux, sur l'état et la vérification des affectations d'E/S. Ajout d'informations sur la fonction de sauvegarde simple.</p> <p><b>Page 229</b> : Ajout d'informations sur les comparaisons de données.</p> <p><b>Pages 232 et 233</b> : Mémoire E/S corrigée pour l'UC dans les titres du tableau.</p> <p><b>Page 233</b> : Ajout d'un nouveau tableau</p> <p><b>Page 240</b> : Ajout d'informations sur l'état et la vérification des affectations d'E/S.</p> <p><b>Page 277</b> : Ajout d'informations sur la Passerelle série au tableau.</p> <p><b>Page 285</b> : Ajout d'une nouvelle section, 6-3-4, sur la Passerelle série.</p> <p><b>Pages 345 et 364</b> : Ajout d'informations au tableau.</p>