

Post-processeur SMC

Pour AUTOMGEN V6 et automates SMC



CHAPITRE N°1.	3
INTRODUCTION	3
CHAPITRE N°2.	4
INSTALLATION	4
CHAPITRE N°3.	7
PRISE EN MAIN	7
CHAPITRE N°4.	9
GÉNÉRALITÉS	9
PREMIÈRE PHASE DE TRADUCTION DU GÉNÉRATEUR DE CODE	9
DEUXIÈME PHASE DE TRADUCTION DU GÉNÉRATEUR DE CODE	9
MODULE DE DIALOGUE	9
CHAPITRE N°5.	10
LES QUATRE FICHIERS TEXTES ASSOCIÉS À UNE APPLICATION	10
LA DÉCLARATION UNITAIRE	10
LA DÉCLARATION DE TABLE LINÉAIRE	11
LA DÉCLARATION DE TABLE POUR AFFECTATION AUTOMATIQUE	11
LES TYPES DE VARIABLES AUTOMGEN	11
LE FICHIER MODÈLE « DEFSMC.VUS »	12
LES FICHIERS « .SRT » ET « .END »	13
CHAPITRE N°6.	14
STRUCTURE DU CODE GÉNÉRÉ	14
CHAPITRE N°7.	15
LES DIRECTIVES DE GÉNÉRATION DE CODE	15
;£E : GESTION DE L'ÉTAT DES ÉTAPES DANS LE TEMPS À LA CHARGE DU PROGRAMMEUR	15
;£O : OPTIMISATION DU CODE GÉNÉRÉ	15
CHAPITRE N°8.	16
LE MODULE DE DIALOGUE	16
CHAPITRE N°9.	17
TECHNIQUES AVANCÉES	17
INSERTION DE CODE SMC DANS LES APPLICATIONS	17

CHAPITRE N°10.	18
LIMITE DE COMPATIBILITÉ	18
CHAPITRE N°11.	19
EXEMPLE COMPLET POUR AUTOMATE SMC	19

CHAPITRE N°1.

Introduction

Le post-processeur SMC est un module logiciel composé de plusieurs fichiers exécutables.

Il traduit les applications développées avec l'atelier logiciel AUTOMGEN en code SMC et les transfère dans un automate cible.

Toute la gamme des automates SMC est utilisable avec le post-processeur.

SMC est une marque déposée d'APRIL.

Le post-processeur permet d'effectuer la mise au point des applications depuis l'atelier logiciel AUTOMGEN en donnant accès aux fonctions de visualisation dynamique, de changement d'état des variables et de modification du mode de marche.

CHAPITRE N°2.

Installation

Pour installer le post-processeur SMC sur votre disque dur, procédez de la façon suivante :

1. Placez la disquette repérée « Post-processeur SMC » dans le lecteur de disquette A ou B,
2. Lancez le programme « IRAINST.EXE » sur la disquette,
3. Suivez ensuite les instructions du programme d'installation.



1. Sélectionnez la commande « Cibles ... » du menu « Option »,
2. Choisissez la ligne « SMC » puis cliquez sur « OK »,
3. Choisissez le menu « Options », « Paramètres par défaut », le bouton poussoir « SMCDIA32.EXE »,
4. La boîte de dialogue de paramétrage du module de dialogue s'ouvre .



Pour la version 16 bits il faut suivre la procédure suivante :

1. Lancez l'environnement AUTOMGEN,
2. Sélectionnez la commande « Cibles ... » du menu « Option »,
3. Cliquez sur le bouton « Insérer »,

La fenêtre suivante est alors affichée :

Définition d'une cible

Intitulé du post-processeur

Nom du post-processeur, éventuellement du module de configuration

Nom de l'exécuteur, éventuellement de l'outil de maintenance

? Annuler OK

4. Entrez sur la première ligne l'intitulé du post-processeur. Cette ligne n'est pas fonctionnelle, elle permettra simplement de sélectionner le post-processeur dans l'environnement. Il est pratique d'entrer un nom générique (« *SMC 50* » par exemple) suivi du port de communication utilisé (« *COM2* » par exemple). Entrez « *SMC 50 sur COM1* » ou « *SMC 600 sur COM2* »,
5. Entrez sur la deuxième ligne le nom du générateur de code SMC. Entrez « *PPSMC2.EXE* »,
6. Entrez sur la troisième ligne le nom du module de communication. Entrez « *SMCDIAL.EXE* » (pour utiliser le port COM1 ou entrez « *SMCDIAL.EXE -C2* » pour utiliser le port COM2. Il faut un et un seul espace entre « *SMCDIAL.EXE* » « *-C2* »,

Exemple d'installation du post-processeur SMC 50 utilisé sur COM1:

Définition d'une cible

Intitulé du post-processeur

SMC 50 COM1

Nom du post-processeur, éventuellement du module de configuration

PPSMC2.EXE

Nom de l'exécuteur, éventuellement de l'outil de maintenance

SMCDIAL.EXE

? Annuler OK

Exemple d'installation sur COM2 avec un automate SMC 25 :

Définition d'une cible

Intitulé du post-processeur
SMC 25 COM2

Nom du post-processeur, éventuellement du module de configuration
PPSMC2.EXE

Nom de l'exécuteur, éventuellement de l'outil de maintenance
SMCDIAL.EXE -C2

? Annuler OK

7. L'installation est terminée, cliquez sur le post-processeur que vous souhaitez utiliser puis sur le bouton « OK ».

Si vous voulez vérifier ou modifier l'installation d'un post-processeur, cliquez sur la ligne correspondante, puis sur le bouton « Modifier ».

Remarque :

- Pour pouvoir connecter AUTOMGEN aux automates SMC, il faut utiliser un boîtier de conversion RS232 vers boucle de courant.

CHAPITRE N°3.

Prise en main

Ce chapitre donne le déroulement précis permettant de compiler le fichier d'exemple et de l'exploiter sur un automate SMC.

Si vous n'avez pas installé le programme d'exemple, relancez la procédure d'installation pour le faire.

1. Lancez l'environnement AUTOMGEN,
2. Cliquez sur l'option « Ouvrir Folio » du menu « Fichier »,
3. A l'aide du sélecteur de fichiers, ouvrez le folio « SMC.GR7 » qui se trouve dans le sous-répertoire « EXAPISMC » du répertoire où est installé AUTOMGEN (probablement « C:\AUTOMV6 »¹),
4. Cliquez sur l'option « Cibles ... » du menu « Option », sélectionnez la ligne « SMC sur COM1 » ou « SMC sur COM2 »² puis le bouton « OK »,
5. L'exemple a été configuré pour un automate SMC 50. Si vous utilisez un automate de ce type, passez directement au point 9,
6. Choisissez l'option « Editeur de texte » du menu « Boîte à outils » et ouvrez le fichier « DEFSMC.VSY » qui se trouve dans le répertoire d'installation d'AUTOMGEN,
7. Modifiez la directive « \$SMC=50 » de ce fichier en « SMC XX » où « XX » est le type de votre automate SMC,
8. Sélectionnez l'option « Quitter » du menu « Fichier » et répondez oui à la question « Voulez-vous sauvegarder DEFSMC.VSY »,
9. Cliquez sur l'option « Compiler » du menu « Compiler »,

La compilation est lancée : elle se termine par l'appel du post-processeur SMC.

Si le post-processeur n'est pas lancé, c'est probablement que l'installation n'a pas été réalisée correctement, reprenez point par point le chapitre « Installation ».

Si tout s'est bien passé, un fichier « SMC.COD » a été généré dans le même répertoire que le fichier « SMC.GR7 ». C'est ce fichier qui sera relu et transféré dans l'automate.

10. Cliquez sur la commande « Charger » du menu « Exécuter »,
11. Une fenêtre « Module de dialogue SMC » s'ouvre et le transfert est réalisé³,
12. Cliquez sur la commande « Initialiser » du menu « Exécuter »,

¹si le fichier n'existe pas, vous n'avez pas sélectionné l'option « Installer les exemples » dans la procédure d'installation. Relancez la procédure d'installation dans ce cas.

²si aucune de ces lignes n'apparaît, l'installation n'a pas été réalisée correctement, retournez au chapitre « Installation ».

³si ce n'est pas le cas, vérifiez que l'installation est en accord avec le port de communication utilisé (voir le chapitre « Installation »). Effacez la mémoire de l'automate en enlevant la pile puis en le mettant hors tension. Vérifiez que la pile de sauvegarde n'est pas défectueuse.

13. Cliquez sur la commande « Exécuter » du menu « Exécuter », l'automate passe en RUN,

14. Cliquez sur la commande « Visualiser » du menu « Debug », la visualisation dynamique est active sur le folio « SMC ».

Ceci termine le chapitre « Prise en Main ».

CHAPITRE N°4.

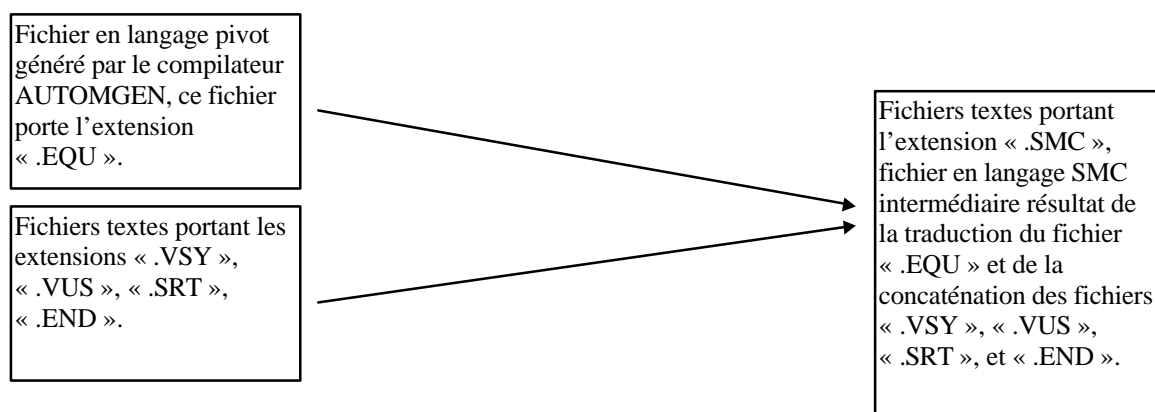
Généralités

Le post-processeur se compose de différents modules détaillés ci-dessous. Pour rendre les explications plus compréhensibles, seuls les fichiers principaux et accessibles au programmeur seront mentionnés.

Première phase de traduction du générateur de code

« PPSMC32.EXE » est l'exécutable qui réalise la première passe de traduction. Il traduit le langage pivot d'AUTOMGEN représenté par un fichier portant l'extension « .EQU » en fichier SMC intermédiaire qui porte l'extension « .SMC ». En outre, ce générateur de code cherche des fichiers textes qui portent les extensions « .VSY », « .VUS », « .SRT », « .END » et les recopie dans le fichier de sortie s'ils existent.

Le schéma ci-dessous donne une description de la tâche accomplie par « PPSMC2.EXE » :



Deuxième phase de traduction du générateur de code

« SMCCO32.EXE » est l'exécutable qui réalise la deuxième phase de traduction. Il traduit le fichier « .SMC » en un fichier « .COD » qui contient un code machine qui peut être transféré sur les automates SMC. Ce traducteur effectue deux tâches : remplacer le nom des variables AUTOMGEN par le nom des variables SMC et résoudre les sauts dans le programme. Un fichier « .LAP » est également généré, il contient la liste des instructions en langage SMC booléen et calcul.

Module de dialogue

« SMCDIA32.EXE » est le module de dialogue appelé par l'environnement AUTOMGEN sous WINDOWS. Il est utilisé pour les fonctions de téléchargement, de visualisation dynamique, de changement d'état des variables et de forçage des modes de marche de l'automate.

CHAPITRE N°5.

Les quatre fichiers textes associés à une application

Comme il a été dit au chapitre précédent, le générateur de code récupère quatre fichiers textes lors de la première phase de traduction.

Si vous êtes débutant dans l'utilisation d'AUTOMGEN ou du post-processeur SMC, nous vous conseillons d'utiliser ces fichiers sans les modifier. Dans un deuxième temps, et si vous désirez réaliser des applications particulières (optimisation de la taille du programme, ajout de morceaux de programmes en langage SMC), vous pourrez étudier ce chapitre ainsi que le chapitre « Techniques avancées ».

Le générateur de code cherche les fichiers portant l'extension « .VSY », « .VUS », « .SRT » et « .END ».

Pour chacun de ces quatre fichiers, le générateur de code recherche :

- d'abord un fichier portant le même nom que le premier folio de l'application,
- si ce fichier n'existe pas, c'est le fichier portant le nom « DEFSCMC » et qui se trouve dans le répertoire où a été installé AUTOMGEN qui est sélectionné,
- si ni l'un ni l'autre de ces fichiers existent, alors le générateur de code considère cette partie comme vide et ne génère aucun message d'erreur.

Ces fichiers peuvent recevoir des commentaires placés à droite d'un caractère « ; » (point virgule).

Les fichiers « .VSY » et « .VUS » sont des fichiers de déclaration de variables. Ils donnent la correspondance entre les variables AUTOMGEN et les variables SMC.

Le fichier « .VSY » donne la correspondance pour les variables dites « Système ». Ce fichier contient également des directives de compilation (voir chapitre « Paramétrage » et « Techniques avancées »). Le fichier « DEFSCMC.VSY » est fourni en modèle dans le répertoire où est installé AUTOMGEN.

Le fichier « .VUS » donne la correspondance pour les variables utilisées dans l'application. Le fichier « DEFSCMC.VUS » est fourni en modèle dans le répertoire où est installé AUTOMGEN.

Il existe plusieurs syntaxes permettant de déclarer ces affectations.

La déclaration unitaire

Elle donne la correspondance entre une variable AUTOMGEN et une variable automate.

La syntaxe est :

variable AUTOMGEN = variable automate

Exemple :

#m200=m100 ; le mot AUTOMGEN 200 est attribué au mot SMC 100

#i10=e16 ; l'entrée 10 d'AUTOMGEN est attribuée à l'entrée SMC E16

La déclaration de table linéaire

Elle donne la correspondance entre une série de variables AUTOMGEN et une série de variables AUTOMATE.

La syntaxe est :

longueur de la table : première variable AUTOMGEN = première variable automate

Exemple :

##16:m100=m100 ; les mots AUTOMGEN 100 à 115 sont attribués aux mots SMC
; 100 à 115

La déclaration de table pour affectation automatique

Elle laisse le soin au générateur de code d'affecter un ou plusieurs types de variables dans une table de variables automate. Le générateur de code affecte automatiquement les variables suivant ses besoins et si la table n'est pas saturée.

La syntaxe est :

type(s) de variable AUTOMGEN = première variable SMC, dernier numéro de variable SMC

Un type est représenté par un nom de variable sans numéro. Si plusieurs types sont à préciser, alors ils doivent être séparés par le caractère « & ».

Exemples :

###x&bx=r0,127 ; affecte automatiquement les bits d'étapes d'AUTOMGEN dans les
; bits SMC R0 à R127

###m&c=m100,127 ; affecte les mots et les compteurs d'AUTOMGEN dans les mots
; SMC 100 à 127

Ce type de déclaration est moins prioritaire que les types « # » et « ## ».

Plusieurs déclarations de ce type peuvent être utilisées pour un même type de variable AUTOMGEN. Dans ce cas, le générateur de code remplira dans l'ordre de déclaration et suivant ses besoins, une ou plusieurs tables.

Les types de variables AUTOMGEN

Les types de variables AUTOMGEN sont un sur-ensemble de ceux décrits dans le manuel d'utilisation du module principal dans la partie « Compilateur ».

En voici la liste exhaustive :

x	étape, état actuel
bx	étape, état futur
b	bit, état actuel
bb	bit, état futur

u	bit utilisateur, état actuel
bu	bit utilisateur, état futur
i	entrée, état actuel
bi	entrée, état futur
o	sortie, état actuel
bo	sortie, état futur
m	mot
c	compteur
t	temporisation, état de fin
bt	temporisation, état de lancement
tconsi	temporisation, mot de consigne
tempo	numéro de temporisation

Pour les variables booléennes d'AUTOMGEN, deux variables booléennes SMC sont utilisées afin de gérer les états dans le temps. Pour les entrées et les sorties d'AUTOMGEN, il est possible d'affecter une même variable SMC dans le cas où aucun test de front n'est utilisé sur ces types de variables.

Le fichier modèle « DEFSMC.VUS »

Détail du fichier « DEFSMC.VUS » présent dans le répertoire où est installé AUTOMGEN.

```
; Fichier DEFSMC.VUS - AUTOMGEN V6
```

```
; Définition des variables bits
; modifier suivant les besoins et les possibilités de l'automate
```

```
###x&bx=b3,255
###i&bo&bb&bu&b&u&t&bt&ux&dx&ui&di&uo&do&du&uu&ub&db&tt&ut&dt=r0,127
```

```
; Définition des entrées / sorties
; exemple pour 1 carte de 16 entrées en 1
; et 1 carte de 16 sorties en 2
```

```
##16:bi0=e0
```

```
##16:o0=a16
```

```
; Définition des mots et des compteurs
; modifier suivant les besoins et les possibilités de l'automate
###m&c&tconsi=m5,127
```

```
; Définition des temporisations
; modifier suivant les besoins et les possibilités de l'automate
##16:tempo0=t0
```

Fixe la valeur de l' « adc »

Définition des variables booléennes autres qu'entrées sorties

Définition des entrées

Définition des sorties

Définition des mots et compteurs

Définition des temporisations

Les fichiers « .SRT » et « .END »

Ces fichiers permettent d'insérer du code SMC au sein d'une application AUTOMGEN.

Le code contenu dans le fichier « .SRT » est exécuté à chaque cycle, avant le code généré par la compilation de l'application issue d'AUTOMGEN.

Le code contenu dans le fichier « .END » est exécuté à chaque cycle, après le code généré par la compilation de l'application issue d'AUTOMGEN.

Le fichier « DEFSMC.SRT » est fourni en modèle dans le répertoire d'installation d'AUTOMGEN. Ce fichier doit au minimum réaliser l'initialisation des variables de l'application. L'état du bit 0 d'AUTOMGEN est géré dans ce fichier, il doit obligatoirement être à l'état 1 au premier cycle de scrutation de l'application.

La syntaxe utilisée dans ces deux fichiers est une syntaxe de type assembleur équivalente à celle du langage SMC. Les compléments de syntaxe suivants sont à prendre en compte :

- la directive « \$ADC=xxx » fixe la valeur de l'ADC (adresse de séparation entre le langage booléen et le langage calcul dans les automates SMC, pour plus de détails consultez les notices de programmation SMC). Par défaut cette valeur est fixée à 1024,
- la directive « \$SEQ » marque le début d'une zone booléenne,
- la directive « \$CAL » débute une zone de calcul,
- la directive « \$PRD » débute une zone de prédisposition de variables,
- la directive « \$SMC=xxx » fixe le type d'automate,
- les variables booléennes peuvent être utilisées en bistable ou en monostable indépendamment des conventions du langage SMC. Le caractère « ! » placé après le signe « = » force la variable à être bistable (mise à un ou mise à zéro), le caractère « ? » placé après le signe « = » force la variable à être monostable (affectation ou affectation complémentée),
- la syntaxe « SS.ccccccc » permet d'écrire une séquence de sécurité (nécessaire sur les automates SMC 25 et 600), « ccccccc » représente un nom de programme sur 8 caractères maximum,
- pour les sauts, il faut utiliser la syntaxe suivante : « @nom de label » marque la destination,
« =J_nom de label_ » marque le saut, Exemple :

e32=j_fin_

...

@fin

Pour plus de détails concernant ces compléments de syntaxe observez le fichier « SMC.SMC » généré par le post-processeur.

Un fichier « DEFSMC.END » est fourni en modèle.

CHAPITRE N°6.

Structure du code généré

Le programme généré par AUTOMGEN adopte les conventions suivantes :

- tout le programme est compilé sous la forme d'une série d'équations booléennes,
- des appels de type « calcul immédiat » sont générés pour les opérations sur mots.

CHAPITRE N°7.

Les directives de génération de code

Le générateur de code peut recevoir des directives permettant d'influer sur la forme du code généré. Ces directives sont à écrire dans le fichier « .VSX » de l'application.

Une seule directive peut être écrite par ligne, la forme est la suivante : « ;*fx* » où *x* est la directive de compilation à écrire.

;fE : gestion de l'état des étapes dans le temps à la charge du programmeur

Cette directive demande au générateur de code de ne pas générer le code d'évolution des étapes dans le temps. Ce code consiste à recopier les bits *bx* dans les bits *x*. Si cette directive est présente, alors il faut écrire le code d'évolution des étapes dans le fichier « .END ».

Cette directive peut être utile pour optimiser le programme généré en réécrivant un code de copie des bits *bx* vers *x* plus efficace (copie de table de bits).

;fO : optimisation du code généré

Cette directive demande au post-processeur d'optimiser le code généré.

CHAPITRE N°8.

Le module de dialogue

« SMCDIA32.EXE » est le module de dialogue appelé par l'environnement pour le téléchargement des programmes, la visualisation dynamique, le changement d'état des variables et le changement d'état des modes de marche.

Le mode pas à pas n'est pas supporté par le module de dialogue.

Les types de variables SMC suivants sont visualisables : entrées, sorties, bits, mots, temporisations.

Le module de dialogue utilise un fichier généré par le générateur de code et qui porte l'extension « .DBG ». Ce fichier donne la correspondance entre les variables AUTOMGEN de l'application et les variables automate.

CHAPITRE N°9.

Techniques avancées

Ce chapitre décrit des possibilités du post-processeur qui permettent de créer des applications plus performantes.

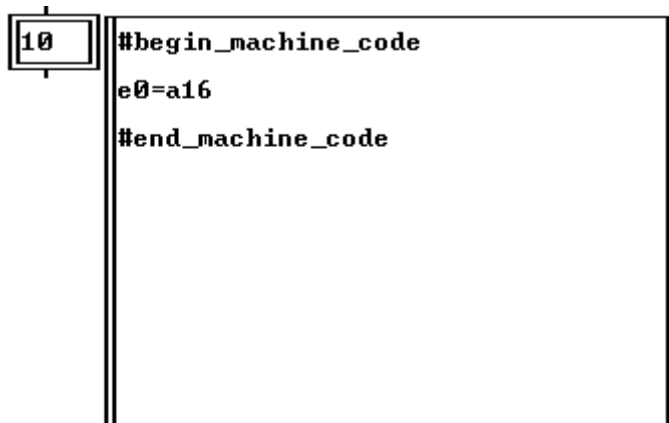
Insertion de code SMC dans les applications

Comme nous l'avons vu au chapitre 5, les fichiers « .SRT » et « .END » associés à une application peuvent recevoir des lignes de code en langage SMC.

Une autre technique peut être utilisée pour insérer du code SMC dans une boîte de code AUTOMGEN.

Les directives « #BEGIN_MACHINE_CODE » et « #END_MACHINE_CODE » permettent de définir une section de code constructeur.

Exemple :



```
#begin_machine_code
e0=a16
#end_machine_code
```

CHAPITRE N°10.

Limite de compatibilité

Ce chapitre détaille les limites d'utilisation du post-processeur SMC par rapport aux possibilités d'AUTOMGEN et des automates SMC.

- l'adressage indirect n'est pas utilisable (utiliser des sections encode SMC),
- les sous-programmes ne sont pas utilisables,
- les calculs en virgule flottante ne sont pas utilisables,
- les calculs sur variables 32 bits ne sont pas utilisables,
- les instructions numériques sont réduites à celles disponibles sur l'automate cible,
- certaines instructions du langage littéral bas niveau ne sont pas utilisables.

Si le générateur de code SMC rencontre une forme de programme intraduisible, alors un message d'erreur de type « combinaison d'instruction et/ou de type d'adressage non supporté à l'adresse XXXX » est généré. XXXX représente une adresse dans le fichier « .EQU » Si vous souhaitez connaître avec précision l'instruction qui a généré cette erreur, utilisez l'utilitaire « CODELIST.EXE⁴ » pour obtenir un listing du fichier « .EQU ».

⁴le chapitre « Techniques avancées » du manuel de l'utilisateur associé au module principal d'AUTOMGEN décrit l'utilisation de l'utilitaire « CODELIST.EXE »

CHAPITRE N°11.

Exemple complet pour automate SMC

Ce chapitre contient le dossier de documentation de l'exemple « SMC » généré avec AUTOMGEN.

Cet exemple est présent dans le sous-répertoire « EXAPISMC » du répertoire où est installé AUTOMGEN.

SMC



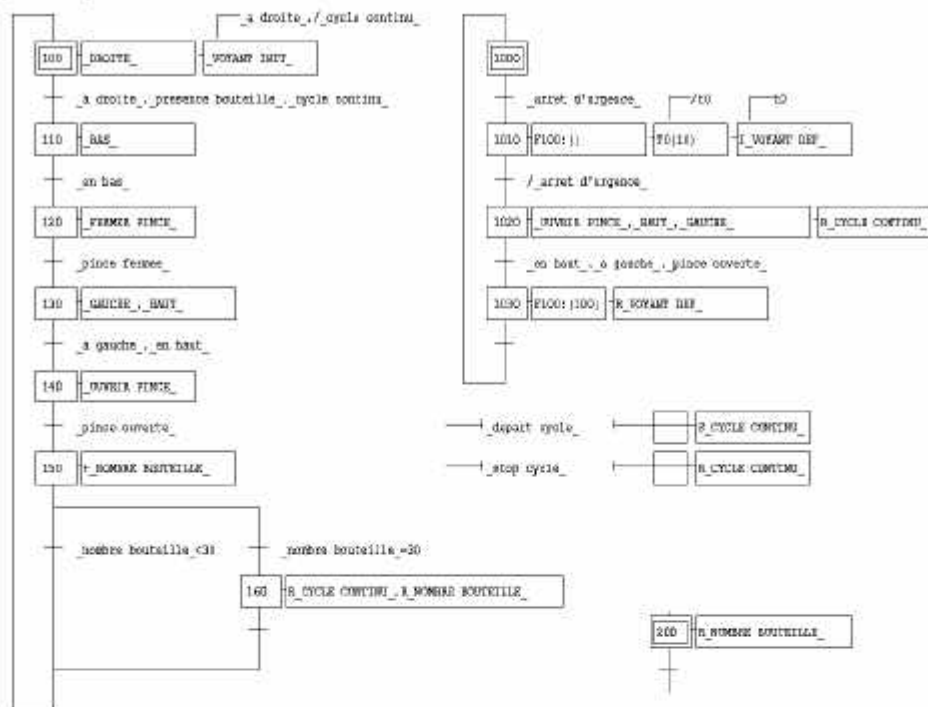
IRAI
30110 LA GRAND COMBE

SMC

Cible : .992

Date de création : 23/03/2010 09:19 00

/* Programme d'exemple */



AUTOMGEN	AD1 n°1	AD2 n°2	SYMBOLE	COMMENTAIRE OU NOM (UNE PILULE OU LA VARIABLE EST UTILISEE)
I0000		B100	a droite	capteur varis 1 sortie
I0001		B101	a gauche	capteur varis 1 renvoie
I0002		B102	en bas	capteur varis 2 sortie
I0003		B103	en haut	capteur varis 2 renvoie
I0004		B104	pinne fermee	capteur pince fermee
I0005		B105	pinne ouverte	capteur pince ouverte
I0006		B106	pression bouteille	capteur bouteille en attente
I0007		B107	depart cycle	bp depart de cycle
I0008		B108	stop cycle	bp stop
I0009		B109	arret d'urgence	bp arret d'urgence
M0000	M0		nombre bouteille	nombre de bouteilles passées
O0000	A16		ENTREE	sortir le varis 1
O0001	A17		ENTREE	rentrer le varis 1
O0002	A18		ENT	sortir le varis 2
O0003	A19		ENT	rentrer le varis2
O0004	A20		FERMER PINCE	fermer la pince
O0005	A21		OUVRIR PINCE	ouvrir la pince
O0006	A22		VOYANT VERT	voyant vert
O0007	A23		VOYANT ROUGE	voyant rouge
P0000	B12	B20		
P0001	B0	B1		
U0000	B0	B1	cycle d'attente	etat cycle d'attente
X0000	B3	B4		
X0010	B5	B6		
X0020	B7	B8		
X0030	B9	B10		
X0040	B11	B12		
X0050	B13	B14		
X0060	B15	B16		
X0070	B17	B18		
X0080	B19	B20		
X0090	B21	B22		
X0100	B23	B24		
X0110	B25	B26		
	B27	B28		
SMC - Page 1 - La liste des références croisées				

Fichier « SMC.SMC »

```
; ***** Variables systèmes ***** VSY
; pas de fichier C:\Automv6\Exapismc\Smc.VSY pour l'application,
utilisation du fichier C:\AUTOMV6\defsmc.VSY
; Fichier DEFSCMC.VSY - AUTOMGEN V6

; A modifier suivant le type de l'automate SMC : 25, 50 ou 600
$SMC=50
; Valeur de l'adc, modifier suivant les besoins
$ADC=768

; ;f0 valide l'otimisation du code généré

#Zf=r473
#Cf=r473
#Of=r473
#ac=m4
#true=b2
; ***** Variables utilisateur ***** VUS
; pas de fichier C:\Automv6\Exapismc\Smc.VUS pour l'application,
utilisation du fichier C:\AUTOMV6\defsmc.VUS
; Fichier DEFSCMC.VUS - AUTOMGEN V6

; Définition des variables bits
; modifier suivant les besoins et les possibilités de l'automate

###x&bx=b3,255
###i&bo&bb&bu&b&u&t&bt&ux&dx&ui&di&uo&do&du&uu&ub&db&tt&ut&dt=r0,127

; Définition des entrées / sorties
; exemple pour 1 carte de 16 entrées en 1
; et 1 carte de 16 sorties en 2

##16:bi0=b100
##16:o0=a16

; Définition des mots et des compteurs
; modifier suivant les besoins et les possibilités de l'automate
###m&c&t&consi=m5,127

; Définition des temporisations
; modifier suivant les besoins et les possibilités de l'automate
##16:tempo0=t0
; ***** Correspondance bits/étapes
#b100=_x100_
#bb100=_bx100_
#b103=_x110_
#bb103=_bx110_
#b105=_x120_
#bb105=_bx120_
#b107=_x130_
#bb107=_bx130_
#b109=_x140_
#bb109=_bx140_
#b112=_x150_
#bb112=_bx150_
#b115=_x160_
#bb115=_bx160_
#b116=_x200_
#bb116=_bx200_
#b102=_x1000_
#bb102=_bx1000_
```

```

#b104=_x1010_
#bb104=_bx1010_
#b106=_x1020_
#bb106=_bx1020_
#b108=_x1030_
#bb108=_bx1030_
; ***** Correspondance bits/bits utilisateur
#b101=_u1000_
#bb101=_bu1000_
; ***** Programme de démarrage ***** SRT
; pas de fichier C:\Automv6\Exapismc\Smc.SRT pour l'application,
utilisation du fichier C:\AUTOMV6\defsmc.SRT
; Fichier DEFSSMC.SRT - AUTOMGEN V6

; pour les SMC 25 ou 600 une séquence de sécurité
; modifier le nom derrière '.' si nécessaire

$IF SMC=25
SS.SMC
$ENDIF
$IF SMC=600
SS.SMC
$ENDIF

; fabrique un bit toujours vrai
_true_+/_true_!=_true_
; le bit 0 est mis à 1 au premier cycle pour les étapes initiales
_true_!=_b0_
; saute au restant du programme
_true_=j_suite_
; label pour le rebouclage du programme
@debut
; le bit 0 est mis à 0 pour les autres cycles
_true_!=/_b0_

@suite
; valide les appels calculs
_true_=r472
; ***** Prédispositions
$PRD
_tconsi0_=10
$SEQ
; ***** Code généré par le post-processeur *****

; ***** Module : C:\AUTOMV6\EXAPISMC\SMC.GR7
_bi0_. _bi6_. _u1000_!=/_bx100_
_b0_!=_bx100_
_x160_=?_bu99_
r472=l_calcl_
$CAL
@calcl
->[0,0],_m200_<30
$SEQ
/r473=?_b8_
_x150_. _b8_=?_bu98_
_bu99+_bu98_!=_bx100_
_bi9_!=/_bx1000_
_b0_!=_bx1000_
_x1030_!=_bx1000_
_bi2_!=/_bx110_
_x100_. _bi0_. _bi6_. _u1000_!=_bx110_
/_bi9_!=/_bx1010_
_x1000_. _bi9_!=_bx1010_
_bi4_!=/_bx120_
_x110_. _bi2_!=_bx120_
_bi3_. _bi1_. _bi5_!=/_bx1020_

```

```

_x1010_./_bi9_!=_bx1020_
_bi1_._bi3_!=/_bx130_
_x120_._bi4_!=_bx130_
_true_!=/_bx1030_
_x1020_._bi3_._bi1_._bi5_!=_bx1030_
_bi5_!=/_bx140_
_x130_._bi1_._bi3_!=_bx140_
_bi7_=?_b110_
r472=l_calc3_
$CAL
@calc3
->[0,0],_m200_<30
$SEQ
/r473=?_b30_
r472=l_calc2_
$CAL
@calc2
->[0,0],_m200_=30
$SEQ
/r473=?_b29_
_b29_+_b30_!=/_bx150_
_x140_._bi5_!=_bx150_
_bi8_=?_b113_
_true_!=/_bx160_
r472=l_calc4_
$CAL
@calc4
->[0,0],_m200_=30
$SEQ
/r473=?_b35_
_x150_._b35_!=_bx160_
_true_!=/_bx200_
_b0_!=_bx200_
_bi0_./_u1000_=?_b118_
/_t0_=?_b121_
_t0_=?_b123_
/_b119_=j_line99_
_true_!=/_bx140_
_true_!=/_bx130_
_true_!=/_bx120_
_true_!=/_bx110_
_true_!=/_bx100_
_true_!=/_bx160_
_true_!=/_bx150_
@line99
/_b124_=j_line108_
_true_!=/_bx140_
_true_!=/_bx130_
_true_!=/_bx120_
_true_!=/_bx110_
_true_!=/_bx100_
_true_!=/_bx160_
_true_!=/_bx150_
@line108
; ***** Module : (ACTION)
_b118_._bx100_=?_b117_
_b121_._bx1010_=?_b120_
_b123_._bx1010_=?_b122_
_bx100_=?_o0_
_b117_=?_o6_
_bx110_=?_o2_
_bx1010_=?_b119_
_b120_=?_bt0_
_b122_=?_b1_
/_b1_=j_label10_
_o7_=?/_o7_

```

```

@label0
_bx120_=?_o4_
_bx1020+_bx140_=?_o5_
_bx1020+_bx130_=?_o3_
_bx1020+_bx130_=?_o1_
_bx1020+_b113+_bx160_!=/_bu1000_
_bx1030_=?_b124_
_bx1030_!=/_o7_
_b110_!=_bu1000_
_bx150_.r472=l_calc5_
$CAL
@calc5
_m200_<-_m200_+1
$SEQ

_bx160+_bx200_.r472=l_calc6_
$CAL
@calc6
_m200_<-0
$SEQ

; ***** Evolution des bits d'étapes
_bx100_=?_x100_
_bx110_=?_x110_
_bx120_=?_x120_
_bx130_=?_x130_
_bx140_=?_x140_
_bx150_=?_x150_
_bx160_=?_x160_
_bx200_=?_x200_
_bx1000_=?_x1000_
_bx1010_=?_x1010_
_bx1020_=?_x1020_
_bx1030_=?_x1030_
; ***** Evolution des bits utilisateurs
_bu1000_=?_u1000_
; ***** Evolution des temporisations
_tempo0_.bt0_.tconsi0_.0=_t0_
; ***** Programme de fin ***** END
; pas de fichier C:\Automv6\Exapismc\Smc.END pour l'application,
utilisation du fichier C:\AUTOMV6\defsmc.END
; DEFSMC.END - AUTOMGEN V6

; rebouclage du programme
->s_debut_

```

Fichier « SMC.LAP »

```

8000 : B6 04 02 2B 05 F6 04 02
8008 : 17 43 2A 03 B7 04 02 B7
8010 : 40 00 S0000 : B2+/B2=!B2
8012 : B6 04 02 2A 03 B7 06 02
801A : 02 B7 40 01 S0001 : B2=!R2
801E : B6 04 02 2A 03 7E 80 3A
8026 : CE 20 22 B7 40 02 S0002 : B2=J4
802C : B6 04 02 2A 04 43 B7 06
8034 : 02 43 02 B7 40 03 S0003 : B2=!/R2
803A : B6 04 02 2A 03 B7 07 D8
8042 : 02 B7 40 04 S0004 : B2=R472
140A : 00 0A 00 00 00 00 00 00
1412 : 00 00 00 00 00 00 00 00 : M5=10

; MODULE : C:\AUTOMV6\EXAPISMC\SMC.GR7

8046 : B6 04 64 B4 04 6A B4 06
804E : 00 2A 04 43 B7 04 04 43
8056 : 02 B7 40 05 S0005 : B100.B106.R0=!/B4
805A : B6 06 02 2A 03 B7 04 04
8062 : 02 B7 40 06 S0006 : R2=!B4
8066 : B6 04 0F B7 06 03 02 B7
806E : 40 07 S0007 : B15=?R3
8070 : B6 07 D8 BD FF F2 CE 00
8078 : 01 B7 40 08 S0008 : R472=L1
8600 : 02 82 40 00 00 00 00 14
8608 : 0C 00 00 00 1E 00 00 01 L0001 : ->[0,0],M6<30
807C : F6 07 D9 17 43 B7 06 04
8084 : 02 B7 40 09 S0009 : /R473=?R4
8088 : B6 04 0D B4 06 04 B7 06
8090 : 05 B7 40 0A S0010 : B13.R4=?R5
8094 : B6 06 03 2B 03 B6 06 05
809C : 2A 03 B7 04 04 B7 40 0B S0011 : R3+R5=!B4
80A4 : B6 04 6D 2A 04 43 B7 04
80AC : 14 43 02 B7 40 0C S0012 : B109=!/B20
80B2 : B6 06 02 2A 03 B7 04 14
80BA : 02 B7 40 0D S0013 : R2=!B20
80BE : B6 04 19 2A 03 B7 04 14
80C6 : 02 B7 40 0E S0014 : B25=!B20
80CA : B6 04 66 2A 04 43 B7 04
80D2 : 06 43 02 B7 40 0F S0015 : B102=!/B6
0000 : B6 04 03 B4 04 64 B4 04
0008 : 6A B4 06 00 2A 03 B7 04
0010 : 06 B7 40 10 S0016 : B3.B100.B106.R0=!B6
80EC : F6 04 6D 17 43 2A 04 43
80F4 : B7 04 16 43 02 B7 40 11 S0017 : /B109=!/B22
80FC : B6 04 13 B4 04 6D 2A 03
8104 : B7 04 16 B7 40 12 S0018 : B19.B109=!B22
810A : B6 04 68 2A 04 43 B7 04
8112 : 08 43 02 B7 40 13 S0019 : B104=!/B8
8118 : B6 04 05 B4 04 66 2A 03
8120 : B7 04 08 B7 40 14 S0020 : B5.B102=!B8
8126 : B6 04 67 B4 04 65 B4 04
812E : 69 2A 04 43 B7 04 18 43
8136 : 02 B7 40 15 S0021 : B103.B101.B105=!/B24
813A : B6 04 15 F6 04 6D 10 2A
8142 : 03 B7 04 18 02 B7 40 16 S0022 : B21./B109=!B24
0000 : B6 04 65 B4 04 67 2A 04
0008 : 43 B7 04 0A 43 B7 40 17 S0023 : B101.B103=!/B10
815A : B6 04 07 B4 04 68 2A 03
8162 : B7 04 0A B7 40 18 S0024 : B7.B104=!B10
8168 : B6 04 02 2A 04 43 B7 04
8170 : 1A 43 02 B7 40 19 S0025 : B2=!/B26

```

```

8176 : B6 04 17 B4 04 67 B4 04
817E : 65 B4 04 69 2A 03 B7 04
8186 : 1A B7 40 1A
818A : B6 04 69 2A 04 43 B7 04
8192 : 0C 43 02 B7 40 1B
8198 : B6 04 09 B4 04 65 B4 04
81A0 : 67 2A 03 B7 04 0C 02 B7
81A8 : 40 1C
81AA : B6 04 6B B7 06 06 02 B7
81B2 : 40 1D
0000 : B6 07 D8 BD FF F2 CE 00
0008 : 02 B7 40 1E
8610 : 02 82 40 00 00 00 00 14
8618 : 0C 00 00 00 1E 00 00 02
81C0 : F6 07 D9 17 43 B7 06 07
81C8 : 02 B7 40 1F
81CC : B6 07 D8 BD FF F2 CE 00
81D4 : 03 B7 40 20
8620 : 02 81 40 00 00 00 00 14
8628 : 0C 00 00 00 1E 00 00 03
81D8 : F6 07 D9 17 43 B7 06 08
81E0 : 02 B7 40 21
81E4 : B6 06 08 2B 03 B6 06 07
81EC : 2A 04 43 B7 04 0E 43 B7
81F4 : 40 22
81F6 : B6 04 0B B4 04 69 2A 03
81FE : B7 04 0E B7 40 23
8204 : B6 04 6C B7 06 09 02 B7
820C : 40 24
820E : B6 04 02 2A 04 43 B7 04
8216 : 10 43 02 B7 40 25
821C : B6 07 D8 BD FF F2 CE 00
8224 : 04 B7 40 26
8630 : 02 81 40 00 00 00 00 14
8638 : 0C 00 00 00 1E 00 00 04
8228 : F6 07 D9 17 43 B7 06 0A
8230 : 02 B7 40 27
8234 : B6 04 0D B4 06 0A 2A 03
823C : B7 04 10 B7 40 28
8242 : B6 04 02 2A 04 43 B7 04
824A : 12 43 02 B7 40 29
8250 : B6 06 02 2A 03 B7 04 12
8258 : 02 B7 40 2A
825C : B6 04 64 F6 06 00 10 B7
8264 : 06 0B 02 B7 40 2B
826A : F6 06 0C 17 43 B7 06 0D
8272 : 02 B7 40 2C
8276 : B6 06 0C B7 06 0E 02 B7
827E : 40 2D
0000 : F6 06 0F 17 43 2A 03 7E
0008 : 82 F2 CE 21 80 B7 40 2E
8290 : B6 04 02 2A 04 43 B7 04
8298 : 0C 43 02 B7 40 2F
829E : B6 04 02 2A 04 43 B7 04
82A6 : 0A 43 02 B7 40 30
82AC : B6 04 02 2A 04 43 B7 04
82B4 : 08 43 02 B7 40 31
82BA : B6 04 02 2A 04 43 B7 04
82C2 : 06 43 02 B7 40 32
82C8 : B6 04 02 2A 04 43 B7 04
82D0 : 04 43 02 B7 40 33
82D6 : B6 04 02 2A 04 43 B7 04
82DE : 10 43 02 B7 40 34
82E4 : B6 04 02 2A 04 43 B7 04
82EC : 0E 43 02 B7 40 35
0000 : F6 06 10 17 43 2A 03 7E

S0026 : B23.B103.B101.B105=!B26
S0027 : B105=!/B12
S0028 : B9.B101.B103=!B12
S0029 : B107=?R6
S0030 : R472=L2
L0002 : ->[0,0],M6<30
S0031 : /R473=?R7
S0032 : R472=L3
L0003 : ->[0,0],M6=30
S0033 : /R473=?R8
S0034 : R8+R7=!/B14
S0035 : B11.B105=!B14
S0036 : B108=?R9
S0037 : B2=!/B16
S0038 : R472=L4
L0004 : ->[0,0],M6=30
S0039 : /R473=?R10
S0040 : B13.R10=!B16
S0041 : B2=!/B18
S0042 : R2=!B18
S0043 : B100./R0=?R11
S0044 : /R12=?R13
S0045 : R12=?R14
S0046 : /R15=J54
S0047 : B2=!/B12
S0048 : B2=!/B10
S0049 : B2=!/B8
S0050 : B2=!/B6
S0051 : B2=!/B4
S0052 : B2=!/B16
S0053 : B2=!/B14

```

```

0008 : 83 62 CE 21 B6 B7 40 36 S0054 : /R16=J62
8302 : B6 04 02 2A 04 43 B7 04
830A : 0C 43 02 B7 40 37 S0055 : B2=!/B12
8310 : B6 04 02 2A 04 43 B7 04
8318 : 0A 43 02 B7 40 38 S0056 : B2=!/B10
831E : B6 04 02 2A 04 43 B7 04
8326 : 08 43 02 B7 40 39 S0057 : B2=!/B8
832C : B6 04 02 2A 04 43 B7 04
8334 : 06 43 02 B7 40 3A S0058 : B2=!/B6
833A : B6 04 02 2A 03 B7 04 04
8342 : 02 B7 40 3B S0059 : B2=!B4
8346 : B6 04 02 2A 04 43 B7 04
834E : 10 43 02 B7 40 3C S0060 : B2=!/B16
8354 : B6 04 02 2A 04 43 B7 04
835C : 0E 43 02 B7 40 3D S0061 : B2=!/B14

; MODULE : (ACTION)

0000 : B6 06 0B B4 04 04 B7 06
0008 : 11 B7 40 3E S0062 : R11.B4=?R17
836E : B6 06 0D B4 04 16 B7 06
8376 : 12 B7 40 3F S0063 : R13.B22=?R18
837A : B6 06 0E B4 04 16 B7 06
8382 : 13 B7 40 40 S0064 : R14.B22=?R19
8386 : B6 04 04 97 90 B7 40 41 S0065 : B4=?A16
838E : B6 06 11 97 96 B7 40 42 S0066 : R17=?A22
8396 : B6 04 06 97 92 B7 40 43 S0067 : B6=?A18
839E : B6 04 16 B7 06 0F 02 B7
83A6 : 40 44 S0068 : B22=?R15
83A8 : B6 06 12 B7 06 14 02 B7
83B0 : 40 45 S0069 : R18=?R20
83B2 : B6 06 13 B7 06 15 02 B7
83BA : 40 46 S0070 : R19=?R21
83BC : F6 06 15 17 43 2A 03 7E
83C4 : 83 D6 CE 21 EE B7 40 47 S0071 : /R21=J73
0000 : 96 97 43 97 97 43 02 B7
0008 : 40 48 S0072 : A23=?/A23
83D6 : B6 04 08 97 94 B7 40 49 S0073 : B8=?A20
83DE : B6 04 18 2B 03 B6 04 0C
83E6 : 97 95 02 B7 40 4A S0074 : B24+B12=?A21
83EC : B6 04 18 2B 03 B6 04 0A
83F4 : 97 93 02 B7 40 4B S0075 : B24+B10=?A19
83FA : B6 04 18 2B 03 B6 04 0A
8402 : 97 91 02 B7 40 4C S0076 : B24+B10=?A17
8408 : B6 04 18 2B 03 B6 06 09
8410 : 2B 03 B6 04 10 2A 04 43
8418 : B7 06 01 43 02 B7 40 4D S0077 : B24+R9+B16=!/R1
8420 : B6 04 1A B7 06 10 02 B7
8428 : 40 4E S0078 : B26=?R16
842A : B6 04 1A 2A 03 43 97 97
8432 : 43 B7 40 4F S0079 : B26=!/A23
0000 : B6 06 06 2A 03 B7 06 01
0008 : 02 B7 40 50 S0080 : R6=!R1
8442 : B6 04 0E B4 07 D8 BD FF
844A : F2 CE 00 05 02 B7 40 51 S0081 : B14.R472=L5
8640 : 02 01 40 14 0C 00 00 14
8648 : 0C 00 00 00 01 00 00 05 L0005 : M6<-M6+1
8452 : B6 04 10 2B 06 B6 04 12
845A : B4 07 D8 BD FF F2 CE 00
8462 : 06 B7 40 52 S0082 : B16+B18.R472=L6
8650 : 01 00 80 14 0C 00 00 00
8658 : 00 00 00 00 00 00 00 06 L0006 : M6<-0
8466 : B6 04 04 B7 04 03 02 B7
846E : 40 53 S0083 : B4=?B3
8470 : B6 04 06 B7 04 05 02 B7
8478 : 40 54 S0084 : B6=?B5

```

```

847A : B6 04 08 B7 04 07 02 B7
8482 : 40 55
8484 : B6 04 0A B7 04 09 02 B7
848C : 40 56
848E : B6 04 0C B7 04 0B 02 B7
8496 : 40 57
8498 : B6 04 0E B7 04 0D 02 B7
84A0 : 40 58
84A2 : B6 04 10 B7 04 0F 02 B7
84AA : 40 59
84AC : B6 04 12 B7 04 11 02 B7
84B4 : 40 5A
84B6 : B6 04 14 B7 04 13 02 B7
84BE : 40 5B
84C0 : B6 04 16 B7 04 15 02 B7
84C8 : 40 5C
0000 : B6 04 18 B7 04 17 02 B7
0008 : 40 5D
84D4 : B6 04 1A B7 04 19 02 B7
84DC : 40 5E
84DE : B6 06 01 B7 06 00 02 B7
84E6 : 40 5F
84E8 : 02 B6 06 14 2B 10 02 B7
84F0 : 06 0C FE 14 0A FF 10 80
84F8 : 4F B7 05 FF 20 03 BD FF
8500 : F5 20 0F 1B 10 80 05 FF
8508 : 06 14 00 00 14 0A 00 00
8510 : 06 0C 02 B7 44 60
8516 : 0F CE 20 1C CE 80 2C FF
851E : 10 7E CE 86 00 FF 10 7C
8526 : BD FF EF B7 4C 61 02 7E
852E : 80 2C

```

S0085 : B8=?B7
S0086 : B10=?B9
S0087 : B12=?B11
S0088 : B14=?B13
S0089 : B16=?B15
S0090 : B18=?B17
S0091 : B20=?B19
S0092 : B22=?B21
S0093 : B24=?B23
S0094 : B26=?B25
S0095 : R1=?R0
S0096 : T0.R20.M5.0=R12
S0097 : ->S3