



Allen-Bradley

Automate Logix5550

(Réf. 1756-L1, -L1Mx)

**Conversion d'un
programme PLC-5
ou SLC 500 en
programme
Logix5550
Manuel de
référence**

Allen-Bradley Spares

Informations importantes destinées à l'utilisateur

En raison de la diversité des utilisations des produits décrits dans le présent manuel, les personnes qui en sont responsables doivent s'assurer que toutes les mesures ont été prises pour que l'application et l'utilisation des produits soient conformes aux exigences de performance et de sécurité, ainsi qu'aux lois, règlements, codes et normes en vigueur.

Les illustrations, schémas et exemples de programmes contenus dans ce manuel sont présentés à titre indicatif seulement. En raison des nombreuses variables et impératifs associés à chaque installation, la société Allen-Bradley ne saurait être tenue responsable ou redevable (y compris en matière de propriété intellectuelle) des suites d'utilisations réelles basées sur les exemples et schémas présentés dans ce manuel.

La publication SGI-1.1, « *Safety Guidelines for the Application, Installation and Maintenance of Solid State Control* » (disponible auprès de votre agence commerciale Allen-Bradley), décrit certaines différences importantes entre les équipements électroniques et les équipements électromécaniques qui devront être prises en compte lors de l'application de ces produits comme indiqué dans la présente publication.

Toute reproduction partielle ou totale du présent manuel sans autorisation écrite de la société Allen-Bradley est interdite.

Des remarques sont utilisées tout au long de ce manuel pour attirer votre attention sur les mesures de sécurité à prendre en compte :



ATENCION: Actions ou situations risquant d'entraîner des blessures pouvant être mortelles, des dégâts matériels ou des pertes financières.

Les messages « Attention » vous aident à :

- identifier un danger
- éviter ce danger
- en discerner les conséquences

Importante: Informations particulièrement importantes dans le cadre de l'utilisation du produit.

PLC-5, SLC, Logix5550 et RSLogix sont des marques commerciales de Rockwell Automation.

Chapitre 1**Conversion d'un programme
PLC-5 ou SLC 500 en un projet
Logix5550**

Introduction	1-1
Comparaison de l'architecture PLC-5 et SLC 500 avec l'architecture Logix5550	1-2
Exportation d'un programme PLC-5 ou SLC 500	1-3
Utilisation du logiciel de programmation RSLogix5 ou RSLogix500	1-4
Utilisation d'un logiciel de programmation Série 6200	1-6
Utilisation d'un logiciel de programmation Série A.I.	1-7
Conversion d'un programme PLC-5 ou SLC 500	1-8
Visualisation des résultats de conversion	1-9
Visualisation du fichier journal	1-11
Importation d'un fichier texte ASCII dans un projet Logix5550	1-12
Corrections des instructions PCE	1-13
Localisation des instructions PCE	1-14
Corrections des instructions UNK	1-15
Configuration de l'automate et du châssis	1-15
Configuration des E/S	1-16
Configuration complète de MSG	1-17
Autres informations	1-18

Chapitre 2**Conversion de structure de
programme**

Introduction	2-1
Création d'une tâche continue	2-3
Conversion d'interruptions temporisées programmables (STI) ..	2-3
Conversion d'interruptions sur entrées (DII/PII)	2-4

Chapitre 3**Conversion de données**

Introduction	3-1
Identification par les fichiers import/export PLC-5 et SLC 500 des valeurs des tableaux des données	3-2
Identification par les fichiers import/export Logix5550 des types de fichier	3-2
Conversion de données d'entrée (I) et de sortie (O)	3-3
Conversion du type de fichier d'état (S)	3-4
Conversion du type de fichier binaire (B)	3-4
Conversion du type de fichier Temporisateur (T)	3-5
Règles de conversion du temporisateur	3-6
Conversion du type de fichier Compteur (C)	3-7
Conversion du type de fichier Contrôle (R)	3-8
Conversion du type de fichier de nombres entiers (N)	3-9
Conversion du type de fichier de données	

à virgule flottante (F)	3-9
Conversion du type de fichier ASCII (A)	3-10
Conversion du type de fichier à base décimale (D)	3-10
Conversion du type de fichier transfert par blocs (BT)	3-10
Règles de conversion des blocs-transferts	3-11
Conversion des types de fichier M0 et M1	3-12
Conversion du type de fichier Message (MG)	3-12
Règles de conversion de message	3-13
Conversion du type de fichier PID (PD)	3-14
Conversion de type de fichier de chaînes de caractères ASCII (ST)	3-14
Conversion du type de fichier ControlNet (CT)	3-14
Conversion de valeurs constantes	3-15
Conversion d'adresses indirectes	3-16
Conversion de l'adressage indirect sur le numéro de fichier	3-16
Conversion d'adresses indexées	3-18
Conversion d'adresses indexées commandées par le mot d'état du processeur S:24	3-18
Conversion d'adresses indexées qui spécifient des données dans les fichiers (tableaux Logix5550).	3-18
Conversion de symboles	3-18
Conversion des commentaires d'adresse	3-19

Chapitre 4

Conversion des instructions PLC-5

Introduction	4-1
Règles de conversion	4-1
Liste d'instructions	4-2
Instructions de conversion CAR	4-16
Conversion des instructions FOR/NXT/BRK	4-16

Annexe A

Messages de conversion

Introduction	A-1
Messages d'état	A-1
Messages d'information	A-2
Messages de question	A-3

Conversion d'un programme PLC-5 ou SLC 500 en un projet Logix5550

Introduction

Le logiciel de programmation RSLogix500 comprend un outil de conversion qui permet de convertir un fichier import/export PLC-5 ou SLC 500 (extension .PC5 ou SLC) en un fichier import/export Logix5550 complet (extension .L5K). Ce manuel décrit l'outil de conversion qui est fourni avec la version 2.10.00 du logiciel de programmation RSLogix5000.

Important: Actuellement, l'outil de conversion convertit uniquement les instructions à relais. Les fichiers SFC et en texte structuré ne sont pas convertis.



ATTENTION: Après avoir exécuté le processus de conversion, le fichier import/export résultant nécessite encore quelques manipulations. Il convient de configurer les entrées/sorties du système Logix5550 et d'utiliser les instructions BTM, MOV ou COP pour placer ces données configurées dans les structures créées par le processus de conversion.

L'outil de conversion produit un fichier import/export Logix5550 correct du point de vue syntaxe, mais le but exact de l'application d'origine risque d'être perdu. Cette perte peut être due à des différences entre les règles de priorité, l'adressage indexé, l'adressage des E/S, etc. Le fichier journal enregistre ces différences.

Le but de l'outil de conversion est de réduire le volume de travail engendré par la conversion d'un programme PLC-5 ou SLC 500 en un projet Logix5550. L'outil de conversion convertit automatiquement la logique du programme, mais il n'apporte pas de solution complète. En fonction de l'application, des tâches supplémentaires sont requises afin de pouvoir utiliser correctement le programme converti.

Le processus de conversion complète inclut :

Etape de conversion :	Voir page :
Exportation du programme PLC-5 ou SLC 500 en fichier texte ASCII	1-3
Utilisation de l'outil de conversion pour convertir le programme	1-8
Visualisation des résultats de conversion	1-9
Importation du fichier texte ASCII dans un projet Logix5550	1-12
Corrections des instructions PCE	1-13
Corrections des instructions UNK retravaillées	1-15
Configuration de l'automate et du châssis	1-15
Configuration des E/S	1-16
Configuration complète de MSG	1-17
Autres informations	1-18

Comparaison de l'architecture PLC-5 et SLC 500 avec l'architecture Logix5550

L'architecture Logix5550 diffère en de nombreux points de celle des processeurs PLC-5 et SLC 500. L'outil de conversion transcrit cette architecture le mieux possible afin de l'adapter à l'architecture Logix5550.

Important: Compte tenu de ces différences architecturales, il est nécessaire de retravailler le projet Logix5550 afin de s'assurer de son bon fonctionnement.

Quelques unes des principales différences d'architecture sont :

Aspect architectural :	Comparaison :
UC	Les processeurs PLC-5 et SLC 500 sont basés sur des opérations sur 16 bits. Les automates Logix5550 utilisent des opérations sur 32 bits. L'outil de conversion transcrit le programme hérité dans son équivalent à 32 bits.
Système d'exploitation	<p>Les processeurs PLC-5 et SLC 500 reconnaissent des fichiers programme individuels qui peuvent être configurés en tant qu'interruptions temporisées programmables (STI) ou interruptions sur entrée (DII/PII). En outre, le processeur PLC-5 reconnaît de nombreux programmes de commande principaux (MCP). L'automate Logix5550 combine ces aspects dans une organisation de tâches, programmes et sous-programmes. L'outil de conversion transcrit les types de programmes existants en tâches Logix5550 appropriées.</p> <p>Les processeurs PLC-5 et SLC 500 utilisent un fichier de données S pour mémoriser l'état du processeur. L'automate Logix5550 mémorise les données différemment. Au lieu d'accéder à différents emplacements au sein d'un fichier, vous pouvez utiliser des instructions GSV (Lire la valeur système) et SSV (Définir la valeur système) pour préciser l'information d'état requise. Il s'agit d'une différence majeure qui nécessite un travail ultérieur une fois que le programme converti est importé dans l'automate Logix5550.</p> <p>Les processeurs PLC-5 et SLC 500 utilisent également des bits en S : 0 pour les indicateurs d'état arithmétiques. Par exemple, S:0/03 enregistre l'état du signe. L'automate Logix5550 utilise des mots clés pour faire référence à ces indicateurs. Ainsi, au lieu de faire référence à une adresse binaire pour commander une opération arithmétique, vous utilisez le mot clé S:N.</p>

Aspect architectural :	Comparaison :
Entrée et sorties	<p>Les processeurs PLC-5 et SLC 500 configurent la mémoire des E/S dans des fichiers de données d'E/S. Les données d'E/S sont mises à jour de manière synchrone par rapport à la scrutation de programme de sorte que vous savez que vous disposez de valeurs actuelles chaque fois que le processeur démarre une scrutation. L'automate Logix5550 relève les E/S qui sont mises à jour de manière asynchrone par rapport à la scrutation logique. Pour l'automate Logix5550, vous pouvez envisager de créer une mémoire tampon de données d'E/S devant être utilisées avec des valeurs statiques durant l'exécution du programme et mettre à jour la mémoire tampon selon les besoins.</p> <p>Une fois la conversion terminée, vous devez ajouter des instructions afin de copier les données d'E/S dans les tableaux E et S. Il est recommandé d'effectuer ceci au début ou à la fin d'un programme pour mettre les données dans la mémoire tampon afin qu'elles se présentent de manière synchrone par rapport à la scrutation de programme.</p>
Données	<p>Les processeurs PLC-5 et SLC 500 mémorisent toutes les données dans des tableaux de données globales. Vous pouvez accéder à ces données en spécifiant les adresses que vous souhaitez. L'automate Logix5550 reconnaît les données qui sont locales par rapport à un programme donné et globales par rapport à l'ensemble des tâches au sein de l'automate. L'automate Logix5550 peut également partager des données avec d'autres automates ; dans ce cas, vous n'utilisez pas d'adresses mais des points pour accéder aux données de votre choix.</p> <p>Chaque fichier de données PLC-5 et SLC 500 peut enregistrer plusieurs mots de données de même nature. L'automate Logix5550 utilise des tableaux pour enregistrer les données de même nature. L'outil de conversion convertit les fichiers de données PLC-5 et SLC 500 en tableaux Logix5550.</p>
Temporisateurs	<p>Les temporisateurs PLC-5 et SLC 500 sont basés sur leur architecture à 16 bits et peuvent avoir différentes bases de temps. L'automate Logix5550 est basé sur une architecture à 32 bits et reconnaît uniquement une base de temps de 1 ms. Les outils de conversion convertissent ces temporisateurs existants le mieux possible afin de s'adapter à l'architecture Logix5550. Les temporisateurs convertis peuvent nécessiter un travail ultérieur afin de vérifier qu'ils fonctionnent correctement.</p>
Communications	<p>Le processeur PLC-5 reconnaît des instructions de lecture et d'écriture (BTR et BTW) de blocs-transferts, des instructions ControlNet I/O (CIO) et des instructions de message (MSG). Le processeur SLC 500 reconnaît les instructions MSG. L'automate Logix5550 reconnaît les instructions MSG. L'outil de conversion transcrit les instructions existantes BTR, BTW, et MSG en instructions MSG Logix5550. Les instructions ControlNet d'E/S ne sont pas converties. Une fois que vous avez importé le programme converti, vous devez configurer les instructions MSG afin qu'elles fonctionnent correctement et retravailler toutes les instructions ControlNet d'E/S.</p>

Le reste de ce manuel décrit les cas particuliers de conversion d'architecture.

Exportation d'un programme PLC-5 ou SLC 500

Avant de pouvoir convertir un programme PLC-5 ou SLC 500 en son équivalent Logix5550, vous devez d'abord exporter le programme en un fichier texte ASCII portant une extension .PC5 pour un fichier PLC-5, ou une extension .SLC pour un fichier SLC 500. Si vous choisissez de convertir les commentaires et les symboles, vous aurez également besoin du fichier .TXT, qui correspond au format de logiciel de programmation 6200 standard pour un fichier de documentation.

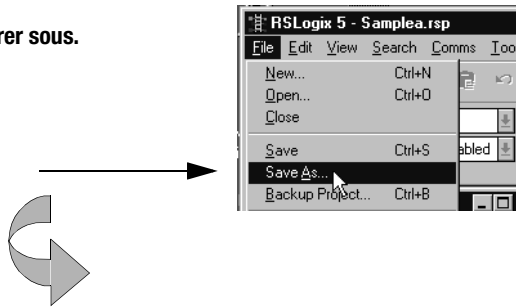
La procédure d'exportation du programme en fichier texte ASCII que vous adopterez dépend du logiciel de programmation que vous utilisez.

Si vous utilisez :	Voir page :
Le logiciel de programmation RSLogix5 ou RSLogix500	1-4
Le logiciel de programmation Série 6200	1-6
Le logiciel de programmation Série A.I.	1-7

Utilisation du logiciel de programmation RSLogix5 ou RSLogix500

Créez un fichier d'extension .PC5 ou .SLC pour le fichier programme :

1. Sélectionnez Fichier → Enregistrer sous.



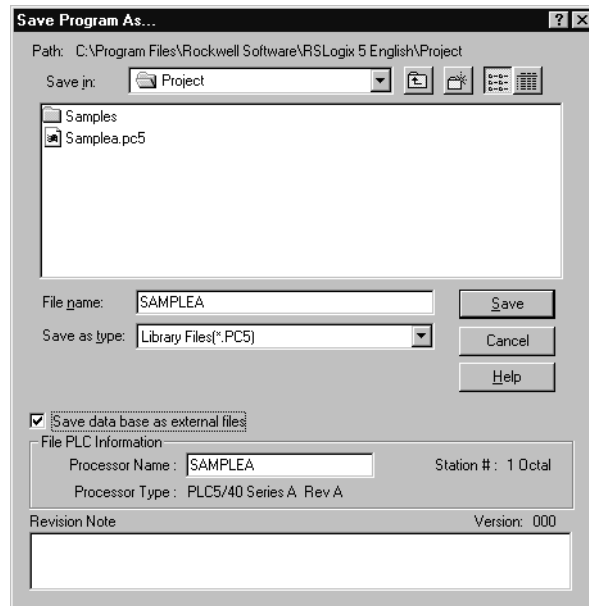
2. Sélectionnez le programme devant être exporté.

Sélectionnez le programme devant être exporté.
 Par défaut, le logiciel indique le répertoire \Project comme destination. Vous pouvez sélectionner ou entrer un autre répertoire de destination.

Vous devez sélectionner le format fichier bibliothèque (.PC5 ou .SLC).

Sélectionnez cette option de manière à ce que les commentaires et les symboles soient inclus dans l'exportation.

Cliquez sur Enregistrer.

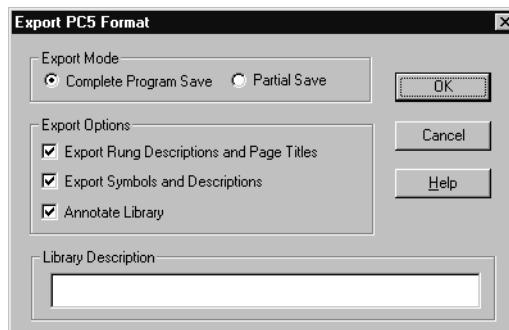


3. Sélectionnez les options d'exportation.

Sélectionnez Enregistrement programme terminé.

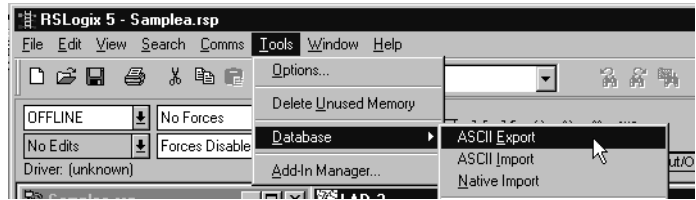
Sélectionnez toutes ces options.

Cliquez sur OK.



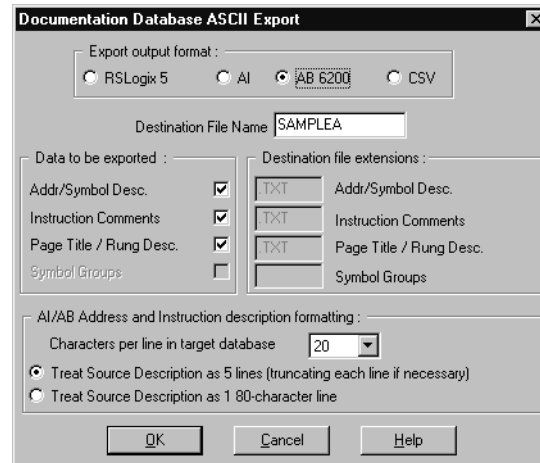
Créez un fichier d'extension .TXT pour les commentaires et les symboles :

1. Sélectionnez Outils → Base de données → Exporter ASCII.



2. Sélectionnez l'endroit où doivent être exportés les commentaires et les symboles.

Sélectionnez le format AB 6200.



Cliquez sur OK.

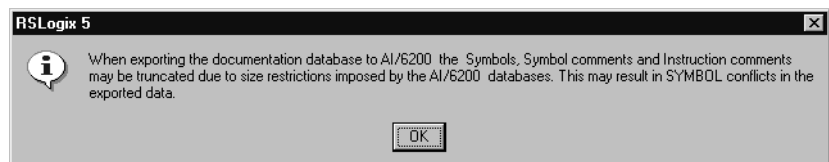
3. Sélectionnez le répertoire où se trouve le fichier .PC5 ou .SLC.



Cliquez sur OK.

4. Validez les avertissements relatifs aux commentaires et symboles.

Cliquez sur OK.



Le logiciel de programmation RSLogix5 enregistre les programmes PLC-5 en utilisant des extensions de fichier .RSP. Le logiciel de programmation RSLogix500 enregistre les programmes SLC 500 en utilisant des extensions de fichier .RSS.

Utilisation d'un logiciel de programmation Série 6200

Pour exporter un programme et ses symboles à l'aide du logiciel de programmation Série 6200 :

1. Placez les fichiers programme dans \IPDS\ARCH\PLC5.
2. Lancez le logiciel de programmation Série 6200.
3. Sélectionnez **F7** : File Utils → **F7** : Export → **F1** : Processor Memory File Only.
4. Pointez le curseur sur le programme que vous souhaitez exporter.
5. Sélectionnez **F3** : Select Source → **F1** : Begin Operation.
6. Une fois l'exportation terminée, appuyez sur une touche quelconque pour continuer.
7. Renommez le fichier journal dans \IPDS\ARCH\PLC5 pour éviter que le fichier ne soit écrasé au cours de l'étape suivante.
8. Sélectionnez **F7** : File Utils → **F7** : Export → **F3** : Comments and Symbols.
9. Sélectionnez le programme à exporter à l'aide du curseur.
10. Sélectionnez **F3** : Select Source → **F1** : Begin Operation.
11. Une fois l'exportation terminée, appuyez sur une touche quelconque pour continuer.
12. Copiez ou déplacez les fichiers .PC5/.SLC et .TXT là où l'outil de conversion Logix5550 les trouvera.

Le logiciel 6200 utilise ces extensions de fichier pour des fichiers programme : .AC\$, .AF5, .B0\$, .B1\$, .D1\$, .IX\$, .LX\$, .OP\$, .P1\$, et .PC\$.

Utilisation d'un logiciel de programmation Série A.I.

Pour exporter un programme et ses symboles à l'aide du logiciel de programmation Série A.I. :

1. Lancez le logiciel de programmation Série A.I.5.
2. Sélectionnez **F1** : Select Program/PLC-5 Address.
3. Placez le curseur sur le programme à exporter et appuyez sur **Enter**.
4. Sélectionnez **F5** : Utility Options → **F1** : Rebuild Damaged Data Base → **F1** : Rebuild current program → **F1** : Yes – Force rebuilding of Index files.
5. Une fois le processus de reconstitution terminée, appuyez sur une touche quelconque pour continuer.
6. Sélectionnez **F4** : Export data base → **F4** : 6200 ASCII.
7. Entrez un nom sans extension pour exporter le fichier de base de données.
8. Une fois l'exportation de la base de données terminée, appuyez sur une touche quelconque pour continuer.
9. Appuyez sur **ESC** pour retourner au menu principal.
10. Sélectionnez **F2** : Offline Programming → **F3** : Edit → **F2** : Block → **F1** : Block Start → **F2** : Copy Block (peu importe ce qui est en fait sélectionné) → **F8** : Save Block.
11. Sélectionnez **F1** : ASCII et entrez un nom sans extension pour le programme exporté.
12. Sélectionnez **F2** : No Rung Descriptions. L'outil de conversion utilise les commentaires à partir du fichier .TXT et non pas du fichier .PC5/.SLC.
13. Sélectionnez **F3** : Entire Program → **F4** : No Annotation → **F5** : No → **F7** : Export. Vous n'avez pas besoin de la « brève description. »

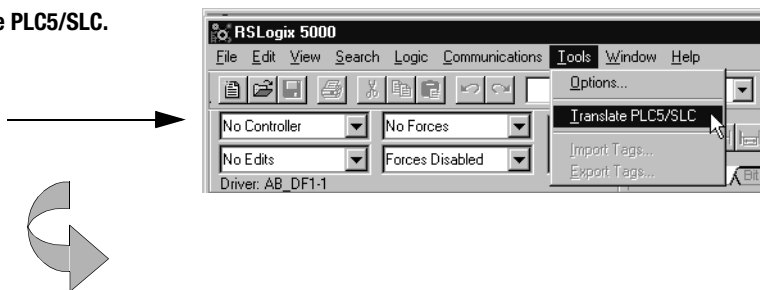
Le logiciel A.I.5 utilise ces extensions de fichier pour les fichiers de programme : .ADR, .CEI, .CET, .CFG, .DSC, .IO2, .IO4, .PRF, .RCK, .RPD, .RPI, .SYM, .X5, .XRF, et .XRI.

Conversion d'un programme PLC-5 ou SLC 500

Une fois que vous avez obtenu le fichier texte ASCII du fichier programme PLC-5 ou SLC 500, vous pouvez convertir le programme en son équivalent Logix5550.

A l'aide du logiciel de programmation RSLogix5000 :

1. Sélectionnez Outils → Traduire PLC5/SLC.



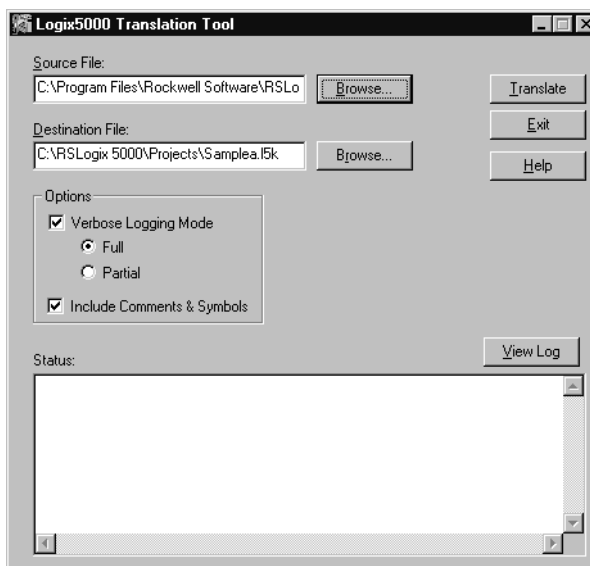
2. Sélectionnez le fichier texte à convertir.

Sélectionnez le fichier texte à convertir.
Il doit avoir une extension .PC5 ou .SLC

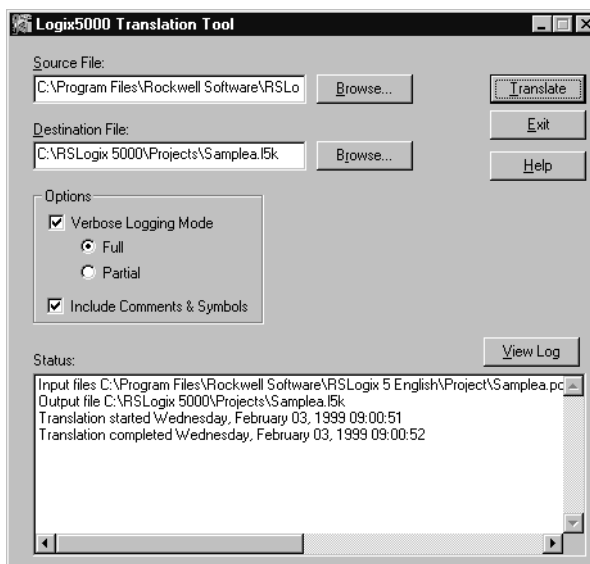
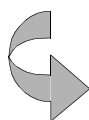
Par défaut, le logiciel indique le répertoire *RSLogix5000\Project* comme destination. Vous pouvez sélectionner ou entrer un autre répertoire de destination.

Remarque : Les deux fichiers .PC5/.SLC et .TXT doivent se trouver dans le même répertoire pour la tâche de conversion.

Sélectionnez les options de conversion.



Cliquez sur Traduire.



Cette boîte de dialogue affiche l'état du processus de conversion.

Cliquez sur Afficher journal pour visualiser le fichier journal.

Les options de conversion suivantes sont disponibles :

Option :	Description :
Mode Enregistrement avec explications	Sélectionnez cette option pour consigner tous les messages de toutes les catégories dans un fichier journal. Dans les autres cas, seulement un petit nombre de messages d'état ainsi que tous les messages relatifs à des questions seront consignés dans le fichier journal. Sélectionnez Complet pour que tous les messages de toutes les catégories figurent dans le fichier journal, en plus des textes supplémentaires descriptifs. Sélectionnez Partiel pour que tous les messages de toutes les catégories figurent dans le fichier journal, sans les textes supplémentaires descriptifs.
Avec commentaires et symboles	Sélectionnez cette option pour préciser si la documentation doit être incluse dans la conversion. Cette option est activée par défaut, de sorte que le fichier de commentaires ASCII .TXT est traité avec le fichier programme du processeur. Important : Le fichier .TXT doit exister pour que le processus de conversion fonctionne lorsque cette option est activée. Important : Les deux fichiers .PC5/.SLC et .TXT doivent se trouver dans le même répertoire.

Visualisation des résultats de conversion

L'outil de conversion crée un fichier import/export complet (extension .L5K) que vous pouvez ensuite importer dans un projet Logix5550.

Pour plus d'informations sur le contenu d'un fichier import/export Logix5550, reportez-vous à *Automate Logix5550 Manuel de référence Import/Export*, publication 1756-6.8.4FR.

Une fois le processus de conversion terminé, le fichier import/export a le format suivant :

```
CONTROLLER <Controller Name>

TAG
    (* Tous les points, alias et descriptions annexes se
    trouvent ici. *)
END_TAG

PROGRAM Continuous (MAIN := mcpMain)

    ROUTINE mcpMain
        JSR <Routine Name>;
        %% Un nombre supérieur d'appels JSR pourraient
        apparaître en fonction du type de processeur.
    END_ROUTINE

    %% Un sous-programme est créé pour chaque programme à
    relais qui s'exécute.

    ROUTINE <Routine Name>
        (* Un programme hérité converti *)
    END_ROUTINE

END_PROGRAM
```

```

PROGRAM Sti (MAIN := <Routine Name>)

    ROUTINE <Routine Name>
        (* Un programme hérité converti *)
    END ROUTINE

    %% Un sous-programme est créé pour chaque programme à
    relais qui s'exécute.
END PROGRAM

TASK Continuous (MODE := CONTINUOUS, WATCHDOG := 500)
    Continuous;
END_TASK

TASK Sti (MODE := PERIODIC, RATE := <Rate>, WATCHDOG := 500)
    Sti;
END_TASK

END_CONTROLLER

```

Les composants du fichier import/export converti sont :

Composant :	Description :
CONTROLLER	Le processus de conversion crée une structure CONTROLLER. Le nom de l'automate découle de la commande PROJECT dans le fichier import/export PLC-5. Si le nom de l'automate est identique à une instruction ou un mot-clé quelconque dans le fichier import/export PLC-5, le processus de conversion ajoute _DUP au nom de l'automate.
TAG	Le processus de conversion crée une structure TAG d'accès automate. Tous les points et alias sont placés dans cette structure TAG globale.
PROGRAM Continuous	Le processus de conversion crée un programme (PROGRAM) désigné par Continuous. De ce programme dérivent tous les sous-programmes.
ROUTINE mcpMAIN	Le processus de conversion crée un sous-programme (ROUTINE) désigné par mcpMAIN. Le sous-programme contient les instructions JSR pour un ou plusieurs sous-programmes considérés comme étant des sous-programmes principaux. Les sous-programmes principaux sont déterminés par les données d'état du processeur PLC-5 qui identifient les programmes de commande principaux.
ROUTINE	Le processus de conversion crée un sous-programme (ROUTINE) pour chaque PLC-5 fichier programme. Il est possible que des appels JSR ou des informations d'état du processeur précisent qu'un même sous-programme est requis par plusieurs programmes. Si c'est le cas, le processus de conversion crée des sous-programmes doubles, un pour chaque programme qui requiert le sous-programme.
PROGRAM Sti	Le processus de conversion crée ce programme pour le programme STI, s'il existe, pour le processeur PLC-5.
TASK Continuous	Le processus de conversion crée une tâche (TASK) afin de préciser comment les programmes doivent être exécutés. Cette tâche est toujours continue et fait référence à Continuous PROGRAM.
TASK Sti	La conversion crée cette tâche afin d'exécuter le programme STI. Il s'agit d'une tâche périodique qui fait référence à Sti PROGRAM.

Visualisation du fichier journal

Chaque procédure de conversion génère un fichier journal de type ASCII. Ce fichier journal fournit un résumé du processus de conversion et contient des messages formatés qui décrivent les actions et les étapes qui ont été effectuées durant le processus de conversion. Le nombre et le type de messages dépendent des options que vous avez sélectionnées pour le processus de conversion. Ces messages sont consignés dans le fichier dans l'ordre dans lequel les actions de conversion afférentes apparaissent.

Important: Le fichier journal identifie les zones devant être examinées pour des problèmes éventuels.

Vous pouvez ouvrir le fichier journal à partir de l'outil de conversion en sélectionnant le bouton de commande Afficher journal. Vous pouvez également utiliser n'importe quel éditeur de texte standard Windows pour ouvrir le fichier journal. Le nom du fichier journal est le même que le nom de fichier de sortie, mais avec l'extension .LOG.

Les numéros de ligne figurant dans le fichier correspondent aux numéros de ligne dans le fichier programme PLC-5 ou SLC 500 exporté. Un éditeur capable d'afficher les numéros de ligne peut vous aider à vous retrouver dans le fichier programme PLC-5 ou SLC 500 exporté.

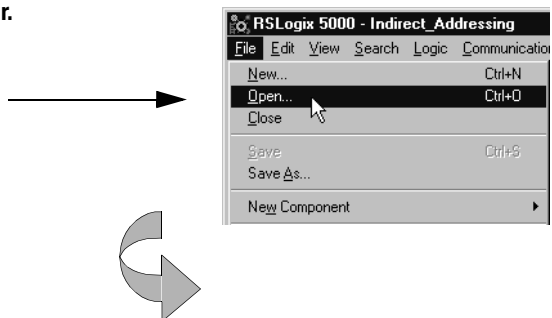
Pour savoir sous quelle forme apparaissent les messages dans le fichier journal, reportez-vous à l'annexe A.

Importation d'un fichier texte ASCII dans un projet Logix5550

Le fichier de sortie du processus de conversion est un fichier import/export Logix5550 portant l'extension .L5K. Importez ce fichier dans un projet Logix5550 à l'aide du logiciel de programmation RSLogix5000.

Utilisez le logiciel de programmation RSLogix5000.

1. Sélectionnez Fichier → Ouvrir.

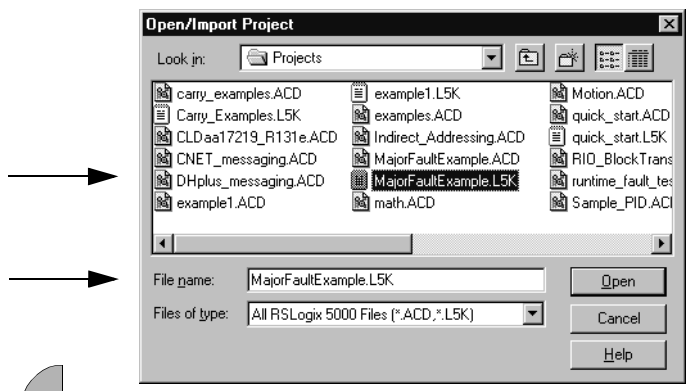


2. Sélectionnez le fichier texte. Le fichier texte doit avoir l'extension .L5K.

Sélectionnez le fichier texte à importer.
 Par défaut, le logiciel indique le répertoire/RSLogix5000\Project. Vous pouvez changer la sélection par défaut par le biais de Outils → Options.

Précisez le nom du fichier à importer.

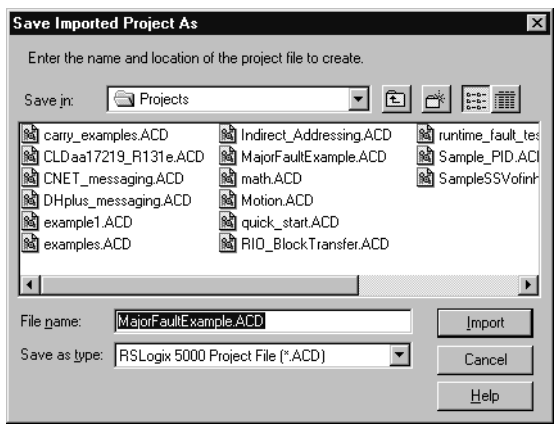
Cliquez sur Ouvrir.



3. Précisez le nom et l'emplacement du projet

Précisez l'emplacement du projet.

Précisez le nom du projet.



Cliquez sur Importer.

Corrections des instructions PCE

Le processus de conversion insère une instruction PCE (Possible Conversion Error) afin d'identifier les erreurs possibles. L'instruction PCE est constituée selon le format suivant (dans le fichier de texte ASCII) :

```
PCE( <Message>, <PCETag> )
```

Où :

Paramètre :	Description :
Message	Identifie le type d'erreur ou d'avertissement qui apparaît. Reportez-vous à l'annexe A pour la liste des messages de conversion possibles.
PCETag	Identifie l'erreur Chaque erreur de conversion reçoit un PCETag unique Le fichier import/export de sortie et le fichier journal comportent tous les deux l'instruction PCE. Vous pouvez chercher n'importe quel fichier en utilisant PCETag pour trouver l'information connexe.

Par exemple :

Une ligne du fichier import/export converti ressemble à ceci :

```
PCE( "3000", "pce00001" ), OTE( B3[0].0 );
```

La ligne correspondante dans le fichier journal ressemble à cela :

```
pce00001
QUES:3000 356:1024 MyProg:MyFirstRoutine:10
Output File reference is not valid
```

Où :

Valeur :	Correspond au :
3000	PCETag 3000
356	ligne 356 dans le fichier original, ASCII PLC-5 ou SLC 500
1024	ligne 1024 dans le fichier ASCII Logix5550 converti
MyProg	programme dans le projet Logix5550 importé
MyFirstRoutine	sous-programme dans le projet Logix5550 importé
10	numéro de ligne dans le projet Logix5550 importé

Une fois que vous avez importé le projet Logix5550 converti, il faut trouver chaque instruction PCE. Une instruction PCE met en évidence une possible erreur de conversion. Vous devez supprimer chaque instruction PCE et la remplacer par la logique appropriée et correcte.

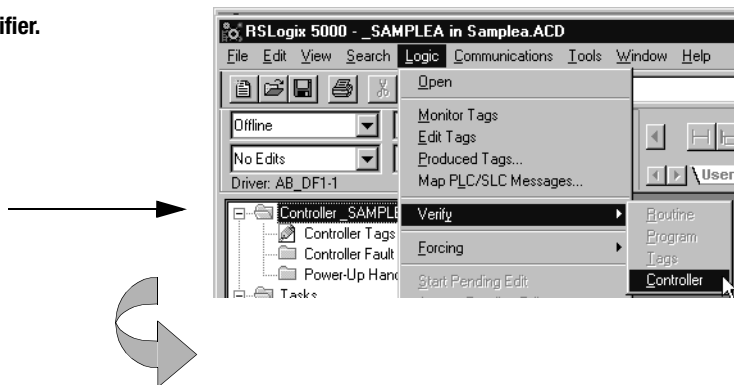
Les instructions PCE peuvent mettre ces erreurs possibles en évidence :

Une instruction PCE peut signifier :	Comment remédier à l'erreur :
L'instruction ne peut pas être convertie	Supprimez l'instruction PCE. Réécrivez le programme en vue de réaliser la fonctionnalité voulue.
Mot état S:24	Dans le processeur PLC-5, cet mot état contient le décalage d'indexation pour l'adresse indexée. Ce mot n'existe pas dans l'automate Logix5550. L'outil de conversion insère une instruction PCE pour chaque occurrence de S:24. Ainsi, une instruction COP par exemple peut contenir deux instructions PCE, une pour la source et l'autre pour la destination. Vérifiez que vous tenez compte de l'utilisation qui est faite de S:24 puis supprimez l'instruction PCE. De même, une instruction « +S24 » sera ajoutée aux mots source et destination de toutes les instructions sur fichiers. Vérifiez à nouveau l'utilisation qui est faite de S:24 dans l'instruction, puis supprimez « +S24. »
Pile faible	Supprimez l'instruction PCE. Utilisez une instruction GSV pour obtenir cette information d'état.
Dépassement mathématique	Supprimez l'instruction PCE. Utilisez le mot-clé S:V dans une instruction sur bits.

Localisation des instructions PCE

Vous pouvez localiser toutes les instructions PCE en vérifiant le contenu du programme.

1. Sélectionnez Logique → Vérifier.



Les résultats apparaissent sur le bas de l'écran :



Cliquez deux fois sur l'erreur afin d'accéder directement à la ligne.

Corrections des instructions UNK

L'outil de conversion transcrit certaines instructions PLC-5 et SLC 500 qui n'ont pas d'équivalent dans l'architecture RSLogix5000. Une fois que vous avez importé ces instructions dans un projet Logix5550, elles apparaissent en tant qu'instructions UNK. Vous devez supprimer chaque instruction UNK et la remplacer par le programme approprié et corrigé.

Vous pouvez également vérifier le programme pour localiser les instructions UNK, comme expliqué précédemment pour la localisation des instructions PCE.

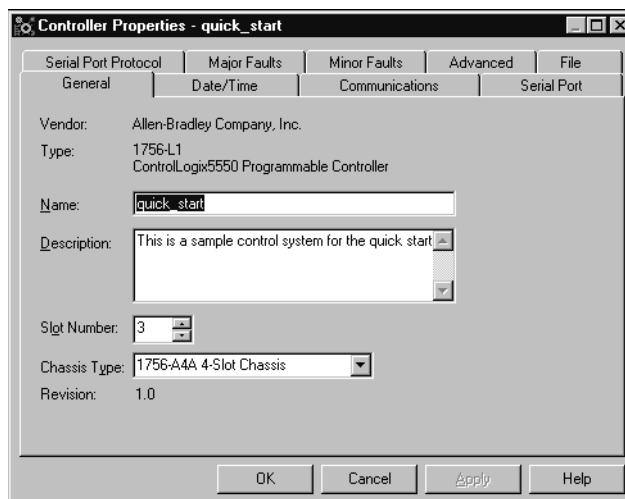
Configuration de l'automate et du châssis

Utilisez la boîte de dialogue Propriété de l'automate pour définir le type du châssis ainsi que le numéro d'emplacement de l'automate.

1. Placez le curseur sur le répertoire Automate.
2. A l'aide du bouton droit de la souris, sélectionnez Propriétés.



3. Configurez l'automate.

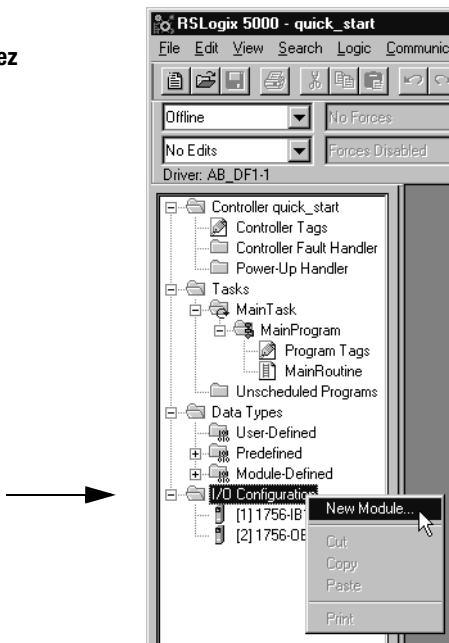


Précisez le numéro d'emplacement de l'automate.

Précisez le type du châssis.

Utilisez ensuite la fenêtre d'organisation de l'automate afin de déterminer les module d'E/S et autres dispositifs pour l'automate.

1. Sélectionnez Configuration des E/S.
2. A l'aide du bouton droit de la souris, sélectionnez Nouveau module.



Configuration des E/S

La base de la structure de fichier dans l'automate Logix5550 est le point. Pour faciliter la conversion, l'outil de conversion crée des points et des tableaux de points pour aligner et configurer les fichiers PLC-5. Par exemple :

Cette adresse PLC-5 :	Correspond à :
N7:500	N7[500]
N17:25	N17[25]
R6:100	R6[100]
I:002	I[2]

Remarquez que les points créés pour les E/S physiques (c'est-à-dire I.2) sont vides à la fin du processus de conversion. A l'aide du logiciel de programmation, additionnez tous les module d'E/S à l'arborescence de l'automate Logix5550. Programmez ensuite les instructions afin de faire correspondre les points d'E/S Logix5550 aux points convertis.

Ainsi, si vous ajoutez un module d'entrées à 16 points à l'emplacement 2 du châssis local, le logiciel de programmation crée les structures de points d'E/S suivantes :

- Local:1.C (informations de configuration)
- Local:1.Data (données de défaut et d'entrée)

Utilisez les instructions BTM, MOV ou COP pour configurer le mot Local:1.Data au point I2 créé par la procédure de conversion.

Une instruction MOV déplace un élément à la fois. Une instruction BTB déplace un groupe de bits, qui vous permet de tenir compte du décalage du bit de départ qui se produit lorsque vous configurez un type de données INT à un type de données DINT. Si des groupes d'E/S consécutifs sont configurés à des éléments consécutifs dans un tableau, une instruction COP est plus efficace.

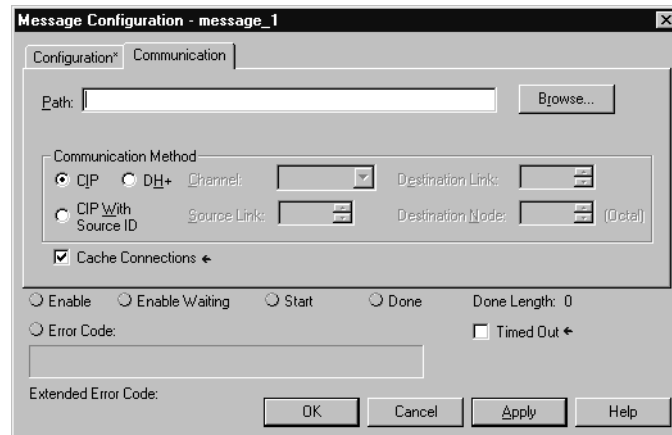
Ainsi, si vous configurez par exemple de I:000 à I:007 aux éléments de Local:1:I.Data[0] à Local:1:I.Data[7], utilisez :

```
COP
Source      I[0]
Destination Local:1:I.Data[0]
Longueur    8
```

Important: Si vous utilisez une instruction MOV, ne mélangez pas les types de données. Si vous mélangez les types de données, la conversion d'un type de données à un autre modifie le bit de signe, ce qui signifie que vous ne pouvez pas être sûr que le bit d'ordre supérieur est réglé comme il convient.

Pour plus d'informations sur la méthode utilisée par l'outil de conversion pour convertir le tableau de données PLC-5 ou SLC 500, reportez-vous au chapitre suivant.

Configuration complète de MSG L'outil de conversion ne convertit que partiellement les instructions MSG. Vous devez utiliser le logiciel de programmation RSLogix 5000 pour configurer chaque instruction MSG en complétant les informations relatives à l'onglet Communications.



Pour plus d'informations sur la configuration des instructions MSG, reportez-vous au manuel *Automate logix5550 – jeu d'instructions*, publication 1756-6.4.1FR. Ce manuel est disponible en format PDF sur le CDROM du logiciel de programmation RSLogix 5000.

Autres informations

Quelques remarques supplémentaires à garder à l'esprit :

- La base de temps des instructions de temporisation est fixée à 1 ms pour l'automate Logix5550. Le processus de conversion règle les présélections des temporisateurs PLC-5 et SLC 500 et valeurs cumulées en conséquence. Par exemple, un dispositif de temporisation PLC-5 ayant une base de temps de 0,01 s et une présélection de 20 est converti à une base de temps de 1 ms et une présélection de 200.
- Les commentaires relatifs aux instructions ne sont pas convertis.
- Le logiciel de programmation RSLogix5000 ne reconnaît pas les interruptions d'entrée programmables (DIIs/PIIs). Un programme DII/PII est converti en tant que programme dans la tâche continue.
- L'automate Logix5550 est basé sur 32 bits. Cela signifie que la plupart des instructions Logix5550 utilisent des mots à 32 bits, contrairement aux mots à 16 bits dans les processeurs PLC-5. Ce qui signifie que les instructions qui utilisent des masques sont susceptibles de fonctionner différemment après la conversion.
- Le processus de conversion crée des points d'alias pour les commentaires d'adressage. Ces alias sont ensuite utilisés à la place des points convertis.

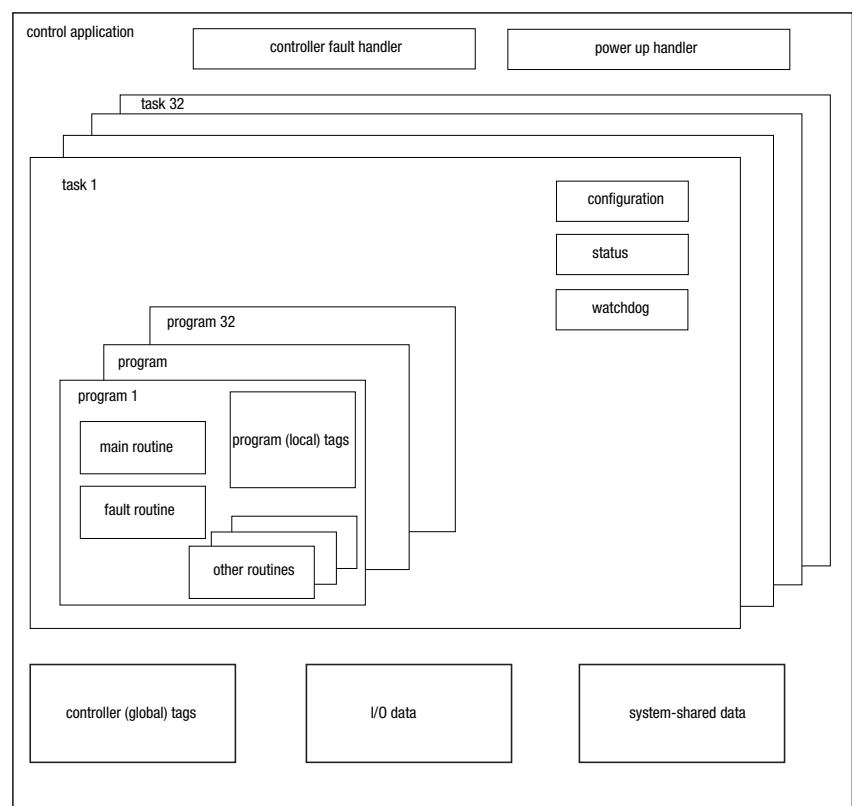
Les alias occupent de l'espace mémoire dans l'automate Logix5550, vous pouvez supprimer ceux que vous ne prévoyez pas d'utiliser. Utilisez le logiciel de programmation RSLogix5000 pour supprimer les alias une fois que vous avez importé le projet.

Conversion de structure de programme

Introduction

L'automate Logix5550 utilise un modèle d'exécution différent des autres processeurs PLC-5 ou SLC 500. L'automate Logix5550 utilise :

- des tâches pour configurer l'exécution de l'automate
- des programmes pour regrouper les données et la logique
- des sous-programmes pour envelopper le code exécutable écrit dans un seul langage de programmation



40012

Important: Actuellement, l'outil de conversion convertit uniquement des instructions à relais. Les fichiers SFC et en texte structuré ne sont pas convertis.

Une tâche fournit des informations de planification et de priorité pour un ou plusieurs programmes qui s'exécutent en fonction de critères spécifiques. Vous pouvez configurer des tâches continues ou périodiques.

Type de tâche :	Nombre accepté par l'automate Logix5550 :
continue	1
périodique	31 s'il y a une tâche continue 32 s'il n'y a pas de tâche continue

Une tâche peut avoir un maximum de 32 programmes séparés, chacun avec ses propres sous-programmes exécutables et points d'accès programme. Une fois une tâche lancée (activée), tous les programmes attribués à cette tâche s'exécutent selon l'ordre dans lequel ils sont groupés. Les programmes ne peuvent apparaître qu'une seule fois dans la fenêtre d'organisation de l'automate et ne peuvent pas être partagés par des tâches multiples. Les programmes planifiés dans une tâche sont exécutés complètement, du premier au dernier.

Un sous-programme est un jeu d'instructions logiques dans un langage de programmation unique, tel que le programme à relais. Les sous-programmes fournissent le code exécutable pour le projet dans l'automate. Un sous-programme est similaire à un fichier programme ou un sous-programme dans un processeur PLC ou SLC.

Chaque programme a un sous-programme principal. C'est le sous-programme qui est exécuté en premier lorsque l'automate lance la tâche associée et appelle le programme associé. Utilisez la logique, les instructions JSR par exemple, pour appeler d'autres sous-programmes.

Vous pouvez également définir un sous-programme de gestion des défauts de programme en option. L'automate exécute ce sous-programme s'il rencontre un défaut d'exécution d'instruction dans un des sous-programmes du programme associé.

Lorsque l'outil de conversion transcrit le programme PLC-5 ou SLC 500, considérez les structures de programme suivantes :

Etape de conversion :	Voir page :
Création d'une tâche continue	2-3
Conversion d'interruptions STI	2-3
Conversion d'interruptions DII et PII	2-4

Création d'une tâche continue

L'automate Logix5550 reconnaît une tâche continue qui fonctionne en mode auto-déclenchement. Elle se relance d'elle-même après chaque exécution. Cette tâche fonctionne comme tâche du niveau de priorité le plus bas de l'automate (un niveau en dessous de la tâche périodique la plus basse). Ceci signifie que toute tâche périodique interrompt la tâche continue.

L'outil de conversion crée automatiquement une tâche continue désignée par Continuous (continu) avec un chien de garde par défaut de 500 ms. Cette tâche constitue la tâche principale du projet converti.

Dans cette tâche continue, l'outil de conversion crée un sous-programme principal désigné par mcpMain, qui liste un ou plusieurs sous-programmes. Ce sous-programme principal contient les instructions JSR relatives à chacun des autres sous-programmes.

- Dans les processeurs PLC-5, le premier programme de commande maître (MCP) devient le sous-programme principal. L'outil de conversion utilise le fichier d'état du PLC-5 pour déterminer le premier MCP.
- Dans les processeurs SLC 500, le programme 2, qui est le programme à relais principal, devient le sous-programme principal.

Chacun des sous-programmes restant dans un programme est considéré comme sous-programme. Les sous-programmes sont locaux dans l'accès au programme (c'est-à-dire qu'ils sont uniquement accessibles par le sous-programme principal et par les autres sous-programmes au sein du programme courant). De ce fait, il est possible que des programmes à relais apparaissent en tant qu'un même sous-programme dans de nombreux programmes.

Le processus de conversion ne définit pas un sous-programme par défaut en tant que programme. Vous devez définir vous-même si nécessaire le sous-programme par défaut.

Conversion d'interruptions temporisées programmables (STI)

Le mot d'état 31 du processeur contient le numéro du programme à relais, s'il existe, qui est désigné pour être utilisé en tant qu'interruption temporisée programmable (STI). L'outil de conversion convertit ce fichier programme en sous-programme principal d'une tâche périodique désignée par Sti.

L'outil de conversion récupère l'intervalle STI du fichier d'état du processeur. Si nécessaire, l'outil de conversion convertit l'intervalle en une base de temps de 1 ms. Après la conversion, il vous faudra modifier les propriétés relatives à la tâche afin de préciser sa priorité.

Le bit d'état S:2/1 du processeur permet de désactiver et d'activer l'interruption temporisée programmable (STI). L'automate Logix5550 ne le permet pas. L'outil de conversion génère une instruction PCE s'il rencontre la moindre référence à S:2/1.

Conversion d'interruptions sur entrées (DII/PII)

L'automate Logix5550 ne reconnaît pas les interruptions sur entrées (DII ou PII). Si le processeur PLC-5 présente une PII ou le processeur SLC 500 une DII, l'outil de conversion la convertit en un sous-programme dans la tâche continue. Il faut modifier le programme Logix5550 pour appeler le sous-programme converti.

Le mot d'état 46 du processeur identifie le fichier programme devant être utilisé avec une interruption DII ou PII. L'outil de conversion génère une instruction PCE et la place dans le sous-programme DII/PII converti.

Conversion de données

Introduction

L'automate Logix5550 est basé sur une architecture à 32 bits contrairement à l'architecture à 16 bits des processeurs PLC-5 et SLC 500. En vue d'obtenir une conversion transparente et des performances optimales, de nombreuses valeurs de tableaux de données sont converties en valeurs à 32 bits (valeurs DINT).

Type de fichier PLC-5 ou SLC :	Type de tableau Logix5550 :	Base de numération :	Remarques :	Voir page :
O	INT	BINAIRE		3-3
I	INT	BINAIRE		3-3
S	INT	HEX		3-4
B	DINT	BINAIRE	La valeur à 16 bits est copiée à l'emplacement 32 bits et affectée d'un signe.	3-4
T	TIMER			3-5
C	COUNTER		Une instruction PCE est générée lorsque des bits de dépassement supérieur (.OV) et de dépassement inférieur (.UN) sont rencontrés.	3-7
R	CONTROL			3-8
N	DINT	DECIMAL	La valeur à 16 bits est copiée à l'emplacement 32 bits et affectée d'un signe.	3-9
F	REAL			3-9
A	INT	HEX		3-10
D	DINT	HEX	La valeur à 16 bits est copiée à l'emplacement 32 bits et complétée de zéro.	3-10
BT	MESSAGE			3-10
M0	INT			3-12
M1	INT			3-12
MG	MESSAGE			3-12
PD	PID			3-14
ST	N/A	N/A	Ce type de fichier n'est pas reconnu ; le tableau des données est ignoré.	3-14
CT	N/A	N/A	Ce type de fichier n'est pas reconnu ; le tableau des données est ignoré.	3-14

Identification par les fichiers import/export PLC-5 et SLC 500 des valeurs des tableaux des données

Les fichiers import/export PLC-5 et SLC 500 utilisent des instructions DATA pour identifier les types de fichier :

```
DATA <fichier_de_référence>:<numéro_du_dernier_élément>
<valeur_de_données>
```

Où :

Ce champ :	Spécifie :
<i>fichier_de_référence</i>	le type de fichier. Par exemple, N identifie un type de fichier de nombres entiers.
<i>numéro_du_dernier_élément</i>	la taille du fichier. Le processus de conversion utilise cette valeur pour déterminer le nombre d'éléments devant être placés dans le tableau utilisé pour ce fichier. Par exemple, DATA N7:9 signifie que le fichier 7 est un fichier de nombres entiers comportant 10 éléments.
<i>valeur_de_données</i>	le contenu du fichier. Par exemple : DATA N7:2 10 11 12 montre que le fichier 7 est un fichier de nombres entiers comportant trois éléments. Les valeurs de ces éléments sont : N7:0 10 N7:1 11 N7:2 12

Identification par les fichiers import/export Logix5550 des types de fichier

Les fichiers import/export Logix5550 utilisent des déclarations de point pour initialiser les valeurs. Par exemple :

Ce fichier de table données et éléments :	Convertis en :	Spécifient :
F8 avec 1 élément	REAL : = 3,25	une valeur unique, réelle
N7 avec 3 éléments	INT[3] : = {1,2,3}	un tableau d'entiers avec trois éléments
T4 avec 2 éléments	TIMER[2] : = {{1,2,3}, {4,5,6}}	un tableau comportant deux structures de temporisation ; chaque structure de temporisation comportant trois éléments

Conversion de données d'entrée (I) et de sortie (O)

Le processus de conversion des tableaux de données d'E/S tente de suivre la structure des tables-images des entrées et sorties dans le processeur PLC-5 et SLC 500. Pour ce faire, le processus de conversion crée un tableau unidimensionnel pour les données d'entrées et un tableau unidimensionnel pour les données de sortie. La taille des tables-images des entrées et sorties dans le processeur PLC-5 et SLC 500 détermine la taille de ces tableaux de conversion.

Le processus de conversion crée des tableaux unidimensionnels INT pour les fichiers d'entrée et de sortie. Les noms de points sont respectivement I et O. Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA O:177 0X0000 0X0000	point O type INT[128] (base de numération := binaire) := {16#0000, ... }
DATA I : 037 0X0000 0X0000	point I type INT[32] (base de numération := binaire) := {16#0000, ... }

Le processeur PLC-5, le processeur SLC 500 et l'automate Logix5550 utilisent des plans d'adressage différents pour les données d'E/S :

Automate :	Adressage des E/S :
Processeur PLC-5	base 8 (octale)
Processeur SLC 500	base 10 (décimale)
Automate Logix5550	base 10 (décimale)

Pour conserver l'adresse d'origine, le processus de conversion crée des points d'alias basés sur l'adresse matérielle. Par exemple :

Automate :	Adresse d'origine :	Adresse convertie :	Nom du point d'alias :
Processeur PLC-5	I:007	I[7]	I_07
	O:010	O[8]	O_010
	I:021/05	I[17].05	I_021_Bit05
	O:035/15	O[29].13	O_035_Bit015
Processeur SLC 500	I:007	I[7]	I_07
	O:010	O[10]	O_010
	I:021/05	I[21].05	I_21_Bit05
	O:035/15	O[35].15	O_35_Bit015

Conversion du type de fichier d'état (S)

Le processus de conversion crée un tableau unidimensionnel INT pour le fichier d'état S. Le point est désigné par S. Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA S:127	point S
0X0000 0X0000 ...	type INT[128] (base de numération := Hex.) := {16#0000, ... }
...	

Quelques exemples d'adresses d'état (S) et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
S:3	S[3]
S:1/15	S[1],15
S:24	S24

Important: Chaque adresse d'état S génère une instruction PCE.

Des considérations spécifiques doivent être prises en compte pour certaines données du fichier d'état :

Cette donnée d'état :	Est traitée comme suit :
Données d'état MCP	Le processeur PLC-5 reconnaît de 1 à 16 programmes de commande principaux (MCP). Chaque MCP utilise trois mots pour les données d'état. Les mots d'état de 80 à 127 contiennent cette information.
Données d'état STI	Le processeur PLC-5 évolué est également en mesure de reconnaître une interruption temporisée programmable. Le fichier d'état du processeur contient l'intervalle relatif à l'interruption ainsi que le numéro du fichier programme devant être exécuté. Le mot d'état 31 contient le nombre de fichiers programme ; le mot d'état 30 contient l'intervalle de temps de l'interruption.
Données d'état DII/PII	Les processeurs PLC-5 et SLC 500 reconnaissent une interruption sur entrée. Le mot d'état 46 contient le numéro du fichier programme devant être exécuté. L'automate Logix5550 ne reconnaît pas cette caractéristique. Si le fichier import/export contient les données d'état PII, le fichier programme PII est converti et placé en tant que sous-programme dans le programme continu. Le processus de conversion place également une instruction PCE dans le sous-programme converti afin d'indiquer que le sous-programme a été utilisé pour une PII.
Adressage indexé	Le mot d'état 24 contient l'index d'adressage en cours utilisé pour l'adressage indexé. L'automate Logix5550 n'utilise pas cette valeur d'index. Durant la conversion, le processus crée un point pour S24: S24 INT (base de numération := décimal) := <valeur>

Conversion du type de fichier binaire (B)

Lors de la conversion d'un fichier B, les valeurs 16 bits sont converties en valeurs 32 bits en remplissant les 16 bits supérieurs avec des zéros. Cette méthode de conversion permet aux instructions qui manipulent des fichiers B de fonctionner correctement, à l'exception des instructions BSL et BSR. Ces instructions nécessitent d'être corrigées, étant donné que les bits de signe qui auraient été déplacés dans un autre mot de 16 bits peuvent uniquement être déplacés dans les 16 bits supérieurs (ou inférieurs) du même mot de 32 bits dans l'architecture Logix5550.

Le processus de conversion crée un tableau unidimensionnel DINT pour le fichier B. Le point est désigné par B x (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA B3:15	point B3
153 227 ...	type DINT[16] (base de numération := binaire) := {153, 227, ... }
...	

Quelques exemples d'adresses B et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
B3.4/1	B3[4].1
B3/65	B3[4].1

Conversion du type de fichier Temporisateur (T)

Les temporisateurs dans les processeurs PLC-5 et SLC 500 consistent en une valeur de présélection à 16 bits, une valeur cumulée à 16 bits et une base de temps de 1 s ou 10 ms. Les temporisateurs dans l'automate Logix5550 consistent en une valeur de présélection à 32 bits, une valeur cumulée à 32 bits et une base de temps de 1 ms.

Le processus de conversion crée un tableau unidimensionnel de structures TIMER pour le fichier T. Le point est désigné par T x (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine. Chaque élément du tableau correspond à une structure TIMER, constituée de trois mots DINT de 32 bits. Voici une comparaison du temporisateur PLC-5/SLC 500 et du temporisateur Logix5550 :

Mot :	Bits PLC-5/SLC 500 :	Bits Logix5550 :	Mnémonique :	Description :
0	15	31	EN	activation
0	14	30	TT	temporisation en cours
0	13	29	DN	terminé
0	N/A	28	FS	première scrutation (utilisation de SFC)
0	N/A	27	LS	dernière scrutation (utilisation de SFC)
0	N/A	26	OV	dépassement supérieur
0	N/A	25	ER	erreur
1	N/A	s/o	PRE	valeur présélectionnée
2	N/A	s/o	ACC	valeur cumulée

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA T4:1 0 x E000 1 123	point T4 type TIMER[2] := {16#E0000000, 1000, 123000} Les valeurs .PRE et .ACC sont converties à partir d'une base de temps de 1 s.

Quelques exemples d'adresses T et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
T4:1	T4[1]
T4:1/15 T4:1/EN T4:1.0/EN	T4[1].EN
T4:1.1 T4:1.PRE	T4[1].PRE
T4:1.2 T4:1.ACC	T4[1].ACC

Important: Chaque adresse faisant référence à une valeur .PRE ou .ACC génère une instruction PCE. La base de temps d'une instruction PLC-5 ou SLC 500 étant susceptible de changer lors de la conversion en une instruction Logix5550, il est important d'examiner l'ensemble des références directes à ces paramètres afin de vérifier que ce programme fonctionne toujours correctement.

Règles de conversion du temporisateur

- Les valeurs .PRE et .ACC sont converties en valeurs équivalentes pour une base de temps de 1 ms.
- La première base de temps rencontrée pour un temporisateur individuel est utilisée pour la conversion des valeurs prédéterminées et cumulées chaque fois que le temporisateur intervient. Si le temporisateur intervient plusieurs fois, mais avec des bases de temps différentes, le processus de conversion insère une instruction PCE pour chaque occurrence de ce temporisateur.
- Chaque référence de programme faite à une valeur .PRE ou .ACC est remplacée par une instruction PCE.

Conversion du type de fichier Compteur (C)

Le processus de conversion crée un tableau unidimensionnel de structures COUNTER pour le fichier C. Le point est désigné par C_x (où *x* correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine. Chaque élément du tableau correspond à une structure COUNTER, constituée de trois mots DINT de 32 bits. Voici une comparaison du compteur PLC-5/SLC 500 et du compteur Logix5550 :

Mot :	Bits PLC-5/SLC 500 :	Bits Logix5550 :	Mnémonique :	Description :
0	15	31	CU	Comptage progressif
0	14	30	CD	Comptage dégressif
0	13	29	DN	terminé
0	12	28	OV	dépassement supérieur
0	11	27	UN	dépassement inférieur
1	N/A	N/A	PRE	valeur prédéterminée
2	N/A	N/A	ACC	valeur cumulée

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA C5:4	point C5
0xF800 500 0	type COUNTER[5] := {{16#F8000000, 500, 0}, ... }
...	

Les valeurs .PRE et .ACC ne sont soumises à aucune manipulation spécifique durant la conversion.

Quelques exemples d'adresses C et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
C:52	C5[2]
C5:2/15	C5[2].CU
C5:2/CU	
C5:2.0/CU	
C:52.1	C5[2].PRE
C5:2.PRE	
C:52.2	C5[2].ACC
C5:2.ACC	

Conversion du type de fichier Contrôle (R)

Le processus de conversion crée un tableau unidimensionnel de structures CONTROL pour le fichier R. Le point est désigné par Rx (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine. Chaque élément du tableau correspond à une structure CONTROL, constituée de trois mots DINT de 32 bits. Voici une comparaison de la structure de commande PLC-5/SLC 500 et de la structure de commande Logix5550 :

Mot :	Bits PLC-5/SLC 500 :	Bits Logix5550 :	Mnémonique :	Description :
0	15	31	EN	activer
0	14	30	EU	file d'attente
0	13	29	DN	terminé
0	12	28	EM	vide
0	11	27	ER	erreur
0	10	26	UL	déchargé
0	9	25	IN	désactivé
0	8	24	FD	trouvé
1	N/A	N/A	LEN	longueur
2	N/A	N/A	POS	position

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA R6:19	tag R6
0xFFFF00 0 0	type CONTROL[20] := {{16#FF000000, 0,0 }, ... }
...	

Les valeurs .LEN et .POS ne sont soumises à aucune manipulation spécifique durant la conversion.

Quelques exemples d'adresses R et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
R6:3	R6[3]
R6:3/15	R6[3].EN
R6:3/EN	
R6:3.0/EN	
R6:3.1	R6[3].LEN
R6:3.LEN	

Conversion du type de fichier de nombres entiers (N)

Le processus de conversion crée un tableau unidimensionnel INT pour le fichier d'état N. Le point est désigné par Nx (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA N7:99	point N7
153 227 ...	type INT[100] (base de numération := décimal) := {153, 227, ... }
...	

Quelques exemples d'adresses N et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
N7:0	N7[0]
N7:1/2	N7[1].2

Conversion du type de fichier de données à virgule flottante (F)

Le processus de conversion crée un tableau unidimensionnel REAL pour le fichier d'état F. Le point est désigné par Fx (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA F8:6	point F8
1.23 4.56 ...	type REAL[7] := {1.23, 4.56, ... }
...	

Quelques exemples d'adresses F et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
F8:3	F8[3]

Conversion du type de fichier ASCII (A)

Le processus de conversion crée un tableau unidimensionnel INT pour le fichier d'état A. Le point est désigné par Ax (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA A9:1	point A9
24930 25444	type INT[2] := {24930, 25444}

Quelques exemples d'adresses A et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
A9:4	A9[4]
A9:5/6	A9[5].6

Conversion du type de fichier à base décimale (D)

Le processus de conversion crée un tableau unidimensionnel INT pour le fichier d'état D. Le point est désigné par Dx (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA D10:2	point D10
256 512 768	type INT[3] := {256, 512, 768}

Quelques exemples d'adresses D et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
D10:0	D10[0]

Conversion du type de fichier transfert par blocs (BT)

Le processus de conversion crée une structure de MESSAGE individuelle pour chaque élément dans le fichier BT (et non pas un tableau de structures). Les points MESSAGE ne peuvent pas être des tableaux d'éléments. Le nom du point est MGx (où x correspond au numéro de fichier de données PLC-5 ou SLC 500).

Seule l'information relative au message local est convertie, c'est-à-dire le type de message, le message proprement dit ainsi que la longueur du message. Après la conversion, utilisez le logiciel de programmation pour configurer le message.

Par exemple, dans le fichier texte ASCII :

Cette instruction DATA :	Convertit en :
DATA BT9:1	point A_MSG_TEMPBUF2
...	type INT[1] (base de numération: = HEX,
...	Description := « Point tampon temporaire créé durant la conversion » := [0]
	point MG9_1
	type MESSAGE (DF1DHFLAG := 0, ProduceCount := 0 LocalTag := A_MSG_TEMPBUF2, RequestedLength := 10, MessageType := Bloc-transfert Lecture);

Le processus de conversion crée une mémoire tampon temporaire dans laquelle ou à partir de laquelle sont exécutées les instructions MSG Logix5000 réelles. Le processus de conversion utilise ensuite une instruction FAL pour copier les données vers/à partir du point local réel.

Quelques exemples d'adresses BT et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
BT11:5	MG11_5
BT11:5.RLEN	MG11_5.RLEN

Règles de conversion des blocs-transferts

- Le type de message, bloc-transfert lecture ou bloc-transfert écriture est sélectionné en fonction de l'instruction par bloc-transfert PLC-5.
- L'indicateur DF1DHFlag est toujours mis à 0.
- Le point local est réglé sur le point spécifié par l'instruction de bloc-transfert PLC-5.

Conversion des types de fichier M0 et M1

Le processus de conversion crée un tableau unidimensionnel INT pour M0x et M1x (où x correspond au numéro de fichier de données SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine.

Par exemple, dans le fichier de texte ASCII :

Cette instruction SLOT :	Convertit en :
SLOT 4 1747-SN SCAN_IN 32 SCAN_OUT 32	point M0_4
ISR 0 M0_SIZE 3300 M1_SIZE 3300 G_FILE 8	type INT[3300] () := [0, 0, ...] point M1_4 type INT[3300] () := [0, 0, ...]

Quelques exemples d'adresses M0/M1 et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
M0:0/1	M0_0[1]
M1:1/1	M1_1[1]

Conversion du type de fichier Message (MG)

Un fichier MG est converti en un point de type MESSAGE. Cependant, seule l'information relative au message local est convertie, c'est-à-dire le type de message, le message proprement dit ainsi que la longueur du message. Après la conversion, utilisez le logiciel de programmation pour configurer le message.

Le processus de conversion crée une structure de MESSAGE individuelle pour chaque élément dans le fichier MG (et non pas un tableau de structures). Les points MESSAGE ne peuvent pas être des tableaux d'éléments. Le nom du point est MGx (où x correspond au numéro de fichier de données PLC-5 ou SLC 500). Voici une comparaison de la structure MG PLC-5/SLC 500 et de la structure MESSAGE Logix5550 :

Type de message :	Type de message Logix5550 :
TYPEDREAD	Lecture PLC5
TYPEDWRITE	Ecriture PLC5
PLC3_WORDRANGEREAD	Lecture plage de mots PLC3
PLC3_WORDRANGEWRITE	Ecriture PLC3
PLC2_UNPROTECTEDREAD	Lecture non protégée PLC2
PLC2_UNPROTECTEDWRITE	Ecriture PLC2
SLC_TYPEDREAD	Lecture SLC
SLC_TYPEDWRITE	Ecriture SLC

Par exemple, dans le fichier de texte ASCII :

Cette instruction MSG :		Convertit en :
MG9:0		point A_MSG_TEMPBUF1
PLC-5 MSG		type INT[1] (base de numération:= HEX,
type de message	lecture non protégée PLC-2	Description := « Point tampon temporaire créé durant la conversion » : =[0]
adresse table		
de données locale	N7:0	point MG9_0
taille en éléments	1	
port	1A	type MESSAGE (DF1DHFLAG := 0,
adresse cible	10	LocalTag := A_MSG_TEMPBUF1,
station cible	2	RequestedLength := 1,
local		MessageType := PLC2 Unprotected Read);

Le processus de conversion crée une zone mémoire tampon temporaire dans laquelle ou à partir de laquelle sont exécutées les instructions MSG Logix5000 réelles. Le processus de conversion utilise ensuite une instruction FAL pour copier les données vers/à partir du point local réel. Voici les lignes d'un fichier .L5K qui sont générées par une simple instruction PLC-5 MSG, en fonction des déclarations ci-dessus :

```
XIO(MG9_0.EN) RES(R_MSG_CTL1) FAL(R_MSG_CTL1, 10, 0, ALL,
A_MSG_TEMPBUF1[0 + R_MSG_CTL1.POS], N7[10 +
R_MSG_CTL1.POS]) MSG(MG9_0);
```

Voici les lignes d'un fichier .PC5 qui a généré ce qui précède :

```
SOR XIO MG9:0/EN MSG MG9:0 EOR
```

Quelques exemples d'adresses MG et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
MG9:5	MG9_5
MG9:5.ERR	MG9_5.ERR

Règles de conversion de message

- Le type de message approprié est choisi, en fonction de l'instruction de message.
- L'indicateur DF1DHFlag est toujours mis à 0.

Conversion du type de fichier PID (PD)

Un fichier PD est converti en un point de type PID. Une instruction PID qui utilise un fichier de contrôle N n'est pas complètement converti. Dans ce cas, le fichier N est converti avec l'instruction PID, mais l'instruction échouera durant la vérification de programme. Utilisez alors votre logiciel de programmation pour configurer l'information de contrôle.

Le processus de conversion crée un tableau unidimensionnel de structures PID pour le fichier PD. Le point est désigné par PD x (où x correspond au numéro du fichier de données PLC-5 ou SLC 500). Le nombre d'éléments contenus dans le tableau converti est le même que le nombre d'éléments contenus dans le fichier de données d'origine. Chaque élément du tableau est une structure PID.

Par exemple, dans le fichier de texte ASCII :

Cette instruction DATA :	Convertit en :
DATA PD10:10	point PD10
256 0 0 0 0 0	type PID10[11].1 := {536870912, 0, 0, 0, 0, 0, 0,
0 0 0 0 0 0	0, 0, 0, 0, 0, 0.1, 0
0 0.1 0 0 0 0	0, 0, 0, 0, 0, 0, 0,
0 0 0 0 0 0	0, 0, 0, 0, 0, 0, [0,
0 0 15 10 1 0	0, 0, 0, 0, 0, 0, 0,
0 0 0 0 0 0	0, 0, 0, 0, 0, 0, 0]]
0 0 0 0 0 0	...
0 0 0 0 0 0	
0 0	
...	

Quelques exemples d'adresses PD et leurs équivalents Logix5550 :

Adresse d'origine :	Adresse convertie :
PD10:1	PD10[1]
PD10:1/15	PD10[1].EN
PD10:1/EN	
PD10:1.0/15	
PD10:1.2	PD10[1].SP

Conversion de type de fichier de chaînes de caractères ASCII (ST)

Le type de données ASCII n'est pas reconnu par l'automate Logix5550. Le processus de conversion ne convertit aucun fichier ST. Le processus de conversion remplace toute instruction qui fait référence à un type de données ST par une instruction PCE.

Conversion du type de fichier ControlNet (CT)

Le type de données ControlNet n'est pas reconnu par l'automate Logix5550. Le processus de conversion ne convertit aucun fichier CT. Le processus de conversion remplace toute instruction qui fait référence à un type de données CT par une instruction PCE.

Conversion de valeurs constantes

Le processus de conversion conserve les constantes. Le format des constantes converties varie légèrement afin de se conformer aux exigences de format de Logix5550.

Par exemple :

Ce type de constante :	Exemple PLC-5/SLC 500 :	Conversion :	Règle de conversion :
nombre entier	&N49	49	supprimer &N, le cas échéant
	-49	-49	copier le reste de la constante
Binaire	&B00110001	2#00110001	remplacer &B par 2# copier le reste de la constante
ASCII	&A1	16#0031	convertir en constante hex
	&Amx	16#6D78	
Hexadécimal	&H0031	16#0031	remplacer &H, 0x ou 0X par 16# copier le reste de la constante
	0x 0032	16#0032	
	0X0033	16#0033	
DCB	&D0049	16#0031	convertir en constante hex
Octal	&O61	8#61	remplacer &O par 8# copier le reste de la constante
Nombre à virgule flottante	-12.34E-12	-12.34E-12	cette syntaxe est complètement compatible copier la constante telle qu'elle est
	3,45	3,45	

Conversion d'adresses indirectes

L'adressage indirect intervient lorsqu'une partie d'adresse est remplacée par une référence à une autre adresse. Les processeurs PLC-5 et SLC 500 peuvent utiliser une référence d'adresse pour définir ces parties d'adresse :

- numéro de fichier
- mot ou numéro d'élément
- nombre de bits (uniquement pour les adresses de type B)

L'outil de conversion reconnaît les adresses indirectes, sauf s'il s'agit d'une caractéristique de tableau. Les caractéristiques de tableau sont converties en alias.

Par exemple :

Type	Exemple PLC-5/SLC 500 :	Conversion	Règle de conversion :
Numéro de fichier	N[N7:0]:5	N/A	L'outil de conversion ne peut pas convertir un numéro de fichier indirect. Une instruction PCE est générée.
Mot ou numéro d'élément	N12:[N7:0]	N12[N7_0]	N7:0 convertit en point de tableau N7[0]. L'alias N7_0 remplace l'adresse indirecte.
	N12:[T4:1.PRE]	N12[T4_1_PRE]	T4:1.PRE convertit en point de tableau T4[1].PRE. L'alias T4_1_PRE remplace l'adresse indirecte.
Numéro de bit	B3/[N7:0]	B3[N7_0 / 16].[N7_0 AND 15]	Le processus de conversion doit convertir correctement le mot et le bit dans ce mot. L'alias N7_0 remplace l'adresse indirecte.

Conversion de l'adressage indirect sur le numéro de fichier

L'adressage indirect sur le numéro de fichier peut être réellement implémenté après le processus de conversion, si les fichiers de données d'origine sont consécutifs. Par exemple, un processeur PLC-5 dispose de 5 fichiers programme contenant chacun une « recette » de traitement thermique.

Élément :	Description :
0	numéro de recette
1	chaleur segment 1 : temps exprimé en minutes
2	chaleur segment 1 : température en F°
3	chaleur segment 2 : temps exprimé en minutes
4	chaleur segment 2 : température en F°
5	temps de refroidissement à la température ambiante en minutes

Dans le fichier de texte ASCII :

DATA N10:5

0 5 350 15 200 60

DATA N11:5

1 10 400 25 300 15

DATA N12:5

2 5 500 20 350 90

DATA N13:5

3 50 300 120 150 90

DATA N14:5

4 10 700 30 500 240

Ces données sont converties en :

N10 : DINT[6] (base de numération:= décimal):= [0, 5, 350, 15, 200, 60];

N11 : DINT[6] (base de numération:= décimal):= [1, 10, 400, 25, 300, 15];

N12 : DINT[6] (base de numération:= décimal):= [2, 5, 500, 20, 350, 90];

N13 : DINT[6] (base de numération:= décimal):= [3, 50, 300, 120, 150, 90];

N14 : DINT[6] (base de numération:= décimal):= [4, 10, 700, 30, 500, 240];

Utilisez un éditeur de texte pour modifier ces fichiers de nombres entiers en un tableau bidimensionnel :

RECETTES : DINT[6, 6] (base de numération:= décimal):= [0, 5, 350, 15, 200, 60,

1, 10, 400, 25, 300, 15,

2, 5, 500, 20, 350, 90,

3, 50, 300, 120, 150, 90,

4, 10, 700, 30, 500, 240];

Supposons qu'il y ait une référence à une adresse indirecte à N[N7:0]:0 pour la lecture du numéro de recette. Dans le projet converti, utilisez RECETTES[N7_0, 0], où N7_0 correspond à la forme convertie de N7:0. Il vous faut vérifier le contrôle de limitation de zones adressables, parce que les numéros de fichier d'origine sont compris entre 10 et 14, mais que le premier index du tableau bidimensionnel est compris dans une plage de allant de 0 à 4.

Conversion d'adresses indexées

Les adresses indexées dans les processeurs PLC-5 et SLC 500 se distinguent par le fait qu'elles sont précédées d'un caractère #.

Conversion d'adresses indexées commandées par le mot d'état du processeur S:24

Le mot d'état S:24 du processeur contient la valeur d'indexation courante devant être ajoutée à la référence d'adressage. Le processus de conversion ajoute la valeur de S:24 aux valeurs indexées qu'elle convertit et place une instruction PCE dans le fichier import/export de sortie.

Par exemple :

Cette adresse :	Convertit en :
# N7:2	N7[2 + S24]

Conversion d'adresses indexées qui spécifient des données dans les fichiers (tableaux Logix5550)

Les adresses indexées sont également utilisées avec des instructions de fichier pour traiter des fichiers de données. Ces instructions utilisent une structure CONTROL afin de déterminer la valeur d'indexation – la position en cours au sein du fichier.

L'automate Logix5550 enregistre les données dans des tableaux et non pas dans des fichiers. Les adresses indexées des instructions de fichier PLC-5 et SLC 500 sont converties en points de tableau, sans ajout de la valeur du mot d'état S:24.

Par exemple :

Cette instruction :	Convertit en :
AVE # N10:0 N11:0 R6:0 6 0	AVE(N10[0], 0, N11[0], R6[0], 6, 0)

Conversion de symboles

Le processus de conversion convertit un symbole en une description.

Le fichier import/export PLC-5 et SLC 500 utilise les instructions SYM pour identifier les symboles :

`SYM <référence_d'adresse> <libellé>`

Où :

Le champ :	Spécifie :
<i>référence_d'adresse</i>	l'adresse Le processus de conversion crée un point qui correspond à l'adresse réelle.
<i>libellé</i>	le texte du symbole Le processus de conversion convertit le texte du symbole en une description.

Les processeurs PLC-5 et SLC 500 reconnaissent quelques formats de symboles qui ne sont pas pris en compte par l'automate Logix5550. Dans ces cas, le processus de conversion modifie le texte du symbole.

Par exemple :

Point Logix5550 :	Instruction SYM :	Point modifié :
N7 : INT[9] (base de numération := décimal)	SYM N7:2 Kitty	N7:INT[9] (base de numération := décimal, commentaire[2] := «Kitty»)
B3 : INT[5] (base de numération := binaire)	SYM B3:4/5 Puppy	B3:INT[5] (base de numération := binaire, commentaire[4].5 := «Puppy»)
T4 : TIMER[2]	SYM T4:0 Ducky SYM T4:1 2ndDuck	T4 : TIMER[2] (commentaire[0] := «Ducky», commentaire[1] := «_2ndDuck»)
N/A	SYM N[N7:0]:0 Pig	Ce format d'adresse n'est pas pris en compte dans le processus de conversion. Aucun point n'est créé.

Si une référence d'adresse dispose à la fois d'un symbole et d'un commentaire d'adressage, le processus de conversion concatène le symbole à la fin du commentaire d'adressage.

Conversion des commentaires d'adresse

Le processus de conversion transforme les commentaires d'adresse en descriptions.

Le fichier import/export PLC-5 et SLC 500 utilise les instructions AC pour identifier les commentaires d'adresse :

```
AC [mot_clé_de_formatage] <référence_d'adresse>
<"texte_de_commentaire">
```

Où :

Le champ :	Spécifie :
<i>mot_clé_de_formatage</i>	le format du texte de commentaire. Les processeurs PLC-5 et SLC 500 reconnaissent les commandes de formatage pour les textes de commentaires. Le processus de conversion ignore ces mots clés de formatage.
<i>référence_d'adresse</i>	l'adresse Le processus de conversion crée un point qui correspond à l'adresse réelle.
<i>libellé</i>	le texte de commentaire Le processus de conversion transforme le texte de commentaire en une description.

Par exemple :

Point Logix5550 :	Instruction AC :	Point modifié :
N7 : INT[9] (base de numération := décimal)	AC N7:2 Kitty	N7 : INT[9] (base de numération := décimal, commentaire[2] := « Kitty »)
B3 : INT[5] (base de numération := binaire)	AC B3:4/5 Puppy	B3 : INT[5] (base de numération := binaire, commentaire[4].5 := « Puppy »)

Si une référence d'adresse dispose à la fois d'un symbole et d'un commentaire d'adressage, le processus de conversion concatène le symbole à la fin du commentaire d'adressage.

Conversion des instructions PLC-5

Introduction

Ce chapitre explique comment l'outil de conversion transcrit les instructions individuelles. Les zones devant être prises en considération sont identifiées aux endroits appropriés.

Important: Actuellement, l'outil de conversion transcrit uniquement les instructions à relais. Les fichiers SFC et en texte structuré ne sont pas convertis.

Règles de conversion

Lors de la conversion des instructions, l'outil de conversion suit les règles suivantes :

- Les paramètres PLC-5 et SLC 500 utilisent 16 bits. Ils seront étendus à 32 bits pour les paramètres Logix5550.
- Les constantes sont converties au format binaire.
- Toutes les références à S:0/0, S:0/1, S:0/2 et S:0/3 sont respectivement remplacées par les mots clés Logix5550 S:C, S:V, S:Z et S:N.
- Chaque référence aux bits .OV et .UN d'un type de fichier COUNTER génère une instruction PCE.
- Toute constante qui représente un port série est toujours convertie à 0, le port série Logix5550.
- Si un mélange de types de données intervient entre les types nombre entier et réel, un message de conversion de type est enregistré dans le fichier journal de conversion et une instruction PCE est générée dans le fichier import/export de sortie.
- Des messages de fichier journal sont insérés pour toutes les expressions converties afin d'encourager l'utilisateur à vérifier que la priorité des opérations a été correctement convertie.

Liste d'instructions

Le tableau suivant énumère les instructions PLC-5 et SLC 500 et inclut des commentaires là où cela s'avère nécessaire pour identifier les indications de conversion.

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
ABL	Test tampon ASCII de ligne	PLC-5 SLC 500	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.
			Contrôle	
ABS	Valeur absolue	SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Destination	
ACB	Nombre de caractères ASCII dans la mémoire tampon	PLC-5	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.
			Contrôle	
ACI	Conversion de chaîne ASCII en nombre entier	PLC-5 SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Destination	
ACL	Effacement mémoire tampon ASCII	SLC 500	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.
			Transmettre mémoire tampon	
ACN	Concaténation de chaînes ASCII	PLC-5 SLC 500	Source A	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Source B	
ACS	Cosinus d'arc	PLC-5 SLC 500	Source A	
			Destination	
ACT	Action SFC	PLC-5	N/A	Ignore en tant que partie de la section SFC.
ADD	Addition	PLC-5 SLC 500	Source A	
			Source B	
			Destination	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :	
AEX	Extraction d'une chaîne ASCII	PLC-5 SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.	
			Index		
			Nombre		
			Destination		
AFI	Toujours faux	PLC-5	N/A		
AGA	Flux AGA	PLC-5	N/A	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.	
AHL	Lignes d'échange réglage/réinitialisation ASCII	PLC-5 SLC 500	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.	
			Masque ET		Ne convertit pas S:24 pour l'indexation. Utilise la valeur .POS de la commande.
			Masque OU		Ne convertit pas S:24 pour l'indexation. Utilise la valeur .POS de la commande.
			Contrôle		
			Etat canal		
AIC	Conversion d'un nombre entier en chaîne ASCII	PLC-5 SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.	
			Destination		
AND	ET logique	PLC-5 SLC 500	Source A		
			Source B		
			Destination		
ARD	Lecture de caractères ASCII	PLC-5 SLC 500	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.	
			Destination		Ne convertit pas S:24 pour l'indexation. Utilise la valeur .POS de la commande.
			Contrôle		
			Longueur chaîne		
			Caractères lus		
ARL	Lecture de ligne de caractères ASCII	PLC-5 SLC 500	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.	
			Destination		Ne convertit pas S:24 pour l'indexation. Utilise la valeur .POS de la commande.
			Contrôle		
			Longueur chaîne		
			Caractères lus		

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
ASC	Recherche de chaîne ASCII	PLC-5 SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Index	
			Recherche	
			Résultat	
ASN	Sinus d'arc	PLC-5 SLC 500	Source	
			Destination	
ASR	Comparaison de chaînes ASCII	PLC-5 SLC 500	Source A	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Source B	
ATN	Tangente d'arc	PLC-5 SLC 500	Source	
			Destination	
AVE	Moyenne	PLC-5	Fichier	Ne convertit pas S:24 pour l'indexation.
			Destination	Insère un 0 en tant que dimension pour varier.
			Fichier de contrôle	
			Longueur	
			Position	
AWA	Ecriture avec ajout ASCII	PLC-5 SLC 500	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.
			Source	Ne convertit pas S:24 pour l'indexation. Utilise la valeur .POS de la commande.
			Contrôle	
			Longueur chaîne	
			Caractère envoyé	
AWT	Ecriture ASCII	PLC-5 SC500	Canal	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme. Le canal est mis à zéro.
			Source	Ne convertit pas S : 24 pour l'indexation. Utilise la valeur .POS de la commande.
			Contrôle	
			Longueur chaîne	
			Caractères envoyé	
BND	Fin de dérivation	PLC-5 SLC 500	N/A	Convertit en crochets fermant (]).
BRK	BRK	PLC-5	N/A	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
BSL	Décalage binaire à gauche	PLC-5 SLC 500	Fichier	Ne convertit pas S:24 pour l'indexation. Enregistre un message dans le fichier journal.
			Fichier de contrôle	
			Adresse de bit	
			Longueur	
BSR	Décalage binaire à droite	PLC-5 SLC 500	Fichier	Ne convertit pas S:24 pour l'indexation. Enregistre un message dans le fichier journal.
			Fichier de contrôle	
			Adresse de bit	
			Longueur	
BST	Début de dérivation	PLC-5 SLC 500	N/A	Convertit en crochet ouvrant (]).
BTD	Distribuer les bits	PLC-5	Source	
			Bit source	
			Destination	
			Bit de destination	
			Longueur	
BTR	Bloc-transfert lecture	PLC-5	Rack	Ignore le paramètre de rack. Convertit l'instruction en instruction MSG et génère une instruction PCE.
			Groupe	Ignore le paramètre de groupe.
			Module	Ignore le paramètre de module.
			Bloc de contrôle	
			Fichier de données	Utilisez ce fichier de données afin de déterminer les attributs LocalTag. Ajoute les instructions RES et FAL pour effectuer les réglages pour la conversion de 16 bits en 32 bits.
			Longueur	Ignore le paramètre de longueur.
			Continu	Ignore le paramètre continu.
BTW	Bloc-transfert écriture	PLC-5	Rack	Ignore le paramètre de rack. Convertit l'instruction en instruction MSG et génère une instruction PCE.
			Groupe	Ignore le paramètre de groupe.
			Module	Ignore le paramètre de module.
			Bloc de contrôle	
			Fichier de données	Utilisez ce fichier de données afin de déterminer les attributs LocalTag. Ajoute les instructions RES et FAL pour effectuer les réglages pour la conversion de 16 bits en 32 bits.
			Longueur	Ignore le paramètre de longueur.
			Continu	Ignore le paramètre continu.

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
CIO	Transfert d'E/S ControlNet	PLC-5	Bloc de contrôle	Enregistre un message dans le fichier journal et génère une instruction PCE.
CLR	Effacer	PLC-5 SLC 500	Destination	
CMP	Comparer	PLC-5	Expression	Contrôlez l'expression convertie afin de vérifier si l'ordre de priorité est correct.
COP	Copier	PLC-5 SLC 500	Source	Ne convertit pas S:24 pour l'indexation. Si les types de source et de destination différent, enregistre un message dans le fichier journal.
			Destination	Ne convertit pas S:24 pour l'indexation.
			Longueur	
COS	Cosinus	PLC-5 SLC 500	Source	
			Destination	
CPT	Calculer	PLC-5 SLC 500	Destination	
			Expression	Contrôlez l'expression convertie afin de vérifier si l'ordre de priorité est correct.
CTD	Décompter	PLC-5 SLC 500	Compteur	
			Présélection	
			Accumulation	
CTU	Compter	PLC-5 SLC 500	Compteur	
			Présélection	
			Accumulation	
DCD	Décode 4 de 1 à 16	SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Destination	
DDT	Détection de diagnostic	PLC-5	Source	Ne convertit pas S:24 pour l'indexation. Suit l'instruction DDT avec les instructions MOV et FAL sur les branches parallèles afin de vérifier que les bits corrects sont traités.
			Référence	Ne convertit pas S:24 pour l'indexation.
			Resultat	Ne convertit pas S:24 pour l'indexation.
			Contrôler comparaison	
			Longueur	
			Position	
			Contrôler résultat	
			Longueur	
			Position	
DEG	Degrés	PLC-5 SLC 500	Source	
			Destination	
DFA	Avertisseur d'erreur de diagnostic	PLC-5	N/A	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
DIV	Diviser	PLC-5 SLC 500	Source A	
			Source B	
			Destination	
DTR	Transition de données	PLC-5	Source	
			Masque	
			Référence	
ENC	Code 1 de 16 à 4	SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Destination	
EOC	Fin de compression SFC	PLC-5	N/A	Ignore en tant que partie de la section SFC
EOR	Fin de ligne	PLC-5 SLC 500	N/A	Aucune action n'est entreprise.
EOT	Fin de transition	PLC-5	N/A	Ignore en tant que partie de la section SFC.
ESE	Fin de la section SFC	PLC-5	N/A	Ignore en tant que partie de la section SFC.
EOP	Fin de programme SFC	PLC-5	N/A	Ignore en tant que partie de la section SFC.
EQU	Egal à	PLC-5 SLC 500	Source A	
			Source B	
ERI	Erreur dans une instruction d'entrée	PLC-5	N/A	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
ERO	Erreur dans une instruction de sortie	PLC-5	N/A	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
ESI	Fin de branche simultanée SFC	PLC-5	N/A	Ignore en tant que partie de section SFC
FAL	Fichier arithmétique	PLC-5	Contrôle	
			Longueur	
			Position	
			Mode	
			Destination	
Expression	Utilise la valeur .POS pour l'indexation, pas S:24. Contrôlez l'expression convertie afin de vérifier si l'ordre de priorité est correct.			

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
FBC	Comparer des bits de fichiers	PLC-5	Source	Ne convertit pas S:24 pour l'indexation. Suit l'instruction DDT avec les instructions MOV et FAL sur les branches parallèles afin de vérifier que les bits corrects sont traités.
			Référence	Ne convertit pas S:24 pour l'indexation.
			Résultat	Ne convertit pas S:24 pour l'indexation.
			Contrôler comparaison	
			Longueur	
			Position	
			Contrôler résultat	
			Longueur	
			Position	
FFL	Chargement FIFO	PLC-5 SLC 500	Source	
			FIFO	Ne convertit pas S:24 pour l'indexation.
			Fichier de contrôle	
			Longueur	
			Position	
FFU	Déchargement FIFO	PLC-5 SLC 500	FIFO	Ne convertit pas S:24 pour l'indexation.
			Destination	
			Fichier de contrôle	
			Longueur	
			Position	
FLL	Remplissage d'un fichier	PLC-5 SLC 500	Source	
			Destination	
			Longueur	Ne convertit pas S:24 pour l'indexation.
FOR	Boucle FOR	PLC-5	Etiquette	Ne convertit pas le numéro d'étiquette. Vous devez modifier l'instruction FOR convertie. Voir page 4-18.
			Index	
			Valeur initiale	
			Valeur finale	
			Taille de l'incrément	
FRD	Depuis DCB	PLC-5 SLC 500	Source	
			Destination	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
FSC	Rechercher et comparer des fichiers	PLC-5	Contrôle	
			Longueur	
			Position	
			Mode	
			Expression	
GEQ	Supérieur ou égal	PLC-5 SLC 500	Source A	
			Source B	
GRT	Supérieur à	PLC-5 SLC 500	Source A	
			Source B	
HSC	Compteur rapide	SLC 500	Compteur	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Présélection	
HSD	Désactivation interruption HSC	SLC 500	Type	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Compteur	
			Présélection	
			Accumulation	
HSE	Activation interruption HSC	SLC 500	Compteur	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
HSL	Charge HSC	SLC 500	Compteur	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Source	
			Longueur	
IDI	Entrée immédiate de données	PLC-5	Décalage fichier de données	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
			Longueur	
			Destination	
IDO	Sortie immédiate de données	PLC-5	Décalage fichier de données	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
			Longueur	
			Destination	
IID	Désactivation interruption d'E/S	SLC 500	Emplacements	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
IIE	Activation interruption d'E/S	SLC 500	Emplacements	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
IIM	Entrée immédiate avec masque	SLC 500	Emplacements Masque Longueur	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
IIN	Entrée immédiate	PLC-5	RRG	
INT	Interruption d'E/S	SLC 500	N/A	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
IOT	Sortie immédiate	PLC-5	RRG	
JMP	Saut (à l'étiquette)	PLC-5 SLC 500	Etiquette	Convertit l'étiquette « n » à « label_n » étant donné qu'une étiquette Logix5550 ne peut pas être un nombre.
JSR	Saut à sous-programme	PLC-5 SLC 500	Programme à relais Paramètres d'entrées Paramètres de retour	Convertit en un nom de sous-programme.
LAB	Etiquette	PLC-5	N/A	Ignore en tant que partie de section SFC.
LBL	LBL	PLC-5 SLC 500	Etiquette	Convertit l'étiquette « n » à « label_n » étant donné qu'une étiquette Logix5550 ne peut pas être un nombre. Vous devez modifier l'instruction FOR convertie. Voir page 4-18.
LEQ	Inférieur ou égal	PLC-5 SLC 500	Source A Source B	
LES	Inférieur à	PLC-5 SLC 500	Source A Source B	
LFL	Chargement LIFO	PLC-5 SLC 500	Source LIFO Fichier de contrôle Longueur Position	Ne convertit pas S:24 pour l'indexation.
LFU	Déchargement LIFO	PLC-5 SLC 500	LIFO Destination Fichier de contrôle Longueur Position	Ne convertit pas S:24 pour l'indexation.
LIM	Test sur limites	PLC-5 SLC 500	Limite basse Test Limite haute	
LN	Logarithme naturel	PLC-5 SLC 500	Source Destination	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
LOG	Logarithme base 10	PLC-5 SLC 500	Source Destination	
MCR	Relais de contrôle maître	PLC-5 SLC 500	N/A	
MEQ	Masque égal	PLC-5 SLC 500	Opérande source Masque source Opérande de comparaison	
MOD	Modulo de division	PLC-5 SLC 500	N/A	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
MOV	Transfert	PLC-5 SLC 500	Source Destination	
MSG	Message	PLC-5 SLC 500	Type	Enregistre un message et génère une instruction PCE. Ajoute les instructions RES et FAL pour effectuer les réglages pour la conversion de 16 bits en 32 bits. Vous devez configurer les paramètres de communication MSG.
MUL	Multiplier	PLC-5 SLC 500	Source A Source B Destination	
MVM	Transfert avec masque	PLC-5 SLC 500	Opérande source Masque source Destination	
NEG	Annuler	PLC-5 SLC 500	Source Destination	
NEQ	Différent de	PLC-5 SLC 500	Source A Source B	
NOP	Pas d'opération	PLC-5	N/A	
NOT	NON logique (inversion)	PLC-5 SLC 500	Source Destination	
NSE	Branche suivante de sélection SFC	PLC-5	N/A	Ignore en tant que partie de section SFC.
NSI	Branche suivante simultanée SFC	PLC-5	N/A	Ignore en tant que partie de section SFC.
NXB	Dérivation suivante	PLC-5 SLC 500	N/A	Convertit en une virgule (,).
NXT	Suivant	PLC-5	Étiquette	Ne convertit pas le numéro d'étiquette. Vous devez modifier l'instruction FOR convertie. Voir page 4-18.
ONS	Front (sur ligne)	PLC-5	Bit source	
OR	OU logique	PLC-5 SLC 500	Source A Source B Destination	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
OSF	Impulsion front descendant	PLC-5	Bit de stockage	
			Bit de sortie	Combine un bit de sortie et un mot de sortie.
			Mot de sortie	
OSR	Impulsion front montant	PLC-5 SLC 500	Bit de stockage	Dans le cas d'une instruction SLC 500, convertit en une instruction ONS.
			Bit de sortie	Combine un bit de sortie et un mot de sortie.
			Mot de sortie	
OTE	Activation de sortie	PLC-5 SLC 500	Bit de destination	
OTL	Verrouillage de sortie	PLC-5 SLC 500	Bit de destination	
OTU	Déverrouillage de sortie	PLC-5 SLC 500	Bit de destination	
PID	PID	PLC-5 SLC 500	Bloc de contrôle	Vérifiez les paramètres de configuration PID.
			Valeur PV	
			Valeur tieback	
			Valeur CV	
RAC	Accumulateur remise à zéro du compteur rapide	SLC 500	Compteur	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Source	
RAD	Degrés en radians	PLC-5 SLC 500	Source	
			Destination	
REF	Référence SFC	PLC-5	N/A	Ignore en tant que partie de section SFC.
REF	Rafraîchissement des E/S	SLC 500	Canal 0	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Canal 1	
RES	Réinitialisation	PLC-5 SLC 500	Référence fichier	
RET	Retour (de sous-programme)	PLC-5 SLC 500	Paramètres de retour	
RPI	RAZ interruption en attente	SLC 500	Emplacements	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
RTO	Temporisateur rémanent activé	PLC-5 SLC 500	Temps	
			Base de temps	Convertit la base de temps en 1 milliseconde.
			Présélection	Remplace par « ? ». Vous devez modifier l'instruction RTO convertie.
			Accumulation	Remplace par « ? ». Vous devez modifier l'instruction RTO convertie.
SBR	Sous-programme	PLC-5 SLC 500	Paramètres d'entrées	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :
SCL	Echelle	SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Vitesse	
			Décalage	
			Destination	
SCP	Echelle avec paramètres	SLC 500	Entrée	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.
			Mini. entrée	
			Maxi. entrée	
			Mini. mis à l'échelle	
			Maxi. mis à l'échelle	
			Sortie mise à l'échelle	
SDS	Séquenceur dirigé intelligent	PLC-5	N/A	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
SEL	Branche de sélection SFC	PLC-5	N/A	Ignore en tant que partie de section SFC
SFR	RAZ SFC	PLC-5	Numéro de fichier	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.
			Redémarre à l'étape	
SIM	Branche SFC simultanée	PLC-5	N/A	Ignore en tant que partie de section SFC.
SIN	Sinus	PLC-5 SLC 500	Source	
			Destination	
SOC	Début de compression SFC	PLC-5	N/A	Ignore en tant que partie de section SFC.
SOP	Début de programme SFC	PLC-5	N/A	Ignore en tant que partie de section SFC.
SOR	Début de ligne	PLC-5 SLC 500	N/A	Début une sortie sur une nouvelle ligne.
SQI	Entrée séquenceur	PLC-5 SLC 500	Fichier	Ne convertit pas S:24 pour l'indexation.
			Masque	
			Source	
			Fichier de contrôle	
			Longueur	
			Position	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :	
SQL	Chargement séquenceur	PLC-5 SLC 500	Fichier	Ne convertit pas S:24 pour l'indexation.	
			Source		
			Fichier de contrôle		
			Longueur		
			Position		
SQO	Sortie séquenceur	PLC-5 SLC 500	Fichier	Ne convertit pas S:24 pour l'indexation.	
			Mask de destination		
			Destination		
			Fichier de contrôle		
			Longueur		
SQR	Racine carrée	PLC-5 SLC 500	Source		
			Destination		
			Fichier de contrôle		
			Longueur		
			Position		
SRT	Trier	PLC-5	Fichier de tri	Ne convertit pas S:24 pour l'indexation.	
			Fichier de contrôle		Insère un 0 en tant que dimension à faire varier.
			Longueur		
			Position		
			Destination		
STD	Ecart type	PLC-5	Fichier	Insère un 0 en tant que dimension à faire varier.	
			Destination		
			Fichier de contrôle		
			Longueur		
			Position		
STD	Désactivation Interruption temporisée programmable	SLC 500	N/A	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.	
STE	Activation Interruption temporisée programmable	SLC 500	N/A	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.	
STP	Etape SFC	PLC-5	N/A	Ignore en tant que partie de section SFC.	
STS	Début Interruption temporisée programmable	SLC 500	Fichier	Pas d'équivalent Logix5550. Enregistre un message dans le fichier journal et génère une instruction PCE.	
			Temps		
SUB	Soustraire	PLC-5 SLC 500	Source A		
			Source B		
			Destination		
SUS	Interruption	SLC 500	ID interruption	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.	

Instruction :	Nom :	Processeur :	Paramètre :	Précisions :	
SVC	Exécution des communications	SLC 500	Canal 0	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.	
			Canal 1		
SWP	Permutation	SLC 500	Source	Convertit, mais RSLogix5000 ne reconnaît pas cette instruction. Lorsque vous importez le fichier converti, elle apparaît en tant qu'instruction UNK et vous devez reprendre le programme.	
			Longueur		
TAN	Tangente	PLC-5 SLC 500	Source		
			Destination		
TND	Fin temporaire	PLC-5 SLC 500	N/A		
TOD	Conversion en DCB	PLC-5 SLC 500	Source		
			Destination		
TOF	Temporisateur au déclenchement	PLC-5 SLC 500	Temporisateur		
			Base de temps		Convertit la base de temps en 1 milliseconde.
			Présélection		Remplace par « ? ». Vous devez modifier l'instruction RTO convertie.
			Accumulation		Remplace par « ? ». Vous devez modifier l'instruction RTO convertie.
TON	Temporisateur à l'enclenchement	PLC-5 SLC 500	Temporisateur		
			Base de temps		Convertit la base de temps en 1 milliseconde.
			Présélection		Remplace par « ? ». Vous devez modifier l'instruction RTO convertie.
			Accumulation		Remplace par « ? ». Vous devez modifier l'instruction RTO convertie.
TRC	Transition SFC	PLC-5	N/A	Ignore en tant que partie de section SFC.	
UID	Désactivation interruption utilisateur	PLC-5	N/A		
UIE	Activation interruption utilisateur	PLC-5	N/A		
XIC	Examine si contact fermé (On)	PLC-5 SLC 500	Bit source		
XIO	Examine si contact ouvert (Off)	PLC-5 SLC 500	Bit source		
XOR	OU exclusif	PLC-5 SLC 500	Source A		
			Source B		
			Destination		
XPY	X à la puissance Y	PLC-5 SLC 500	Source A		
			Source B		
			Destination		

Instructions de conversion CAR

L'outil de conversion ne convertit pas les instructions CAR. L'instruction PCE est insérée dans le fichier import/export de sortie pour chaque instruction CAR rencontrée. Les instructions CAR incluent AGA (flux AGA), SDS (séquenceur dirigé intelligent), et DFA (avertisseur d'erreur de diagnostic).

Conversion des instructions FOR/NXT/BRK

La structure des commandes FOR/NXT/BRK a changé dans l'architecture Logix5550. Dans le processeur PLC-5, les instructions FOR et NXT encadrent une section de codes qui doivent être répétés plusieurs fois, alors que l'instruction BRK fournit un moyen de rompre la séquence de répétition de code. Dans l'architecture RSLogix, l'instruction FOR appelle un sous-programme donné un nombre de fois déterminé, de sorte qu'aucune instruction NXT n'est requise. L'instruction BRK fonctionne quasiment comme dans le processeur PLC-5.

Etant donné que ce changement d'architecture est significatif, vous devrez probablement envisager une restructuration de votre programme.

Messages de conversion

Introduction

Le processus de conversion crée un fichier journal qui renseigne sur l'avancement de la conversion. Le processus de conversion crée :

Message de conversion :	Voir page :
Message d'état	A-1
Message d'information	A-2
Message de question	A-3

Messages d'état

Les messages d'état enregistrent des événements importants durant le processus de conversion. Les messages d'état sont toujours consignés dans un fichier journal, quel que soit le mode d'enregistrement.

Les messages d'état utilisent le format suivant :

STAT:<code> <texte>

Où :

Paramètre :	Description :
<i>code</i>	identifie le message d'état
<i>texte</i>	décrit l'événement

Code :	Texte :	Quand est-il enregistré ?
200	Fichier d'entrées <nom_de_fichier>.	Avant que la moindre activité de conversion n'ait eu lieu.
201	Fichier de sortie <nom_de_fichier>.	Avant que la moindre activité de conversion n'ait eu lieu.
202	Conversion démarrée <date et heure>.	Avant que la moindre activité de conversion n'ait eu lieu.
203	Conversion terminée <date et heure>.	Une fois que toutes les conversions auront été effectuées.
204	Fichier non accepté ; fin.	Avant que la moindre activité de conversion n'ait eu lieu.
205	Instruction de modification de contrôle trouvée ; fin.	Lorsqu'une instruction SDZ, SIZ ou SRZ est rencontrée.
206	Impossible d'ouvrir <nom_de_fichier>.	Au moment où le fichier .TXT existant est ouvert.
207	Impossible de lire <nom_de_fichier>.	Au moment où le fichier .TXT existant est lu.
208	Impossible d'écrire dans <nom_de_fichier>.	Au moment où le fichier import/export Logix5550 est créé.
299	<état_général>.	N'importe quel autre message d'état non converti dans la liste précédente.

Messages d'information

Les messages d'information enregistrent les détails relatifs au processus de conversion. Ces messages d'état sont uniquement consignés dans un fichier journal lorsque vous sélectionnez le mode d'enregistrement avec explications.

Les messages d'information utilisent le format suivant :

```
INFO:<code>[<ligne_entrée>:<ligne_sortie>][<programme>[:<routine>[:<ligne>]]] <texte>
```

Où :

Paramètre :	Description :
<i>code</i>	identifie le message d'information
<i>ligne_entrée</i>	numéro de ligne dans le fichier original, ASCII PLC-5 ou SLC 500
<i>ligne_sortie</i>	numéro de ligne dans le fichier converti ASCII Logix5550
<i>programme</i>	programme dans le projet importé Logix5550
<i>routine</i>	sous-programme dans le projet importé Logix5550
<i>ligne</i>	numéro de ligne dans le projet Logix5550 importé
<i>texte</i>	décrit le message

Code :	Texte :	Quand est-il enregistré ?
1	L'expression IOA a été ignorée.	Chaque fois qu'une commande IOA est ignorée dans le fichier .TXT hérité.
2	L'expression IOC a été ignorée.	Chaque fois qu'une commande IOC est ignorée dans le fichier .TXT hérité.
3	L'expression IOS a été ignorée.	Chaque fois qu'une commande IOS est ignorée dans le fichier .TXT hérité.
4	L'expression FCI a été ignorée.	Chaque fois qu'une commande FCI est ignorée dans le fichier .TXT hérité.
5	L'expression FCN a été ignorée.	Chaque fois qu'une commande FCN est ignorée dans le fichier .TXT hérité.
6	L'expression RACK a été ignorée.	Chaque fois qu'une commande RACK est ignorée dans le fichier .TXT hérité.
7	L'expression SLOT a été ignorée.	Chaque fois qu'une commande SLOT est ignorée dans le fichier .TXT hérité.
8	La section SFC a été ignorée.	Chaque fois qu'une section SFC est ignorée dans le fichier .TXT hérité.
9	La section Texte structurée a été ignorée.	Chaque fois qu'une section ST est ignorée dans le fichier .TXT hérité.
10	L'expression de forçage a été ignorée.	Chaque fois qu'une commande FORCE est ignorée dans le fichier .TXT hérité.
11	L'expression filtres d'entrée a été ignorée.	Chaque fois qu'une commande INPUT FILTERS est ignorée dans le fichier .TXT hérité.
12	L'expression multi-point a été ignorée.	Chaque fois qu'une commande MULTI POINT est ignorée dans le fichier .TXT hérité.
13	L'expression Comptabilité PLC-2 a été ignorée.	Chaque fois qu'une commande PLC2 est ignorée dans le fichier .TXT hérité.
14	L'expression Comptabilité PLC-5 a été ignorée.	Chaque fois qu'une commande PLC5 est ignorée dans le fichier .TXT hérité.
15	L'expression Configuration Canal a été ignorée.	Chaque fois qu'une commande CHANNEL CONFIGURATION est ignorée dans le fichier .TXT hérité.
16	L'expression CONFIG a été ignorée.	Chaque fois qu'une commande CONFIG est ignorée dans le fichier .TXT hérité.

Code :	Texte :	Quand est-il enregistré ?
17	Type de données non accepté ; table de donnée ignorée.	Chaque fois qu'une commande DATA est rencontrée pour un type de fichier non reconnu (ST, CT ou SC).
18	La section En-tête Programme a été exécutée.	Chaque fois que la commande START est exécutée.
19	L'expression Emplacement a été exécutée.	Chaque fois que la commande SLOT est exécutée.
20	La section Projet a été exécutée.	Chaque fois que la commande PROJECT est exécutée.
21	La section Données a été exécutée.	Chaque fois que la commande DATA est exécutée.
22	La section Relais a été exécutée.	Chaque fois que la commande LADDER est exécutée.
23	Les types de fichier Source et Destination de l'instruction COP ne correspondent pas.	Chaque fois qu'une instruction COP ayant des types de source et de destination différents est rencontrée.
24	Une instruction RAC a été trouvée.	Chaque fois qu'une instruction RAC est rencontrée.
25	Valeurs de RAZ insuffisantes dans la table de donnée.	Chaque fois qu'une commande DATA est rencontrée avec des valeurs d'initialisation qui ne correspondent pas aux dimensions indiquées.
26	La taille de la table de données a été accrue pour accepter les valeurs de RAZ trouvées.	Chaque fois qu'une commande DATA est rencontrée avec des valeurs d'initialisation qui dépassent les dimensions indiquées.
27	Un commentaire d'adresse associé à un fichier a été ignoré.	Chaque fois qu'un commentaire d'adresse associé à un fichier est rencontré.
28	Le type de données BT ou MG n'a pas assez d'information, et a donc été ignoré.	Chaque fois qu'un type BT ou MG auquel manquent des attributs essentiels est rencontré.
29	Le message de type ASCCI a été ignoré.	Chaque fois qu'un message de type ASCII est rencontré.
30	L'instruction IO a été ignorée.	Chaque fois qu'une commande IO est rencontrée dans le fichier .TXT hérité.
99	Conversion annulée <date et heure>.	Chaque fois que l'utilisateur annule la conversion en appuyant sur la touche d'annulation.

Messages de question

Les messages de question mettent en évidence des éléments qui requièrent une attention particulière de l'utilisateur pour la suite du traitement. Les messages de question sont toujours consignés dans un fichier journal, quel que soit le mode d'enregistrement.

Les messages de question utilisent le format suivant :

QUES:<code>[[<fichier_type><ligne_entrée>:<ligne_sortie>][<programme>[:<routine>[:<ligne>]]] <texte>

Où :

Paramètre :	Description :
<i>code</i>	identifie le message d'information
<i>ligne_entrée</i>	numéro de ligne dans le fichier original, ASCII PLC-5 ou SLC 500
<i>ligne_sortie</i>	numéro de ligne dans le fichier converti ASCII Logix5550
<i>programme</i>	programme dans le projet Logix5550 importé
<i>sous-programme</i>	programme dans le projet Logix5550 importé
<i>ligne</i>	numéro de ligne dans le projet Logix5550 importé
<i>texte</i>	décrit le message

Code :	Texte :	Quand est-il enregistré ?
100	L'adresse renvoie à un type non accepté. Elle n'est pas convertie.	Chaque fois qu'une référence d'adressage du type STRING, ControlNet ou Etat SFC est rencontrée.
101	L'adresse renvoie à un champ de bit .VA de compteur. Elle n'est pas convertie.	Chaque fois qu'une référence à un champ .JA de compteur est rencontrée.
102	L'adresse renvoie à un champ de dépassement supérieur (.OV) ou inférieur (.UN). La conversion doit être validée.	Chaque fois qu'une référence à un champ .OV ou .UN de compteur est rencontrée.
103	Attention : La référence de fichier S n'est pas équivalente au fichier d'état.	Chaque fois qu'une référence au fichier S est rencontrée.
104	Instruction non acceptée.	Chaque fois qu'une instruction non reconnue par RSLogix5000 est rencontrée.
105	La désactivation de l'interruption temporisée programmable (STD) doit être validée.	Chaque fois qu'une instruction STD est rencontrée.
106	L'adresse renvoie à un numéro de fichier indirect. Elle n'est pas convertie.	Chaque fois qu'une référence d'adresse associée à un numéro de fichier indirect est rencontrée.
107	Le 6 ^{ème} paramètre de l'instruction FAL convertie peut comporter une erreur de priorité.	Chaque fois qu'une instruction FAL à laquelle est associée une expression est rencontrée.
108	La référence de l'adresse peut comporter un index incorrect. La conversion doit être validée.	Chaque fois qu'un index d'un tableau ne peut pas être déterminé.
109	L'instruction a été convertie, mais doit être validée.	Chaque fois qu'une instruction BTR, BTW ou MSG est convertie.
110	L'adresse renvoie au champ (.ACC) d'un temporisateur. La conversion doit être validée.	Chaque fois qu'une référence à un champ .ACC de temporisateur est rencontrée.
111	L'adresse renvoie au champ (.PRE) de présélection d'un temporisateur. La conversion doit être validée.	Chaque fois qu'une référence à un champ .PRE de temporisateur est rencontrée.

A

adresses
 indexées 3-18
 indirectes 3-16
 valeurs constantes 3-15
adresses indirectes 3-16
alias
 commentaires d'adressage 3-19
 symboles 3-18
automate 1-15

B

bloc-transfert
 instructions 4-5
blocs-transferts
 fichiers 3-10

C

châssis 1-15
commentaires d'adressage 3-19
complément des instructions
 MSG 1-17
configuration de l'automate et du
 châssis 1-15
configuration des E/S 1-16
conversion
 adresses indexées 3-18
 adresses indirectes 3-16
 aperçu 1-1
 commentaires d'adressage 3-19
 complément des
 instructions MSG 1-17
 configuration de l'automate et
 du châssis 1-15
 configuration des E/S 1-16
 DII 2-4
 données 3-1
 exportation d'un
 programme PLC-5 ou
 SLC 500 1-3
 fichier binaire (B) 3-4
 fichiers 3-18
 fichiers A 3-10
 fichiers BT 3-10
 fichiers Compteur (C) 3-7
 fichiers CT 3-14
 fichiers D 3-10

 fichiers d'entrée (I) 3-3
 fichiers d'état (S) 3-4
 fichiers de sortie (O) 3-3
 fichiers F 3-9
 fichiers journal 1-11
 fichiers M0/M1 3-12
 fichiers MG 3-12
 fichiers N 3-9
 fichiers PD 3-14
 fichiers R 3-8
 fichiers S 3-18
 fichiers ST 3-14
 fichiers Temporisateur (T) 3-5
 importation de fichiers
 convertis 1-12
 information 1-18
 instructions 4-1
 instructions CAR 4-16
 instructions de
 blocs-transferts 4-5
 instructions FOR/NXT/BRK 4-16
 instructions MSG 4-11
 instructions PCE 1-13
 instructions UNK 1-15
 messages A-1
 messages d'état A-1
 messages d'information A-2
 messages de question A-3
 options 1-9
 PII 2-4
 programme PLC-5 ou
 SLC 500 1-8
 résultats 1-9
 STI 2-3
 structure de programme 2-1
 symboles 3-18
 tâches continues 2-3
 valeurs constantes 3-15
 vérification 1-14
 voir conversion 1-1
conversion options 1-9

D

DII 2-4
données 3-1
données de fichier 3-18

E

exportation
 programme PLC-5 ou
 SLC 500 1-3
 utilisation d'un logiciel 6200 1-6
 utilisation du logiciel A.I. 1-7
 utilisation du logiciel RSLogix
 1-4
 extension L5K 1-1
 extension PC5 1-1
 extension RSP 1-5
 extension RSS 1-5
 extension TXT 1-3

F

fichier binaire (B) 3-4
 fichier de chaînes de caractères 3-14
 fichier de données à
 virgule flottante 3-9
 fichier Temporisateur 3-5
 fichiers A 3-10
 fichiers à base décimale 3-10
 fichiers ASCII 3-10
 fichiers binaires 3-4
 fichiers BT 3-10
 fichiers Compteur (C) 3-7
 fichiers ControlNet 3-14
 fichiers CT 3-14
 fichiers D 3-10
 fichiers d'entrée (I) 3-3
 fichiers d'état 3-18
 fichiers d'état (S) 3-4
 fichiers de compteur 3-7
 fichiers de contrôle 3-8
 fichiers de message 3-12
 fichiers de nombres entiers 3-9
 fichiers de sortie (O) 3-3
 fichiers F 3-9
 fichiers journal 1-11, A-1
 fichiers M0/M 3-12
 fichiers MG 3-12
 fichiers N 3-9
 fichiers PD 3-14
 fichiers PID 3-14
 fichiers R 3-8
 fichiers S 3-18
 fichiers ST 3-14

fichiers Temporisateur (T) 3-5
 formats de fichier
 L5K 1-1
 logiciel 6200 1-6
 logiciel A.I 1-7
 PC5 1-1
 RSP 1-5
 TXT 1-3
 formats de fichiers
 RSS 1-5

I

importation 1-12
 informations 1-18
 instructions 4-1
 instructions CAR 4-16
 instructions FOR/NXT/BRK 4-16
 instructions MSG 1-17, 4-11
 instructions PCE 1-13
 instructions UNK 1-15
 interruptions sur entrées
 voir DII/PII 2-4
 interruptions temporisées
 programmables
 voir STI 2-3

L

logiciel de programmation
 RSLogix 1-4
 logiciel programmation de
 Série 6200 1-6
 logiciel programmation de
 Série A.I 1-7

M

messages d'état A-1
 messages d'information A-2
 messages de question A-3
 modèle d'exécution 2-1

P

PII 2-4
 programmes 2-2

R

résultats
 fichiers journal 1-11, A-1
 structure de fichier 1-9
retravaillées
 instructions PCE 1-13
 instructions UNK 1-15

S

sous-programmes 2-2
STI 2-3
structure de programme 2-3
 aperçu 2-1
 DII 2-4
 modèle d'exécution 2-1
 PII 2-4
 programmes 2-2
 sous-programmes 2-2
 STI 2-3
 tâche 2-2
symboles 3-18

T

tableaux 3-18
tâches 2-2
tâches continues 2-3
type de fichier
 A (ASCII) 3-10
 BT (blocs-transferts) 3-10
 CT (ControlNet) 3-14
 D (à base décimale) 3-10
 F (virgule flottante) 3-9
 M0/M1 (spécifiquement) 3-12
 MG (message) 3-12
 PD (PID) 3-14
 S (état) 3-18
 ST (chaîne de caractères) 3-14
types de fichier
 fichiers d'état (état) 3-4
 N (de nombres entiers) 3-9
 R (de contrôle) 3-8
 T (temporisation) 3-5
types de fichiers
 I (entrée) 3-3
 O (sortie) 3-3

V

valeurs constantes 3-15
vérification 1-14

Rejoignez-nous sur : www.rockwellautomation.com

Quels que soient vos besoins, dans le monde entier, Rockwell fédère un ensemble de marques leaders en automatisation industrielle : Allen-Bradley et ses solutions de contrôle, Reliance Electric et ses systèmes de transmission de puissance, Dodge et ses produits de transmission mécanique, ainsi que Rockwell Software et ses logiciels. Rockwell Automation propose une approche unique et flexible pour aider ses clients à obtenir un avantage concurrentiel certain, avec l'aide de milliers de partenaires, distributeurs et intégrateurs système agréés à travers le monde.



Siège mondial : Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204 USA, Tél. : (1) 414 382-2000, Fax : (1) 414 382-4444
Siège européen : Rockwell Automation, Avenue Hermann Debroux, 46, 1160 Bruxelles, Belgique, Tél. : (32) 2 663 06 00, Fax : (32) 2 663 06 40
Belgique : Rockwell Automation, De Kleetlaan 2B, B-1831 Diegem, Tél. : (32) 2 716 84 11, Fax : (32) 2 725 07 24
Canada : Rockwell Automation, 135 Dundas Street, Cambridge, Ontario, N1R 5X1, Tél. : (1) 519 623-1810, Fax : (1) 519 623 8930
France : Rockwell Automation, 36, avenue de l'Europe, 78941 Vélizy Cedex, Tél. : 33 (0)1 30 67 72 00, Fax : 33 (0)1 34 65 32 33
Suisse : Rockwell Automation, Gewerbepark, CH-5506 Mägenwil, Tél. : (41) 62 889 77 77, Fax : (41) 62 889 77 66

