

Fiche AC25801-01/02 : TSX 47

Alimentation et processeur

par Louis Boudillon (ancien de la Télémécanique)

Les langages

Avec cette version de processeur, le concepteur dispose de 3 niveaux de langages (dénommés PL7/3) pour décrire les « traitements » à appliquer sur les « objets » (signaux d'E-S, variables internes logiques ou numériques, étapes-grafcet, temporisateur, compteur, etc...) ; ces objets sont typiques du métier de l'électricien ou de l'automaticien.

Ces 3 langages correspondent à la norme CEI 61131-3 (celle-ci décrit deux autres langages pour l'automaticien) :

1 - le « ladder » (LD de la norme) : les « expressions logiques » sont décrites, par le concepteur, sous forme de schémas électriques, organisés en « réseaux » (grille de N lignes et de M colonnes avec contacts, blocs-fonction, bobines,... reliés en série (fonction ET) ou en parallèle (fonction OU).

Les blocs-fonction permettent de « simuler » des organes électromécaniques, comme un compteur. Un bloc-fonction « Message » permet par exemple de déclencher « électriquement » l'impression d'un texte sur une imprimante

Ces réseaux s'affichent sur l'écran graphique de la console de programmation. Ils sont même animés (changement de couleur ou inversion video) en phase de mise au point ; certaines variables peuvent même être « forcées » par le programmeur (par exemple pour vérifier l'effet du changement d'état d'un capteur, sans que celui-ci n'évolue ; ou pour « tester » un signal de sortie)

L'enchaînement des « réseaux » est linéaire, du début à la fin, et relancé systématiquement, « simulant » ainsi le fonctionnement des schémas électriques. L'exécution des calculs est donc périodique : les valeurs des signaux d'entrée sont « échantillonnées » et les valeurs des signaux de sortie sont mises à jour en conséquence quelque temps après (quelques millisecondes).

2 - le littéral, (ST : Structured Text dans la norme) : les « expressions logiques » sont décrites, par le concepteur, sous forme de phrases textuelles, avec une syntaxe et des mots-clefs, analogues à un langage de type informatique. Bien sûr, tout autre type d'expression est possible, avec des « structures itératives » à plusieurs niveaux, mais sur l'ensemble des « objets », avec leurs spécificités, de l'automaticien (communs avec le LD, le premier niveau, y compris les blocs-fonction)

Il bénéficie des mêmes possibilités pour la mise au point (animation d'écran et forçage de variables), ainsi que d'autres « services », notamment des services documentaires.

Réseaux « ladder » et phrases « littérales » peuvent co-exister au sein d'une même application, découpée alors en « modules » homogènes dans leur langage.

3 - le Grafcet (SFC : Sequential Function Chart dans la norme) : il permet une description sous forme graphique, animée en mise au point, du fonctionnement désiré de l'application : enchaînement conditionnel « d'étapes », reliées par des « transitions », séries ou parallèles, avec des « actions » associées. Les transitions et actions sont décrites en LD ou ST.

Les évolutions entre étapes sont calculées en une seule fois par cycle de traitement, d'une manière transparente à l'utilisateur : seules les « transitions » et « actions » concernées (en fonction de l'état actuel) sont évaluées ou exécutées : ceci permet d'alléger les ressources du processeur.

En conséquence, le grafcet apporte non seulement une facilité dans la conception d'une application, mais il procure une amélioration en performances temporelles (par rapport à une programmation purement en LD ou littéral) ; il reste toutefois optionnel. mise en forme graphique, animée en mise au point, du fonctionnement désiré de l'application (enchaînement conditionnel d'états : les transitions, séries ou parallèles ; leurs actions associées). Les transitions et actions sont décrites en LD ou ST.

Le grafcet apporte non seulement une facilité dans la conception d'une application, mais il procure une amélioration en performances temporelles (réduction du TRL par rapport à une programmation purement en LD ou littéral)

Remarques :

Bien que le ladder bénéficie de « dérogations », comme les « sauts conditionnels », qui lui permettent de programmer sous forme de schémas électriques des structures algorithmiques, le littéral est mieux adapté aux fonctions évoluées assurées par un API moderne (conduite évoluée, aide à l'exploitation, statistiques de production, ...).

Le ST reste néanmoins accessible à un « électricien » (après une formation minimum), surtout avec les « services » que le constructeur (ici Télémécanique) a prévu, notamment pour les mises au point.

Dans les « grosses » applications, le concepteur structurera son application, outre les Grafcet, en modules programmés soit en LD, soit en littéral, suivant le besoin d'accessibilité par le personnel de maintenance. Par exemple, les « transitions » et les « actions » d'un grafcet sont indifféremment programmables en LD ou ST.

La tâche :

En pratique, elle comprend quatre phases, qui sont « exécutées » séquentiellement par le processeur :

- la première correspond à des « fonctions-système », programmées (en langage informatique) par le constructeur ; elles assurent un certain nombre de services (gestion des états du processeur, gestion du temps, communications avec la console de programmation, etc.) ; elles sont ignorées de l'utilisateur
- la seconde correspond à la « scrutation des signaux d'entrée » : une « image » des valeurs de ceux-ci est mise à jour dans la mémoire du processeur ; cette phase aussi est ignorée de l'utilisateur
- la troisième correspond à l'interprétation des « réseaux » eux-même : c'est donc le « programme de l'utilisateur »
- enfin, la quatrième correspond à « l'application des signaux de sortie » : l'image des valeurs de ceux-ci, calculées par les « réseaux », en fonction des « souhaits » du concepteur, est « convertie » en signal électrique ; cette dernière phase est également ignorée de l'utilisateur

Seule la phase « traitement » est donc connue de l'utilisateur, les autres sont transparentes (mais leur temps d'exécution doit être pris en compte). Le fonctionnement sur « images » permet de garantir la cohérence des informations en toutes circonstances (respect des propriétés fondamentales, utilisées par l'électricien pour « simplifier » ses schémas, telles que $a/a = 0$ et $b+/b = 1$, même après coupure-secteur).

La performance essentielle qualifiant le fonctionnement de l'API est le TRL : Temps de Réponse Logiciel. Il est à comparé au temps de réponse d'un circuit câblé équivalent (qui dépend de la technologie, généralement rapide) : c'est le temps maximum entre l'évolution d'un signal d'entrée et la variation du signal de sortie impliquée, calculée par le programme ; il doit permettre de respecter la « dynamique » de l'application (par exemple, la distance d'arrêt d'un mobile en mouvement). Le TRL est de deux fois le temps de cycle de la tâche.

Le multitâche :

Alors que les premiers API, et encore actuellement les tout-petits modèles, n'avait qu'un fonctionnement « simple », dit monotâche (le même cycle s'exécutait indéfiniment, à vitesse maximum dépendant de la « longueur » du programme (nombre et complexité des réseaux), ce TSX47-40 présente un fonctionnement « multi-tâches ».

L'application est structurée en 5 tâches, toutes ayant la même structure cyclique divisée en 4 phases décrites ci-dessus.

Ces tâches sont « lancées » sur événement (tâche la plus prioritaire), périodiquement (tâche rapide et tâche principale, la seule concernée par le Grafcet), ou sur demande (2 tâches auxiliaires, moins prioritaires). Chaque tâche dispose de ses propres entrée-sorties, que doit configurer le concepteur.

A chaque tâche correspond donc un TRL (Temps de Réponse Logiciel), Les besoins de performances des différents organes mécaniques ou opératifs guideront les choix du concepteur, le grafcet réduisant par nature le TRL de la tâche principale

Les modes de marche :

Par ailleurs, de nombreuses fonctions-système, à la disposition du concepteur, permettent de gérer les « modes de marche » d'une installation industrielle (marche automatique, marches manuelles, modes dégradés, gestion des pertes d'énergie, ...), facilitant ainsi l'exploitation et sa sûreté (sécurité et disponibilité).

Ces fonctions sont souvent en relation avec des fonctions console, celle-ci étant raccordée soit directement au processeur, soit par l'intermédiaire d'un réseau (ici TELWAY ou UNITELWAY ; FIPWAY ou ETHWAY sur d'autres modèles de la gamme).

Avec un TSX47, on se rapproche des fonctionnalités et des performances obtenues en Informatique Industrielle (surtout avec des architectures en réseaux), avec toutefois une mise en œuvre accessible aux exploitants industriels, peu formés aux techniques informatiques.

