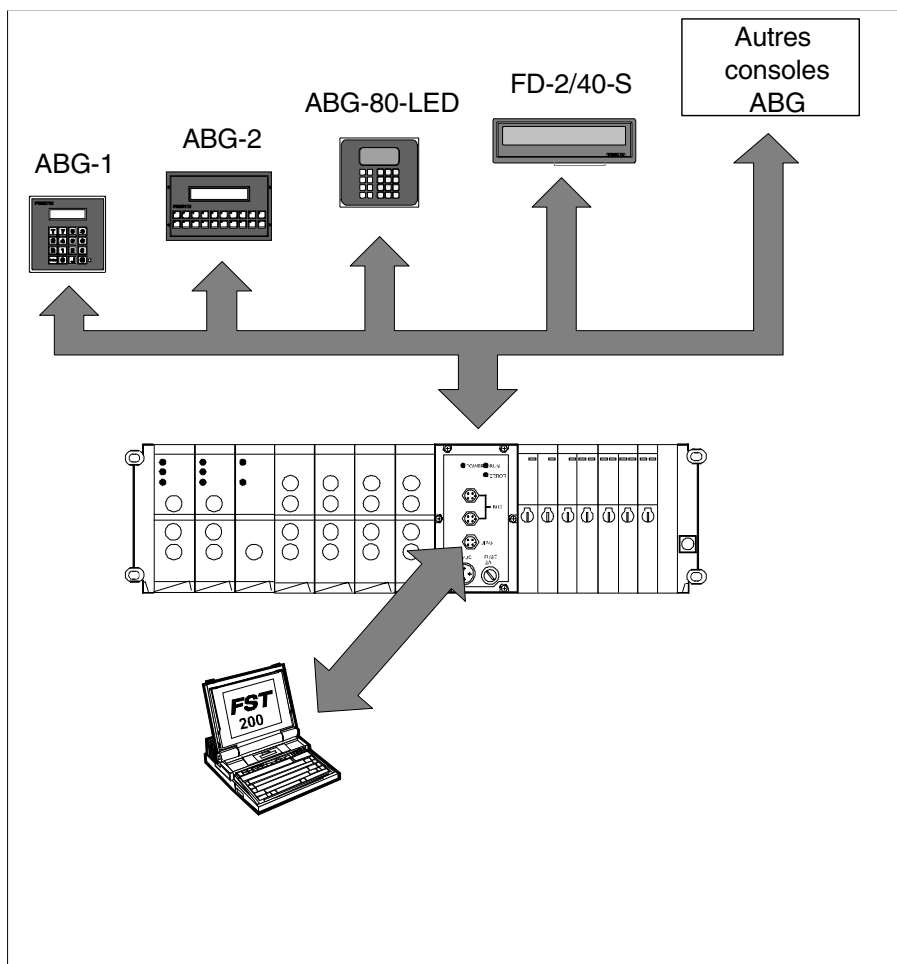


# Outils logiciels FESTO

## Affichage et commande avec l'automate SF 3

### Manuel d'utilisation



Le présent manuel "Affichage et commande avec l'automate SF 3" suppose la connaissance du progiciel Festo FST200 "Liste d'instructions et schémas à contacts du SF 3" (Version 3.2 ou plus récente), et en particulier des connaissances sur l'utilisation des blocs logiciels.

Ce manuel décrit de quelle manière un utilisateur familiarisé avec la programmation de l'automate SF 3 peut :

- créer des blocs logiciels servant de driver pour les consoles d'affichage et de commande puis les insérer dans un programme utilisateur,
- créer ou modifier les textes affichés,
- réaliser des tests à l'aide d'émulateurs d'affichage ou des tests d'écran.

Festo se réserve le droit d'apporter des modifications constituant un progrès technique.

Auteurs : D. Ruckwied, E. Klotz, P. Mauch  
Rédaction : H. J. Drung, M. Holder  
Maquette: Festo AG & Co., Abt. PV-IDM  
Mise en page : DUCOM

Edition : Août 1997

© (Festo AG & Co., D-73726 Esslingen, 1997)  
Toute communication ou reproduction de ce document, toute exploitation ou communication de son contenu sont interdites, sauf autorisation expresse. Tout manquement à cette règle est illicite et expose son auteur au versement de dommages et intérêts. Tous droits réservés, particulièrement le droit de déposer des modèles d'utilité ou des modèles de présentation.

IBM<sup>®</sup>      marque déposée de  
International Business Machines  
Corporation

Microsoft<sup>®</sup>      marque déposée de  
Microsoft Corporation

Affichage et commande - SF 3  
Drivers d'affichage et de clavier pour  
l'automate SF 3

Réf. :              377 745  
Titre :             Manuel d'utilisation  
Désignation :    P.BE-ABGSF3-F

**Sommaire**

|                                      |      |
|--------------------------------------|------|
| Remarques concernant ce manuel ..... | VI   |
| Fil directeur .....                  | VIII |

**1. Présentation**

|                                                                              |            |
|------------------------------------------------------------------------------|------------|
| <b>"Affichage et commande avec l'automate SF 3" .....</b>                    | <b>1-1</b> |
| 1.1 Consoles d'affichage et de commande installées sur l'automate SF 3 ..... | 1-3        |
| 1.1.1 Les différents types de consoles ABG .....                             | 1-6        |
| 1.1.2 Principe de fonctionnement des blocs logiciels .....                   | 1-8        |
| 1.1.3 Affichage de textes .....                                              | 1-10       |
| 1.2 Aperçu des blocs logiciels pour l'automate SF 3 .....                    | 1-12       |
| 1.2.1 Choix des blocs logiciels .....                                        | 1-12       |
| 1.2.2 Fichiers MAK pour d'autres affichages et automates .....               | 1-14       |
| 1.2.3 Logiciels utilisés, matériel et accessoires .....                      | 1-15       |
| 1.3 Différenciation SF 3 - SB202 .....                                       | 1-16       |

**2. Editeur d'affichage FST200 .....**

|                                                                        |      |
|------------------------------------------------------------------------|------|
| 2.1 Appel de l'éditeur d'affichage .....                               | 2-5  |
| 2.2 Création d'un fichier d'affichage .....                            | 2-7  |
| 2.2.1 Ecran de l'éditeur .....                                         | 2-11 |
| 2.2.2 Mode édition .....                                               | 2-14 |
| 2.2.3 Mode texte .....                                                 | 2-15 |
| 2.2.4 Mode caractères de commande .....                                | 2-19 |
| 2.2.5 Fonctions supplémentaires de l'éditeur .....                     | 2-23 |
| 2.2.6 Quitter l'édition .....                                          | 2-26 |
| 2.3 Effacement d'un fichier d'affichage .....                          | 2-28 |
| 2.4 Impression d'un fichier d'affichage .....                          | 2-29 |
| 2.5 Chargement de la RAM d'affichage par l'interface RS232 (V24) ..... | 2-31 |
| 2.6 Programmation de l'EPROM .....                                     | 2-33 |

|           |                                                                                                                |            |
|-----------|----------------------------------------------------------------------------------------------------------------|------------|
| <b>3.</b> | <b>Création des blocs logiciels<br/>à partir des fichiers MAK</b>                                              | <b>3-1</b> |
| 3.1       | Principes généraux pour la création<br>de blocs                                                                | 3-5        |
| 3.2       | Création d'un bloc à l'aide de l'éditeur<br>d'affichage                                                        | 3-8        |
| 3.3       | Création d'un bloc à l'aide de la<br>fonction Include module<br>(relier un module)                             | 3-13       |
| <b>4.</b> | <b>Description des blocs logiciels<br/>Appel des blocs dans le programme<br/>utilisateur</b>                   | <b>4-1</b> |
| 4.1       | Instructions générales concernant<br>l'appel de blocs                                                          | 4-5        |
| 4.1.1     | Initialisation de blocs ou d'affichages                                                                        | 4-6        |
| 4.1.2     | Appel cyclique de blocs                                                                                        | 4-7        |
| 4.1.3     | Analyse des bits mémoire<br>d'état/messages d'erreur                                                           | 4-8        |
| 4.1.4     | Blocs logiciels et interpréteur de<br>commandes                                                                | 4-9        |
| 4.2       | Utilisation du bloc comme driver<br>d'affichage, Textes dans le SF 3<br>(textes internes)                      | 4-10       |
| 4.3       | Utilisation du bloc comme driver<br>d'affichage, textes dans la console                                        | 4-16       |
| 4.4       | Utilisation du bloc comme driver de<br>saisie pour la scrutation du clavier<br>ou la mémoire tampon de clavier | 4-23       |
| 4.5       | Utilisation du bloc comme driver<br>de saisie pour la scrutation<br>du clavier / les témoins LED               | 4-30       |
| 4.6       | Bloc multitâche                                                                                                | 4-36       |

|           |                                                      |            |
|-----------|------------------------------------------------------|------------|
| <b>5.</b> | <b>Mise en service des consoles ABG</b>              | <b>5-1</b> |
| 5.1       | Mise en service avec émulateurs<br>d'affichage ..... | 5-4        |
| 5.1.1     | Remarques sur les émulateurs .....                   | 5-5        |
| 5.1.2     | Installation des émulateurs .....                    | 5-6        |
| 5.1.3     | Appel et arrêt d'un émulateur .....                  | 5-8        |
| 5.1.4     | Description des émulateurs .....                     | 5-10       |
| 5.2       | Test des textes d'affichage .....                    | 5-15       |
| 5.3       | Conseils pratiques .....                             | 5-17       |

## **Annexe A**

|                                            |            |
|--------------------------------------------|------------|
| <b>Résumé des blocs logiciels</b> .....    | <b>A-1</b> |
| Blocs logiciels de l'automate SF3 .....    | A-3        |
| Description des fichiers MAK (exemple) ... | A-4        |

## **Annexe B**

|                                                           |            |
|-----------------------------------------------------------|------------|
| <b>Annexe technique</b> .....                             | <b>B-1</b> |
| Réglages des consoles d'affichage et de<br>commande ..... | B-3        |
| Câbles de liaison .....                                   | B-4        |
| Messages d'erreur .....                                   | B-8        |
| Définition des concepts .....                             | B-15       |

**Remarques concernant ce manuel**

Ce manuel d'utilisation reprend les concepts spécifiques et les abréviations suivantes :

| Concept/Abréviation | Signification                                      |
|---------------------|----------------------------------------------------|
| ABG                 | Console d'affichage et de commande                 |
| FD                  | Festo Display                                      |
| DIAG                | Interface de diagnostic du SF 3 (interface RS-232) |

Des informations plus détaillées sur les consoles d'affichage et de commande sont données dans la documentation jointe aux produits.

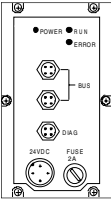
| Manuels d'utilisation pour un système SF 3 |                                                                                                                                                                 |                                                                                                                                           |                                                                                                       | Consoles d'affichage et de commande                                                                                                 |
|--------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| Manuel                                     | <p>Terminal de distributeurs programmable type 02 à 05 avec automate SF 3</p>  | <p>Outil logiciel FESTO FST200 V3.2 ou plus récente</p>  | <p>Affichage et commande avec l'automate SF 3</p>                                                     | <p>Documentation produit</p>                       |
|                                            | <p>TN 165490 (type 02)<br/>TN 165491 (type 03)</p>                                                                                                              | <p>TN 165 494</p>                                                                                                                         | <p><b>Le présent manuel :<br/>TN 377 745</b></p>                                                      |                                                                                                                                     |
| Contenu                                    | <p>Informations spécifiques sur l'installation, la mise en service, la programmation et le diagnostic</p>                                                       | <p>Informations sur l'installation des logiciels et sur la programmation STL et LDR</p>                                                   | <p>Informations spécifiques sur la création des textes d'affichage et des blocs logiciels utiles.</p> | <p>Informations sur l'installation, les réglages spécifiques et sur le diagnostic de chaque console d'affichage et de commande.</p> |
|                                            |                                                                                                                                                                 |                                                                                                                                           |                                                                                                       |                                                                                                                                     |

Fig. 0/1 :   Manuels d'utilisation concernant les systèmes SF3, y compris l'affichage et la commande



## Fil directeur

Ce paragraphe doit servir de fil directeur. Il propose une procédure très utile pour relier une console d'affichage et de commande à un automate SF3.

| Procédure |                                                                                                                                                                                                               |                                  |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------|
| 1         | Définir le type de console ABG et son mode de fonctionnement                                                                                                                                                  | Chapitre 1.1                     |
| 2         | Sélectionner les fichiers MAK pour la création de blocs logiciels                                                                                                                                             | Chapitre 1.2                     |
| 3         | Créer un fichier comportant les textes qui doivent s'afficher                                                                                                                                                 | Chapitre 2.2                     |
| 4         | Pour les textes stockés dans la console ABG : charger les textes à afficher dans la console ABG                                                                                                               | Chapitre 2.5                     |
| 5         | Créer les blocs logiciels :<br>dans l'éditeur d'affichage - avec la fonction Generate CMP/CFM for standard mode (créer CMP/CFM pour le mode standard),<br>avec la fonction Include module (relier un module). | Chapitre 3.2<br><br>Chapitre 3.3 |
| 6         | Relier les blocs logiciels au programme utilisateur                                                                                                                                                           | Partie 4                         |
| 7         | Tester l'affichage et les blocs logiciels<br>– à l'aide d'un émulateur d'affichage (textes dans l'automate SF 3),<br>– à l'aide de la fonction Display test (test d'affichage) (texte dans la console ABG)    | Chapitre 5.1<br>Chapitre 5.2     |

# **Partie 1**

## **Présentation**

### **"Affichage et commande avec l'automate SF 3"**

**Sommaire**

|            |                                                                           |             |
|------------|---------------------------------------------------------------------------|-------------|
| <b>1.1</b> | <b>Consoles d'affichage et de commande installées sur l'automate SF 3</b> | <b>1-3</b>  |
| 1.1.1      | Les différents types de consoles ABG                                      | 1-6         |
|            | Les consoles d'affichage et de commande et leurs fonctions                | 1-7         |
| 1.1.2      | Principe de fonctionnement des blocs logiciels                            | 1-8         |
| 1.1.3      | Affichage de textes                                                       | 1-10        |
|            | Choix du mode de fonctionnement                                           | 1-11        |
| <b>1.2</b> | <b>Aperçu des blocs logiciels pour l'automate SF 3</b>                    | <b>1-12</b> |
| 1.2.1      | Choix des blocs logiciels                                                 | 1-12        |
| 1.2.2      | Fichiers MAK pour d'autres affichages et automates                        | 1-14        |
| 1.2.3      | Logiciels utilisés, matériel et accessoires                               | 1-15        |
| <b>1.3</b> | <b>Différenciation SF 3 - SB202</b>                                       | <b>1-16</b> |

Ce chapitre donne un aperçu des différentes possibilités d'affichage et de commande avec l'automate SF 3.

### 1.1 Consoles d'affichage et de commande installées sur l'automate SF 3

L'interface de diagnostic de l'automate SF 3 peut être utilisée pour la programmation et le diagnostic à l'aide d'un PC mais aussi pour la communication avec la console d'affichage et de commande (ABG).

Le raccordement d'une console ABG facilite considérablement à l'utilisateur la commande de l'automate SF 3, car la plupart des consoles ABG disposent d'un afficheur et d'un clavier. Une commande simple et conventionnelle à partir d'interrupteurs et de touches est également possible par l'intermédiaire des entrées/sorties TOR, tout comme la signalisation des états de fonctionnement par des voyants.

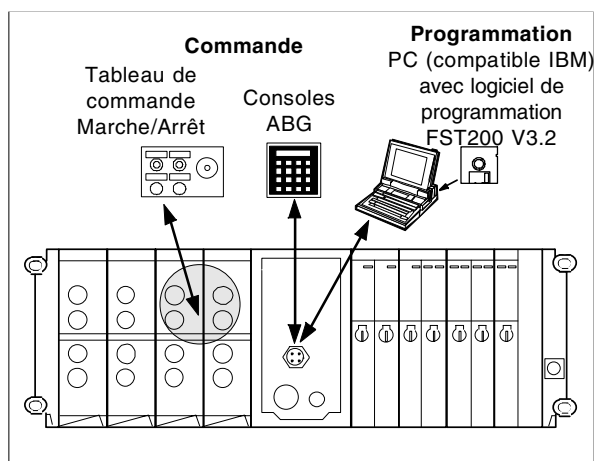


Fig. 1/1 : Structure du système "Affichage et commande" avec un automate SF 3

Les consoles d'affichage et de commande sont destinées aux applications suivantes :

- Affichage de textes définis par l'utilisateur (p. ex. affichage de messages),
- Affichage de valeurs (p. ex. nombre de pièces, temps, etc.),
- Saisie de paramètres (saisies numériques, diagnostic machine, gestion d'exécution).

Les consoles d'affichage et de commande sont gérées à l'aide de blocs de programme ou de blocs fonctionnels (CMP/CFM) spécifiques. Ces blocs logiciels permettent d'établir les fonctions de communication avec l'automate SF 3 et avec certaines consoles d'affichage et de commande. C'est pour cette raison que ces blocs logiciels sont également appelés drivers.

Le répertoire \LIB du progiciel FST contient à cet effet plusieurs fichiers MAK (fichiers "Make"). Ces fichiers MAK "faire" permettent de créer – quel que soit le type de console ABG – les blocs logiciels appropriés.

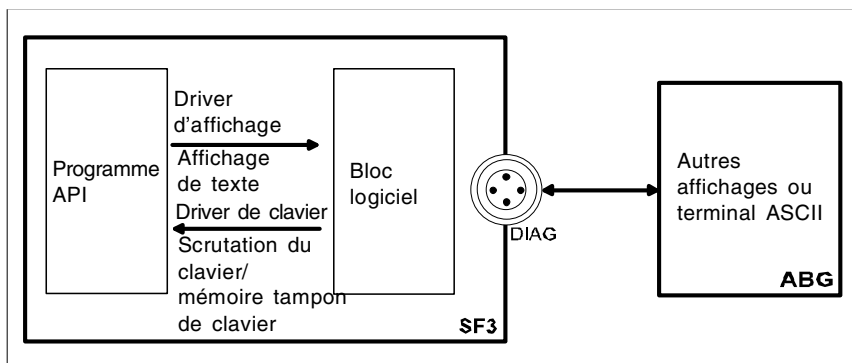


Fig. 1/2 : Blocs logiciels pour l'affichage et la commande

Le progiciel FST contient de plus les fonctions et les programmes suivants destinés à simplifier la création de textes ainsi que la programmation et la mise en service de l'affichage :

- éditeur d'affichage pour la création des textes, qui seront affichés ensuite sur la console ABG,
- fonctions permettant de créer à partir des fichiers MAK les blocs logiciels de l'affichage. Suivant l'afficheur, on a :
  - a) dans le menu "Utilities" (utilitaires), la fonction  
Include module (relier un module)
  - b) dans l'éditeur d'affichage, la fonction  
Generate CMP/CFM for standard mode  
(créer CMP/CMF pour le mode standard),
- des émulateurs pour simuler les consoles ABG Festo pendant l'étape de mise en service (le PC doit être raccordé),
- un programme de test des affichages, grâce auquel il est possible d'appeler les textes mémorisés dans l'afficheur.  
D'autre part, l'éditeur d'affichage permet d'afficher les textes contenus dans le PC.

### 1.1.1 Les différents types de consoles ABG

Il existe différents types de consoles d'affichage et de commande pour l'automate SF3, disposant des fonctions suivantes :

- Affichage de textes
  - affichage de textes mémorisés dans l'automate SF 3 (textes internes)
  - affichage de textes mémorisés dans la console ABG (textes externes)
  - affichage de variables (valeurs numériques) d'un programme en cours d'exécution
- Scrutation du clavier (contrôle direct des touches, le bit mémoire correspondant est scruté cycliquement par le programme)
- Mémoire tampon de clavier (saisie possible jusqu'à cinq caractères dans une mémoire tampon/un registre, scrutation indirecte)
- Touches avec LED (éteintes/allumées)

Les afficheurs permettent la représentation de la plupart des caractères alphanumériques. Suivant le type de console ABG, ils disposent de 1 à 4 lignes avec 16 à 40 caractères. Les textes peuvent être mémorisés aussi bien dans l'automate SF 3 ("Mode standard") que dans la console ABG elle-même.

Les saisies au clavier sont scrutées par un système programmé de contrôle cyclique des touches. La plupart des consoles disposent d'une mémoire tampon de clavier, qui assure le stockage intermédiaire des valeurs numériques saisies.

De plus, des LED supplémentaires peuvent être associées aux touches (p. ex. sur la console ABG2).

## Les consoles d'affichage et de commande et leurs fonctions

| Type de console<br>ABG                                                                                                                                        | Fonction | Propriétés                                                                                                                                                                                                                                                                        |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| E.ABG-1<br>(E.ABG-FPC,<br>-LED, -EL, -VF)                                                                                                                     | A/C/P    | Afficheur LCD, 2 lignes de 16 caractères,<br>Touches de fonction, clavier numérique 10<br>touches (extension de mémoire possible<br>avec E.ABG-MEM)                                                                                                                               |
| E.ABG-2                                                                                                                                                       | A/C/L    | Témoins LED, 2 lignes de 40 caractères,<br>Touches de fonction avec LED intégrées                                                                                                                                                                                                 |
| E.ABG-80-LED                                                                                                                                                  | A/C/P    | Témoins LED, 4 lignes de 20 caractères,<br>Touches de fonction, clavier numérique                                                                                                                                                                                                 |
| FD-1/40-P<br>FD-1/40-S                                                                                                                                        | A        | Témoins LED, 1 ligne de 40 caractères                                                                                                                                                                                                                                             |
| E.FD-2/40-P<br>E.FD-2/40-S                                                                                                                                    | A        | Témoins LED, 2 lignes de 40 caractères                                                                                                                                                                                                                                            |
| Autres affichages<br>ou<br>terminaux ASCII                                                                                                                    | A/C/P    | suivant l'affichage :<br>Affichage de textes, de variables, de<br>caractères de commande, appel de textes,<br>Saisie de caractères ASCII quelconques ou<br>de valeurs numériques pour les variables. La<br>communication s'effectue par l'intermédiaire<br>de l'interface RS-232. |
| Consoles PCS                                                                                                                                                  | A/C/P    |                                                                                                                                                                                                                                                                                   |
| Fonctions :   A = affichage,<br>C = scrutation du clavier,<br>P = mémoire tampon de clavier,<br>L = LED<br>(fait en juin 1997, sous réserve de modifications) |          |                                                                                                                                                                                                                                                                                   |



### 1.1.2 Principe de fonctionnement des blocs logiciels

Dans toutes les consoles d'affichage et de commande disponibles sur le marché, les blocs logiciels sont appelés cycliquement par un programme. La communication entre le programme qui appelle et le bloc logiciel met en jeu des opérandes, comme p. ex. les registres, les bits mémoire ou les unités fonctionnelles spécifiques (FU).

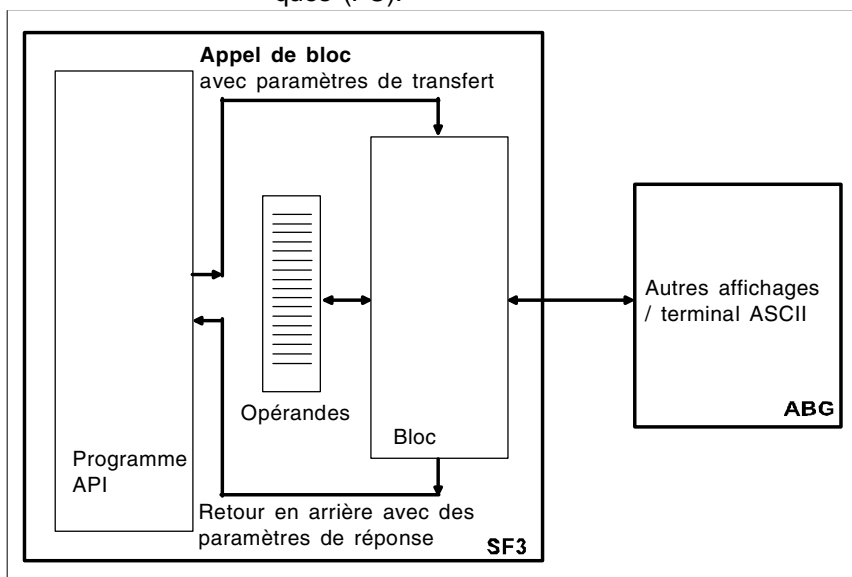


Fig. 1/3 : Principe du driver d'affichage et du driver de clavier

Les consoles ABG sont raccordées sur l'interface de diagnostic (RS232) du SF 3. Le bloc logiciel gère la communication avec l'afficheur par l'intermédiaire de cette interface RS-232.

On peut distinguer deux types de blocs logiciels d'après la fonction qu'ils assurent :

- les blocs **clavier** ou drivers de **saisie**  
Ce sont des blocs logiciels destinés à la scrutation/au contrôle des fonctions attribuées à des touches sur les consoles.
- les blocs **texte** ou drivers **d'affichage**  
Ce sont des blocs logiciels destinés aux fonctions d'affichage de textes sur les consoles.

### 1.1.3 Affichage de textes

Suivant le type d'afficheur, il existe différentes possibilités pour afficher du texte sur l'écran de la console ABG :

- Textes stockés dans l'automate SF 3 (textes internes)

Ce mode est appelé mode standard et est en général accessible sur tous les afficheurs par l'interface série. Les textes peuvent être modifiés aussi rapidement que facilement mais ils occupent de l'espace mémoire dans l'automate SF 3 et mobilisent les capacités de calcul de l'automate (ce qui réduit les performances du système).

- Textes stockés dans l'afficheur (textes externes)
  - mode RAM – pour les afficheurs disposant d'une mémoire RAM intégrée
  - mode EPROM – pour les afficheurs disposant d'une mémoire EPROM
  - mode bus de terrain – pour les afficheurs disposant d'une mémoire RAM ou EPROM et d'une interface bus de terrain.

Dans ce cas, le nombre de données qui peuvent être échangées entre le SF 3 et l'afficheur est limité. Les textes n'occupent aucun espace mémoire dans l'automate SF 3 ce qui diminue le temps de cycle du programme. En revanche, les textes ne peuvent pas être modifiés aussi rapidement.

### **Choix du mode de fonctionnement**

Au cours de la phase de mise en service, il est possible de décider si le texte doit être mémorisé dans l'automate SF 3 ou dans l'afficheur. La décision tiendra compte des réponses aux questions suivantes :

- Quels sont les modes de fonctionnement compatibles avec l'affichage ?
- Le traitement du texte doit-il être rapide ?
- L'automate SF 3 dispose-t-il d'une mémoire suffisante pour les textes ?
- Le transfert des textes entre l'automate et l'afficheur va-t-il augmenter de façon excessive le temps de cycle du programme ?

Pendant la phase de développement, il est préférable de mémoriser les textes dans l'automate SF 3 (Mode standard, textes dans l'afficheur). La modification des textes demande dans ce cas moins de temps.

## 1.2 Aperçu des blocs logiciels pour l'automate SF 3

Les fichiers MAK de création des blocs logiciels font partie du progiciel FST et sont stockés dans le répertoire de la bibliothèque (\LIB) au moment de l'installation du progiciel. La fonction Include module (relier un module) permet alors d'inclure directement dans le projet en cours les blocs logiciels à partir du répertoire \LIB.

### 1.2.1 Choix des blocs logiciels

Pour choisir le bloc logiciel approprié, procéder de la manière suivante :

1. Chercher dans les tableaux suivants la ligne correspondant à votre afficheur et dans les colonnes "Fonction" et "Description du mode de fonctionnement" le mode de fonctionnement désiré.
2. Sélectionner un des fichiers MAK sur la ligne correspondante. Les fichiers comportant un trait de soulignement ( \_ ) servent à créer des blocs CMP, ceux comportant un accent circonflexe (^) à créer des blocs CFM.
3. La commentaire sur les fichiers MAK indique comment le bloc logiciel doit être créé :
  - avec l'éditeur d'affichage du FST200, à l'aide de la fonction Generate CMP/CFM for standard mode (créer CMP/CFM pour le mode standard)
  - avec le logiciel FST, à l'aide de la fonction Include module (relier module).

Vous trouverez des informations détaillées ainsi que la procédure à suivre dans les chapitres suivants :

- Création des blocs logiciels  
⇒ Partie 3.
- Description des blocs logiciels  
Appel des blocs dans le programme utilisateur  
⇒ Partie 4.

Attribution des fichiers MAK aux affichages :

| Afficheur                                                                                                                                                                                    | Fonction                                               | Description du mode de fonctionnement                                       | Fichiers MAK correspondants                                                                                          |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|-----------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| E.ABG-1<br>E.ABG-FPC<br>E.ABG-LED<br>E.ABG-EL<br>E.ABG-VF                                                                                                                                    | Affichage                                              | Driver pour autres affichages,<br>Texte dans le SF 3                        | 3FD_XABG.MAK<br>3FD^XABG.MAK <sup>1)</sup>                                                                           |
|                                                                                                                                                                                              |                                                        | Driver d'affichage<br>E.ABG-LED, E.ABG-80-LED,<br>Texte dans la console ABG | 3FD_216S.MAK<br>3FD^216S.MAK <sup>2)</sup>                                                                           |
|                                                                                                                                                                                              | Scrutation du clavier<br>Mémoire tampon de clavier     | Driver de clavier E.ABG-LED,<br>Valeur intégrée                             | 3_EABG.MAK<br>3^EABG.MAK <sup>2)</sup>                                                                               |
| E.ABG-2                                                                                                                                                                                      | Afficheur                                              | Driver pour autres affichages,<br>Texte dans le SF 3                        | 3FD_XABG.MAK<br>3FD^XABG.MAK <sup>1)</sup>                                                                           |
|                                                                                                                                                                                              | Scrutation du clavier et activation des LED de touches | Driver de scrutation du clavier/commande des LED E.ABG-2                    | 3_EABG2.MAK<br>3^EABG2.MAK <sup>2)</sup>                                                                             |
| E.FD-1/40S<br>E.FD-1/40P<br>E.FD-2/40S<br>E.FD-2/40F                                                                                                                                         | Affichage                                              | Driver pour autres affichages,<br>Textes dans le SF3                        | 3FD_XABG.MAK<br>3FD^XABG.MAK <sup>1)</sup>                                                                           |
|                                                                                                                                                                                              |                                                        | Driver d'affichage<br>FD-X40/S, Textes dans la console ABG                  | 3FD_240S.MAK<br>3FD^240S.MAK <sup>2)</sup>                                                                           |
| E.ABG-80-LED                                                                                                                                                                                 | Affichage                                              | Driver pour autres affichages,<br>Textes dans le SF 3                       | 3FD_XABG.MAK<br>3FD^XABG.MAK <sup>1)</sup>                                                                           |
|                                                                                                                                                                                              |                                                        | Driver d'affichage E.ABG-LED,<br>E.ABG-80-LED, Textes dans la console ABG   | 3FD_216S.MAK<br>3FD^216S.MAK <sup>2)</sup>                                                                           |
|                                                                                                                                                                                              | Scrutation du clavier<br>Mémoire tampon de clavier     | Driver de clavier E.ABG-80-LED                                              | 3_ABG80.MAK<br>3^ABG80.MAK <sup>2)</sup>                                                                             |
| Autres affichages / Terminal ASCII                                                                                                                                                           | Affichage et saisie                                    | Driver de saisie/ affichage de caractères (bloc multitâche)                 | 3_ETMV5.MAK<br>3^ETMV5.MAK <sup>2)</sup>                                                                             |
| Ligne de données (autres affichages)                                                                                                                                                         | Affichages                                             | Driver pour autres affichages quelconques, Textes dans le SF 3              | 3FD_XABG.MAK<br>3FD^XABG.MAK <sup>1)</sup>                                                                           |
| Consoles PCS                                                                                                                                                                                 | Affichage et saisie                                    | Driver PCS<br>Consoles de commande                                          | 3_PCS.MAK<br>3^PCS.MAK <sup>2)</sup>                                                                                 |
| <sup>1)</sup> Création de blocs CMP/CFM avec l'éditeur d'affichage, à l'aide de la fonction "Generate CMP (or CFM) for standard mode (créer CMP/CFM pour le mode standard)" (paragraphe 3.2) |                                                        |                                                                             | <sup>2)</sup> Création de blocs CMP/CFM à l'aide de la fonction "Include module (relier un module)" (paragraphe 3.3) |
| fait en juin 1997, sous réserve de modifications                                                                                                                                             |                                                        |                                                                             |                                                                                                                      |

Lors de l'installation du FST200, dans la version 3.2 ou une version plus récente – révision 8 (version 8/97) les fichiers MAK sont chargés automatiquement sur le PC (dans le répertoire \LIB). Pour spécifier la version du logiciel FST, appuyer sur la touche F10 du menu principal. La version s'inscrit alors dans la ligne d'état (p. ex. 3.2-0008).

Les fichiers MAK doivent ensuite individuellement être reliés sous forme de blocs logiciels au projet en cours à l'aide des fonctions Generate CMP/CFM for standard mode (créer CMP/CFM pour le mode standard) ou Include module (relier un module).

### **1.2.2 Fichiers MAK pour d'autres affichages et automates**

Les autres affichages ou automates peuvent également utiliser des fichiers MAK. Ces derniers ne sont cependant pas fournis avec le progiciel FST200. Les drivers d'affichage joints sont importés par le programme d'installation lui-même.

Les drivers d'affichage ne faisant pas partie du progiciel FST200, doivent être importés par le logiciel FST dans le répertoire \LIB (répertoire de la bibliothèque). Se renseigner dans ce cas à l'aide de la documentation du driver d'affichage.

### 1.2.3 Logiciels utilisés, matériel et accessoires

La mise en service d'une console ABG nécessite :

- un automate SF 3 avec une version de logiciel V1.2 et un équipement de terminal de distributeurs quelconque.  
(Le bloc de saisie/affichage de caractères à travers l'interface de diagnostic (3\_ETMV5.MAK/3^ETMV5.MAK) requiert la version de logiciel V1.5. Le numéro de version peut être affiché à l'aide de la fonction Online mode (mode Online),
- un logiciel de programmation FST200 avec manuel d'utilisation,
- une console d'affichage et de commande avec câble de raccordement (voir Annexe B).



### 1.3 Différenciation SF 3 - SB202

Tous les blocs de programme disponibles pour le SB202, le sont également pour le SF 3. De plus, le SF 3 bénéficie de blocs fonctionnels (CFM) destinés au pilotage de consoles d'affichage et de commande. La désignation du fichier MAK permet de reconnaître à quels types d'automates et de blocs le fichier est destiné (exemple) :

| <b>2FD_240S.MAK<br/>3FD^240S.MAK</b> | <b>Signification de la désignation du fichier</b>                                                                                                                                                                 |
|--------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>2<br/>3</b>                       | Type d'automate (1=101, 2=202c/SB202, 3=SF 3, 4=404, 5=405)                                                                                                                                                       |
| <b>FD</b>                            | Type d'écran, ici :<br>FestoDisplay (FD)                                                                                                                                                                          |
| <b>_<br/>^</b>                       | _ le fichier permet de créer un bloc de programme (CMP)<br>^ le fichier permet de créer un bloc fonctionnel (CFM)                                                                                                 |
| <b>2</b>                             | Dépend du type d'écran, ici :<br>Nombre de lignes :<br>1= mode RAM/EPROM : afficheur une ligne FD-1/40-S<br>2= mode RAM/EPROM : afficheur deux lignes FD-2/40-S<br>X= mode Standard : pour FD-1/40-S et FD-2/40-S |
| <b>40</b>                            | Dépend de l'écran, ici :<br>Nombre de colonnes (40=40 colonnes)                                                                                                                                                   |
| <b>S</b>                             | Dépend de l'écran, ici :<br>Type de transmission (S=série ; P=parallèle ; F=bus de terrain)                                                                                                                       |
| <b>.MAK</b>                          | Fichier permettant de créer un bloc logiciel.                                                                                                                                                                     |

Dans certains cas les champs de dialogue apparaissant lors de la création des blocs diffèrent légèrement entre les automates SF 3 et SB202.

## **Partie 2**

### **Editeur d'affichage FST200**

**Sommaire**

|            |                                            |             |
|------------|--------------------------------------------|-------------|
| <b>2.1</b> | <b>Appel de l'éditeur d'affichage</b>      | <b>2-5</b>  |
| <b>2.2</b> | <b>Création d'un fichier d'affichage</b>   | <b>2-7</b>  |
| 2.2.1      | Ecran de l'éditeur                         | 2-11        |
|            | Généralités                                | 2-12        |
| 2.2.2      | Mode édition                               | 2-14        |
| 2.2.3      | Mode texte                                 | 2-15        |
|            | Saisie et effacement de texte              | 2-16        |
|            | Insertion et écrasement de                 |             |
|            | caractères de texte                        | 2-16        |
|            | Caractères blancs                          | 2-17        |
|            | Saisie de caractères blancs                | 2-18        |
| 2.2.4      | Mode caractères de commande                | 2-19        |
|            | Saisie de caractères de commande           | 2-19        |
|            | Modification de la séquence de             |             |
|            | commande                                   | 2-22        |
| 2.2.5      | Fonctions supplémentaires de l'éditeur     | 2-23        |
|            | Rechercher                                 | 2-23        |
|            | Copie d'un texte d'affichage               | 2-24        |
|            | Insérer et effacer le texte d'affichage    |             |
|            | (numéro de texte)                          | 2-25        |
| 2.2.6      | Quitter l'édition                          | 2-26        |
|            | Save and leave the editor                  |             |
|            | (sauvegarder puis quitter l'éditeur)       | 2-26        |
|            | Save and continue (sauvegarde              |             |
|            | intermédiaire)                             | 2-26        |
|            | Abort edit (annuler l'édition)             | 2-27        |
| <b>2.3</b> | <b>Effacement d'un fichier d'affichage</b> | <b>2-28</b> |
| <b>2.4</b> | <b>Impression d'un fichier d'affichage</b> | <b>2-29</b> |
|            | Format d'impression                        | 2-30        |

|            |                                                                     |      |
|------------|---------------------------------------------------------------------|------|
| <b>2.5</b> | <b>Chargement de la RAM d'affichage par l'interface RS232 (V24)</b> | 2-31 |
|            | Opération de chargement                                             | 2-33 |
| <b>2.6</b> | <b>Programmation de l'EPROM</b>                                     | 2-33 |
|            | Généralités                                                         | 2-33 |
|            | Programmation de l'EPROM –                                          |      |
|            | Méthode 1                                                           | 2-35 |
|            | Sélection du programmeur EPROM                                      | 2-37 |
|            | Autres programmeurs EPROM                                           | 2-38 |
|            | Intellec-Hex-Dump                                                   | 2-38 |
|            | Intellec-Hex-Dump to File<br>(comme fichier)                        | 2-39 |
|            | Sélection de la configuration<br>de l'interface                     | 2-39 |
|            | Sélection du type d'EPROM                                           | 2-40 |
|            | Programmation de l'EPROM                                            | 2-41 |
|            | Programmation de l'EPROM –                                          |      |
|            | Méthode 2                                                           | 2-42 |

L'éditeur d'affichage FST permet en premier lieu d'utiliser l'afficheur de textes Festo. Il est également possible d'utiliser des afficheurs de textes d'autres fabricants si ceux-ci sont équipés d'une interface série.

L'éditeur d'affichage est appelé à partir du logiciel FST. Les fonctions disponibles dépendent du type d'afficheur :

| Affichage                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Référence                                                                                            | Fonctions disponibles |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|-----------------------|
| E.ABG-1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | 8345 (E.ABG-EL,<br>N° de série > 1999)<br>8345 (E.ABG-LED,<br>N° de série > 4100)<br>8346 (E.ABG-VF) | 1,2,3,4,5             |
| E.ABG-80-LED                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | 271357                                                                                               | 1, 2, 3, 4, 5         |
| E.FD-1/40P                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 8208                                                                                                 | 1, 2, 5               |
| E.FD-1/40S                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 8206                                                                                                 | 1, 2, 3               |
| E.FD-2/40S                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 8207                                                                                                 | 1, 2, 3, 4, 5         |
| E.FD-2/40F                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | 8172                                                                                                 | 1, 2, 3, 4, 5         |
| E.ABG-2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | 268994                                                                                               | 1, 2, 3               |
| Ligne de données                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | au choix                                                                                             | 1, 2, 3               |
| 1 = Création d'un fichier d'affichage avec données de texte et de commande.<br>2 = Impression des fichiers d'affichage créés.<br>3 = Création d'un bloc de programme logiciel (CMP) ou d'un bloc fonctionnel (CFM), pour transférer des textes au cours d'un programme, de l'automate SF 3 à l'afficheur de textes.<br>4 = Chargement d'un fichier d'affichage dans la mémoire RAM de l'afficheur par l'interface RS232 (V24).<br>5 = Transfert/Programmation d'un fichier d'affichage dans l'EPROM de l'afficheur. |                                                                                                      |                       |

## 2.1 Appel de l'éditeur d'affichage

Un fichier d'affichage fait partie d'un projet. Il est impératif de sélectionner un projet avant de lancer l'éditeur d'affichage.

L'éditeur d'affichage ne peut être lancé qu'à partir du logiciel FST. Activer pour cela dans le menu "Utilities" (utilitaires) la fonction Configuration. Saisir sur l'écran Program call (appel du programme) (F4) les indications correspondantes (voir fig. suivante).



Ces saisies sont effectuées lors de l'installation du logiciel FST par le programme d'installation.

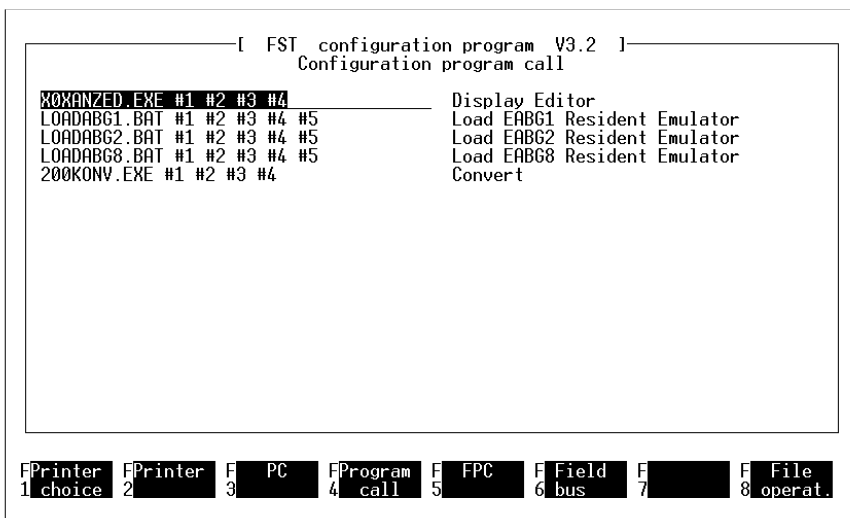


Fig. 2/1 : Indications sur l'exécution du programme

Le programme d'installation saisit toutes les informations nécessaires. Les paramètres (#1 #2 #3 #4) permettent à l'éditeur d'affichage de reconnaître la configuration de l'ordinateur, le chemin d'accès au projet actuel et le type d'automate utilisé (voir Manuel FST Chapitre 3.10.1).

Le programme d'installation entre comme nom de programme l'éditeur d'affichage. Activer dans le logiciel FST dans le menu "Utilities" (utilitaires) la fonction Program execution (exécution du programme), le nom apparaît dans la fenêtre de sélection (voir fig. suivante).

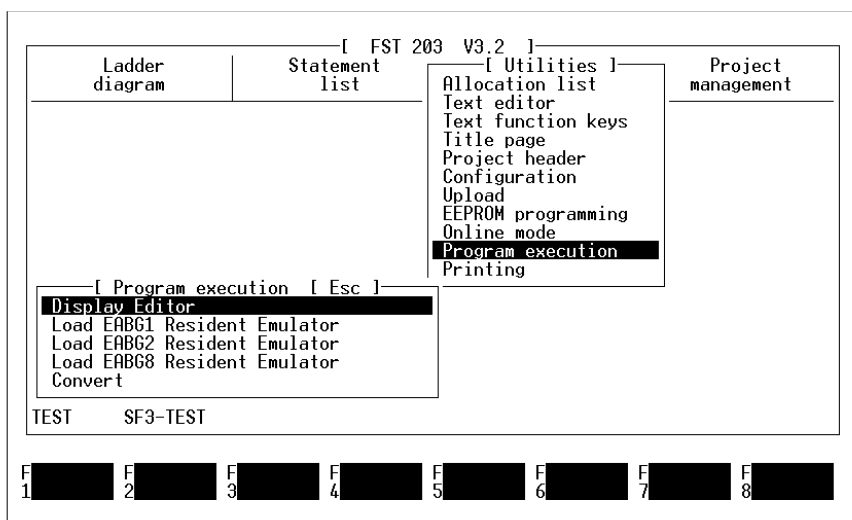


Fig. 2/2 : Appel de programmes

Activer cette entrée pour lancer le programme. La touche F9 permet de demander une aide contextuelle, comme dans le logiciel FST.

## 2.2 Création d'un fichier d'affichage

L'éditeur d'affichage permet en plus de l'édition de textes d'exécuter d'autres fonctions. Les différentes fonctions apparaissent une fois que l'éditeur d'affichage a été sélectionné (voir fig. suivante). Il est possible :

- d'éditer des fichiers d'affichage nouveaux ou existants,
- d'effacer des fichiers d'affichage existants,
- d'imprimer des fichiers d'affichage,
- de charger des fichiers d'affichage dans une EPROM ou dans la mémoire RAM de l'afficheur,
- de créer un bloc de programme ou un bloc fonctionnel pour le fonctionnement standard,
- de quitter l'éditeur d'affichage.

Pour créer un fichier d'affichage, activer la fonction Edit file (éditer fichier).

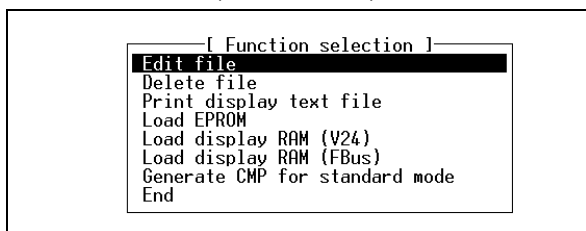


Fig. 2/3 :Menu de sélection des fonctions de l'éditeur d'affichage



La suite de la procédure est variable selon que le projet sélectionné :

- ne contient pas encore de fichier d'affichage,
- contient déjà un fichier d'affichage.

**Si le projet ne contient pas de fichier d'affichage :**

Si aucun fichier n'existe pour le projet activé, la fonction New file name (nouveau nom de fichier) apparaît dans le menu de sélection des fichiers.



Fig. 2/4 : Menu de sélection des fichiers

La fonction New file name (nouveau nom de fichier) permet de créer un nouveau fichier dans le projet actuel. La fenêtre suivante apparaît après activation de cette fonction.

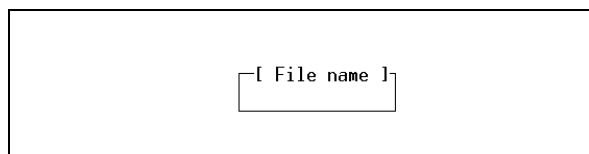


Fig. 2/5 : Saisie du nom du fichier

Entrer un nom de fichier de huit caractères maximum. Il ne peut contenir que des lettres, des chiffres et le trait de soulignement ( \_ ). Confirmer la saisie, la sélection du type d'afficheur apparaît.

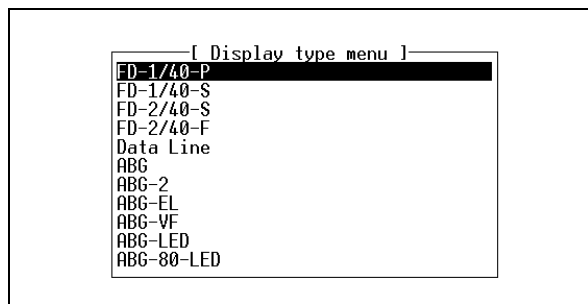


Fig. 2/6 : Sélection du type d'afficheur

La sélection du type d'afficheur permet de communiquer au programme s'il s'agit d'un afficheur à une ou plusieurs lignes et quels caractères de commande peuvent être reconnus. Il est ensuite possible d'éditer le fichier (voir chapitre 2.2.2).

**Si le projet contient déjà un fichier d'affichage :**

Si le projet contient déjà un fichier d'affichage, le menu de sélection des fichiers suivant apparaît :

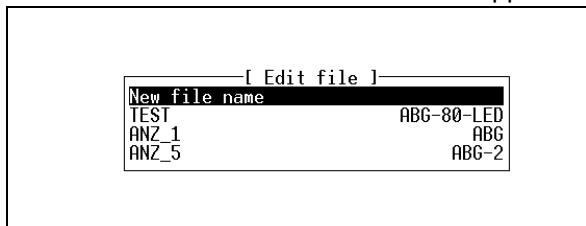


Fig. 2/7 : Sélection du fichier

A côté du nom du fichier figure le type d'afficheur pour lequel les fichiers ont été créés. Le fichier à éditer peut alors être sélectionné. La fonction New file name (nouveau nom de fichier) permet d'ajouter un autre fichier au projet.

### 2.2.1 Ecran de l'éditeur

L'éditeur est activé après qu'un fichier a été sélectionné ou créé. L'écran d'édition dépend du type d'afficheur sélectionné. Le nom du fichier d'affichage actuel figure dans l'en-tête. La figure suivante représente l'écran d'édition pour un afficheur de textes à deux lignes.

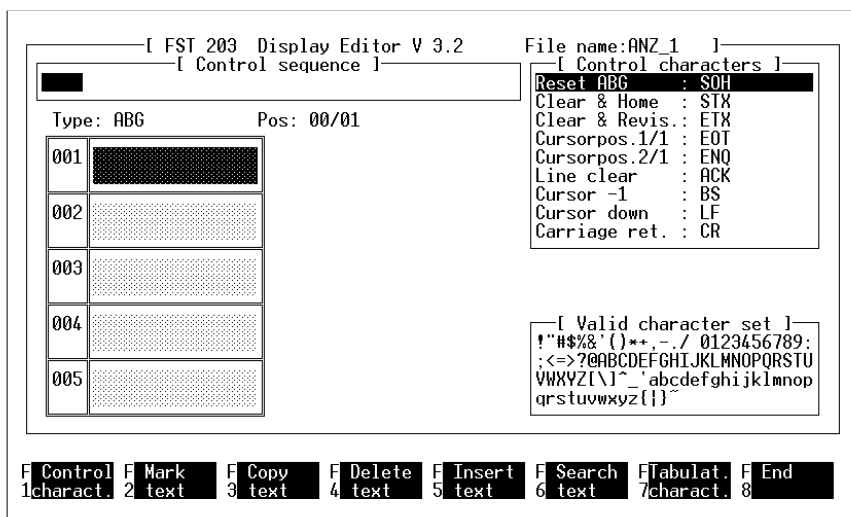


Fig. 2/8 : Ecran d'édition d'un afficheur à deux lignes

On peut voir quatre fenêtres :

- la fenêtre de séquence de commande (en haut à gauche) :  
Indique les caractères de commande insérés au point actuel du curseur de la fenêtre de texte. L'ensemble de ces caractères de commande forme la séquence de commande.
- le menu des caractères de commande (en haut à droite) :  
Contient les caractères de commande autorisés pour le type d'afficheur avec un texte explicatif.
- la fenêtre de texte :  
Entrer ici le texte qui doit apparaître ensuite sur l'afficheur.
- le jeu de caractères valide (en bas à droite) :  
Indique le jeu de caractères valide pour l'afficheur.

Le clavier et la souris offrent les mêmes possibilités que dans le logiciel FST.

### **Généralités**

Un fichier d'affichage comprend du texte et des caractères de commande. Entrer en général le texte dans la fenêtre de texte à la position où il devra plus tard apparaître sur l'afficheur.

Les caractères de commande de certains afficheurs permettent de représenter le texte sous différentes formes (p. ex. souligner certains caractères). Il est donc possible d'attribuer à chaque position du texte plusieurs caractères de commande. Le nombre maximal de caractères de commande dépend de la mémoire de l'afficheur et de la longueur de la fenêtre de séquence de commande. L'ensemble des caractères de commande d'une position forme la séquence de commande (voir fig. suivante).

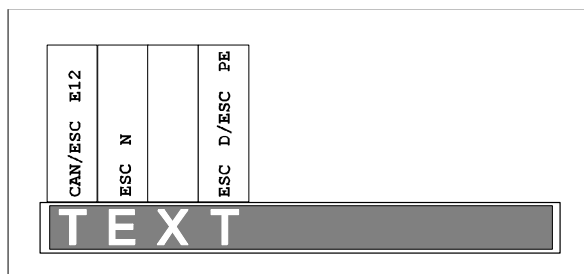


Fig. 2/9 :Caractères de texte et de commande

La fenêtre de séquence de commande indique la séquence qui se rapporte au caractère du texte actuel. Si le curseur se trouve à la position E de TEXT, le caractère de commande ESC N figure dans la fenêtre de séquence.

### Transmission des textes :

1. Les positions sont traitées successivement en commençant par la première position (à gauche) de la première rangée.
2. Lorsqu'une position contient une séquence de commande, celle-ci est transmise avant le caractère de texte.
3. Le caractère de texte est transmis après la séquence de commande (si elle existe).
4. La position suivante est traitée. Cette opération se répète jusqu'à ce que toutes les positions de texte soient traitées.

## 2.2.2 Mode édition



A l'appel de l'éditeur, le mode texte est actif. Utiliser la touche de fonction F1 ou cliquer sur la fenêtre correspondante pour changer de mode d'édition :

- texte ou
- caractères de commande.

Selon le mode, il est possible d'insérer ou de modifier du texte ou des caractères de commande dans le fichier. Le tableau ci-dessous donne une brève description. Des descriptions plus détaillées figurent dans les paragraphes suivants.

| Caractères de texte                                                                                                                           | Caractères de commande                                                                                                                                                                                                  |
|-----------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Avant la saisie du caractère :</b><br>Sélectionner le numéro de texte où l'on désire saisir du texte.                                      | <b>Avant la saisie du caractère :</b><br>Placer le curseur sur le caractère de texte où l'on désire saisir un caractère de commande.                                                                                    |
| <b>Activer la fonction :</b><br>Enter text (saisir texte) (F1), ou activer la fenêtre de texte avec la souris si elle n'est pas déjà activée. | <b>Activer la fonction :</b><br>Control character (caractère de commande) (F1), ou activer la fenêtre de séquence ou celle des caractères de commande avec la souris.                                                   |
| <b>La saisie s'effectue :</b><br>à l'aide du clavier.                                                                                         | <b>La saisie s'effectue :</b><br>à l'aide de la souris en sélectionnant le caractère de commande dans le menu des caractères de commande (cliquer 2 fois), ou à l'aide des touches de direction et de la touche Entrée. |
| <b>Le caractère saisi figure :</b><br>dans la fenêtre de texte.                                                                               | <b>Le caractère saisi figure :</b><br>dans la fenêtre de séquence.                                                                                                                                                      |
| <b>Complément :</b><br>aucune particularité                                                                                                   | <b>Complément :</b><br>Le caractère de texte concerné est marqué en couleur. En cas d'écran monochrome, le caractère blanc devient noir.                                                                                |

### 2.2.3 Mode texte



Activer le mode texte à l'aide de la fonction Enter text (saisir texte) (F1) ou en cliquant sur la fenêtre de texte à l'aide de la souris. Les fonctions d'édition disponibles dans ce mode sont décrites dans les paragraphes suivants :

- Saisie de caractères de texte.
- Insertion et effacement de caractères de texte. Lors de l'effacement d'un caractère de texte, la séquence de commande qui lui est attribuée est également effacée.
- Saisie de caractères blancs.
- Fonctions supplémentaires.

L'en-tête d'une fenêtre activée est mis en surbrillance.

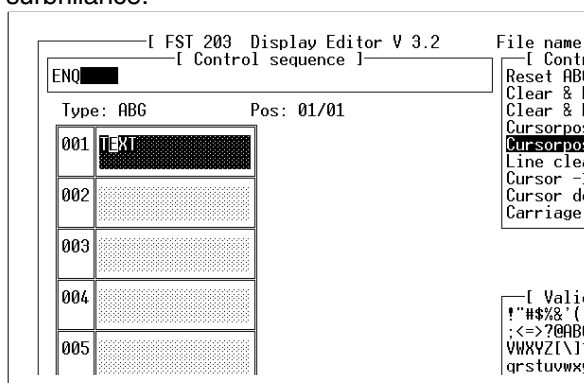


Fig. 2/10 : En-tête de la fenêtre de texte

L'en-tête de la fenêtre de texte contient des informations sur le type d'afficheur sélectionné et sur la position du curseur dans la fenêtre de texte.



### **Saisie et effacement de texte**



Sélectionner à l'aide des touches de direction ou de la souris le numéro du texte où l'on désire saisir ou modifier le texte. On peut ensuite éditer le texte de la manière habituelle. Pour effacer un caractère, utiliser la touche DEL (Suppr). La séquence de commande qui lui est attribuée est également effacée (voir paragraphe 2.2.4).

### **Insertion et écrasement de caractères de texte**



La touche INS (=Inser) permet de passer du mode insertion au mode reffappe. La forme du curseur indique quel est le mode actif. Dans le mode insertion, il a la forme d'un rectangle, en mode reffappe, celle d'un trait. Lors de l'insertion, les caractères de texte et les caractères de commande qui leur sont attribués sont décalés vers la droite.

Les caractères de texte et de commande qui dépassent à droite sont perdus !

Le mode reffappe permet d'écraser des caractères de texte. Les séquences de commande ne sont ni modifiées, ni effacées.

### Caractères blancs

Le caractère blanc est un caractère de texte particulier. L'appel d'un nouveau texte dans l'afficheur ne modifie pas les positions auxquelles un caractère blanc est transmis. L'ancien texte n'est pas modifié à ces endroits.

Il est ainsi possible de placer des textes quasi les uns sur les autres et de construire un texte avec plusieurs numéros de texte (voir fig. suivante).

Le caractère blanc n'est pas utilisable avec tous les types d'afficheur.

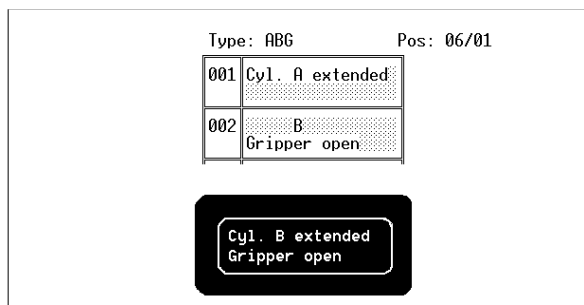


Fig. 2/11 : Superposition de textes

La figure montre que des caractères du premier texte peuvent être en partie écrasés (ici la lettre A est écrasée par la lettre B).

Si l'affichage est effacé par un caractère de commande, et si un texte est ensuite transmis, le caractère blanc ne se distingue pas de l'espace. Pourtant même dans ce cas, il faut préférer le caractère blanc à l'espace pour économiser de l'espace mémoire. Après création d'un fichier d'affichage, tous les textes d'affichage sont décrits avec le caractère blanc (voir fig. 2/8).



### **Saisie de caractères blancs**

Pour écraser un caractère de texte avec un caractère blanc, activer la fonction F7 Tabulator character (caractère blanc). Les caractères de commande ne sont pas modifiés.

### 2.2.4 Mode caractères de commande

Ce mode permet de saisir des caractères de commande. Leur nombre et leur capacité dépend de l'afficheur utilisé. D'autres fonctions d'édition sont disponibles dans ce mode :

- saisir ou effacer un caractère de commande ou une séquence de commande (plusieurs caractères de commande) pour un caractère de texte,
- modifier une séquence de commande en insérant ou effaçant des caractères de commande.

Les séquences de commande peuvent également être effacées en mode texte.

#### Saisie de caractères de commande



Placer d'abord en mode texte le curseur à l'endroit du fichier texte où l'on désire saisir ou modifier des caractères de commande. Activer la fonction Control character (caractère de commande) (p. ex. avec F1). Sélectionner ensuite le caractère de commande dans la fenêtre des caractères de commande.

Cette fenêtre indique selon le type d'afficheur d'abord uniquement une partie des caractères de commande autorisés pour cet afficheur. Il est possible d'atteindre les autres caractères de commande en faisant défiler la fenêtre.

Le caractère de commande sélectionné apparaît ensuite dans la fenêtre de séquence de commande. Pour les caractères de commande qui ont besoin d'une indication numérique, un champ de saisie s'ouvre dans la fenêtre de séquence (voir fig. suivante).

**FST 203 Display Editor V 3.2**

File name: ANZ\_1

**Control sequence**

ENQ

Type: ABG Pos: 06/01

|     |                 |
|-----|-----------------|
| 001 | Cyl. A extended |
| 002 | B Gripper open  |
| 003 | Reject No.      |
| 004 |                 |
| 005 |                 |

**Control characters**

Reset ABG : SOH  
 Clear & Home : STX  
 Clear & Revis.: ETX  
 Cursorpos.1/1 : FOT  
 Cursorpos.2/1 : ENQ  
 Line clear : ACK  
 Cursor -1 : BS  
 Cursor down : LF  
 Carriage ret. : CR

**Valid character set**

! " # \$ % & ' ( ) \* + , - . / 0 1 2 3 4 5 6 7 8 9 :  
 ; < = > ? @ A B C D E F G H I J K L M N O P Q R S T U  
 V W X Y Z [ \ ] ^ \_ ` a b c d e f g h i j k l m n o p  
 q r s t u v w x y z { | } ~

**Keyboard Legend:**

|         |        |        |          |          |          |            |       |
|---------|--------|--------|----------|----------|----------|------------|-------|
| F Enter | F Mark | F Copy | F Delete | F Insert | F Search | F Tabulat. | F End |
| 1 text  | 2 text | 3 text | 4 text   | 5 text   | 6 text   | 7 charact. | 8     |

Fig. 2/12 : Champ de saisie dans la fenêtre de séquence

La plage de valeurs autorisée pour la saisie figure dans la fenêtre des caractères de commande. Confirmer la saisie à l'aide de la touche Entrée.

Une fois le caractère de commande saisi, le caractère de texte qui lui est attribué, donc le caractère de texte à la position actuelle du curseur est marqué en couleur. En fonction du type d'écran utilisé, la procédure est différente, comme le montre le tableau suivant.

| Type d'écran                                                                                     | Monochrome             | Couleur              |
|--------------------------------------------------------------------------------------------------|------------------------|----------------------|
| Représentation des caractères de texte, pour lesquels des caractères de commande ont été saisis. | noir sur gris          | blanc sur bleu clair |
| Représentation des caractères de texte, pour lesquels aucun caractère de commande n'existe.      | blanc intense sur gris | noir sur gris        |

Lorsqu'un caractère de texte marqué en couleur est effacé, les caractères de commande qui lui sont attribués sont également effacés.

Il est possible de saisir plusieurs caractères de commande pour chaque caractère de texte ou chaque position du curseur dans la fenêtre de texte. La quantité est limitée par la taille de la fenêtre de séquence et dépend donc de la longueur des caractères de commande. Les caractères de commande sont séparés par un slash (/).

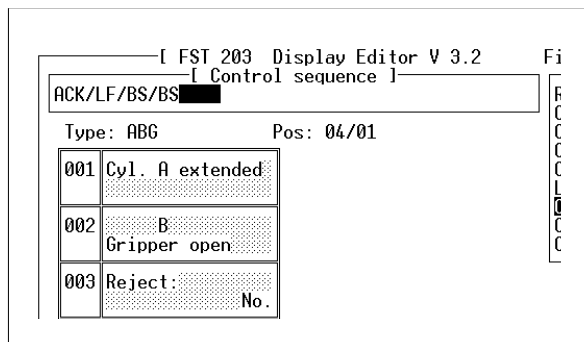


Fig. 2/13 : Séquence de commande

La position du curseur est mise en surbrillance. Si la fenêtre de séquence est saturée en caractères de commande, il est impossible de saisir d'autres caractères de commande pour le caractère de texte actuel.

### Modification de la séquence de commande

Placer le curseur en mode texte sur le caractère de texte où l'on désire modifier la séquence de commande. La séquence de commande figure alors dans la fenêtre de séquence (voir fig. 2/13).

Activer ensuite la fonction Control character (caractère de commande). Il est alors possible de modifier la séquence de commande à l'aide du clavier ou de la souris, comme le montre le tableau suivant.

| Effacer/insérer                                                            | A l'aide du clavier                                                                                                                                                                                                                           | A l'aide de la souris                                                                                                                                                                                                                                                                                             |
|----------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Effacement d'un caractère de commande                                      | Sélectionner dans la fenêtre de séquence le caractère de commande concerné à l'aide des touches de direction. Appuyer alors sur la touche DEL. La touche Retour permet d'effacer des caractères placés avant la position actuelle du curseur. | Placer le curseur de la souris sur le slash (/) à droite du caractère de commande à effacer, et appuyer sur le bouton gauche de la souris.                                                                                                                                                                        |
| Insertion d'un caractère de commande avant la position actuelle du curseur | Sélectionner dans la fenêtre de séquence le caractère de commande avant lequel on désire insérer un caractère. Sélectionner le caractère de commande dans le menu des caractères de commande, et appuyer sur la touche Entrée.                | Placer le curseur de la souris sur le caractère de commande devant lequel on désire insérer un caractère, et appuyer sur le bouton gauche de la souris. Déplacer le curseur de la souris dans le menu des caractères de commande sur le caractère à insérer. Appuyer deux fois sur le bouton gauche de la souris. |

### 2.2.5 Fonctions supplémentaires de l'éditeur



En mode texte sont disponibles des fonctions supplémentaires qui facilitent l'édition d'un fichier. Elles concernent toujours un seul texte d'affichage (le texte d'un numéro de texte). Les fonctions sont les suivantes :

- Insertion d'un texte d'affichage (F5);
- Effacement d'un texte d'affichage (F4);
- Copie d'un texte d'affichage (F3);
- Recherche d'un texte d'affichage (F6).

Utiliser pour cela les touches de fonction de la figure 2/12.

#### Rechercher

Cette fonction facilite la recherche d'un texte d'affichage. Au lieu de rechercher un texte en faisant défiler les pages à l'écran, activer cette fonction. La fenêtre suivante s'ouvre. Entrer le numéro de texte désiré.

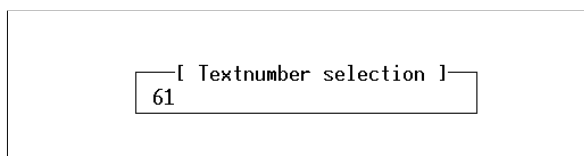


Fig. 2/14 : Rechercher

Confirmer la saisie. Le texte recherché apparaît tout en haut de la fenêtre de texte.



### Copie d'un texte d'affichage

Sélectionner d'abord le texte d'affichage à copier. Placer le curseur sur le texte à copier et activer la fonction Mark text (sélectionner texte). Il est possible également de sélectionner le texte d'affichage en cliquant avec la souris sur le numéro de texte. Le numéro de texte est ensuite mis en surbrillance.

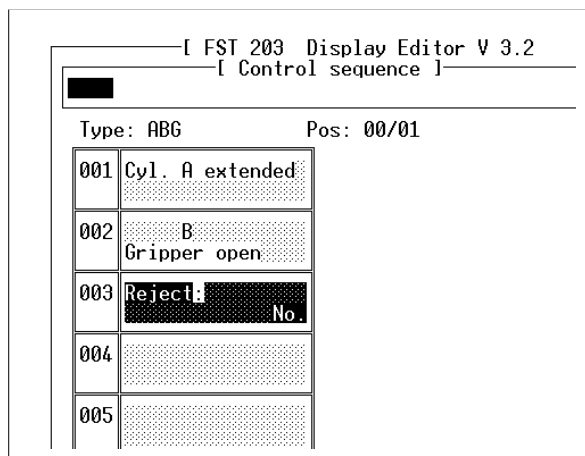


Fig. 2/15 : Sélection de texte

Déplacer le curseur à l'endroit où le texte doit être copié. Activer ensuite la fonction Copy text (copier texte).

Si du texte ou des caractères de commande existent déjà pour ce numéro de texte, la question suivante apparaît sur l'écran :

Text number N already exists (le numéro de texte N existe déjà).

Overwrite? (l'écraser) (Y/N)

Yes La copie est effectuée.

Le texte sélectionné est copié dans le numéro de texte indiqué par N.

No La copie est annulée.

### **Insérer et effacer le texte d'affichage (numéro de texte)**



Placer le curseur sur le texte à effacer ou le texte devant lequel on désire insérer un texte d'affichage (numéro de texte). Activer alors la fonction Delete text (effacer texte) (F4) ou Insert text (insérer texte) (F5).

A partir de la position actuelle du curseur, les numéros de texte sont incrémentés ou décrémentés de 1 selon qu'on efface ou qu'on insère.

## 2.2.6 Quitter l'édition



La fonction End (quitter) permet de quitter l'éditeur. Une fois la fonction activée, la fenêtre des File operations (commandes de fichiers) apparaît.

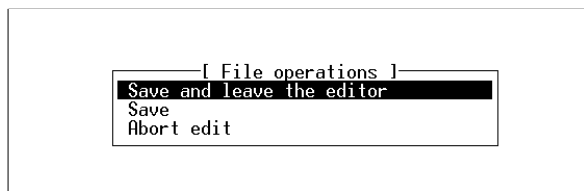


Fig. 2/16 : Commandes de fichiers

### **Save and leave the editor (sauvegarder puis quitter l'éditeur)**

Le fichier d'affichage est enregistré et on quitte l'éditeur. La sélection des fonctions de l'éditeur d'affichage réapparaît sur l'écran.

### **Save and continue (sauvegarde intermédiaire)**

Le fichier d'affichage est seulement enregistré. On peut ensuite continuer à éditer au point où on était. Cette fonction permet d'éviter d'éventuelles pertes de données (p. ex. à la suite de coupures de courant). Dans un tel cas, le dernier état enregistré peut être chargé à nouveau et on peut continuer à travailler.

En cas de modification d'un fichier et d'un nouvel enregistrement, l'éditeur d'affichage attribue à la nouvelle version une extension .ANZ. L'ancienne version est par sécurité sauvegardée avec l'extension .BAK.

### **Abort edit (annuler l'édition)**

Cette fonction permet de retourner dans le menu de sélection des fonctions sans enregistrer le fichier d'affichage. Si cette fonction est activée, la question suivante apparaît dans une autre fenêtre :

Are you sure? (Etes-vous sûr ?) (Y/N)

Yes Les modifications apportées depuis le dernier enregistrement sont rejetées et le programme retourne à la sélection des fonctions.

No Retour à l'éditeur.

### 2.3 Effacement d'un fichier d'affichage

Un fichier d'affichage fait partie d'un projet. S'assurer d'avoir activé dans le logiciel FST le projet dans lequel on désire éditer des fichiers d'affichage.

Pour effacer un fichier d'affichage, activer dans le menu de sélection des fonctions de l'éditeur d'affichage la fonction Delete file (supprimer fichier). Le menu de sélection des fichiers indique alors les fichiers d'affichage du projet actuel de la manière suivante :



Fig. 2/17 : Effacement d'un fichier

Sélectionner ici le fichier à effacer. Après confirmation du choix, la question suivante apparaît dans une fenêtre :

Are you sure? (Etes-vous sûr ?) (Y/N)

Yes Le fichier sélectionné et la copie de sauvegarde enregistrée avec l'extension \*.BAK sont effacés.

No Le programme retourne dans le menu de sélection des fonctions.

## 2.4 Impression d'un fichier d'affichage

La fonction Print display text file (imprimer fichier d'affichage) permet de sortir sur une imprimante un fichier d'affichage du projet activé.

Les fichiers d'affichage ne peuvent pas être imprimés à l'aide de la fonction Print project (imprimer projet).

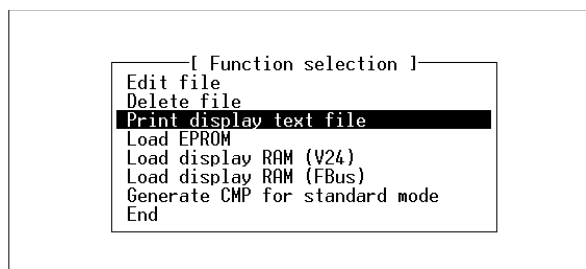


Fig. 2/18 : Impression d'un fichier d'affichage

Les réglages actifs dans le logiciel FST concernant la configuration de l'imprimante sont utilisés. L'en-tête du projet créé avec le logiciel FST apparaît sur chaque page imprimée.

S'assurer d'abord que :

- l'imprimante est allumée et est correctement raccordée à l'ordinateur,
- les réglages de l'imprimante effectués dans le logiciel FST correspondent à la configuration réelle (voir aussi le manuel du FST).

Pour imprimer, activer dans le menu de sélection des fonctions l'option Print display text file (imprimer fichier d'affichage). Le menu de sélection des fichiers apparaît à nouveau. Sélectionner alors le fichier à imprimer. L'opération d'impression commence alors. A la fin de l'impression, le menu de sélection des fonctions réapparaît.

### Format d'impression

Les numéros de texte d'un fichier d'affichage sont imprimés par ordre croissant. Le texte apparaît sous la forme d'une reproduction de l'afficheur de textes. Il indique les séquences de commande qui en font partie en indiquant les coordonnées des lignes / colonnes (voir fig. suivante).

| FESTO Software Tool FST                                  |                                                                                                                                                                                                                                                                         |
|----------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <This is the project page header of the current project> |                                                                                                                                                                                                                                                                         |
| Text no. 001                                             | <div style="text-align: center;"> <sup>1</sup><br/>           0123456789012345<br/> <sup>1</sup><br/>           2 Cyl. A extended         </div>                                                                                                                        |
| Text no. 002                                             | <div style="text-align: center;"> <sup>1</sup><br/>           0123456789012345<br/> <sup>1</sup><br/>           2 B<br/>           Gripper open         </div>                                                                                                          |
| Text no. 003                                             | <div style="text-align: center;"> <sup>1</sup><br/>           0123456789012345<br/> <sup>1</sup><br/>           2 Reject: No.         </div> <div style="text-align: left; padding-top: 5px;">           C-Block to line/column<br/>           01/06 ENQ         </div> |
| SF3 Project:TEST Aug 7, 1997 Page 1                      |                                                                                                                                                                                                                                                                         |

Fig. 2/19 : Exemple d'impression

## **2.5 Chargement de la RAM d'affichage par l'interface RS232 (V24)**

Pour charger un fichier d'affichage dans la mémoire RAM de l'afficheur, procéder aux préparatifs suivants :

- Connecter le PC à l'afficheur à l'aide d'un câble de liaison approprié. D'autres indications concernant le choix du câble figurent au chapitre 1 et dans l'annexe B.
- Vérifier que les réglages de communication correspondent dans le logiciel FST et dans l'afficheur.
- Certains afficheurs exigent un réglage spécifique des interrupteurs ou des cavaliers pour charger les textes (voir manuels d'utilisation du matériel pour les afficheurs et l'annexe B).
- Certains afficheurs ont besoin d'un module d'extension de mémoire pour enregistrer des textes d'affichage (textes externes) (p. ex. E.ABG-1).

Activer dans le menu de sélection des fonctions la fonction Load display RAM (V24) (charger RAM d'affichage V24). Le menu fichier s'ouvre alors permettant de sélectionner le fichier à charger. La sélection de l'interface a ensuite lieu (voir fig. suivante).



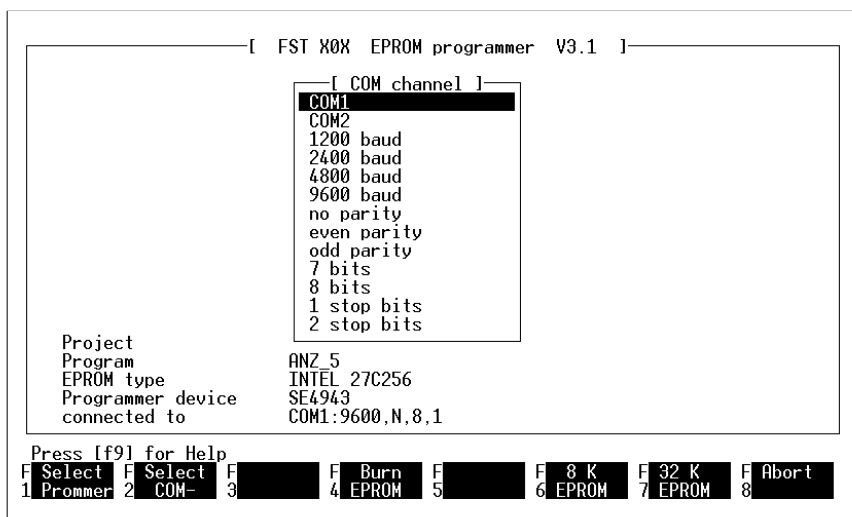


Fig. 2/20 : Sélection de l'interface

Régler en activant les différentes options successivement l'adresse de l'interface utilisée puis sa configuration. Sur l'écran est entrée la configuration de l'interface choisie (configuration de l'interface actuelle). L'opération de chargement s'effectue après activation de la fonction Load text (charger texte).

La configuration de l'interface sélectionnée doit correspondre au réglage effectué dans l'afficheur de textes. Veiller à ce que l'afficheur de textes se trouve en mode programmation (voir manuel d'utilisation de l'afficheur de textes) !

### **Opération de chargement**

Différents messages d'état s'affichent au cours du chargement. Pour les consoles E.ABG1, (EL, LED, VF) et E.ABG-80-EL, des messages supplémentaires s'affichent.

## **2.6 Programmation d'une EPROM**

L'éditeur d'affichage permet la programmation d'une EPROM pour l'enregistrement des textes d'affichage.

### **Généralités**

La programmation d'une EPROM s'effectue en deux étapes. Les fichiers d'affichage sont d'abord convertis en un fichier de format INTEL-HEX. Le contenu de ce fichier est enregistré à l'aide d'un programmeur EPROM dans l'EPROM. Cette opération s'appelle aussi "Programmation".

La programmation exige un logiciel spécifique adapté au programmeur. Certains modèles courants sont implémentés directement par l'éditeur d'affichage.

De plus, presque tous les programmeurs EPROM implémentent les fichiers créés en format INTELHEX. Ces fichiers peuvent être créés à l'aide de l'éditeur d'affichage et chargés dans l'EPROM à l'aide du logiciel du programmeur EPROM.

La programmation d'une EPROM pour l'affichage peut se diviser en deux méthodes :

– **Méthode 1 (pour toutes les consoles ABG avec EPROM)**

Programmation d'une EPROM pour tous les afficheurs qui implémentent cette méthode (y compris EPROM 8k pour afficheurs E.ABG-EL, -LED, -VF).

Ces EPROM sont programmées à l'aide de la fonction Load EPROM (charger EPROM) dans le menu principal de l'éditeur d'affichage (FST202C : possible également via le menu Utilities (utilitaires)).

– **Méthode 2 (pour la console E.ABG-1)**

Programmation des EPROM en mode OEM pour les afficheurs E.ABG-EL, -LED et -VF. Les textes pour l'afficheur et le système d'exploitation de l'afficheur sont combinés dans une EPROM 32k.

Ce fichier combiné est créé à l'aide de la fonction EPROM Utility (utilitaire EPROM) ("ABG-EL/VF OEM EPROM Utility Program", intégré à partir de la version 7/97 – V.8 – dans l'éditeur d'affichage). Lancer le programme dans l'éditeur d'affichage du FST200 (touche F3 EPROM Utility (utilitaire EPROM)). La touche F9 permet d'appeler d'autres textes d'aide.



### Programmation d'une EPROM – Méthode 1

Activer dans l'éditeur d'affichage ou (uniquement avec FST202C) dans Utilities (utilitaires) la fonction Load EPROM (charger EPROM), la liste des fichiers s'affiche sur l'écran. Sélectionner le fichier qui doit être enregistré dans l'EPROM. Une fois sélectionné, le fichier est converti dans un format de données compatible avec l'afficheur.

Le processus de conversion peut être suivi sur une fenêtre dans laquelle s'affichent au fur et à mesure les numéros de texte convertis.

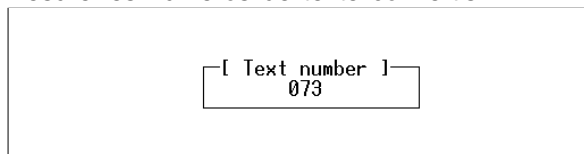


Fig. 2/21 : Suivi de l'opération de conversion

Lorsque le processus de conversion est terminé, l'écran suivant apparaît.

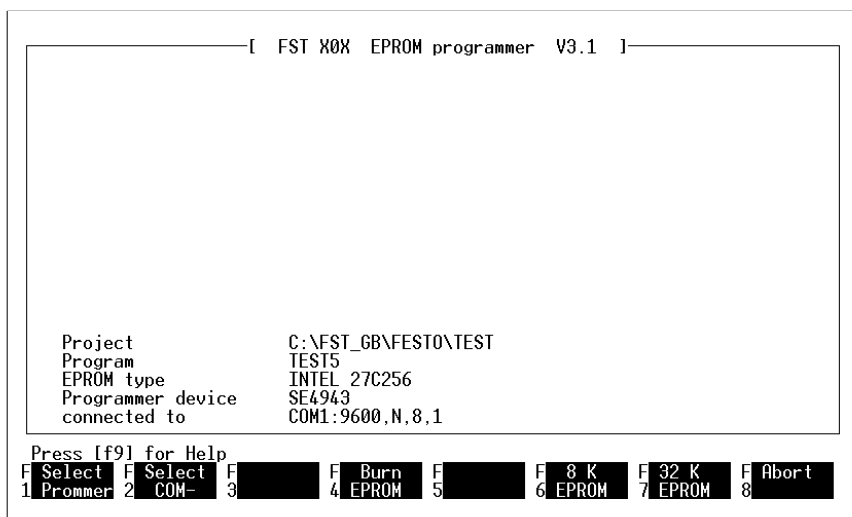


Fig. 2/22 : Ecran de chargement de l'EPROM

Sur la moitié inférieure de l'écran s'affichent le nom du projet activé et la configuration de l'interface actuelle. Les fonctions représentées permettent :



- d'indiquer le type du programmeur EPROM (F1),
- de configurer l'interface de l'ordinateur utilisée (F2),
- de sélectionner le type d'EPROM (F6),
- de lancer la programmation d'une EPROM (F4),
- de retourner au menu de sélection des fonctions (F8).

### Sélection du programmeur EPROM

Une fois la fonction F1 activée, la fenêtre de sélection suivante apparaît :

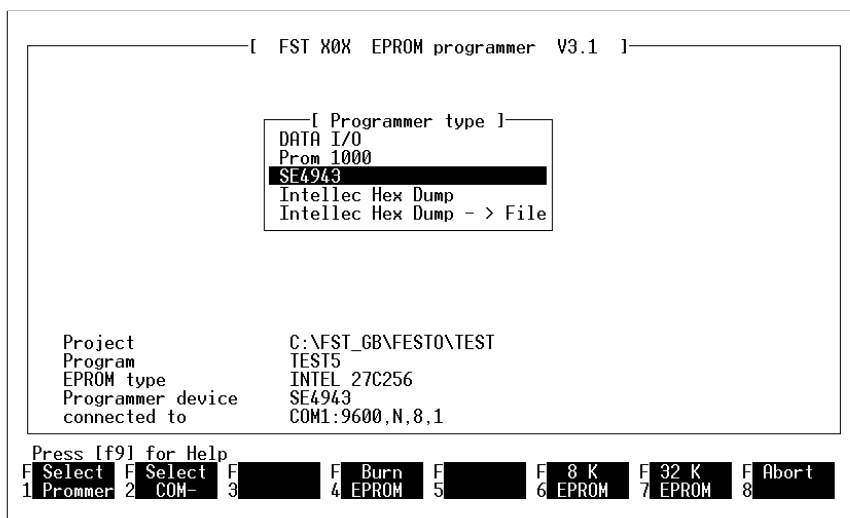


Fig. 2/23 : Sélection du programmeur

Le logiciel FST supporte les programmeurs EPROM DATA I/O, PROM1000 et SE4943. L'utilisation d'un de ces programmeurs rend inutiles un logiciel supplémentaire ainsi que d'autres réglages sur le programmeur EPROM. Le programmeur concerné est sélectionné par activation de l'entrée correspondante.

Veiller cependant à ce que les paramètres de l'interface indiqués correspondent au réglage du programmeur.

### **Autres programmeurs EPROM**

Il est également possible d'utiliser n'importe quel autre programmeur EPROM si on travaille avec le format Intellec-Hex. Utiliser pour cela les deux dernières options dans la fenêtre Programmer type (type de programmeur).

Lire avec soin le manuel d'utilisation du programmeur EPROM concerné.

### **Intellec-Hex-Dump**

Le fichier d'affichage est converti en format Intellec-Hex et transféré directement sur l'interface spécifiée.

### **Intellec-Hex-Dump to File (comme fichier)**

Lorsque cette option est activée, le programme est transféré en format Intellec-Hex et enregistré dans le fichier spécifié. Indiquer pour cela un nom de fichier qui renseigne sur le fichier d'affichage correspondant. Il est ainsi possible d'utiliser le logiciel livré avec le programmeur EPROM pour la programmation d'une EPROM et de pouvoir ainsi rappeler ce fichier. Il est possible de saisir un nom DOS (p. ex. LPT1) pour activer le programmeur EPROM par l'interface parallèle.

### **Sélection de la configuration de l'interface**



Activer la fonction F2 Select COM (sélectionner COM). Sur l'écran apparaît une fenêtre avec une liste permettant de définir l'interface et ses paramètres. Régler en activant successivement les différentes options l'adresse de l'interface utilisée (COM1 ou COM2) et sa configuration. Sur l'écran est immédiatement entrée la configuration sélectionnée.

Si le PC ne dispose que d'une seule interface série, interrompre d'abord la liaison vers l'automate et raccorder le programmeur EPROM à cette même interface.



### Sélection du type d'EPROM



Avant de commencer à programmer l'EPROM, indiquer le type d'EPROM utilisé. Pour cela, activer les fonctions F6 ou F7. Au centre de l'écran apparaît une fenêtre de sélection avec les différents types d'EPROM possibles (voir fig. suivante).

#### REMARQUE :

Le réglage de l'interface du PC doit correspondre à celle du programmeur EPROM.

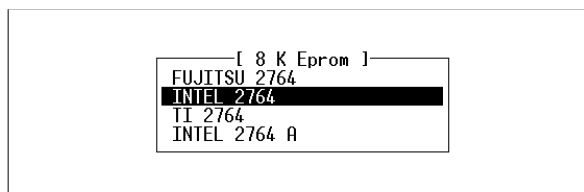


Fig. 2/24 : Sélection du type d'EPROM

Activer l'entrée correspondante pour sélectionner le type d'EPROM désiré. Une nouvelle entrée apparaît immédiatement sur l'écran sous EPROM type (type d'EPROM).

### Programmation d'une EPROM

Après avoir entré les données nécessaires dans les réglages par défaut pour la programmation d'une EPROM, celle-ci peut commencer.

Le PC doit être correctement relié au programmeur EPROM. Un câble de liaison spécifique est nécessaire. Le schéma figurant en annexe B indique l'affectation des broches.



La fonction F4 permet de lancer la programmation ou le gravage. 9



A titre de contrôle, une petite fenêtre compte par ordre croissant les octets de données transférés. Cette opération peut être annulée à tout moment par action sur la touche ESC.



Des erreurs de transmission éventuelles sont dues soit à une liaison défectueuse avec le programmeur EPROM, soit à des entrées incorrectes dans les réglages par défaut de la programmation d'une EPROM. La fonction F9 permet de demander de l'aide.

En cas d'utilisation d'un programmeur EPROM compatible, l'EPROM peut être utilisée immédiatement dans l'afficheur à la fin de la programmation.

### **Programmation d'une EPROM – Méthode 2**

La deuxième méthode ne peut être appliquée que pour les afficheurs E.ABG-EL/-LED/-VF. Ces afficheurs possèdent leur système d'exploitation interne sur une EPROM 32k sur le socle U3 de la carte. Cette EPROM possède encore un espace mémoire suffisant pour les textes. Les textes pour l'afficheur et le système d'exploitation de l'afficheur sont stockés ensemble dans cette EPROM 32 k.

Avantage : En cas d'appareils en série avec textes toujours identiques, cette variante est plus rationnelle et permet d'économiser les frais d'une éventuelle extension de mémoire dans la console ABG.

D'un autre côté, la création et la modification des fichiers texte sont moins souples, cette méthode n'est donc pas recommandée p. ex. lors de la mise en service.



#### **ATTENTION :**

Veiller à enficher une EPROM créée sans erreurs (système d'exploitation) dans le socle de la console ABG.

Le système d'exploitation d'une EPROM pilote le microprocesseur de la console ABG. Des EPROM défectueuses ou inappropriées peuvent détruire durablement l'afficheur !

Procéder comme suit :

1. Créer tous les textes d'affichage pour le projet et les enregistrer.
2. Raccorder la console ABG au PC et s'assurer
  - qu'un câble approprié est utilisé et
  - que les paramètres de l'interface sont identiques des deux côtés.
3. Allumer l'afficheur.
4. Activer dans l'éditeur d'affichage la fonction F3 EPROM Utility (utilitaire EPROM). L'écran suivant apparaît.
5. Si nécessaire, régler sur le PC à l'aide de la touche F3 les paramètres de l'interface correspondants.

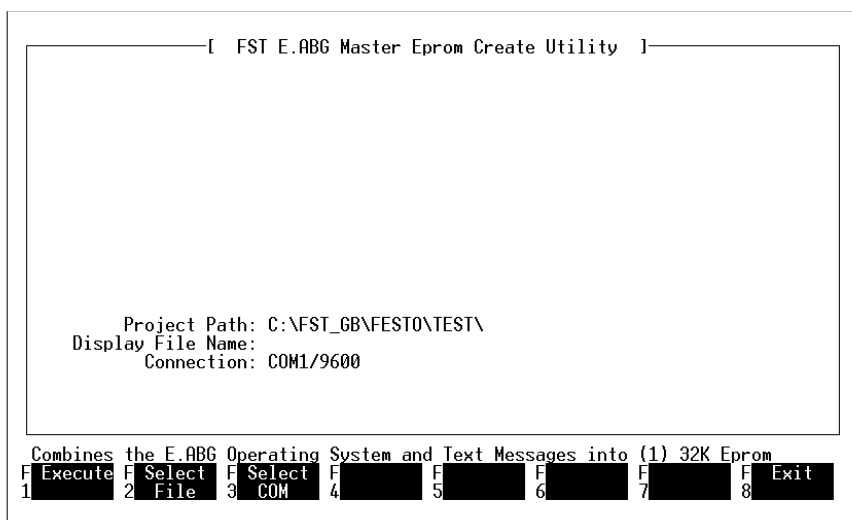


Fig. 2/25 : EPROM pour console ABG



La touche F9 permet d'appeler d'autres textes d'aide.

**F2**

6. Sélectionner à l'aide de la touche F2 le fichier d'affichage à enregistrer dans la console ABG.

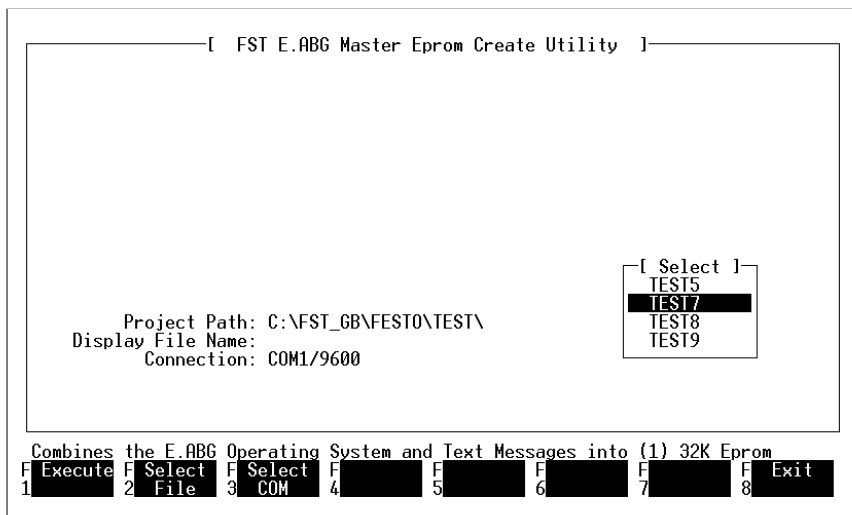


Fig. 2/26 : Sélection du fichier d'affichage

**F1**

Appuyer sur F1 pour créer le fichier pour l'EPROM.

La procédure comprend ensuite les étapes suivantes :

- Le texte d'affichage est compilé (OBJ-File).
- Le système d'exploitation de l'afficheur est chargé et enregistré sur le disque dur.
- Ces deux fichiers sont compilés et enregistrés sur le disque dur.

Pendant cette opération, des messages d'état sont affichés sur l'écran. Un message apparaît dans la ligne d'état pour indiquer que l'opération a réussi.

Le message de fin contient les informations suivantes :

- le chemin d'accès DOS indiquant l'endroit où est stocké le nouveau fichier,
- le nom des nouveaux fichiers Intel-Hex (\*.32k).

L'EPROM pour la console ABG peut alors être programmée avec ce fichier et enfichée dans le socle correspondant de la console ABG. Respecter absolument les indications figurant dans le manuel d'utilisation de la console ABG.

**ATTENTION :**

Veiller à enficher une EPROM créée sans erreurs (système d'exploitation) dans le socle de la console ABG.

Le système d'exploitation d'une EPROM pilote le microprocesseur de la console ABG. Des EPROM défectueuses ou inappropriées peuvent détruire durablement l'afficheur !



## **Partie 3**

### **Création des blocs logiciels à partir des fichiers MAK**



**Sommaire**

|            |                                                                                           |      |
|------------|-------------------------------------------------------------------------------------------|------|
| <b>3.1</b> | <b>Principes généraux pour la création de blocs</b> .....                                 | 3-5  |
|            | Edge detection (analyse du front) ....                                                    | 3-6  |
|            | Sign detection (analyse du signe) ....                                                    | 3-6  |
|            | Output delay (temporisation de l'affichage) .....                                         | 3-7  |
|            | Opérandes (bit mémoire, registre, temporisateur) .....                                    | 3-7  |
| <b>3.2</b> | <b>Création d'un bloc à l'aide de l'éditeur d'affichage</b> .....                         | 3-8  |
|            | Création de blocs CMP/CFM .....                                                           | 3-9  |
|            | Saisie des paramètres de création d'un bloc .....                                         | 3-12 |
| <b>3.3</b> | <b>Création d'un bloc à l'aide de la fonction Include module (relier un module)</b> ..... | 3-13 |
|            | Création de blocs CMP/CFM .....                                                           | 3-15 |
|            | Saisie des paramètres lors de la création d'un bloc .....                                 | 3-19 |

Le mode de création de blocs logiciels qui permettent de relier des consoles d'affichage et de commande est différent selon le type de bloc. On distingue quatre modes de fonctionnement :

- driver d'affichage, textes mémorisés dans le SF3
- driver d'affichage, textes mémorisés dans l'afficheur
- driver de touches ou de clavier (driver de saisie)
- autres drivers de saisie/affichage de caractères.

Pour saisir ou afficher du texte, il est nécessaire de créer deux blocs logiciels pour la console ABG – un driver d'affichage et un driver de clavier. Le tableau suivant indique les fonctions des blocs logiciels à relier et où trouver dans ce chapitre les informations sur leur création.

| Fonction                                         | Fichier MAK                  | Types d'afficheurs                                                                                                                                                                                           | Création                                                                     |
|--------------------------------------------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| Driver d'affichage, Textes dans le SF 3          | 3FD_XABG.MAK<br>3FD^XABG.MAK | <ul style="list-style-type: none"> <li>- E.ABG-1 (E.ABG-LED, E.ABG-EL, E.ABG-VF)</li> <li>- E.ABG-2</li> <li>- E.FD-2/40S</li> <li>- E.ABG-80-LED</li> <li>- ligne de données (autres affichages)</li> </ul> | Création de blocs CMP/CFM à l'aide de l'éditeur d'affichage (paragraphe 3.2) |
| Driver d'affichage, Textes dans l'affichage      | 3FD_216S.MAK<br>3FD^216S.MAK | <ul style="list-style-type: none"> <li>- E.ABG-1 (E.ABG-LED, E.ABG-EL, E.ABG-VF)</li> <li>- E.ABG-80-LED</li> </ul>                                                                                          | Création de blocs CMP/CFM à l'aide de la fonction                            |
|                                                  | 3FD_240S.MAK<br>3FD^240S.MAK | <ul style="list-style-type: none"> <li>- E.FD-2/40S</li> </ul>                                                                                                                                               | "Include module (relier un module)" (paragraphe 3.3)                         |
| Driver de clavier ou de touches                  | 3_EABG.MAK<br>3^EABG.MAK     | <ul style="list-style-type: none"> <li>- E.ABG-1 (E.ABG-LED, E.ABG-EL, E.ABG-VF)</li> </ul>                                                                                                                  |                                                                              |
| Driver de saisie                                 | 3_EABG2.MAK<br>3^EABG2.MAK   | <ul style="list-style-type: none"> <li>- E.ABG-2</li> </ul>                                                                                                                                                  |                                                                              |
|                                                  | 3_ABG80.MAK<br>3^ABG80.MAK   | <ul style="list-style-type: none"> <li>- E.ABG-80-LED</li> </ul>                                                                                                                                             |                                                                              |
| Autres drivers de saisie/affichage de caractères | 3_ETMV5.MAK<br>3^ETMV5.MAK   | <ul style="list-style-type: none"> <li>- Autres affichages/ terminal ASCII</li> </ul>                                                                                                                        |                                                                              |

### 3.1 Principes généraux pour la création de blocs

Pour créer des blocs logiciels, il est nécessaire d'effectuer au préalable quelques saisies de configuration. Il s'agit principalement des opérandes qui sont utilisés par le bloc, mais aussi du paramétrage de la gestion de l'exécution.

Les saisies se font à l'intérieur d'une fenêtre de création de blocs (voir exemple suivant) :

[ FST 203 Display Editor V 3.2 File name:ANZ\_1 ]

[ FST V3.2 Display driver ABG, internal text ]

With edge detection ? (1 = yes / 0 = no) : 0  
Activate sign detection with flag : F 00.01  
Module transmitting text (Busy) : F 00.02  
Character output delay in ms (0,1,2) : 0

F Execute F 2 F 3 F 4 F 5 F 6 F 7 F Abort

Fig. 3/1 : Saisie des paramètres lors de la création de blocs (Exemple)

Créer les blocs logiciels nécessaires

- à l'aide de l'éditeur d'affichage (Chapitre 3.2) ou
- à l'aide de la fonction Include module (relier un module) (Chapitre 3.3)

Les saisies suivantes sont typiques et sont nécessaires lors de la création de la plupart des blocs logiciels :

### **Edge detection (analyse du front)**

Lorsque l'analyse du front est activée, le bloc logiciel empêche l'affichage successif de textes identiques.

L'analyse du front n'agit pas si une variable est affichée en même temps que du texte.

Lors de la création d'un bloc, l'analyse du front est :

- désactivée si la valeur 0 est saisie,
- activée si la valeur 1 est saisie.

### **Sign detection (analyse du signe)**

L'analyse du signe n'a de sens que pour l'affichage de variables. Elle est déterminée par la valeur donnée au bit mémoire au moment de la création du bloc. Si son état logique est :

- 0, le signe n'est pas affiché. La valeur de la variable est comprise entre 0 et 65535.
- 1, la saisie de la variable doit comporter une colonne supplémentaire pour spécifier son signe. La valeur de la variable est alors comprise entre -32768 et +32767.

**Output delay (temporisation de l'affichage)**

Délai attendu avant l'affichage de chaque caractère. Ce paramètre garantit le bon fonctionnement de l'affichage (recommandation pour le E.ABG-2 : 1 ms).

**Opérandes (bit mémoire, registre, temporisateur)**

Les autres saisies, comme p. ex. les bits mémoire et les temporisateurs utilisés par le bloc, dépendent du type de bloc. Vous trouverez des informations à ce sujet dans la description des différents blocs aux chapitres 3.2 et 3.3.

Les opérandes sont saisis dans la liste des affectations.

**Avant de créer un bloc logiciel, vérifier dans la liste des affectations quels opérandes ou domaines d'opérandes sont encore libres.**

**NOTE :**

Aucun message d'erreur n'apparaît au moment de la création du bloc si les saisies dans la liste des affectations sont incorrectes. Par conséquent, et pour chaque bloc créé, procéder systématiquement à un contrôle des points stipulés dans la liste des erreurs (fonction Show error list (afficher liste des erreurs)).

### 3.2 Création d'un bloc à l'aide de l'éditeur d'affichage

Le mode Standard (textes dans le SF 3) est principalement destiné aux consoles d'affichage dépourvues de mémoire de texte, mais il convient aussi à celles disposant de leur propre mémoire de texte. Dans ce mode de fonctionnement, il faut générer un bloc logiciel (CMP/CFM) à partir d'un fichier MAK et d'un fichier de texte. Le bloc logiciel est chargé dans l'automate SF3. Le transfert du bloc sur l'écran d'affichage a lieu lors de l'appel du bloc (voir chapitre 4).

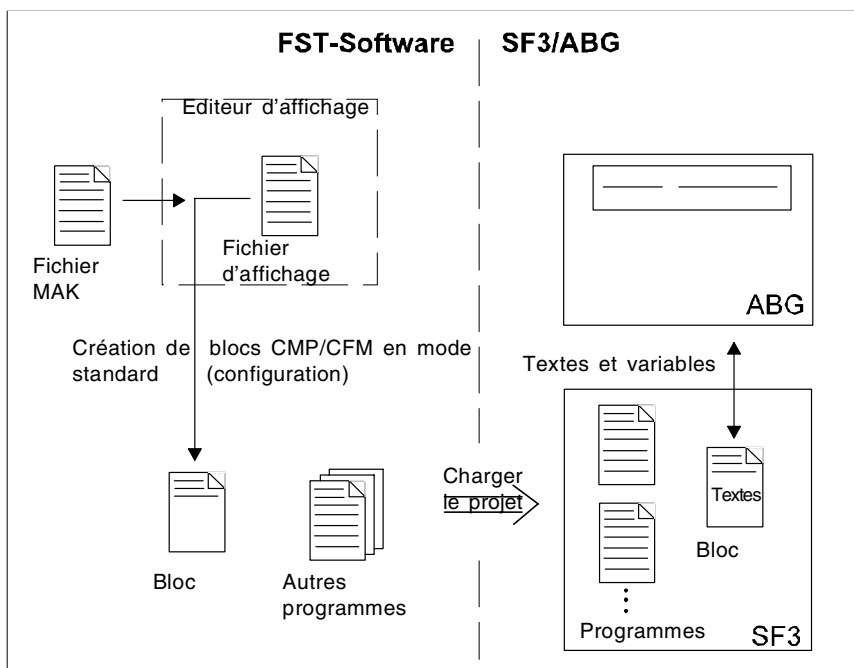


Fig. 3/2 : Création d'un bloc à l'aide de l'éditeur d'affichage

Cette méthode est applicable aux fichiers MAK suivants :

– 3FD\_XABG.MAK / 3FD^XABG.MAK

Pour cela, le fichier MAK doit se trouver dans le répertoire \LIB. Les fichiers MAK présentés dans ce manuel sont chargés automatiquement lors de l'installation du logiciel FST200. Les autres fichiers MAK doivent au préalable être stockés à l'aide de la fonction Import file (importation de fichiers) dans le répertoire (\LIB) de la bibliothèque.

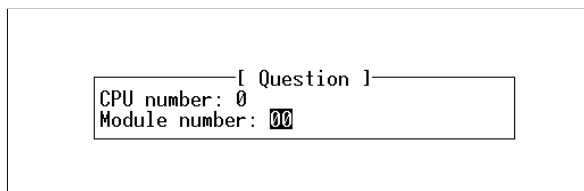
### **Création de blocs CMP/CFM**

1. Activer le projet pour lequel le texte doit être créé (fonction Select project (sélection projet)).
2. Démarrer l'éditeur d'affichage du FST200.
3. Préciser dans l'écran de départ de l'éditeur d'affichage avec la touche de fonction F1, si un bloc de programme ou un bloc fonctionnel va être créé (CMP/CFM). La barre de dialogue indique quelle fonction est activée.
4. Activer alors la fonction Generate CMP/CMF for standard mode (créer des blocs CMP/CFM pour le mode standard). Un menu de sélection des fichiers apparaît. Celui-ci contient tous les fichiers d'affichage du projet activé.





5. Sélectionner le fichier qui doit être inclus dans le bloc. D'autres saisies sont encore nécessaires.



The image shows a screenshot of a software interface. At the top, there is a title bar that says "[ Question 1 ]". Below the title bar, there are two input fields. The first field is labeled "CPU number:" and contains the value "0". The second field is labeled "Module number:" and contains the value "00". The fields are separated by a vertical line.

Fig. 3/3 : Saisie des caractéristiques du bloc

6. Indiquer, dans quelle unité centrale (**l'automate SF3 a toujours le numéro d'UC 0**) et avec quel numéro le bloc logiciel devra ensuite être appelé.

Sélectionner un numéro libre entre 0 et 15 pour les blocs de programmes, et pour les blocs fonctionnels entre 90 et 99.



7. Activer ensuite la fonction Translate (convertir) (F1).

S'il existe déjà un bloc de programme qui possède les mêmes caractéristiques, la question suivante apparaît à l'écran :

Module xx/ZE0 (or function xx/ZE0) already exists in the project directory (le bloc de programme xx/ZE0 (ou la fonction xx/ZE0) existe déjà dans le répertoire du projet) ! Overwrite? (Y/N) (L'écraser ? O/N).

Yes La création du bloc reprend. Le bloc existant sera écrasé.

No La création du bloc est interrompue. Le programme retourne dans le menu de sélection des fonctions.

Une fenêtre affichant les numéros de texte permet de suivre le déroulement de la conversion.

**Saisie des paramètres de création d'un bloc**

Les paramètres à saisir dépendent du type de bloc. Vérifier que les opérandes sont encore libres dans la liste des affectations.

Saisies pour la création d'un bloc logiciel :

- **3FD\_XABG.MAK / 3FD^XABG.MAK**  
"Driver d'affichage ABG, Textes dans le SF 3 (textes internes)"

```
+-----[FST V3.2 display driver ABG, internal texts]-----+
| With edge detection? (1 = yes/0 = no)           :    1      |
| Activate sign detection with flag                : F 00.00   |
| Module transmitting text (busy)                 : F 00.01   |
| Character output delay in ms (0,1,2)            :    0      |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

Paramètres à saisir :

- Edge detection (analyse du front) (voir chapitre 3.1),
- Sign detection (analyse du signe) (voir chapitre 3.1),
- Transmitting text (affichage de texte activé), ce bit mémoire est forcé par le bloc aussi longtemps que du texte ou des variables sont transmis pour être affichés,
- Output delay (temporisation de l'affichage) (voir chapitre 3.1).



Après avoir saisi tous les paramètres, activer la fonction Execute (exécuter) (F1). Ceci termine la procédure de création du bloc et le bloc peut être chargé dans l'automate.

La partie 4 décrit à l'aide de quels paramètres le bloc devra être appelé par le programme.

### 3.3 Création d'un bloc à l'aide de la fonction Include module (relier un module)

La fonction Include module (relier un module) du logiciel FST permet de créer les blocs logiciels suivants à partir des fichiers MAK du répertoire \LIB :

- Driver d'affichage, textes dans l'afficheur (textes externes)
  - 3FD\_216S.MAK / 3FD^216S.MAK
  - 3FD\_240S.MAK / 3FD^240S.MAK
- Driver de clavier / de saisie
  - 3\_EABG.MAK / 3^EABG.MAK
  - 3\_EABG2.MAK / 3^EABG2.MAK
  - 3\_ABG80.MAK / 3^ABG80.MAK
- Driver de saisie/affichage de caractères par l'intermédiaire de l'interface de diagnostic (bloc multitâche)
  - 3\_ETMV5.MAK / 3^ETMV5.MAK
- Autres blocs (calcul 32 bits, driver de console PCS, voir annexe A)

Le fichier MAK correspondant doit se trouver dans le répertoire \LIB. Les fichiers MAK présents dans le manuel sont chargés automatiquement au moment de l'installation du logiciel FST. Les autres fichiers MAK doivent au préalable être stockés à l'aide de la fonction Import file (importation de fichiers) dans le répertoire \LIB de la bibliothèque.

Après l'appel du bloc, l'appel du texte a lieu puis le transfert de variables sur l'écran de la console d'affichage (voir chapitre 4).

Remarque :

Lorsque le texte est stocké dans le SF 3 (textes internes) les drivers d'affichage doivent être créés à partir de l'éditeur d'affichage (voir chapitre 3.2)

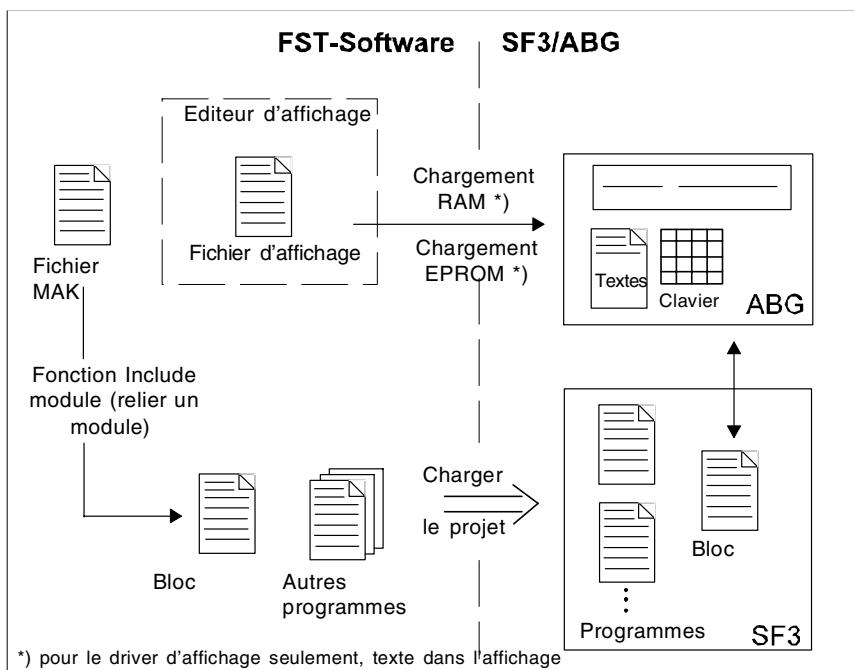


Fig. 3/4 : Création de blocs à l'aide de la fonction Include module (relier un module)

En cas d'utilisation de drivers d'affichage en mode RAM, EPROM et bus de terrain (textes stockés dans l'afficheur), l'affichage doit disposer d'une mémoire et d'une interface appropriées. L'éditeur d'affichage permet de stocker les fichiers d'affichage en mémoire (voir chapitre 2.5 et 2.6). Les textes se trouvent ensuite dans la console.

Le chargement des textes en mémoire sur l'écran de l'affichage s'effectue ensuite en appelant le numéro de texte à l'aide du bloc logiciel approprié.

### **Création de blocs CMP/CFM**

1. Charger le projet/programme pour lequel les blocs doivent être créés.
2. Lancer l'éditeur d'affichage du FST 200.
3. Activer dans le gestionnaire de projets la fonction Include module (relier un module). La liste des fichiers MAK du répertoire \LIB apparaît.

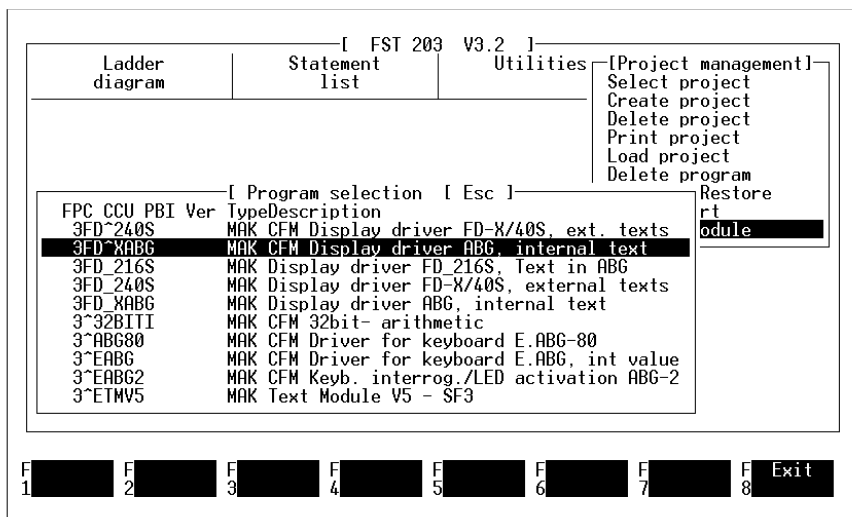


Fig. 3/5 : Include module (relier un module) (sélection)

- Sélectionner le fichier MAK à partir duquel le bloc logiciel doit être créé. Activer l'option désirée en cliquant à deux reprises ou en la mettant en surbrillance et en appuyant sur la touche Entrée.

Ensuite, une fenêtre de saisie apparaît pour l'utilisation du nouveau bloc.

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                   |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| [ FST 203 V3.2 ]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                   |
| Ladder<br>diagram                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Statement<br>list |
| Utilities [Project management]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                   |
| Select project<br>Create project<br>Delete project<br>Print project<br>Load project<br>Delete program<br>Restore<br>rt                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                   |
| [ Program selection [ Esc ]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                   |
| Prog./Module [P/B/F] <span style="border: 1px solid black; display: inline-block; width: 50px; height: 1.2em; vertical-align: middle;"></span><br>Program/Module No. 0<br>Version No. 1<br>Description No comment                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                   |
| <div style="display: flex; justify-content: space-between; padding: 5px;"> <span>F Include</span> <span>F <span style="background-color: black; color: black;">          </span></span> <span>F <span style="background-color: black; color: black;">          </span></span> <span>F <span style="background-color: black; color: black;">          </span></span> <span>F <span style="background-color: black; color: black;">          </span></span> <span>F <span style="background-color: black; color: black;">          </span></span> <span>F <span style="background-color: black; color: black;">          </span></span> <span>F <span style="background-color: black; color: black;">          </span></span> </div> <div style="display: flex; justify-content: space-between; padding: 5px;"> <span>1 <span style="background-color: black; color: black;">          </span></span> <span>2 <span style="background-color: black; color: black;">          </span></span> <span>3 <span style="background-color: black; color: black;">          </span></span> <span>4 <span style="background-color: black; color: black;">          </span></span> <span>5 <span style="background-color: black; color: black;">          </span></span> <span>6 <span style="background-color: black; color: black;">          </span></span> <span>7 <span style="background-color: black; color: black;">          </span></span> <span>8 <span style="background-color: black; color: black;">          </span></span> </div> |                   |

Fig. 3/6 : Données relatives au module

5. Entrer dans Prog./module, si le fichier MAK doit servir à créer un bloc de programme (B) ou un bloc fonctionnel (F).
6. Entrer un numéro libre pour le bloc logiciel. Suivant le type de bloc logiciel, ce numéro est compris entre 0 et 15 (CMP) ou entre 90 et 99 (CFM).
7. Pour différencier les versions, il est possible d'indiquer un numéro de version entre 1 et 9. Entrer sous Description une description du bloc logiciel, généralement un commentaire sur le fichier MAK.





8. Activer Include (relier) (F1), lorsque toutes les saisies sont valides.

S'il existe déjà un bloc de programme qui possède les mêmes caractéristiques, la question suivante apparaît à l'écran :

Module xx/ZE0 (or function xx/ZE0) already exists in the project directory (le bloc de programme xx/ZE0 (ou la fonction xx/ZE0) existe déjà dans le répertoire du projet) ! Overwrite (Y/N) (L'écraser ? O/N)

Yes La création du bloc reprend. Le bloc existant sera écrasé.

No La création du bloc est interrompue. Le programme retourne dans le menu de sélection des fonctions.

Les blocs logiciels utilisent des opérandes pour le transfert de paramètres. Une fenêtre de saisie apparaît.

Les chapitres suivants ou l'annexe A fournissent une description des opérandes utiles pour les blocs logiciels particuliers.

### Saisie des paramètres lors de la création d'un bloc

Les paramètres à saisir dépendent du type de bloc. Vérifier que les opérandes sont encore libres dans la liste des affectations.

Saisies pour la création de blocs logiciels :

- **3FD\_216S.MAK / 3FD^216S.MAK**

"Driver d'affichage FD-X/40S, Texte dans la console ABG (textes externes)"

```
+----[FST V3.2 display driver ABG, text in ABG]----- +
| With edge detection? (1 = yes/0 = no)           :    1 |
| Activate sign detection with flag                : F  00.00 |
| Module transmitting text (busy)                  : F  00.01 |
| Character output delay in ms (0,1,2)             :    0 |
+-----+-----+
```

Paramètres à saisir (voir chapitre 3.1) :

- Edge detection (analyse du front)
- Sign detection (analyse du signe)
- Transmitting text (affichage de texte actif). Ce bit mémoire est forcé par le bloc aussi longtemps que du texte ou des variables sont transmis pour être affichés,
- Output delay (temporisation de l'affichage)

- **3FD\_240S.MAK / 3FD^240S.MAK**

"Driver d'affichage FD-X/40S, Texte dans la console ABG (textes externes)"

```

+--[FST V3.2 display driver FD-X/40S, external texts]--+
| With edge detection? (1 = yes/0 = no)           : 1      |
| Activate sign detection with flag                : F 00.00 |
| Module transmitting text (busy)                  : F 00.01 |
| Character output delay in ms (0,1,2)             : 0      |
+-----+

```

Les paramètres à indiquer sont les mêmes que pour le 3FD\_216S.MAK / 3FD^216S.MAK (voir chapitre 3.1).

- **3\_EABG.MAK / 3^EABG.MAK**

"Driver de clavier E.ABG, Partie entière"

```

+--[FST V3.2 Driver for keyboard E.ABG, integer value]--+
| Auxiliary flag word (flag 0..31)                 : FW 10    |
| Register for keyboard buffer (0..127)            : R 010    |
+-----+

```

Paramètres à indiquer (voir chapitre 3.1) :

- Auxiliary flag word (mot mémoire pour bit mémoire auxiliaire). Les bits mémoire Fx.0 à Fx.5 sont forcés lorsque les touches qui leur sont attribuées sont actionnées. Le bit mémoire Fx.6 permet d'afficher une modification de la mémoire tampon de clavier. En cas d'effacement du bit mémoire Fx.7 le bloc est entièrement réinitialisé.  
(x ... mot mémoire spécifié)
- Register for keyboard buffer (registre de la mémoire tampon de clavier). Le bloc logiciel mémorise le contenu de la mémoire tampon de clavier dans le registre spécifié.

- **3\_EABG2.MAK / 3^EABG2.MAK**

"Scrutation du clavier/commande LED ABG-2"

```
+----[FST V3.2   Key interrog./LED activation ABG-2]----+
| 1. Flag word for key flags      (0..31)      : FW 10  |
| 2. Flag word for key/LED flags(0..31)      : FW 11  |
| 3. Flag word for LED flags      (0..31)      : FW 12  |
+-----+-----+-----+-----+-----+-----+-----+
```

Paramètres à indiquer (voir chapitre 3.1) :

- Flag word for key flags (mot mémoire pour bit mémoire de touche). Les bits mémoire Fx.0 à Fx.15 sont forcés en actionnant la touche qui leur est attribuée et sont réinitialisés en relâchant cette touche.  
(x ... mot mémoire spécifié)
- Flag word for key/LED flags (mot mémoire pour bit mémoire de touche/LED). Les bits mémoire Fy.0 à Fy.3 sont forcés en actionnant la touche qui leur est attribuée et sont réinitialisés en relâchant cette touche. Le bit mémoire Fy.4 est forcé lorsque le bloc logiciel reçoit un avis de réception de la demande d'état (Fz.12) de la part de la console E.ABG-2. Le bloc de programme maintient le bit mémoire Fy.5 forcé, jusqu'à l'envoi d'une instruction de commande des LED. Les bits mémoire Fy.8 à Fy.15 sont destinés à la commande des LED.  
(y ... mot mémoire spécifié)
- Flag word for LED flags (mot mémoire pour bit mémoire LED). Les bits mémoire Fz.0 à Fz.11 sont destinés à la commande des LED. Le forçage du bit mémoire Fz.12 permet d'interroger l'état de l'afficheur. Le forçage du bit mémoire Fz.13 provoque la réinitialisation du bloc.  
(z ... mot mémoire spécifié)

- **3\_ABG80.MAK / 3^ABG80.MAK**

"Driver pour clavier E.ABG-80"

```
+-----[FST V3.2 Driver for keyboard E.ABG-80]----- +
| Auxiliary flag word                (flag 0..31) : FW 10 |
| Register for keyboard buffer      (0..127)      : R   010 |
+-----+-----+-----+-----+-----+-----+-----+
```

Paramètres à indiquer (voir chapitre 3.1) :

- Auxiliary flag word (mot mémoire pour bit mémoire auxiliaire). Les bits mémoire Fx.0 à Fx.5 et Fx.12 à Fx.15 sont forcés en actionnant la touche qui leur est attribuée. Le bit mémoire Fx.6 permet d'afficher une modification de la mémoire tampon de clavier. En cas d'effacement du bit mémoire Fx.7 le bloc est entièrement réinitialisé.  
(x ... mot mémoire spécifié)
- Register for keyboard buffer (registre de la mémoire tampon de clavier). Le bloc logiciel mémorise le contenu de la mémoire tampon de clavier dans le registre spécifié.

### • 3\_ETMV5.MAK / 3^ETMV5.MAK

"Module de texte V5 - SF3"

```
+-----[FST V3.2 Text module V5 - SF3]-----+
| Watchdog Timer for input           : T 15      |
| Flag for Input status              : F 15.00    |
| Flag to abort input                : F 15.01    |
| Data Register for ASCII char. (0..127) : R 015    |
| Output variables with sign detection : F 15.02    |
+-----+
```

Paramètres à saisir (voir chapitre 3.1) :

- Watchdog Timer for input (watchdog temporisateur lors de la saisie). Le temporisateur contrôle le délai de saisie des valeurs numériques pour une variable. En cas de dépassement du délai spécifié dans le programme, le bit mémoire du timeout est forcé et le bloc logiciel est abandonné.
- Bit mémoire pour timeout de saisie (flag for Input status). Ce bit mémoire est forcé si la valeur de la variable n'est pas spécifiée dans le délai prédéfini.
- Flag to abort input (bit mémoire pour annuler la saisie de caractères). Si le bit mémoire est forcé, la saisie de caractères ASCII est interrompue puis le programme en cours reprend.
- Data register for ASCII char. (registre pour caractères ASCII). Ce registre garde en mémoire le ou les deux derniers caractères ASCII indiqués.
- Output variables with sign detection (affichage de variables avec analyse du signe)



Après avoir saisi tous les paramètres, activer la fonction Execute (exécuter) (F1). Ceci termine la procédure de création de blocs. Le bloc peut être chargé dans l'automate.

Le chapitre 4 précise quels paramètres permettent d'appeler un bloc à partir d'un programme.

## **Partie 4**

### **Description des blocs logiciels**

#### **Appel des blocs dans le programme utilisateur**



**Sommaire**

|            |                                                                                                                             |      |
|------------|-----------------------------------------------------------------------------------------------------------------------------|------|
| <b>4.1</b> | <b>Instructions générales concernant l'appel de blocs</b>                                                                   | 4-5  |
| 4.1.1      | Initialisation de blocs ou d'affichages                                                                                     | 4-6  |
| 4.1.2      | Appel cyclique de blocs                                                                                                     | 4-7  |
| 4.1.3      | Analyse des bits mémoire d'état/<br>messages d'erreur                                                                       | 4-8  |
| 4.1.4      | Blocs logiciels et interpréteur de<br>commandes                                                                             | 4-9  |
| <b>4.2</b> | <b>Utilisation du bloc comme driver<br/>d'affichage, Textes dans<br/>le SF 3 (textes internes)</b>                          | 4-10 |
|            | Mode de fonctionnement                                                                                                      | 4-10 |
|            | Transfert de paramètres                                                                                                     | 4-12 |
|            | Exemples d'application                                                                                                      | 4-13 |
|            | Messages d'erreur                                                                                                           | 4-15 |
| <b>4.3</b> | <b>Utilisation du bloc comme driver<br/>d'affichage, textes dans<br/>la console</b>                                         | 4-16 |
|            | Mode de fonctionnement                                                                                                      | 4-16 |
|            | Réglages et opérandes                                                                                                       | 4-18 |
|            | Transfert de paramètres                                                                                                     | 4-19 |
|            | Exemples d'application                                                                                                      | 4-20 |
|            | Messages d'erreur                                                                                                           | 4-22 |
| <b>4.4</b> | <b>Utilisation du bloc comme driver<br/>de saisie pour la scrutation<br/>du clavier ou la mémoire tampon<br/>de clavier</b> | 4-23 |
|            | Mode de fonctionnement                                                                                                      | 4-23 |
|            | Réglages et opérandes pour la<br>console E.ABG-1                                                                            |      |
|            | (3_EABG.MAK / 3^EABG.MAK)                                                                                                   | 4-25 |
|            | Réglages et opérandes pour la<br>console E.ABG-80-LED                                                                       |      |
|            | (3_ABG80.MAK/3^ABG80.MAK)                                                                                                   | 4-26 |
|            | Transfert de paramètres                                                                                                     | 4-28 |
|            | Exemple d'application                                                                                                       | 4-29 |

|            |                                                                                                        |             |
|------------|--------------------------------------------------------------------------------------------------------|-------------|
|            | Messages d'erreur.....                                                                                 | 4-29        |
|            | Remarques .....                                                                                        | 4-29        |
| <b>4.5</b> | <b>Utilisation du bloc comme driver de saisie pour la scrutation du clavier / les témoins LED.....</b> | <b>4-30</b> |
|            | Mode de fonctionnement .....                                                                           | 4-30        |
|            | Réglages et opérandes pour la console E.ABG-2                                                          |             |
|            | (3_EABG2.MAK / 3^EABG2.MAK)...                                                                         | 4-32        |
|            | Transfert de paramètres.....                                                                           | 4-34        |
|            | Exemple d'application .....                                                                            | 4-34        |
|            | Messages d'erreur.....                                                                                 | 4-35        |
|            | Remarques .....                                                                                        | 4-35        |
| <b>4.6</b> | <b>Bloc multitâche .....</b>                                                                           | <b>4-36</b> |
|            | Modes de fonctionnement .....                                                                          | 4-36        |
|            | Opérandes .....                                                                                        | 4-39        |
|            | Transfert de paramètres.....                                                                           | 4-40        |
|            | Affichage de texte (paramètre de saisie P1 = 0) .....                                                  | 4-41        |
|            | Affichage de variables non formatées (paramètre de saisie P1 = 1).....                                 | 4-41        |
|            | Affichage de variables formatées (paramètre de saisie P1 = 1).....                                     | 4-42        |
|            | Affichage de caractères de commande (paramètre de saisie P1 = 2).....                                  | 4-44        |
|            | Saisie de valeurs pour les opérandes de bits multiples sans timeout (paramètre de saisie P1 = 3).....  | 4-45        |
|            | Saisie de caractères ASCII (paramètre de saisie P1 = 5).....                                           | 4-47        |
|            | Affichage de textes internes à la console ABG (paramètre de saisie P1 = 6).....                        | 4-49        |
|            | Remarques .....                                                                                        | 4-50        |

Les chapitres suivants comportent des descriptions plus détaillées des fichiers MAK actuellement disponibles ainsi que des blocs logiciels destinés à l'affichage et à la commande.

Respecter dans tous les cas en premier lieu les instructions générales du chapitre 4.1.

| Fichiers MAK                                                     | Type de bloc                                                                                                              | Chapitre |
|------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|----------|
| 3FD_XABG.MAK<br>3FD^XABG.MAK                                     | Driver d'affichage,<br>Textes dans le SF 3                                                                                | 4.2      |
| 3FD_216S.MAK<br>3FD^216S.MAK<br><br>3FD_240S.MAK<br>3FD^240S.MAK | Driver d'affichage,<br>Textes dans l'affichage                                                                            | 4.3      |
| 3_EABG.MAK<br>3^EABG.MAK<br><br>3_ABG80.MAK<br>3^ABG80.MAK       | Driver de<br>clavier/touches (driver<br>de saisie pour<br>la scrutation du clavier<br>et la mémoire tampon<br>de clavier) | 4.4      |
| 3_EABG2.MAK<br>3^EABG2.MAK                                       | Driver de saisie pour la<br>scrutation du clavier et<br>des témoins LED                                                   | 4.5      |
| 3_ETMV5.MAK<br>3^ETMV5.MAK                                       | Driver de saisie et<br>d'affichage de<br>caractères (bloc<br>multitâche)                                                  | 4.6      |

#### 4.1 Instructions générales concernant l'appel de blocs

Les blocs logiciels faisant fonction de drivers d'affichage et de clavier sont appelés par un programme utilisateur.

Les points suivants sont à étudier lors de la création du programme utilisateur :

- initialisation de l'automate SF 3 (à l'aide du bloc fonctionnel CFM 0, effacement des opérands internes),
- initialisation de l'affichage,
- appel cyclique des blocs,
- inscriptions dans la liste des affectations,
- communication par l'intermédiaire d'opérands spécifiques, de registres et de bits mémoire,
- analyse des messages d'erreur.

Pour toutes les consoles d'affichage et de commande disponibles sur le marché, les blocs logiciels doivent être appelés par un programme. La communication s'effectue à l'aide de registres, de bits mémoire et d'unités fonctionnelles spécifiques (FU).

#### **4.1.1 Initialisation de blocs ou d'affichages**

Lors de la mise en service de l'automate, les opérandes prennent généralement des valeurs arbitraires. C'est pourquoi il est nécessaire de définir toutes les valeurs de sortie des opérandes dès le début d'un programme. C'est le cas également des paramètres ou des opérandes des blocs logiciels utilisés pour l'affichage.

Par conséquent, une première initialisation automatique a lieu lors du premier appel d'un bloc logiciel. Si les opérandes d'un bloc logiciel doivent être réinitialisés ultérieurement au cours de l'exécution du programme, il est nécessaire de programmer pour cela le bit mémoire correspondant du bloc.

Certains affichages peuvent être entièrement réinitialisés à l'aide de bits mémoire (p. ex. E.ABG-2).

### 4.1.2 Appel cyclique de blocs

Certaines fonctions importantes du programme doivent être exécutées régulièrement (de manière cyclique). Le contrôle des touches ayant pu être actionnées en fait partie.

Recommandation :

Appeler cycliquement les blocs logiciels correspondants (driver de saisie), p. ex. à partir du programme principal P0. Dans le cas où le clavier ne possède pas de mémoire tampon, ceci permet de s'assurer que toutes les touches actionnées ont bien été prises en compte.



Si le clavier possède une mémoire tampon, il suffit de scruter la touche Entrée <CR> puis de reprendre la valeur de la mémoire tampon (p. ex. sur la console ABG-80 : jusqu'à cinq chiffres).

### 4.1.3 Analyse des bits mémoire d'état/messages d'erreur

Les blocs logiciels renvoient des messages d'état ou des messages d'erreur par l'intermédiaire de bits mémoire ou d'opérandes spécifiques. Ceux-ci doivent ensuite être analysés par le programme utilisateur.

Il existe trois manières d'interroger des messages d'erreur :

- Dans le mode terminal du mode Online à l'aide de l'interpréteur de commandes (CI) : Les données suivantes sont transmises lors de l'interrogation du numéro d'erreur par 'DE' :

$$>DF=nn$$

1

numéro d'erreur

- En mode Online avec un message en toutes lettres :  
Dans le menu "Display SF 3 information (affichage infos SF 3)" avec la touche de fonction F6 (erreur/diagnostic)

Example :

| Error no. | Error description |
|-----------|-------------------|
|-----------|-------------------|

----- $\alpha$ -----

100

Incorrect number of  
parameters

- Dans le programme utilisateur STL/LDR à l'aide de l'analyse des paramètres de réponse.  
Lorsque le paramètre de réponse FU32 = 0, le paramètre de réponse FU33 indique le numéro d'erreur.

#### **4.1.4 Blocs logiciels et interpréteur de commandes**

Si au moment de l'appel de bloc, l'interpréteur de commandes est activé (système de test), le bloc est aussitôt abandonné et le programme en cours reprend. Par conséquent, aucun affichage de texte n'a lieu ni aucune mise à jour de la mémoire tampon de clavier, des bits mémoire de touches ou de la commande des LED.

Le bloc logiciel exécute simplement un contrôle des paramètres puis enregistre le cas échéant un numéro d'erreur.

Pour avoir accès à l'interface RS-232 (DIAG) (ou à l'affichage de textes), quitter le système de test à l'aide de la commande Exit ('X'). Il est possible d'activer à nouveau le système de test en saisissant 'CTRL T'.



## 4.2 Utilisation du bloc comme driver d'affichage, textes dans le SF 3 (textes internes)

( p. ex. pour E.ABG-1, E.ABG-80-LED, E.ABG-2, E.FD-2/40S)

### Mode de fonctionnement

Le bloc logiciel reçoit par l'intermédiaire des paramètres de transfert les numéros des textes mémorisés dans le bloc et éventuellement aussi une variable. Les paramètres peuvent être transmis sous forme de constantes ou à partir d'opérandes de bits multiples.

Le bloc assure le transfert de l'ensemble des textes à afficher et des variables vers la console ABG. De plus, certains bits mémoire ou opérandes spécifiques sont utilisés pour la gestion de l'exécution (état du transfert, type de représentation, erreurs).

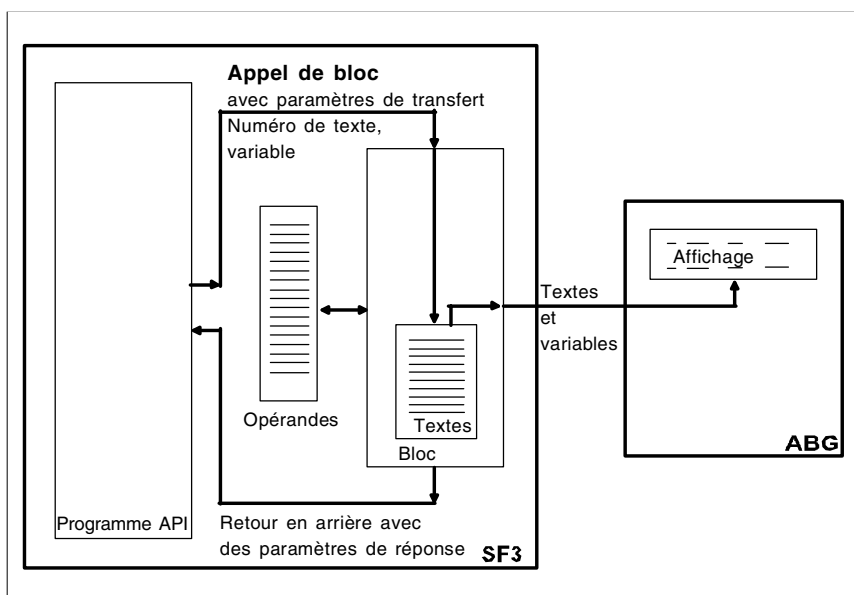


Fig. 4/1 : Principe du driver d'affichage (textes dans le SF 3)

Le traitement du bloc logiciel est séquentiel, c.-à-d. qu'en cas d'affichage le programme en cours reprend seulement après achèvement de la procédure d'envoi. Les autres programmes ne sont pas concernés, leur exécution cyclique se poursuit.

#### Réglages et opérandes

- Analyse du front (voir chapitre 3.1)
- Temporisation de l'affichage (voir chapitre 3.1)
- Inscriptions dans la liste des affectations :

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                  |
|----------------------|--------------------------|-------------------------------------------|
| Fx.y                 | Var_sign                 | Value output sign mode (VZ) <sup>1)</sup> |
| Fx.y                 | Busy                     | Module transmitting text <sup>2)</sup>    |

Rem.: x = 0...31 numéro de mot mémoire

y = 0...15 numéro de bit mémoire

- 1) Lorsque le bit mémoire est forcé, les chiffres sont signés.  
Plage de valeurs : Fx.y = 0 0...65535  
Fx.y = 1 -32768...32767.

- 2) Le bit mémoire reste forcé pendant toute la transmission du texte vers l'affichage.

### Transfert de paramètres

| Paramètre | Valeur                      | Fonction                                                       |
|-----------|-----------------------------|----------------------------------------------------------------|
| 1         | 1...255                     | Numéro de texte                                                |
| 2         | 0...65535<br>-32768...32768 | Données non signées<br>Données signées                         |
| 3         | 1...5                       | Nombre de chiffres<br>affichés<br>Valeur par défaut = 5        |
| 4         | 32...126                    | Caractère d'introduction<br>Valeur par défaut = 32<br>(espace) |

Les paramètres peuvent être transmis sous forme de constantes ou à partir d'opérandes de bits multiples.

Suivant le paramètre d'appel, il est possible d'afficher :

- du texte seulement (adressage de 255 textes au maximum). Si la détection du front est activée, l'affichage de deux textes identiques successifs est impossible ce qui évite un scintillement de l'affichage.
- des textes avec données **non** formatées<sup>\*)</sup>  
Le texte affiché peut comporter jusqu'à 5 caractères (valeur par défaut). Le caractère d'introduction est une espace (valeur par défaut).
- des textes avec données formatées<sup>\*)</sup>  
Le nombre de chiffres affiché peut être limité.

<sup>\*)</sup> Si la représentation des données est signée une place doit être réservée devant les données pour le signe Moins éventuel.  
La détection du front n'intervient pas en cas d'affichage de textes avec données !

### Exemples d'application

Exemple 1 :

Affichage du texte n 10

(affichage d'un texte spécifié par le paramètre 1)

```
THEN CMP 2      'Display driver ABG,  
                'internal texts  
                WITH V10 "Text number 10
```

Si la détection du front est activée, le système vérifie au préalable si le numéro de texte actuel est identique au numéro du précédent texte affiché. Si c'est le cas, le texte n'est pas affiché de nouveau.

Exemple 2 :

Affichage de texte avec données (non formatées)

```
THEN CMP 2      'Display driver ABG,  
                'internal texts  
                WITH V5   "Text number 5 - FESTO  
                WITH V-123 "Value
```

Le texte n 5 sélectionné avec le paramètre 1, puis la valeur du paramètre 2 sont affichés sur cinq chiffres avec une espace comme caractère d'introduction. La valeur peut également être affichée avant le texte ou être insérée dans le texte en utilisant une séquence de positionnement du curseur.

Dans la représentation signée, un chiffre est réservé devant la valeur pour le signe Moins éventuel.

Affichage de l'exemple (bit mémoire de signe forcé) :

FESTO - 123

Exemple 3 :

Affichage de texte avec données (formatées)

```
THEN CMP      2      'Display driver ABG,
                  'internal texts
            WITH V5      "Text number 5 - result:
            WITH R0      "Value
            WITH V3      "Print width 3
                  "characters
            WITH V'0'    "Leading sign
```

Le 3e paramètre permet de définir le nombre de chiffres pour l'affichage de données formatées. Si le nombre de chiffres à afficher est supérieur au nombre spécifié dans le 3e paramètre, l'écran n'affiche que des caractères '\$'. Si la valeur du 3e paramètre est supérieure à 5, la valeur par défaut 5 est retenue sans qu'un message d'erreur soit émis.

Edition (exemple, bit mémoire de signe réinitialisé) :

| Avec R0 = | Résultat |
|-----------|----------|
| 1         | 001      |
| 10        | 010      |
| 100       | 100      |
| 1000      | \$\$\$   |
| 10000     | \$\$\$   |

### Messages d'erreur

Le bloc logiciel génère les erreurs 100, 101, 104.

| Erreur | Description                                                                            |
|--------|----------------------------------------------------------------------------------------|
| 100    | Nombre de paramètres erroné                                                            |
| 101    | Erreur dans le paramètre 1 : le numéro de texte indiqué vaut 0 ou est supérieur à 255. |
| 104    | Erreur dans le paramètre 4 : le caractère de remplissage indiqué n'est pas valable     |

### 4.3 Utilisation du bloc comme driver d'affichage, textes dans la console

(p. ex. pour E.ABG-1, E.ABG-80-LED, E.FD-2/40S)

#### Mode de fonctionnement

Le bloc logiciel reçoit par l'intermédiaire des paramètres de transfert les numéros des textes mémorisés dans la console ABG et éventuellement aussi une variable. Les paramètres peuvent être transmis sous forme de constantes ou à partir d'opérandes de bits multiples.

Le bloc assure le transfert des numéros de textes et des variables vers la console ABG. De plus, certains bits mémoire ou opérandes spécifiques sont utilisés pour la gestion de l'exécution (état du transfert, type de représentation, erreurs).

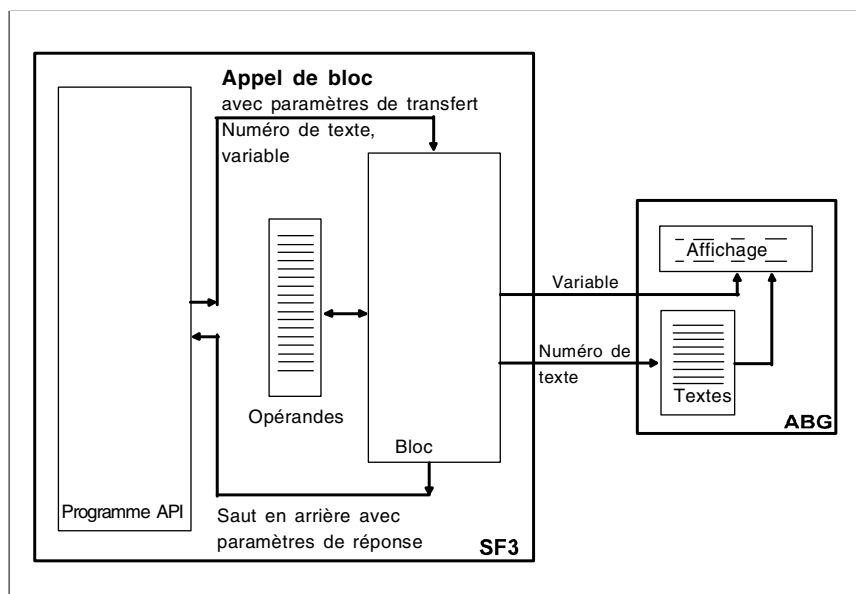


Fig. 4/2 : Principe du driver d'affichage (textes dans la console ABG)

Le traitement du bloc logiciel est séquentiel, c.-à-d. qu'en cas d'affichage le programme en cours reprend seulement après achèvement complet de la procédure d'envoi. Les autres programmes ne sont pas concernés, leur exécution cyclique se poursuit.

Le bloc logiciel affiche le numéro de texte sélectionné par le paramètre 1 ainsi qu'éventuellement un mot de données à travers l'interface de diagnostic.

Remarque concernant l'emploi de certaines consoles d'affichage ou terminaux ASCII :  
L'affichage d'un numéro de texte correspond à une séquence "Escape" ou à une séquence de 2 octets et se décompose de la manière suivante :

| Fichier MAK                   | Affichage de numéros de texte                                                                                                                              |
|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3FD_216S.MAK/<br>3FD^216S.MAK | <GS><Text no.><br><GS> = caractères de commande ASCII 'GS' (29 déc.)<br><Text-Nr.> = Valeur binaire, p. ex. 65 déc.                                        |
| 3FD_240S.MAK/<br>3FD^240S.MAK | <ESC><Z><Text no.><br><ESC> = caractères de commande ASCII 'Escape' (27 déc.)<br><Z> = caractère ASCII 'Z'<br><Text-Nr.> = à 3 chiffres décimal p. ex. 075 |



### Réglages et opérandes

- Analyse du front (voir chapitre 3.1)
- Temporisation de l'affichage (voir chapitre 3.1)
- Inscriptions dans la liste des affectations,

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                  |
|----------------------|--------------------------|-------------------------------------------|
| Fx.y                 | Var_sign                 | Value output sign mode (VZ) <sup>1)</sup> |
| Fx.y                 | Busy                     | Module transmitting text <sup>2)</sup>    |

Rem.: x = 0...31 numéro de mot mémoire  
y = 0...15 numéro de bit mémoire

- <sup>1)</sup> Lorsque le bit mémoire est forcé, les chiffres comportent un signe.

Plage de valeurs : Fx.y = 0      0...65535  
Fx.y = 1      -32768...32767.

- <sup>2)</sup> Le bit mémoire reste forcé pendant toute la transmission du texte vers l'affichage.

### Transfert de paramètres

| Paramètre | Valeur                        |                               | Fonction                                                       |
|-----------|-------------------------------|-------------------------------|----------------------------------------------------------------|
|           | 3FD_216S.MAK/<br>3FD^216S.MAK | 3FD_240S.MAK/<br>3FD^240S.MAK |                                                                |
| 1         | 1...999                       | 1...255                       | Numéro de texte                                                |
| 2         | 0...65535<br>-32768...32768   |                               | Données non signées<br>Données signées                         |
| 3         | 1...5                         |                               | Nombre de chiffres affichés<br>Valeur par défaut = 5           |
| 4         | 32...126                      |                               | Caractère d'introduction<br>Valeur par défaut = 32<br>(espace) |

Les paramètres peuvent être transmis sous forme de constantes ou à partir d'opérandes de bits multiples.

Suivant les paramètres d'appel utilisés, il est possible d'afficher :

- du texte seulement (adressage de 999 ou 255 textes au max.).
- des textes avec données **non** formatées<sup>\*)</sup>  
Le texte affiché peut comporter jusqu'à 5 caractères (valeur par défaut). Le caractère d'introduction est une espace (valeur par défaut).
- des textes avec données **formatées**<sup>\*)</sup>  
Le nombre de chiffres affiché peut être limité.

<sup>\*)</sup> Si la représentation des données est signée une place doit être réservée devant les données pour le signe Moins éventuel.  
La détection du front n'intervient pas en cas d'affichage de textes avec données !

### Exemples d'application

Exemple 1 :

Affichage de textes (affichage du texte spécifié par le paramètre 1)

```
THEN CMP 3          'Display driver ABG,  
                   'external texts  
      WITH V10      "Text number 10
```

Exemple 2 :

Affichage de texte avec données (non formatées)

```
THEN CMP 3          'Display driver ABG,  
                   'external texts  
      WITH V5       "Text number 5 - FESTO  
      WITH V-123    "Value
```

Le numéro de texte sélectionné avec le paramètre 1, puis la valeur du paramètre 2 à cinq chiffres et espaces d'introduction sont édités. La valeur peut également être placée avant le texte ou être insérée dans le texte en utilisant une séquence de positionnement du curseur.

Dans la représentation signée, un chiffre est réservé devant la valeur pour le signe Moins éventuel.

Affichage correspondant à l'exemple (bit mémoire de signe forcé) :

```
FESTO -    123
```

Exemple 3 :  
Affichage de texte avec données (formatées)

```
THEN CMP 3      'Display driver ABG,  
                'external texts  
      WITH V5   "Text number 5 - result:  
      WITH R0   "Value  
      WITH V3   "Pint width 3 characters  
      WITH V'0#' "Leading sign
```

Le 3e paramètre permet de définir le nombre de chiffres pour l’affichage de données formatées. Si le nombre de chiffres à afficher est supérieur au nombre spécifié par le 3e paramètre, l’écran n’affiche que des caractères ‘\$’. Si la valeur du 3e paramètre est supérieure à 5, la valeur par défaut 5 est retenue sans qu’un message d’erreur soit généré.

Edition (exemple, bit mémoire de signe réinitialisé) :

| Avec R0 = | Résultat |
|-----------|----------|
| 1         | 001      |
| 10        | 010      |
| 100       | 100      |
| 1000      | \$\$\$   |
| 10000     | \$\$\$   |

**Messages d'erreur**

Le bloc logiciel génère les erreurs 100, 101, 104.

| Erreur | Description                                                                                                             |
|--------|-------------------------------------------------------------------------------------------------------------------------|
| 100    | Nombre de paramètres erroné                                                                                             |
| 101    | Erreur dans le paramètre 1 :<br>La plage de valeurs valide est<br>1 à 255 (pour le FD_240)<br>1 à 999 (pour le FD_216). |
| 104    | Erreur dans le paramètre 4 : le caractère<br>de remplissage n'est pas valable                                           |

#### 4.4 Utilisation du bloc comme driver de saisie pour la scrutation du clavier ou la mémoire tampon de clavier

( p. ex. pour E.ABG-1, E.ABG-80-LED)

##### Mode de fonctionnement

Lorsque la touche de la console ABG est actionnée, le bloc logiciel force le bit mémoire correspondant. Celui-ci peut être analysé ou réinitialisé par le programme en cours.

Le contenu de la mémoire tampon de clavier (valeur saisie sur la console) est chargé dans le registre correspondant. De plus, certains bits mémoire sont utilisés pour la gestion de l'exécution (modification des valeurs, initialisation).

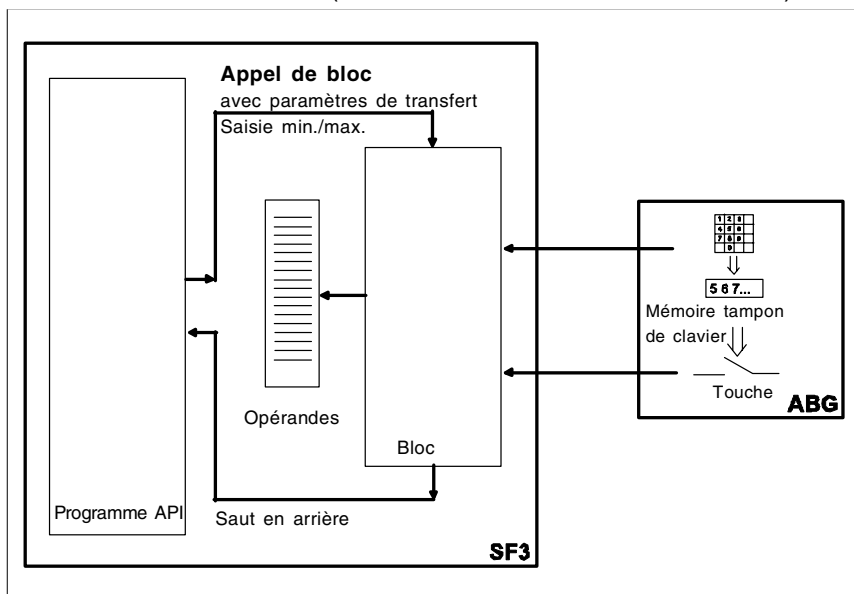


Fig. 4/3 : Principe du driver de saisie (scrutation du clavier et mémoire tampon de clavier)

Les blocs logiciels sont destinés à l'échange cyclique de données avec les consoles d'affichage et de commande ABG-1 ou ABG-80-LED.

Le traitement des blocs logiciels s'effectue en mode Scan, c.-à-d. que la scrutation du clavier et la gestion de la mémoire tampon sont instantanées. Le programme en cours reprend après l'appel du bloc.

Pour mettre à jour en permanence les bits mémoire de touches et la mémoire tampon de clavier, il est indispensable d'appeler les blocs cycliquement.

### Réglages et opérandes pour la console E.ABG-1 (3\_EABG.MAK / 3^EABG.MAK)

– Inscriptions dans la liste des affectations,

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                |
|----------------------|--------------------------|-----------------------------------------|
| Fx.0                 | Enter                    | E.ABG : Enter key pressed <sup>1)</sup> |
| Fx.1                 | F1                       | E.ABG : F1 key pressed <sup>1)</sup>    |
| Fx.2                 | F2                       | E.ABG : F2 key pressed <sup>1)</sup>    |
| Fx.3                 | F3                       | E.ABG : F3 key pressed <sup>1)</sup>    |
| Fx.4                 | F4                       | E.ABG : F4 key pressed <sup>1)</sup>    |
| Fx.5                 | Clear                    | E.ABG : CLEAR key pressed <sup>1)</sup> |
| Fx.6                 | Change                   | E.ABG : Value changed <sup>2)</sup>     |
| Fx.7                 | Initbuff                 | E.ABG : Init if flag = 0 <sup>3)</sup>  |
| Ry                   | Buffer                   | E.ABG : Keyboard buffer <sup>4)</sup>   |

Rem.: x = 0...31      numéro de mot mémoire  
           y = 0...127    numéro de mot du registre

- 1) Les bits mémoire Fx.0 à Fx.5 sont forcés lorsque les touches qui leur sont attribuées sont actionnées. Les bits mémoire doivent être réinitialisés dans le programme utilisateur.
- 2) Le bit mémoire de modification de valeur est forcé, lorsque la valeur de la mémoire tampon de clavier a changé. C'est le cas en particulier lorsque :
  - la touche CLEAR a été actionnée (la mémoire tampon de clavier est effacée)
  - l'une des touches du pavé numérique 0 à 9 a été actionnée.
- 3) Le bit mémoire est forcé après le premier appel du driver et lors de l'abandon du système de test. Une réinitialisation complète du bloc logiciel intervient dès que le bit mémoire est réinitialisé.
- 4) Ce registre permet de mémoriser la valeur saisie et sa position (mémoire tampon de clavier). La mémoire tampon de clavier est effacée, lorsque :
  - la touche CLEAR a été actionnée
  - la valeur saisie sort de la plage de valeurs autorisée.



### Réglages et opérandes pour la console E.ABG-80-LED (3\_ABG80.MAK/3^ABG80.MAK)

– Inscriptions dans la liste des affectations,

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                   |
|----------------------|--------------------------|--------------------------------------------|
| Fx.0                 | Enter                    | E.ABG-80 : Enter key pressed <sup>1)</sup> |
| Fx.1                 | F1                       | E.ABG-80 : F1 key pressed <sup>1)</sup>    |
| Fx.2                 | F2                       | E.ABG-80 : F2 key pressed <sup>1)</sup>    |
| Fx.3                 | F3                       | E.ABG-80 : F3 key pressed <sup>1)</sup>    |
| Fx.4                 | F4                       | E.ABG-80 : F4 key pressed <sup>1)</sup>    |
| Fx.5                 | Clear                    | E.ABG-80 : CLEAR key pressed <sup>1)</sup> |
| Fx.6                 | Change                   | E.ABG-80 : Value changed <sup>2)</sup>     |
| Fx.7                 | Initbuff                 | E.ABG-80 : Init if flag = 0 <sup>3)</sup>  |
| Fx.12                | F5                       | E.ABG-80 : F5 key pressed <sup>1)</sup>    |
| Fx.13                | F6                       | E.ABG-80 : F6 key pressed <sup>1)</sup>    |
| Fx.14                | F7                       | E.ABG-80 : F7 key pressed <sup>1)</sup>    |
| Fx.15                | F8                       | E.ABG-80 : F8 key pressed <sup>1)</sup>    |
| Ry                   | Buffer                   | E.ABG-80 : Keyboard buffer <sup>4)</sup>   |

Rem.: x = 0...31      numéro de mot mémoire  
y = 0...127      numéro de mot du registre

Légendes des tableaux <sup>1)</sup> à <sup>4)</sup> voir page suivante.

- 1) Les bits mémoire Fx.0 à Fx.5 et Fx.12 à Fx.15 sont forcés lorsque la touche qui leur est attribuée est actionnée.  
Les bits mémoire doivent être réinitialisés dans le programme utilisateur.
- 2) Le bit mémoire de modification de valeur est forcé, lorsque la valeur de la mémoire tampon de clavier a changé. C'est le cas lorsque :
  - la touche CLEAR a été actionnée (la mémoire tampon de clavier est effacée)
  - l'une des touches du pavé numérique 0 à 9 a été actionnée.
- 3) Le bit mémoire est forcé après le premier appel du driver et lors de l'abandon du système de test.  
Une réinitialisation complète du bloc logiciel intervient dès lors que le bit est effacé (les bits mémoire et les registres sont réinitialisés).
- 4) Ce registre mémorise la valeur saisie avec sa position (mémoire tampon de clavier).  
La mémoire tampon de clavier est effacée lorsque :
  - la touche CLEAR a été actionnée
  - la valeur indiquée sort de la plage de valeurs autorisée.

### Transfert de paramètres

Lors de l'appel d'un bloc logiciel sans transfert de paramètres, le système vérifie que la suite de chiffres saisie est comprise dans la plage de valeurs 0....65535.

Si **un seul** paramètre est transmis lors de l'appel, celui-ci correspond à la valeur de saisie maximale. Si **deux** paramètres sont transmis, ils définissent un plage de valeurs autorisée.

| Nombre de paramètres | Valeur de saisie minimale              | Valeur de saisie maximale              |
|----------------------|----------------------------------------|----------------------------------------|
| 0                    | 0                                      | 65535                                  |
| 1                    | 0                                      | Valeur du paramètre 1                  |
| 2                    | Plus faible valeur des deux paramètres | Plus grande valeur des deux paramètres |

Pour afficher du texte ou des caractères de commande sur l'affichage, un driver doit être créé à partir des fichiers MAK 3FD\_XABG.MAK ou 3FD^XABG.MAK. L'éditeur d'affichage du FST200 est pour cela indispensable.

**Exemple d'application**

L'objectif est d'entrer une suite de chiffres comprise dans la plage de valeurs autorisée entre 10000 et 10999 et de la valider en appuyant sur la touche ENTREE.

```

STEP
THEN LOAD      V0
              TO      R10      'E.ABG: Keyboard buffer

STEP
THEN CMP 0      'driver for keyboard
              'E.ABG, integer value
              WITH V10000 "min. input value
              WITH V10999 "max. input value
              .
              .
IF            F10.0      'E.ABG: ENTER key
THEN RESET    F10.0      'E.ABG: ENTER key

```

Si plus de deux paramètres sont spécifiés, seuls les deux premiers sont pris en compte.

**Messages d'erreur**

Le bloc logiciel ne délivre aucun message d'erreur.

**Remarques**

Il est important que le bloc logiciel ne soit appelé cycliquement que par un seul programme à la fois.

Si le bloc reçoit des codes ASCII (codes clavier) inconnus, ceux-ci sont ignorés.

#### 4.5 Utilisation du bloc comme driver de saisie pour la scrutation du clavier / les témoins LED

( p. ex. pour la console E.ABG-2)

##### Mode de fonctionnement

Le bloc logiciel force les bits mémoire correspondant aux touches actionnées sur la console ABG et les réinitialise lorsque ces touches sont relâchées. La saisie de valeurs numériques est possible grâce au contrôle par le programme des bits mémoire de touches.

Les LED des touches s'allument ou s'éteignent en fonction de l'état des bits mémoire de LED. De plus, certains bits mémoire sont utilisés pour la gestion de l'exécution (interrogation d'état, avis de réception d'état, initialisation).

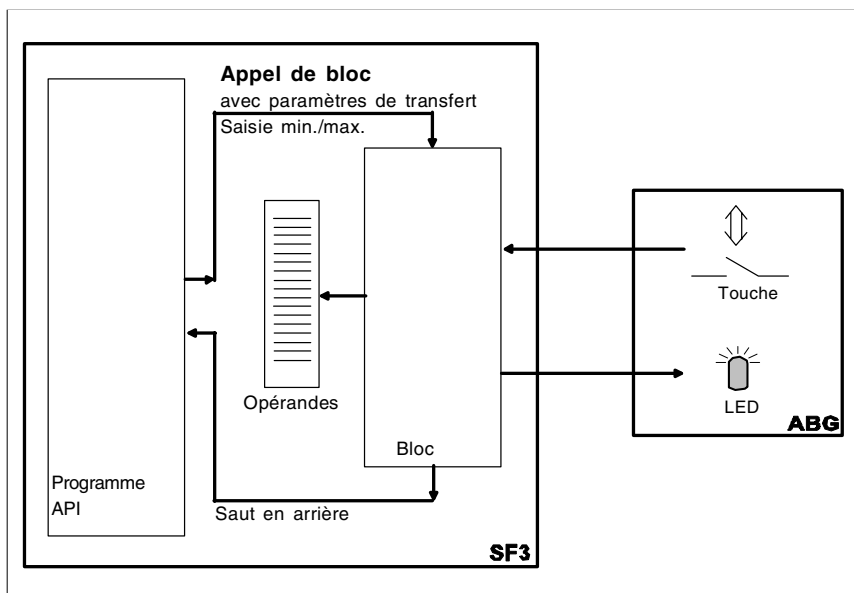


Fig. 4/4 : Principe du driver de saisie (scrutation du clavier et témoins LED)

Le bloc logiciel est destiné à l'échange cyclique de données avec la console d'affichage et de commande ABG-2.

Le traitement du bloc logiciel s'effectue en mode Scan, c.-à-d. que la commande des LED des touches (procédure d'envoi) ou la scrutation des touches (procédure de réception) sont instantanés. Le programme en cours reprend après l'appel du bloc.

Pour mettre en permanence à jour les touches et les LED et assurer la circulation correcte des données, il est nécessaire d'appeler le bloc logiciel cycliquement.

Seules les données réellement utiles sont transmises (détection du front en interne dans le bloc logiciel permettant l'activation/la désactivation des LED).

### Réglages et opérandes pour la console E.ABG-2 (3\_EABG2.MAK / 3^EABG2.MAK)

– Inscriptions dans la liste des affectations,

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                         |
|----------------------|--------------------------|--------------------------------------------------|
| Fx.0                 | F_1_key                  | E.ABG2 : F1 key pressed <sup>1)</sup>            |
| Fx.1                 | F_2_key                  | E.ABG2 : F2 key pressed <sup>1)</sup>            |
| ...                  | ...                      | E.ABG2 : F... key pressed <sup>1)</sup>          |
| Fx.14                | F15_key                  | E.ABG2 : F15 key pressed <sup>1)</sup>           |
| Fx.15                | F16_key                  | E.ABG2 : F16 key pressed <sup>1)</sup>           |
| Fy.0                 | F17_key                  | E.ABG2 : F17 key pressed <sup>1)</sup>           |
| Fy.1                 | F18_key                  | E.ABG2 : F18 key pressed <sup>1)</sup>           |
| Fy.2                 | F19_key                  | E.ABG2 : F19 key pressed <sup>1)</sup>           |
| Fy.3                 | F20_key                  | E.ABG2 : F20 key pressed <sup>1)</sup>           |
| Fy.4                 | Status_OK                | E.ABG2 : Acknowledgement<br>status <sup>3)</sup> |
| Fy.5                 | Send_LED                 | E.ABG2 : Update LED active <sup>4)</sup>         |
| Fy.8                 | LED_F1                   | E.ABG2 : LED F1 key <sup>2)</sup>                |
| Fy.9                 | LED_F2                   | E.ABG2 : LED F2 key <sup>2)</sup>                |
| ...                  | ...                      | E.ABG2 : LED F... key <sup>2)</sup>              |
| Fy.14                | LED_F7                   | E.ABG2 : LED F7 key <sup>2)</sup>                |
| Fy.15                | LED_F8                   | E.ABG2 : LED F8 key <sup>2)</sup>                |
| Fz.0                 | LED_F9                   | E.ABG2 : LED F9 key <sup>2)</sup>                |
| Fz.1                 | LED_F10                  | E.ABG2 : LED F10 key <sup>2)</sup>               |
| ...                  | ...                      | E.ABG2 : LED F... key <sup>2)</sup>              |
| Fz.10                | LED_F19                  | E.ABG2 : LED F19 key <sup>2)</sup>               |
| Fz.11                | LED_F20                  | E.ABG2 : LED F20 key <sup>2)</sup>               |
| Fz.12                | Status                   | E.ABG2 : Request status <sup>5)</sup>            |
| Fz.13                | Init                     | E.ABG2 : Program module init <sup>6)</sup>       |

Rem. : x = 0...31 numéro de mot mémoire de touche  
 y = 0...31 numéro de bit mémoire de touche/LED  
 z = 0...31 numéro de mot mémoire de LED

Légendes des tableaux <sup>1)</sup> à <sup>6)</sup> voir page suivante.

- 1) Les bits mémoire Fx.0 à Fx.15 et Fy.0 à Fy.3 sont forcés lorsque la touche qui leur est attribuée est actionnée et sont réinitialisés lorsque la touche est relâchée.
- 2) Les bits mémoire Fy.8 à Fy.15 et Fz.0 à Fz.11 sont destinés à la commande des LED. Lorsqu'un bit mémoire change d'état la LED correspondante est activée/désactivée.
- 3) Le bit mémoire est forcé lorsque le bloc logiciel reçoit un avis de réception de la demande d'état (Fz.12) en provenance de la console E.ABG-2 (= O.K.). Le bit mémoire doit être réinitialisé dans le programme utilisateur.
- 4) Ce bit mémoire reste forcé jusqu'à l'émission d'une ligne de commande pour la mise à jour des LED.
- 5) Le forçage du bit mémoire permet d'interroger l'état de l'afficheur. Le bloc logiciel réinitialise le bit mémoire lorsque la commande est acquittée.  
La demande d'état ne doit pas être effectuée cycliquement mais doit être commandée par le temporisateur. Il est ainsi possible de contrôler le bon fonctionnement de l'affichage.
- 6) Le bloc est initialisé dès que le bit mémoire est forcé :
  - les bits mémoire de touches sont réinitialisés
  - la séquence d'initialisation est transmise à l'affichage, toutes les LED sont désactivées
  - les bits mémoire internes au bloc permettant l'analyse du front sont réinitialisés
  - les bits mémoire des avis de réception d'état et de mise à jour des LED sont réinitialisés.L'initialisation est automatique :
  - lors du premier appel du bloc
  - lors de l'abandon du système de test (commande CI 'X')



### Transfert de paramètres

Aucun transfert de paramètres n'a lieu lors de l'appel d'un bloc logiciel.

Si l'interpréteur de commandes CI (système de test) est activé au moment de l'appel d'un bloc, le bloc est aussitôt abandonné puis le programme en cours reprend. Aucune commande des LED des touches n'intervient. Les bits mémoire des touches ne peuvent pas être mis à jour.

Pour afficher du texte ou des caractères de commande sur l'affichage, un driver doit être créé à partir des fichiers 3FD\_XABG.MAK ou 3FD^XABG.MAK. L'éditeur d'affichage du FST200 est pour cela indispensable.

### Exemple d'application

Exemple :



```

THEN CMP 1                                'Key interrogation/LED_
                                           'control ABG-2
IF      F_1_key                            'E.ABG2: F1-key
THEN SET LED_F1                          'E.ABG2: LED F1-key
      RESET LED_F2                        'E.ABG2: LED F2-key
IF      F_2_key                            'E.ABG2: F2-key
THEN RESET LED_F1                        'E.ABG2: LED F1-key
      SET LED_F2                          'E.ABG2: LED F2-key
IF      NOP
THEN PW

```

Les LED des touches s'allument/s'éteignent lorsque les touches de fonction F1 ou F2 correspondantes sont actionnées.

**Messages d'erreur**

Le bloc logiciel ne délivre aucun message d'erreur.

**Remarques**

Il est important que le bloc logiciel ne soit appelé cycliquement que par un seul programme à la fois. Si le bloc reçoit des codes ASCII (codes clavier) inconnus, ceux-ci sont ignorés.

#### **4.6 Bloc multitâche**

(destiné à la saisie/affichage de caractères sur tous types d'afficheurs)

##### **Modes de fonctionnement**

Le bloc logiciel permet la communication (saisie et affichage de caractères) avec tous types de consoles d'affichage et de commande à travers l'interface RS-232 (DIAG) de l'automate SF 3. La communication est assurée par différents opérandes (p. ex. bits mémoire, registres, unités fonctionnelles).

Le mode de fonctionnement du bloc multitâche varie selon le paramètre qui appelle. Il est décrit plus en détail dans les paragraphes suivants. Les modes de fonctionnement suivants sont possibles :

| Fonction                                                                                                                                                                                                    | Signification                                                                                                                                                                                                                                                                  |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Affichage de texte                                                                                                                                                                                          | Une chaîne comportant au maximum 54 caractères ASCII est affichée. Le texte est mémorisé dans le programme utilisateur.                                                                                                                                                        |
| Affichage de variables (non formatées/formatées)                                                                                                                                                            | Affichage des valeurs numériques issues du programme utilisateur, signées ou non. 5 chiffres au maximum.                                                                                                                                                                       |
| Affichage de caractères de commande                                                                                                                                                                         | Une suite de 27 caractères de commande max. (format Hex) est affichée. La suite de caractères de commande est mémorisée dans le programme utilisateur.                                                                                                                         |
| Saisie de valeurs pour les opérandes de bits multiples (avec/sans timeout)                                                                                                                                  | Une valeur est enregistrée sous forme hexadécimale ou sous forme de nombre à 5 chiffres max. La valeur est restituée sous forme hexadécimale, décimale (signée ou non) dans l'opérande FU33.                                                                                   |
| Saisie de caractères ASCII (codes clavier ou lignes de caractères)                                                                                                                                          | Une suite de codes clavier pour la console ABG est enregistrée. La saisie est validée à l'aide de la touche Entrée (<CR>). Suivant l'objet de l'appel, deux caractères ASCII peuvent être renvoyés dans un registre ou jusqu'à 4047 caractères dans des unités fonctionnelles. |
| Affichage de textes internes à la console ABG                                                                                                                                                               | Appel des textes mémorisés dans l'afficheur (textes externes).                                                                                                                                                                                                                 |
| L'usage des fonctions ou des possibilités propres à ce bloc logiciel dépend des particularités de la console d'affichage et de commande.<br>Consulter pour plus d'informations le manuel de la console ABG. |                                                                                                                                                                                                                                                                                |

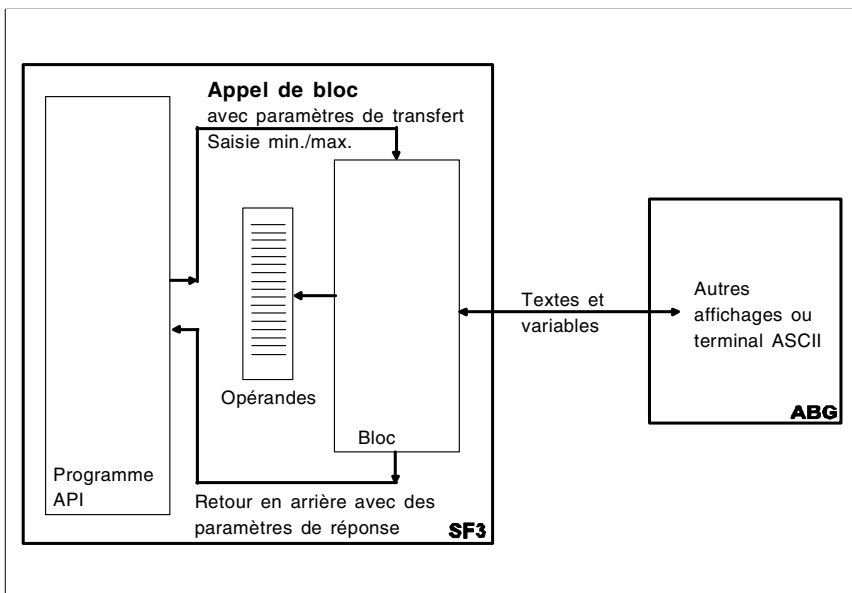


Fig. 4/5 : Principe du bloc multitâche

Le traitement du driver est séquentiel, c.-à-d. qu'en cas d'affichage le programme en cours reprend seulement **après** achèvement complet de la procédure d'envoi. Les autres programmes ne sont pas concernés, leur exécution cyclique se poursuit.

## Opérandes

– Inscriptions dans la liste des affectations,

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                  |
|----------------------|--------------------------|-------------------------------------------|
| Tn                   | Inp_timer                | Timer for input timeout <sup>1)</sup>     |
| TVn                  | Inp_TPRe                 | Timer preselect for input <sup>1)</sup>   |
| TWn                  | Inp_Tword                | Timer word-input Timeout <sup>1)</sup>    |
| Fx.y                 | Time_out                 | Flag for input status <sup>2)</sup>       |
| Fx.y                 | Abort_Inp                | Flag for input abort <sup>3)</sup>        |
| Fx.y                 | Var_sign                 | Value output sign mode <sup>4)</sup>      |
| Rm                   | AsciiVal                 | Return Value of ASCII input <sup>5)</sup> |

Rem. : n = 0...31      numéro de bit de l'état du  
                                  temporisateur/numéro de mot  
                                  présélection du temporisateur,  
                                  mot du temporisateur  
                  x = 0...31      numéro du mot mémoire  
                  y = 0...15      numéro du bit mémoire  
                  m = 0...127    numéro de mot du registre

- 1) Le temporisateur contrôle la saisie de valeurs pour une variable. En cas de dépassement du délai spécifié dans le programme, le bit mémoire du timeout est forcé et le bloc logiciel est abandonné.
- 2) Ce bit mémoire est forcé si la valeur de la variable n'est pas spécifiée dans le délai prédéfini. En cas d'erreur (timeout), le bloc logiciel est abandonné.
- 3) Si le bit mémoire est forcé, la saisie de caractères ASCII est interrompue avant le retour dans le programme en cours.
- 4) Lorsque le bit mémoire est forcé, les chiffres comportent un signe. Plage de valeurs :  
 Fx.y = 0    0...65535  
 Fx.y = 1    -32768...32767
- 5) Ce registre garde en mémoire le ou les deux derniers caractères ASCII saisis, le dernier dans l'octet de poids faible, l'avant dernier dans l'octet de poids fort.

### Transfert de paramètres

La fonction du bloc est déterminée par le 1er paramètre d'appel.

| Catégorie<br>(1er paramètre) | Fonction                                      |
|------------------------------|-----------------------------------------------|
| 0                            | Affichage de textes                           |
| 1                            | Affichage de variables                        |
| 2                            | Affichage de caractères de commande           |
| 3                            | Saisie de valeurs sans timeout                |
| 4                            | Saisie de valeurs avec timeout                |
| 5                            | Saisie de caractères ASCII                    |
| 6                            | Affichage de textes internes à la console ABG |

Selon la catégorie, le système vérifie le nombre de paramètres autorisés. Si ce nombre n'est pas respecté, le message d'erreur 100 est généré.

L'éditeur d'affichage du FST200 n'est pas indispensable pour créer les textes et les caractères de commande à afficher. Avec l'appel du bloc, ces derniers peuvent être définis comme des paramètres en chaîne de caractères et être mémorisés dans le programme qui appelle.

Les paramètres peuvent être transmis sous forme de constantes ou à partir d'opérandes de bits multiples.

**Affichage de texte (paramètre de saisie P1 = 0)**

Paramètre de saisie

P2 = paramètre en chaîne de caractères,  
longueur max. 54 caractères  
(suite de caractères ASCII quelconque  
entre guillemets)

Exemple :

```
THEN CMP 15      'Text module V5 - SF 3
                WITH V0
                WITH      'Text output'
```

Le texte sous forme de paramètre en chaîne de caractères est transféré via l'interface de diagnostic.

**Affichage de variables non formatées  
(paramètre de saisie P1 = 1)**

Paramètre de saisie

P2 = 0...65535 pour les variables non  
signées  
ou = -32768...32767 pour les variables  
signées

Exemple :

```
THEN CMP 15      'Text module V5 - SF 3
                WITH V1
                WITH R0 "Variable value"
```

La variable sous forme de paramètre est transférée non formatée par l'interface de diagnostic, signée ou non en fonction de l'état du bit mémoire d'analyse du signe.



**Affichage de variables formatées  
(paramètre de saisie P1 = 1)**

Paramètre de saisie

|    |                  |                                         |
|----|------------------|-----------------------------------------|
| P2 | = 0...65535      | pour les variables<br>non signées       |
| ou | = -32768...32767 | pour les variables<br>signées           |
| P3 | = 1...5          | nombre de chiffres<br>sur l'affichage   |
| P4 | = 32...126       | caractère d'introduction<br>(en option) |

Exemple :

```
THEN CMP 15      'Text module V5-SF 3
      WITH V1
      WITH R0      "Variable value
      WITH V5      "Number of positions
      WITH V32     "Leading sign (blank)
```

Le texte affiché est toujours formaté lorsque plus de deux paramètres ont été spécifiés.

Les variables peuvent prendre toutes les valeurs des opérandes de bits multiples et des constantes. La plage de valeurs est 0..65535 ou -32768...32767, suivant le type de représentation choisi. Dans la représentation signée, un chiffre est réservé devant la valeur pour le signe Moins éventuel.

Le 3e paramètre permet de définir le nombre de chiffres pour l'affichage de données formatées (entre 1 et 5). Si la valeur numérique comporte un nombre de chiffres supérieur à celui indiqué dans le 3e paramètre, l'écran n'affiche que des caractères '\$' (erreur de format). Si la valeur du 3e paramètre est supérieure à 5, la valeur par défaut 5 est retenue sans qu'un message d'erreur soit généré.

Le 4e paramètre permet de transférer le code ASCII du caractère d'introduction.

Par défaut = une espace (code ASCII 32 déc.). Si les valeurs s'écartent de la plage admissible qui va de 32 déc. à 126 déc. , le message d'erreur 104 (erreur sur le paramètre 4) est généré.

**Affichage de caractères de commande  
(paramètre de saisie P1 = 2)**

Paramètre de saisie

P2= paramètre en chaîne de caractères, de  
54 caractères ASCII max.,  
c.-à-d. 27 caractères hexadécimaux  
(suite de caractères ASCII quelconque  
entre guillemets)

Exemple :

```
THEN CMP 15      'Text module V5 - SF 3  
      WITH V2  
      WITH      * '080A0D'
```

Les caractères de commande transférés sous forme de paramètre en chaîne de caractères doivent être codés au format hexadécimal, c.-à-d. que deux caractères ASCII successifs sont regroupés avant affichage par l'interface de diagnostic et interprétés comme un caractère de commande (valeurs admissibles '0' à '9' et 'A' à 'F'). La suite de caractères résultant de l'exemple serait donc : <BS><LF><CR>.

Si la suite de caractères n'est pas valide, le message d'erreur 102 'Error on parameter 2 (erreur sur le paramètre 2)' est généré et le bloc logiciel est abandonné.

### Saisie de valeurs pour les opérandes de bits multiples sans timeout (paramètre de saisie P1 = 3)

Exemple :

```
THEN CMP 15      'Text module V5-SF 3
      WITH V3     "Value input without
                  'timeout
```

La valeur peut être saisie sous forme hexadécimale (précédée du caractère '\$') ou sous forme d'un nombre à 5 chiffres au maximum avec ou sans signe (précédé du signe '-' pour les valeurs négatives).

Valeurs admissibles :

nombres hexadécimaux :      \$0...\$FFFF

nombres négatifs :            -32768...32767

nombres positifs :            0...65535

Les nombres valides sont réfléchis (écho), les caractères de commande sont écrasés.



En cas de dépassement du nombre de chiffres autorisés ou en cas de saisie de chiffres non valides, le bloc logiciel est abandonné et le paramètre de réponse FU32 <> 0 indique un résultat non valide. Une saisie peut être validée à l'aide de la touche Entrée (<CR>).

Lorsque le résultat est valide, le paramètre de réponse FU32 prend la valeur 0, et la valeur numérique elle-même est indiquée par FU33.

### Saisie de valeurs pour les opérandes de bits multiples avec timeout (paramètre de saisie P1 = 4)

Exemple :

```

STEP
  THEN  LOAD V3000 'Time specification 30s
        TO   TP15  'Input timeout
              'Timer pre.

STEP loop
  IF      NOP
  THEN  CMP 15      'Text module V5-SF 3
        WITH V4     "Value input with
                  "timeout

        LOAD FU32
          TO R0
        LOAD FU33
          TO R1

  IF      F15.0 'Flag
              'timeout-Stat.-E.

  THEN JMP  TO timeout
  OTHRW      NOP

```

La saisie de valeurs est assistée d'un système de contrôle du délai de saisie. Ce système utilise les temporisateurs définis au cours de la configuration (p. ex. le temporisateur 15).



Le temporisateur est lancé automatiquement au moment de l'appel d'un bloc, seule la valeur de présélection doit être spécifiée. Si la saisie n'est pas validée à l'aide de la touche Entrée (<CR>) avant écoulement du temps spécifié dans la présélection du temporisateur, le bit mémoire du timeout est forcé puis le bloc logiciel est abandonné. Le résultat est dans ce cas non valide.

Lorsque le résultat est valide, le paramètre de réponse FU32 prend la valeur 0, et la valeur numérique elle-même est indiquée par FU33.

### Saisie de caractères ASCII (paramètre de saisie P1 = 5)

Chargement de codes clavier de la console ABG

Exemple :

```
THEN    CMP 15      'Text module V5-SF 3
        WITH V5 "load ABG key enc.
IF      R15=K$4B32 "F.-key F2 pressed?
THEN    NOP
OTHRW   PW
```



Il est possible de charger des codes clavier en saisissant une suite de caractères ASCII après appel du bloc logiciel. Les caractères ne sont pas réfléchis (pas d'écho), les saisies sont validées à l'aide de la touche Entrée <CR>. Ce registre, créé lors de la configuration, garde en mémoire le ou les deux derniers caractères ASCII saisis, le dernier dans l'octet de poids faible, l'avant dernier dans l'octet de poids fort.

Le bloc logiciel peut être abandonné prématurément en forçant le bit mémoire Abort (défini lors de la configuration du bloc).

Chargement de chaînes de caractères ASCII

Paramètres de saisie

P2 = 48...4095 numéro de l'opérande spécifique  
P3 = 1...4047 interruption après un certain  
                  nombre de chiffres (en option)

## Exemple

```
THEN CMP 15      'Text module V5-SF 3
      WITH V5     "Read ASCII sequence
      WITH V3000  "Basis address special FU
      WITH V10    "Abort after 10 char.
      CFM 3       "Read special operand
      WITH V3000  "Request number of char.
IF      FU33      "with active test system
      =          V0      "Repeat access
THEN PW
```

Le 2e paramètre de l'appel est interprété comme l'adresse de départ dans le champ des opérandes spécifiques, à partir de laquelle le nombre de caractères reçus (compteur de caractères) et aussi les caractères eux-mêmes sont mémorisés. Un opérande spécifique est attribué à chaque caractère reçu (octet de poids faible = caractère, octet de poids fort = 00h).



Le compteur est initialisé à 0 lors de l'appel du bloc. Les caractères reçus ne sont **pas** réfléchis (pas d'écho). Les saisies sont validées soit à l'aide de la touche <CR> (touche Entrée) soit lorsque le nombre de caractères attendu, spécifié dans le 3e paramètre (en option), est atteint. Si un numéro d'opérande spécifique erroné est saisi, le message d'erreur 102 'Parameter 2 incorrect (paramètre 2 erroné)' est généré.

D'autre part, il est possible de provoquer un abandon prématuré du bloc en forçant le bit mémoire Abort (défini lors de la configuration du bloc).

Le bloc n'est en revanche pas abandonné lorsque la limite de l'opérande (FU4095) a été atteinte avant le nombre de caractères max. attendus.

Le ou les deux derniers caractères ASCII saisis sont mémorisés dans le registre défini lors de la configuration du bloc (le dernier dans l'octet de poids faible, l'avant dernier dans l'octet de poids fort).

#### **Affichage de textes internes à la console ABG (paramètre de saisie P1 = 6)**

Exemple :

```
THEN CMP 15      'Text module V5-SF 3
                WITH V6
                WITH V1  "Text number
```

Le 2e paramètre permet de transférer le numéro du texte mémorisé dans la console ABG et qui doit être affiché. Lors de l'appel, la chaîne de caractères c.-à-d. 01h est transmise par l'interface de diagnostic. Aucun contrôle du numéro de texte n'est effectué.



**Remarques**

Un même bloc logiciel peut être appelé simultanément à partir de plusieurs programmes, à condition que la valeur (fonction) transférée par le paramètre P1 soit **différente**.

Exemple : Le programme P0 affiche du texte (P1 = 0) et le programme P5 attend la saisie de caractères ASCII (P1 = 5).

Si ceci ne peut pas être garanti, effectuer un verrouillage à l'aide de bits mémoire, ou utiliser les programmes pour réaliser l'appel des différents blocs logiciels.

## **Partie 5**

### **Mise en service des consoles ABG**

**Sommaire**

|            |                                                          |      |
|------------|----------------------------------------------------------|------|
| <b>5.1</b> | <b>Mise en service avec émulateurs d'affichage</b> ..... | 5-4  |
| 5.1.1      | Remarques sur les émulateurs .....                       | 5-5  |
| 5.1.2      | Installation des émulateurs .....                        | 5-6  |
| 5.1.3      | Appel et arrêt d'un émulateur .....                      | 5-8  |
| 5.1.4      | Description des émulateurs .....                         | 5-10 |
|            | Emulateur de la console E.ABG-1 ...                      | 5-10 |
|            | Emulateur de la console E.ABG-2 ...                      | 5-12 |
|            | Emulateur de la console E.ABG-80-LED                     | 5-13 |
| <b>5.2</b> | <b>Test des textes d'affichage</b> .....                 | 5-15 |
| <b>5.3</b> | <b>Conseils pratiques</b> .....                          | 5-17 |

Le logiciel FST permet une mise en service aisée du SF 3 avec console d'affichage et de commande grâce aux fonctions suivantes :

- émulateurs d'affichage (pour les textes dans le SF 3 ainsi que les saisies au clavier) et
- test d'affichage (pour les textes dans la console ABG).

Il est ainsi possible de reproduire la fonction des consoles ABG sur le PC ou de vérifier les textes enregistrés dans la console ABG.

Ce chapitre contient également des conseils pratiques qui facilitent la mise en service.

### 5.1 Mise en service avec émulateurs d'affichage

Lors de la mise en service, le PC est relié via l'interface série à l'automate SF 3 pour pouvoir charger et suivre le déroulement des programmes. Pour cela, il peut être nécessaire de contrôler ou de modifier le contenu des opérandes de bit unique ou de bits multiples.

Lorsque l'installation doit être commandée via une console ABG, il est nécessaire de changer fréquemment le branchement (ou de commuter par un boîtier de commutation) entre la console ABG et le PC.

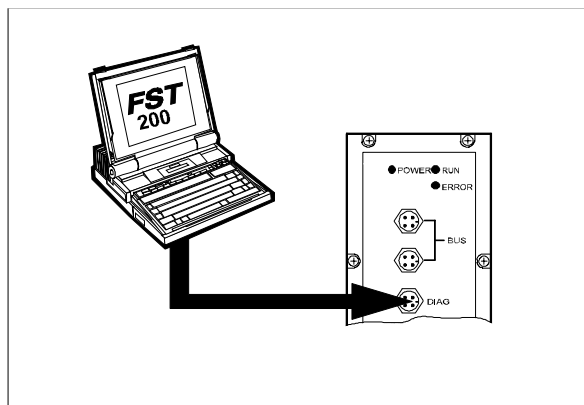


Fig. 5/1 : Raccordement du PC lors de la mise en service du SF3

Cette procédure est possible pour toutes les consoles ABG mais d'utilisation moins aisée. C'est pourquoi des émulateurs ont été développés pour les consoles ABG les plus courantes de Festo.

### 5.1.1 Remarques sur les émulateurs

Le FST200 comprend actuellement trois émulateurs pour les consoles d'affichage et de commande E.ABG-1, E.ABG-2 et E.ABG-80-LED de Festo. Ces émulateurs représentent la console ABG concernée sur l'écran du PC.

Il est ainsi possible de raccorder pendant la mise en service le PC à l'interface RS-232 (DIAG) du SF 3 et :

- de charger et de tester simultanément des programmes utilisateur et
- d'afficher tous les textes du SF 3 (textes internes) sur l'écran via l'émulateur et de scruter les saisies au clavier.

Des informations plus détaillées sur les fonctions des consoles ABG sont jointes dans la documentation produit de chaque console.

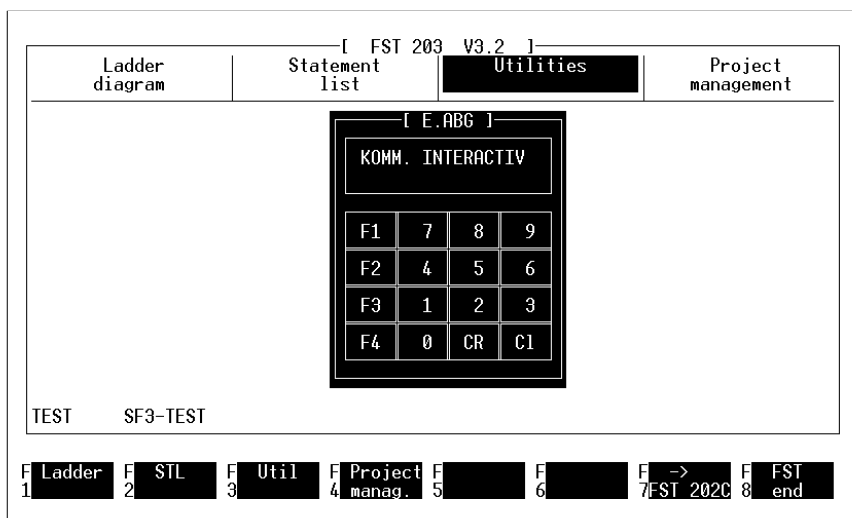


Fig. 5/2 : Mise en service aisée avec l'émulateur

### 5.1.2 Installation des émulateurs

Lors de l'installation du FST200 sur le PC, les émulateurs sont automatiquement installés. Les émulateurs sont des programmes indépendants qui peuvent être lancés à partir du menu "Utilities" (utilitaires) à l'aide de la fonction Program execution (exécution du programme).

1. Sélectionner d'abord le projet.
2. Lancer l'émulateur qui correspond à la console ABG dans le menu "Utilities" (utilitaires) à l'aide de la fonction Program execution (exécution du programme).

Un seul émulateur peut être lancé à la fois.

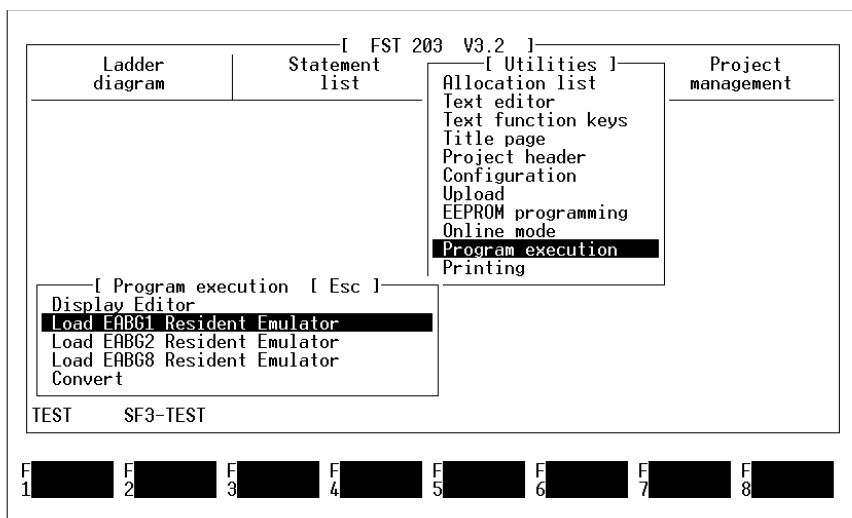


Fig. 5/3 : Appel d'un émulateur (exemple)



L'émulateur est alors chargé dans la mémoire du PC ("résidant en mémoire") et peut être lancé avec la combinaison de touches ALT+ F10. Les programmes résidants restent dans la mémoire du PC jusqu'à ce qu'ils soient explicitement enlevés.



A la fin d'une session du FST, l'émulateur ainsi chargé doit être retiré de la mémoire. Cette opération s'effectue automatiquement à partir de la version 3.2 - révision 8 - 8/97 du logiciel FST200. Pour spécifier la version du logiciel FST, appuyer sur la touche F10 du menu principal. La version s'inscrit alors dans la ligne d'état (p. ex. 3.2-0008 01.08.97).

Pour les versions précédentes ou pour retirer l'émulateur sans quitter le logiciel FST, il est possible d'entrer ou d'exécuter l'instruction **EABG1N-U**

- à l'aide de la fonction Configuration (sous Program termination (fin de programme)), ou
- à l'aide de la fonction Program execution (exécution du programme).



### 5.1.3 Appel et arrêt d'un émulateur

Une fois installé et lancé à l'aide de la fonction Program execution (exécution du programme), un émulateur peut être appelé ou activé à tout moment à partir de l'arrière plan. La combinaison de touches ALT+F10 permet d'appeler l'émulateur résidant en mémoire et la console ABG correspondante apparaît sur l'écran. A cet instant, les autres fonctions du FST sont bloquées.



- Appeler l'émulateur : ALT + F10
- Quitter l'émulateur : ESC

Tous les textes affichés par le SF 3 apparaissent alors sur la console ABG sur l'écran du PC. Les saisies autorisées peuvent alors être effectuées à l'aide du clavier du PC. Celles-ci sont transférées au SF 3 (conformément au programme utilisateur en cours).

A la fin de la saisie, il est possible de quitter l'émulateur à l'aide de la touche ESC. Toutes les fonctions du FST sont à nouveau disponibles.



#### REMARQUE :

Ne jamais quitter l'émulateur avec ESC pendant que le programme affiche des textes, la communication avec le système de test pourrait en être perturbée.

Si cela arrive tout de même :

Quitter le système de test avec F8 et le rappeler immédiatement. La communication est alors rétablie.

#### Remarques et conseils :

Une fois l'émulateur appelé, la console ABG apparaît dans l'angle supérieur droit de l'écran. Les touches de direction permettent de déplacer la console ABG activée sur l'écran. Cette information est conservée pendant la session FST.

Si l'émulateur est activé, il déplace des éléments du logiciel FST sur le disque dur. Pour cela, 300 ko environ sont nécessaires. Lorsque l'émulateur est fermé, les éléments du programme sont à nouveau chargés dans la mémoire vive.

Le système d'exploitation du SF 3 traite les commandes de l'interpréteur de commandes en priorité avant les saisies et les affichages commandés par un programme. Si un émulateur est appelé, toutes les autres fonctions du FST – y compris l'interpréteur de commandes – sont bloquées.

Les émulateurs échangent les données désirées avec le SF 3 via l'interface RS-232. Le PC se comporte comme la console ABG correspondante et communique avec le bloc logiciel actif (driver d'affichage/de clavier) du programme utilisateur.

### 5.1.4 Description des émulateurs

Des émulateurs sont disponibles pour les consoles ABG suivantes :

- E.ABG-1
- E.ABG-2
- E.ABG-80-LED

Ceux-ci sont décrits dans les paragraphes suivants.

#### Emulateur de la console E.ABG-1

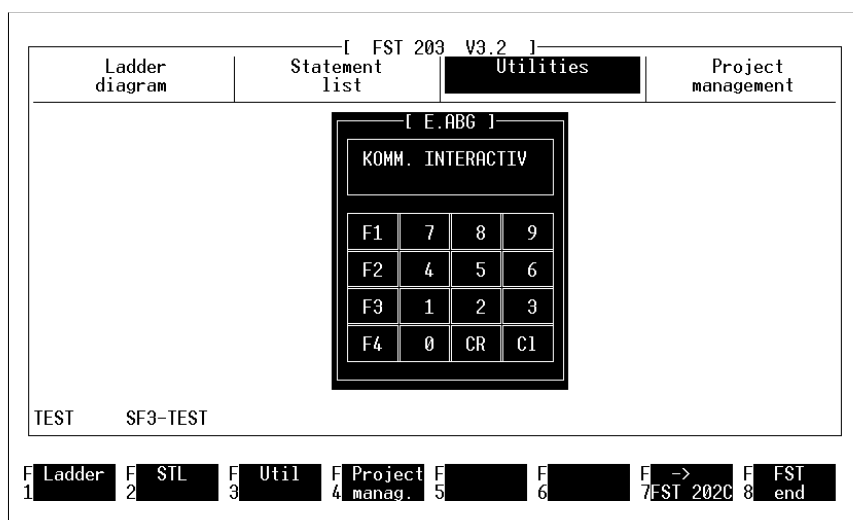


Fig. 5/4 : Emulateur de la console E.ABG-1

Une fois l'émulateur appelé, les touches suivantes du PC sont actives :

**pour les touches numériques :**



**pour les touches de fonction :**



**pour la touche CR :**



**pour la touche Cl :**



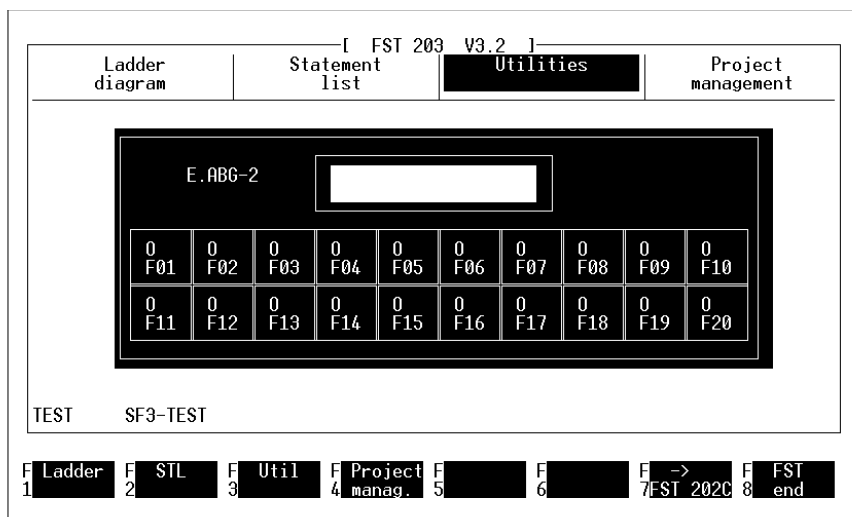
**Emulateur de la console E.ABG-2**

Fig. 5/5 : Emulateur de la console E.ABG-2

Une fois l'émulateur appelé, les touches suivantes du PC sont actives :

**pour la rangée de touches supérieure :**



**pour la rangée de touches inférieure :**



Pour la console ABG-2, les LED des touches sont également émulées. Les cercles sur l'écran pour les touches concernées sont au choix

- sombres avec le bord clair (= LED éteinte),  
ou
- clairs (= LED allumée)

Sur les écrans couleur, la couleur change également.

### Emulateur de la console E.ABG-80-LED

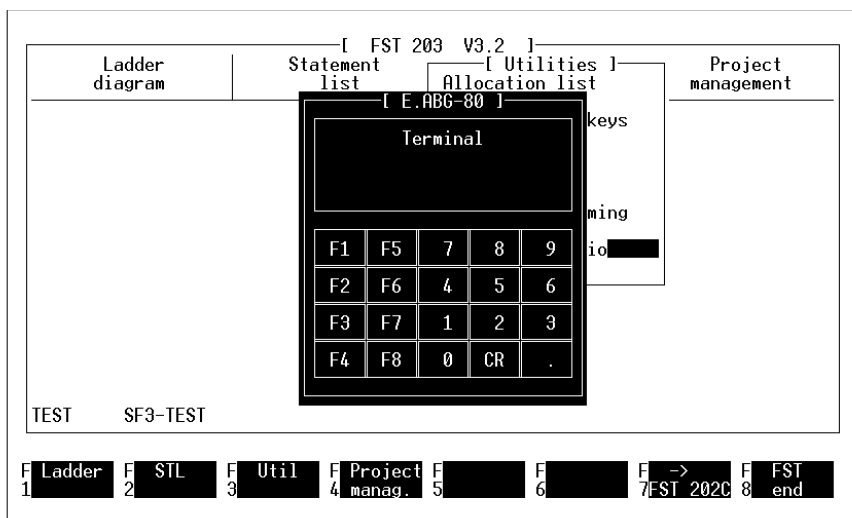


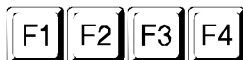
Fig. 5/6 : Emulateur de la console E.ABG-80-LED

Une fois l'émulateur appelé, les touches suivantes du PC sont actives :

**pour les touches numériques :**



**pour les touches de fonction :**



**pour la touche CR :**



**pour la touche "." (<--):**



## 5.2 Test des textes d'affichage



Pour vérifier les textes enregistrés dans la console ABG, les appeler avec la fonction Display test (test d'affichage) (F2) de l'éditeur d'affichage.

L'afficheur est raccordé au PC avec les mêmes réglages que pour le chargement des textes.

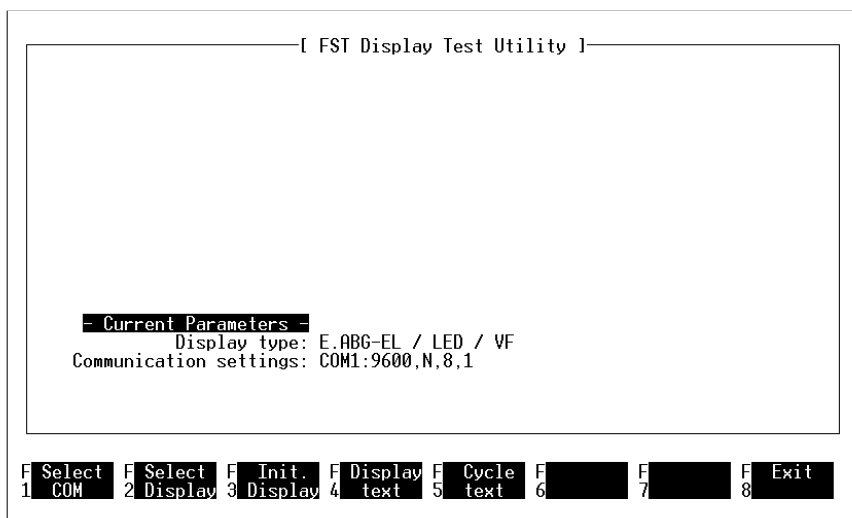


Fig. 5/7 : Test d'affichage



Une fois la fonction Display test (test d'affichage) appelée, les fonctions suivantes sont disponibles :



- **F1 Select COM (sélectionner COM) :**  
La touche F1 permet de sélectionner l'interface série utilisée qui correspond à la configuration de l'afficheur. Appuyer à nouveau sur la touche pour passer au paramètre suivant.



- **F2 Select display (sélectionner afficheur) :**  
La touche F2 permet de sélectionner la console ABG utilisée. Appuyer à nouveau sur la touche pour passer à l'affichage suivant.



- **F3 Initialize display (initialiser afficheur) :**  
Initialise la console ABG raccordée. Le message "Initialization completed (initialisation terminée)" apparaît ensuite.



- **F4 Display text (texte d'affichage) :**  
Appuyer sur F4 pour pouvoir saisir un numéro de texte. Confirmer avec ENTREE pour afficher le texte qui en fait partie sur la console ABG.



- **F5 Cycle text (texte cyclique) :**  
Cette fonction permet d'afficher de façon cyclique des textes avec des numéros partant de 1 et allant jusqu'à un numéro spécifié par l'utilisateur. Entrer le numéro de texte maximal et lancer l'affichage des textes avec ENTREE.



Quitter cette fonction avec ESC.



- **F8 Exit (quitter) :**  
Quitte le programme et retourne à l'éditeur d'affichage.

### 5.3 Conseils pratiques

- Intégrer dès le début les blocs logiciels pour l'affichage et la commande dans le système du programme utilisateur.  
Cette précaution permet d'éviter des conflits entre les numéros de bloc et les entrées dans la liste des affectations.
- Attention, lors de l'appel des blocs logiciels, le temps de cycle s'allonge en raison du temps nécessaire pour la saisie et l'affichage des caractères (env. 1 ms par caractère). En tenir particulièrement compte lors de l'appel du bloc du programme 0 (ou du programme principal).



# **Annexe A**

## **Résumé des blocs logiciels**

**Sommaire**

|                                                                                                                |      |
|----------------------------------------------------------------------------------------------------------------|------|
| Blocs logiciels de l'automate SF3 . . . . .                                                                    | A-3  |
| Description des fichiers MAK (exemple) . . .                                                                   | A-4  |
| Driver pour autres affichages<br>(texte dans le SF 3) . . . . .                                                | A-5  |
| Driver pour l'affichage FD-2/40-S (texte<br>dans la console ABG). . . . .                                      | A-7  |
| Driver pour affichages E.ABG-1 et<br>E.ABG-80-LED (textes dans la console ABG). .                              | A-9  |
| Driver pour clavier ABG-1 /-EL /-VF /-LED<br>(scrutation du clavier et mémoire tampon<br>de clavier) . . . . . | A-11 |
| Driver pour clavier E.ABG-80-LED<br>(scrutation du clavier et mémoire tampon<br>de clavier)l. . . . .          | A-13 |
| Driver pour clavier E.ABG-2<br>(scrutation du clavier et témoins LED) . . .                                    | A-16 |
| Saisie/diffusion de caractères à travers<br>l'interface de diagnostic (bloc multitâche) .                      | A-19 |
| Driver pour console de commande PCS<br>(en liaison avec la logiciel de<br>programmation PCSPRO). . . . .       | A-24 |
| Driver pour arithémétique 32-bits<br>(4 opérations fondamentales). . . . .                                     | A-30 |

**Blocs logiciels de l'automate SF3**

| Fonction                         | Types d'afficheurs compatibles                                                                            | Description sommaire                                         | Fichier MAK | Bloc créé |
|----------------------------------|-----------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|-------------|-----------|
| Driver d'affichage               | E.ABG-1 (-LED, -VF, -EL)<br>E.ABG-80-LED<br>E.FD-2/40S<br>E.ABG-2<br>Ligne de données (autres affichages) | Driver d'affichage E.ABG-1 (-LED, -VF, -EL), textes internes | 3FD_XABG    | CMP       |
|                                  |                                                                                                           |                                                              | 3FD^XABG    | CFM       |
|                                  | E.FD-2/40S                                                                                                | Driver d'affichage FD-X40/S, textes externes                 | 3FD_240S    | CMP       |
|                                  |                                                                                                           |                                                              | 3FD^240S    | CFM       |
|                                  | E.ABG-1 (...)<br>E.ABG-80-LED                                                                             | Driver d'affichage E.ABG-1, E.ABG-80-LED, textes externes    | 3FD_216S    | CMP       |
|                                  |                                                                                                           |                                                              | 3FD^216S    | CFM       |
| Driver de clavier                | E.ABG-1 (-LED, -VF, -EL)                                                                                  | Driver pour clavier E.ABG-1 (-LED, -VF, -EL), Partie entière | 3_EABG      | CMP       |
|                                  |                                                                                                           |                                                              | 3^EABG      | CFM       |
|                                  | E.ABG-80-LED                                                                                              | Driver de clavier E.ABG-80-LED                               | 3_ABG80     | CMP       |
|                                  |                                                                                                           |                                                              | 3^ABG80     | CFM       |
|                                  | E.ABG-2                                                                                                   | Scrutation du clavier/commande des LED, E.ABG-2              | 3_EABG2     | CMP       |
|                                  |                                                                                                           |                                                              | 3^EABG2     | CFM       |
| Driver d'affichage et de clavier | Autres affichages/terminal ASCII                                                                          | Driver de saisie/affichage de caractères                     | 3_ETMV5     | CMP       |
|                                  |                                                                                                           |                                                              | 3^ETMV5     | CFM       |
|                                  | Console PCS                                                                                               | Driver de PCS<br>Console de commande                         | 3_PCS       | CMP       |
|                                  |                                                                                                           |                                                              | 3^PCS       | CFM       |
| Calcul                           | –                                                                                                         | Calcul 32 bits                                               | 3_32BITI    | CMP       |
|                                  |                                                                                                           |                                                              | 3^32BITI    | CFM       |

**Description des fichiers MAK (exemple)**

Les noms des fichiers MAK destinés à la création de drivers (Make-Files) se décomposent de la façon suivante :

| <b>3_EABG.MAK /<br/>3^ETMV5.MAK</b> | <b>Utilisation</b>                                                               |
|-------------------------------------|----------------------------------------------------------------------------------|
| 3                                   | Type d'automate<br>(3 .. SF 3, 2 .. SB202, etc.)                                 |
| _ ou ^                              | _ = fichier MAK pour bloc<br>CMP<br>^ = fichier MAK pour bloc<br>CFM             |
| EABG ou<br>ETMV5                    | Type d'écran compatible ou fonction<br>du driver<br>(ETM = Enhanced Text Module) |
| .MAK                                | Fichier à partir duquel un driver<br>(CMP ou CFM) peut être créé<br>(Make File)  |

**CMP/CFM****3FD\_XABG.MAK/  
3FD^XABG.MAK****Driver pour autres affichages (texte dans le SF 3)**

Le bloc logiciel est configuré dans l'éditeur d'affichage FST à l'aide de la fonction "Generate CMP for standard mode (créer CMP pour mode standard)" ou "Generate CFM for standard mode (créer CFM pour mode standard)" avant d'être relié au projet utilisateur.

Les textes saisis à l'aide de l'éditeur d'affichage FST font partie du bloc ainsi créé.

**Format de saisie (STL)**

```
THEN      CMP {00 .. 15}
(or       CFM {90 .. 99})
          WITH <P1>
          WITH <P2>
          WITH <P3>
          WITH <P4>
```

**Paramètres**

P1 = numéro de texte (1 .. 255)  
P2 = données non signées (0 .. 65535) ou  
données signées (-32768 .. 32768)  
P3 = nombre de chiffres affichés (1 .. 5)  
P4 = caractère d'introduction (32 .. 126)

**Paramètres de réponse**

P1 (FU32) = {-1}      l'opération a réussi  
                 {0}      l'opération a échoué  
P2 (FU33) =            numéro d'erreur (100, 101, 104)



### Opérandes

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| <b>Absolute<br/>(Absolu)</b> | <b>Symbolic<br/>(Symbolique)</b> | <b>Comment<br/>(Commentaire)</b>          |
|------------------------------|----------------------------------|-------------------------------------------|
| Fx.y                         | Var_Sign                         | Value output sign mode (VZ) <sup>1)</sup> |
| Fx.y                         | Busy                             | Module transmitting text <sup>2)</sup>    |

Rem.: x = 0...31 numéro de mot mémoire  
y = 0...15 numéro de bit mémoire

- <sup>1)</sup> Les chiffres comportent un signe lorsque le bit mémoire est forcé.  
Plage de valeurs : Fx.y = 0      0...65535  
Fx.y = 1    -32768...32767.
- <sup>2)</sup> Le bit mémoire reste forcé pendant toute la transmission du texte à l'affichage.

**CMP/CFM****3FD\_240S.MAK/  
3FD^240S.MAK****Driver pour l'affichage FD-2/40-S (texte dans la console ABG)**

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

**Format de saisie (STL)**

```
THEN    CMP  {00 .. 15}  
(or     CFM  {90 .. 99})  
        WITH <P1>  
        WITH <P2>  
        WITH <P3>  
        WITH <P4>
```

**Paramètres**

P1 = numéro de texte (1 .. 999)  
P2 = données non signées (0 .. 65535) ou  
données signées (-32768 .. 32768)  
P3 = nombre de chiffres affichés (1 .. 5)  
P4 = caractère d'introduction (32 .. 126)

**Paramètres de réponse**

P1 (FU32) = {-1}    l'opération a réussi  
                  {0}    l'opération a échoué  
P2 (FU33) =            numéro d'erreur (100, 101, 104)

### Opérandes

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)               |
|----------------------|--------------------------|----------------------------------------|
| Fx.y                 | Var_sign                 | Value output sign mode <sup>1)</sup>   |
| Fx.y                 | Busy                     | Module transmitting text <sup>2)</sup> |

Rem. : x = 0...31    numéro de mot mémoire  
          y = 0...15    numéro de bit mémoire

- <sup>1)</sup> Les chiffres comportent un signe lorsque le bit mémoire est forcé.  
Plage de valeurs : Fx.y = 0      0...65535  
                          Fx.y = 1    -32768...32767.
- <sup>2)</sup> Le bit mémoire reste forcé pendant toute la transmission du texte à l'affichage.

**CMP/CFM****3FD\_216S.MAK/  
3FD^216S.MAK****Driver pour affichages E.ABG-1 et  
E.ABG-80-LED (textes dans la console ABG)**

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

**Format de saisie (STL)**

```
THEN    CMP  {00 .. 15}  
(or     CFM  {90 .. 99})  
        WITH <P1>  
        WITH <P2>  
        WITH <P3>  
        WITH <P4>
```

**Paramètres**

P1 = numéro de texte (1 .. 255)  
P2 = données non signées (0 .. 65535) ou  
données signées (-32768 .. 32768)  
P3 = nombre de chiffres affichés (1 .. 5)  
P4 = caractère d'introduction (32 .. 126)

**Paramètres de réponse**

P1 (FU32) = {-1}    l'opération a réussi  
                  {0}    l'opération a échoué  
P2 (FU33) =            numéro d'erreur (100, 101, 104)

### Opérandes

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| <b>Absolute<br/>(Absolu)</b> | <b>Symbolic<br/>(Symbolique)</b> | <b>Comment<br/>(Commentaire)</b>          |
|------------------------------|----------------------------------|-------------------------------------------|
| Fx.y                         | Var_Sign                         | Value output sign mode (VZ) <sup>1)</sup> |
| Fx.y                         | Busy                             | Module transmitting text <sup>2)</sup>    |

Rem. : x = 0...31    numéro de mot mémoire  
      y = 0...15    numéro de bit mémoire

- <sup>1)</sup> Les chiffres comportent un signe lorsque le bit mémoire est forcé.

Plage de valeurs : Fx.y = 0      0...65535  
                      Fx.y = 1    -32768...32767.

- <sup>2)</sup> Le bit mémoire reste forcé pendant toute la transmission du texte à l'affichage.

**CMP/CFM****3\_EABG.MAK/  
3^EABG.MAK****Driver pour claviers ABG-1 /-EL /-VF /-LED  
(scrutation du clavier et mémoire tampon de  
clavier)**

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

**Format de saisie (STL)**

```
THEN    CMP  {00 .. 15}  
(or     CFM  {90 .. 99})  
        WITH <P1>  
        WITH <P2>
```

**Paramètres**

aucun paramètre : valeurs de saisie autorisées 0 .. 65535

P1 = valeur maximale, si seul P1  
est transmis ou P2 < P1  
valeur minimale, si P1 < P2  
P2 = valeur maximale, si P1 < P2  
valeur minimale, si P2 < P1

Lors de l'appel d'un bloc logiciel sans transfert de paramètres, le système vérifie que la suite de chiffres saisie est comprise dans la plage de valeurs 0....65535. Si un seul paramètre est transmis lors de l'appel, celui-ci correspond à la valeur de saisie maximale. Si en revanche deux paramètres sont transmis, ils définissent une plage de valeurs autorisée pour les saisies.

**Paramètres de réponse**

Le bloc logiciel ne délivre aucun paramètre de réponse

**Opérandes**

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                |
|----------------------|--------------------------|-----------------------------------------|
| Fx.0                 | Enter                    | E.ABG : ENTER key pressed <sup>1)</sup> |
| Fx.1                 | F1                       | E.ABG : F1 key pressed <sup>1)</sup>    |
| Fx.2                 | F2                       | E.ABG : F2 key pressed <sup>1)</sup>    |
| Fx.3                 | F3                       | E.ABG : F3 key pressed <sup>1)</sup>    |
| Fx.4                 | F4                       | E.ABG : F4 key pressed <sup>1)</sup>    |
| Fx.5                 | Clear                    | E.ABG : CLEAR key pressed <sup>1)</sup> |
| Fx.6                 | Change                   | E.ABG : Value changed <sup>2)</sup>     |
| Fx.7                 | Initbuff                 | E.ABG : Init if flag = 0 <sup>3)</sup>  |
| Ry                   | Buffer                   | E.ABG : Keyboard buffer <sup>4)</sup>   |

Rem. : x = 0...31      numéro de mot mémoire  
           y = 0...127    numéro de mot du registre

- <sup>1)</sup> Les bits mémoire Fx.0 à Fx.5 sont forcés lorsque les touches qui leur sont attribuées sont actionnées. Les bits mémoire doivent être réinitialisés dans le programme utilisateur.
- <sup>2)</sup> Le bit mémoire de modification de valeur est forcé, lorsque la valeur de la mémoire tampon de clavier a changé. C'est le cas en particulier lorsque :
  - la touche CLEAR a été actionnée (la mémoire tampon de clavier est effacée)
  - l'une des touches du pavé numérique 0 à 9 a été actionnée.
- <sup>3)</sup> Le bit mémoire est forcé lors du premier appel du driver et lors de l'abandon du système de test. Une réinitialisation complète du bloc logiciel intervient dès que le bit mémoire est effacé.
- <sup>4)</sup> Ce registre permet de mémoriser la valeur saisie et sa position (mémoire tampon de clavier). La mémoire tampon de clavier est effacée, lorsque :
  - la touche CLEAR a été actionnée
  - la valeur saisie sort de la plage de valeurs autorisée.

### Limites du bloc

Il est important que le bloc logiciel ne soit appelé cycliquement que par un seul programme à la fois. Si le bloc reçoit des codes ASCII (codes clavier), ceux-ci sont ignorés.

**CMP/CFM**

3\_ABG80.MAK/  
3^ABG80.MAK

**Drivers pour clavier E.ABG-80-LED  
(scrutation du clavier et mémoire tampon de clavier)**

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

**Format de saisie (STL)**

```
THEN    CMP  {00 .. 15}  
(or     CFM  {90 .. 99})  
        WITH <P1>  
        WITH <P2>
```

**Paramètres**

aucun paramètre : valeurs de saisie autorisées 0 .. 65535

P1 = valeur maximale, si seul P1  
est transmis ou P2 < P1  
valeur minimale, si P1 < P2  
P2 = valeur maximale, si P1 < P2  
valeur minimale, si P2 < P1

Lors de l'appel d'un bloc logiciel sans transfert de paramètres, le système vérifie que la suite de chiffres saisie est comprise dans la plage de valeurs 0....65535. Si un seul paramètre est transmis lors de l'appel, celui-ci correspond à la valeur de saisie maximale. Si en revanche deux paramètres sont transmis, ils définissent une plage de valeurs autorisée pour les saisies.

**Paramètres de réponse**

Le bloc logiciel ne délivre aucun paramètre de réponse



## Opérandes

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| Absolute<br>(Absolu) | Symbolic<br>(Symbolique) | Comment<br>(Commentaire)                   |
|----------------------|--------------------------|--------------------------------------------|
| Fx.0                 | Enter                    | E.ABG-80 : ENTER key pressed <sup>1)</sup> |
| Fx.1                 | F1                       | E.ABG-80 : F1 key pressed <sup>1)</sup>    |
| Fx.2                 | F2                       | E.ABG-80 : F2 key pressed <sup>1)</sup>    |
| Fx.3                 | F3                       | E.ABG-80 : F3 key pressed <sup>1)</sup>    |
| Fx.4                 | F4                       | E.ABG-80 : F4 key pressed <sup>1)</sup>    |
| Fx.5                 | Clear                    | E.ABG-80 : CLEAR key pressed <sup>1)</sup> |
| Fx.6                 | Change                   | E.ABG-80 : Value changed <sup>2)</sup>     |
| Fx.7                 | Initbuff                 | E.ABG-80 : Init if flag = 0 <sup>3)</sup>  |
| Fx.12                | F5                       | E.ABG-80 : F5 key pressed <sup>1)</sup>    |
| Fx.13                | F6                       | E.ABG-80 : F6 key pressed <sup>1)</sup>    |
| Fx.14                | F7                       | E.ABG-80 : F7 key pressed <sup>1)</sup>    |
| Fx.15                | F8                       | E.ABG-80 : F8 key pressed <sup>1)</sup>    |
| Ry                   | Buffer                   | E.ABG-80 : Keyboard buffer <sup>4)</sup>   |

Rem. : x = 0...31      numéro de mot mémoire  
           y = 0...127     numéro de mot du registre

- 1) Les bits mémoire Fx.0 à Fx.5 et Fx.12 à Fx.15 sont forcés lorsque la touche qui leur est attribuée est actionnée. Les bits mémoire doivent être réinitialisés dans le programme utilisateur.
- 2) Le bit mémoire de modification de valeur est forcé, lorsque la valeur de la mémoire tampon de clavier a changé. C'est le cas lorsque :
  - la touche CLEAR a été actionnée (la mémoire tampon de clavier est effacée)
  - l'une des touches du pavé numérique 0 à 9 a été actionnée.

- 3) Le bit mémoire est forcé lors du premier appel du driver et lors de l'abandon du système de test. Une réinitialisation complète du bloc logiciel intervient dès lors que le bit est effacé (les bits mémoire et les registres sont réinitialisés).
- 4) Ce registre mémorise la valeur entrée avec sa position (mémoire tampon de clavier). La mémoire tampon de clavier est effacée lorsque :
  - la touche CLEAR a été actionnée
  - la valeur indiquée sort de la plage de valeurs autorisée.

### Limites du bloc

Il est important que le bloc logiciel ne soit appelé cycliquement que par un seul programme à la fois. Si le bloc reçoit des codes ASCII (codes clavier), ceux-ci sont ignorés.

**CMP/CFM**

3\_EABG2.MAK/  
3^EABG2.MAK

**Drivers pour clavier E.ABG-2  
(scrutation du clavier et témoins LED)**

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

**Format de saisie (STL)**

```
THEN    CMP  {00 .. 15}  
(or     CFM  {90 .. 99})
```

**Paramètres**

Aucun transfert de paramètres n'a lieu lors de l'appel d'un bloc logiciel.

**Paramètres de réponse**

Le bloc logiciel ne délivre aucun paramètre de réponse

## Opérandes

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| Absolute (Absolu) | Symbolic (Symbolique) | Comment (Commentaire)                         |
|-------------------|-----------------------|-----------------------------------------------|
| Fx.0              | F_1_key               | E.ABG2 : F1 key pressed <sup>1)</sup>         |
| Fx.1              | F_2_key               | E.ABG2 : F2 key pressed <sup>1)</sup>         |
| ...               | ...                   | E.ABG2 : F... key pressed <sup>1)</sup>       |
| Fx.14             | F15 key               | E.ABG2 : F15 key pressed <sup>1)</sup>        |
| Fx.15             | F16 key               | E.ABG2 : F16 key pressed <sup>1)</sup>        |
| Fy.0              | F17 key               | E.ABG2 : F17 key pressed <sup>1)</sup>        |
| Fy.1              | F18 key               | E.ABG2 : F18 key pressed <sup>1)</sup>        |
| Fy.2              | F19 key               | E.ABG2 : F19 key pressed <sup>1)</sup>        |
| Fy.3              | F20 key               | E.ABG2 : F20 key pressed <sup>1)</sup>        |
| Fy.4              | Status_OK             | E.ABG2 : Acknowledgement status <sup>3)</sup> |
| Fy.5              | Send_LED              | E.ABG2 : Update LED active <sup>4)</sup>      |
| Fy.8              | LED_F1                | E.ABG2 : LED F1key <sup>2)</sup>              |
| Fy.9              | LED_F2                | E.ABG2 : LED F2 key <sup>2)</sup>             |
| ...               | ...                   | E.ABG2 : LED F... key <sup>2)</sup>           |
| Fy.14             | LED_F7                | E.ABG2 : LED F7 key <sup>2)</sup>             |
| Fy.15             | LED_F8                | E.ABG2 : LED F8 key <sup>2)</sup>             |
| Fz.0              | LED_F9                | E.ABG2 : LED F9 key <sup>2)</sup>             |
| Fz.1              | LED_F10               | E.ABG2 : LED F10 key <sup>2)</sup>            |
| ...               | ...                   | E.ABG2 : LED F... key <sup>2)</sup>           |
| Fz.10             | LED_F19               | E.ABG2 : LED F19 key <sup>2)</sup>            |
| Fz.11             | LED_F20               | E.ABG2 : LED F20 <sup>2)</sup>                |
| Fz.12             | Status                | E.ABG2 : Request status <sup>5)</sup>         |
| Fz.13             | INIT                  | E.ABG2 : Program module init <sup>6)</sup>    |

Rem. : x = 0...31 numéro de mot du bit mémoire de touche

y = 0...31 numéro de mot du bit mémoire de touche/LED

z = 0...31 numéro de mot mémoire de LED

Légendes des tableaux <sup>1)</sup> à <sup>6)</sup> voir page suivante.

- 1) Les bits mémoire Fx.0 à Fx.15 et Fy.0 à Fy.3 sont forcés lorsque la touche qui leur est attribuée est actionnée et sont réinitialisés lorsque la touche est relâchée.
- 2) Les bits mémoire Fy.8 à Fy.15 et Fz.0 à Fz.11 sont destinés à la commande des LED. Lorsqu'un bit mémoire change d'état la touche de LED correspondante est activée/désactivée.
- 3) Le bit mémoire est forcé lorsque le bloc logiciel reçoit un avis de réception de la demande d'état (Fz.12) de la part de la console E.ABG-2 (= O.K.). Le bit mémoire doit être réinitialisé dans le programme utilisateur.
- 4) Ce bit mémoire reste forcé jusqu'à l'émission d'une ligne de commande pour la mise à jour des LED.
- 5) Le forçage du bit mémoire permet d'interroger l'état de l'afficheur. Le bloc logiciel réinitialise le bit mémoire lorsque la commande est réceptionnée. La demande d'état ne doit pas être effectuée cycliquement mais doit être commandée par le temporisateur. Il est ainsi possible de contrôler le bon fonctionnement de l'affichage.
- 6) Le bloc est initialisé dès que le bit mémoire est forcé :
  - les bits mémoire de touches sont réinitialisés
  - la séquence d'initialisation est transmise à l'affichage, toutes les LED sont désactivées
  - les bits mémoire internes au bloc permettant l'analyse du front sont réinitialisés
  - les bits mémoire des avis de réception d'état et de mise à jour des LED sont réinitialisés.L'initialisation est automatique :
  - lors du premier appel du bloc
  - lors de l'abandon du système de test (commande CI 'X')

### Limites du bloc

Il est important que le bloc logiciel ne soit appelé cycliquement que par un seul programme à la fois. Si le bloc reçoit des codes ASCII (codes clavier), ceux-ci sont ignorés.

**CMP/CFM**

3\_ETMV5.MAK/  
3^ETMV5.MAK

**Saisie/affichage de caractères à travers l'interface de diagnostic (bloc multitâche)**

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

Le bloc logiciel permet l'affichage de textes, de variables et de caractères de commande, l'appel de textes mémorisés dans la console ABG, la saisie de caractères ASCII quelconques (ainsi que de codes clavier ABG), ainsi que la saisie de valeurs numériques pour les variables (opérantes de bits multiples) à travers l'interface RS-232 (DIAG) de l'automate SF 3.

**Format de saisie (STL)**

```
THEN    CMP {00 .. 15}
(or     CFM {90 .. 99})
        WITH <P1>
        WITH <P2>
        WITH <P3>
        WITH <P4>
```

**Paramètres**

P1 = (0 .. 6) Mode du bloc,  
 0 = affichage de texte  
 1 = affichage de variables  
 2 = affichage de caractères de commande  
 3 = saisie de valeurs sans timeout  
 4 = saisie de valeurs avec timeout  
 5 = saisie de caractères ASCII  
 6 = affichage de textes internes à la console ABG  
 Voir description ci-dessous

P2 .. P4 = suivant le mode choisi pour P1

Paramètres de réponse

Cas 1

P1 (FU32) = {0} l'opération a réussi ou la valeur saisie est valide  
P2 (FU33) = valeur saisie (modes 3 et 4)  
P2 .. Pnn = caractères reçus, opérandes spécifiques définis par le 2e paramètre de saisie (Mode 5)

Cas 2

P1 (FU32) = {<>0} l'opération a échoué  
P2 (FU33) = numéro d'erreur (100, 101, 102, 103, 104), suivant le mode

Opérandes

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| Absolute (Absolu) | Symbolic (Symbolique) | Comment (Commentaire)                     |
|-------------------|-----------------------|-------------------------------------------|
| Fx.y              | Time_out              | Flag for input status <sup>2)</sup>       |
| Fx.y              | Abort_inp             | Flag for input abort <sup>3)</sup>        |
| Fx.y              | Var_sign              | Value output sign mode <sup>4)</sup>      |
| Rm                | AsciiVal              | Return Value of ASCII input <sup>5)</sup> |
| Tn                | Inp_timer             | Timer for input timeout <sup>3)</sup>     |
| TVn               | Inp_TPRE              | Timer preselect for input <sup>1)</sup>   |
| TWn               | Inp_Tword             | timer word-input timeout <sup>1)</sup>    |

Rem. : n = 0...31      numéro de bit pour l'état du temporisateur/numéro de mot pour la présélection du temporisateur, mot du temporisateur  
                              x = 0...31      numéro de mot du bit mémoire  
                              y = 0...15      numéro du bit mémoire  
                              m = 0...127    numéro de mot du registre

Légendes des tableaux <sup>1)</sup> à <sup>5)</sup> voir page suivante.

- 1) Le temporisateur contrôle la saisie de valeurs pour une variable. En cas de dépassement du délai spécifié dans le programme, le bit mémoire du timeout est forcé et le bloc logiciel est abandonné.
- 2) Ce bit mémoire est forcé si la valeur de la variable n'est pas spécifiée dans le délai prédéfini. En cas d'erreur (timeout), le bloc logiciel est abandonné.
- 3) Si le bit mémoire est forcé, la saisie de caractères ASCII est interrompue avant le retour au programme en cours.
- 4) Les chiffres comportent un signe lorsque le bit mémoire est forcé. Plage de valeurs :  
Fx.y = 0    0...65535  
Fx.y = 1    -32768...32767
- 5) Ce registre garde en mémoire le ou les deux derniers caractères ASCII saisis, le dernier dans l'octet de poids faible, l'avant dernier dans l'octet de poids fort.

### Description des modes

La fonction du bloc est déterminée par le 1er paramètre d'appel.

Selon la catégorie, le système vérifie le nombre de paramètres autorisés. Si ce nombre n'est pas respecté, le message d'erreur 100 est émis.



Les modes suivants peuvent être paramétrés :

| P1 | Fonction                                                            | Autres paramètres de transfert                                                                                                                                                                                                                                                                                    |
|----|---------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0  | Affichage de texte                                                  | P2 = chaîne de caractères ASCII, affichée à travers l'interface de diagnostic                                                                                                                                                                                                                                     |
| 1  | Affichage de variables non formatées                                | P2 = 0...65535 valeurs non signées<br>ou = -32768...32767 valeurs signées                                                                                                                                                                                                                                         |
|    | Affichage de variables formatées                                    | P2 = 0...65535 variables non signées<br>ou = -32768...32767 variables signées<br>P3 = 1...5 nombre de chiffres affichés<br>P4 = 32...126 caractère d'introduction (en option)                                                                                                                                     |
| 2  | Affichage de caractères de commande                                 | P2 = paramètre en chaîne de caractères, long. max. 27 caractères Hex (54 caractères)                                                                                                                                                                                                                              |
| 3  | Saisie de valeurs sans timeout pour les opérandes de bits multiples | Nombres hexadécimaux admissibles : \$0.....\$FFFF<br>Nombres négatifs admissibles : -32768.....32767<br>Nombres positifs admissibles : 0....65535                                                                                                                                                                 |
| 4  | Saisie de valeurs avec timeout pour les opérandes de bits multiples | Nombres hexadécimaux admissibles : \$0.....\$FFFF<br>Nombres négatifs admissibles : -32768.....32767<br>Nombres positifs admissibles : 0....65535                                                                                                                                                                 |
| 5  | Saisie de caractères ASCII                                          | - Enregistrement de codes clavier ABG --> pas d'autres paramètres, Validation avec <CR>, Valeurs dans le registre "AsciiVal"<br>- Enregistrement de chaîne de caractères ASCII --> P2, P3<br>P2 = 48...4095 Numéro d'opérande spécifique<br>P3 = 1...4047 Abandon après le nombre de caractères reçus (en option) |
| 6  | Affichage de textes internes à la console ABG                       | P2 = Numéro du texte mémorisé dans la console ABG<br>Lors de l'appel, la chaîne <1dh> <01h> est émise.                                                                                                                                                                                                            |

L'éditeur d'affichage du FST200 n'est pas indispensable pour créer les textes et les caractères de commande à afficher. Avec l'appel du bloc, ces derniers peuvent être définis comme des paramètres en chaîne de caractères et être mémorisés dans le programme qui appelle.

Les paramètres peuvent être transmis sous forme de constantes ou à partir d'opérandes de bits multiples.

#### Remarques

Un même bloc logiciel peut être appelé simultanément à partir de plusieurs programmes, à condition que la valeur (fonction) transférée par le paramètre P1 soit **différente**.

#### Exemple :

**Le programme P0 affiche du texte (P1 = 0) et le programme P5 attend la saisie de caractères ASCII (P1 = 5).**

Si ceci ne peut pas être garanti, effectuer un verrouillage à l'aide de bits mémoire, ou utiliser les programmes pour l'appel des différents blocs logiciels.

**CMP/CFM**

3\_PCS.MAK/  
3^PCS.MAK

**Driver pour console de commande PCS  
(en liaison avec le logiciel de programmation  
PCSPRO)**

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

Il est utilisé avec les consoles de commande PCS paramétrées (paramétrage à l'aide du logiciel PCSPRO, de la société Lauer) sur l'interface RS-232 (DIAG) de l'automate SF 3.

Le bloc logiciel permet d'une part de rétablir la liaison avec la console PCS lors du redémarrage ou après une interruption de la communication et d'autre part de réguler l'échange de données entre les champs paramétrés des opérandes spécifiques du SF 3 (256 mots) et les instructions reçues par l'interface série du PCS.

**Format de saisie (STL)**

```
THEN    CMP  {00 .. 15}  
(or    CFM  {90 .. 99})  
        WITH <P1>
```

**Paramètre**

aucun paramètre = ...  
P1 = ...

**Paramètres de réponse****Cas 1**

P1 (FU32) = {0} ...

P2 .. Pnn = ...

**Cas 2**

\*\*\*P1 (FU32) = {<> 0} l'opération a échoué

\*\*\*P2 (FU33) = numéro d'erreur (100, 101, 102, 103, 104), suivant le mode

**Opérandes**

Les opérandes suivants sont utilisés par les blocs internes et doivent être spécifiés lors de la configuration du bloc :

**Nombre de répétitions (1..255)**

En cas de réception de données erronées, la recherche des données est répétée. Indiquer le nombre maximal de répétitions de la recherche (1..255). Après réception des données correctes, le comptage reprend au début. Le dépassement du nombre maximal de répétitions entraîne une interruption de la communication.

**Timeout (0,6 s à 2,55 s)**

Pendant la communication entre la console PCS et l'automate SF 3, les données échangées doivent en permanence être remises à jour. Le timeout est la durée maximale séparant l'émission d'un télégramme de réponse par le SF 3 et la réception d'un nouveau paquet d'instructions par le PCS. La valeur saisie doit être comprise entre 60 et 255.

La valeur du timeout s'obtient de la manière suivante : Valeur de saisie fois 10 ms, ce qui permet de spécifier des temps compris entre 0,6 s et 2,55 s (valeur par défaut : 0,6 s).

En cas d'erreur (timeout) le système doit être redémarré.

Les opérandes suivants sont utilisés par le bloc logiciel et inscrits lors de la configuration dans la liste des affectations :

| <b>Absolute<br/>(Absolu)</b> | <b>Symbolic<br/>(Symbolique)</b> | <b>Comment<br/>(Commentaire)</b>      |
|------------------------------|----------------------------------|---------------------------------------|
| Fx.y                         | PCS_Init                         | PCS init flag <sup>1)</sup>           |
| FEn                          | PCS_Start                        | PCS_data (start adress) <sup>2)</sup> |

Rem. : x = 0...31      numéro de mot  
         y = 0...15      numéro de bit mémoire  
         n = 48...3840    numéro de mot de l'opérande  
                             spécifique

- <sup>1)</sup> Bit mémoire d'initialisation  
Le bit mémoire choisi est un bit d'erreur général, qui reflète l'état de la communication. Si ce bit = 1, l'échange de données a été arrêté. La reprise de la communication par l'appel du bloc avec le paramètre 0 a réussi lorsque le bit reprend la valeur 0.
- <sup>2)</sup> Affectation des opérandes spécifiques (FU48 à FU3840)  
L'adresse de base de la plage des mots de données que l'automate SF3 met à disposition de la console PCS (valeur par défaut FU48). Les 256 mots qui suivent sont toujours affectés de manière fixe.

### Description

Le bloc logiciel configuré est destiné aussi bien à l'initialisation qu'à l'échange cyclique de données avec la console de commande PCS. Les différentes fonctions sont déterminées par le transfert de paramètres (voir Transfert de paramètres).

Pour l'échange de données, la console PCS a le rôle du Maître, et l'automate SF 3 celui de l'esclave. La console de commande envoie des paquets d'instructions à l'automate SF 3, lequel les traite pour renvoyer des données de réponse au PCS.

Pour que l'échange de données soit valable, il est nécessaire d'appeler cycliquement le bloc logiciel. L'automate SF 3 prend seulement le rôle du Maître de la communication pour émettre le télégramme Reset. Ce dernier permet de relancer la communication après un redémarrage du SF 3 ou après une interruption de la communication.

Les paquets d'instructions sont formulés par la console PCS, et toutes les opérations à effectuer (lecture, écriture, opérations OU et ET) doivent faire intervenir les opérandes spécifiques dans la plage spécifiée. Seules les données réellement utiles sont transmises, ce qui réduit considérablement leur nombre.

Le bloc logiciel est abandonné après chaque appel, et le programme en cours reprend.

Transfert de paramètres et messages d'erreur

Deux variantes permettent d'appeler les blocs logiciel configurés :

### **Variante 1**

Appel à l'aide d'un paramètre de saisie : P1 = 0.  
Etablir la liaison à l'aide de la console PCS, envoyer le télégramme Reset, appel succédant à un redémarrage du SF 3 ou à une interruption de la communication.

Sortie :

Paramètre

|    |      |      |                                              |
|----|------|------|----------------------------------------------|
|    | FU32 | = 0  | liaison établie avec succès                  |
|    | FU33 | = 0  |                                              |
| ou | FU32 | <> 0 | message d'état (voir ci-dessous)             |
|    | FU33 | = 0  |                                              |
| ou | FU32 | = 0  |                                              |
|    | FU33 | =    | numéro d'erreur<br>(LED Error rouge allumée) |
|    | 100  | =    | nombre de paramètres<br>erroné               |
|    | 101  | =    | erreur sur le paramètre 1                    |

### **Variante 2**

Appel sans paramètres de saisie : Régule l'échange de données entre le champ de données configuré pour les opérandes spécifiques (256 mots) et les paquets d'instructions reçus par l'interface série.

Sortie :

Paramètre

FU32 = 0 l'opération a réussi  
 FU33 = 0  
 ou FU32 <> 0 message d'état (voir ci-dessous)  
 FU33 : nombre de demandes de répétitions  
 ou FU32 = 0  
 FU33 = numéro d'erreur (LED Error rouge allumée)  
 100 = nombre de paramètres erroné

### Message d'état :

| Bit | Etat                                                                                      |
|-----|-------------------------------------------------------------------------------------------|
| 0   | Erreur de parité<br>Défaut de transmission, les instructions ne sont pas traitées         |
| 2   | Dépassement de capacité<br>Défaut de transmission, les instructions ne sont pas traitées  |
| 3   | Erreur de télégramme BCC<br>Défaut de transmission, les instructions ne sont pas traitées |
| 4   | Timeout<br>Interruption de la communication, pas de liaison                               |
| 15  | Phase d'initialisation non validée.                                                       |

### Exemple d'application

#### Extrait de programme STL

```

IF      PCS_Init      'PCS Init flag
THEN    CMP 0          'Driver for PCS...
        WITH V0       "Connection
OTHRW   CMP 0          'Driver for PCS...
```



**CMP/CFM**

3\_32BITI.MAK/  
3^32BITI.MAK

### Driver pour calcul 32-bits (4 opérations fondamentales)

Le bloc logiciel est configuré puis relié au programme utilisateur à l'aide de la fonction "Include module (relier un module)".

Le driver peut être utilisé partout où la précision du calcul standard 16 bits n'est plus suffisante. Il dispose des quatre opérations fondamentales : addition, soustraction, multiplication et division. Les valeurs autorisées sont comprises entre 0 et +4294967295.

#### Format de saisie (STL)

```
THEN    CMP {00 .. 15}
(or     CFM {90 .. 99})
        WITH <P1>
        WITH <P2>
        WITH <P3>
        WITH <P4>
        WITH <P5>
```

#### Paramètres

P1 = opérande 1 de poids faible (16 bits)  
 P2 = opérande 1 de poids fort (16 bits)  
 P3 = fonction du bloc (1 .. 4)  
     1: Addition  
     2: Soustraction  
     3: Multiplication  
     4: Division  
 P4 = opérande 2 de poids faible (16 bits)  
 P5 = opérande 2 de poids fort (16 bits)

Les paramètres P1/P2 ainsi que P4/P5 réunis contiennent toujours une valeur 32 bits (0 à + 4 294 967 295)

**Paramètres de réponse****Cas 1**

|          |   |      |                               |
|----------|---|------|-------------------------------|
| P1 (F32) | = | {0}  | pas d'erreur                  |
|          |   | {21} | dépassement en soustraction   |
|          |   | {41} | dépassement en multiplication |
|          |   | {51} | division par 0                |

En cas de dépassement en addition : toujours 0 ! -->  
contrôler FU35

|           |     |                                                        |
|-----------|-----|--------------------------------------------------------|
| FU35      | {0} | dépassement en addition                                |
| P2 (FU33) | =   | résultat en poids faible (en l'absence de dépassement) |
| P3 (FU34) | =   | résultat en poids fort (en l'absence de dépassement)   |

Les paramètres P2/P3 réunis contiennent le résultat sous forme de valeur 32 bits (0 à + 4 294 967 295)

**Cas 2**

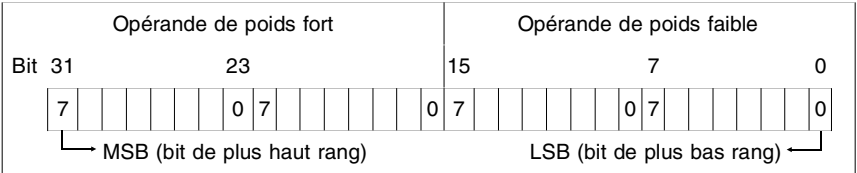
|           |   |                            |                                           |
|-----------|---|----------------------------|-------------------------------------------|
| P1 (FU32) | = | {<> 0}                     | l'opération a échoué                      |
| P2 (FU33) | = | messages d'erreur 100, 103 | (la fonction indiquée n'est pas reconnue) |

Inscriptions dans la liste des affectations

**Aucune** inscription n'est effectuée dans la liste des affectations pendant la configuration d'un bloc.

Description

Les opérandes transférés sous forme de paramètres lors de l'appel d'un bloc sont représentés comme des valeurs 32 bits.



Temps d'exécution

| Opération      | Temps d'exécution (valeur max.) |
|----------------|---------------------------------|
| Addition       | 120 µs                          |
| Soustraction   | 120 µs                          |
| Multiplication | 120 µs                          |
| Division       | 140 µs                          |

Les temps d'exécution indiqués sont des valeurs maximales (les valeurs types sont inférieures d'env. 25 %) et comprennent l'appel complet du bloc.

Exemples d'application

Exemple 1 :

Addition de valeurs 32 bits dans les paires de registres R1:R0 et R3:R2 (fort:faible), stockage du résultat dans la paire de registres R11:R10 (fort:faible)

```
CMP 0                                calcul '32-bits
    WITH R0
    WITH R1
    WITH V1      "Addition
    WITH R2
    WITH R3
IF      FU35      "overrun
    =      V0      "(work around)
THEN LOAD  FU33      "Result is valid
    TO      R10
    LOAD      FU34
    TO      R11
    .
    .
OTHRW   JMP TO overflow  "Overflow!
```

Le dépassement pour l'addition est recherché différemment en fonction de la description des erreurs (voir remarques).

| 1er terme          |          | 2e terme          |          | Etat    | Somme    |                   |          |
|--------------------|----------|-------------------|----------|---------|----------|-------------------|----------|
| R1                 | R0       | R3                | R2       | FU32    | FU35     | FU34              | FU33     |
| 0001 hex           | AEF0 hex | 0001 hex          | 6058 hex | 0 hex   | 0000 hex | 0003 hex          | 0F48 hex |
| 110 320 déc.       |          | 90 200 déc        |          | 0       | 0        | 200 520 déc       |          |
| R1                 | R0       | R3                | R2       | Fu32    | FU35     | FU34              | FU33     |
| 7D2B hex           | 7500 hex | 8321 hex          | 5600 hex | 1 hex*) | 0001 hex | 004C hex          | CB00 hex |
| 2 100 000 000 déc. |          | 2 200 000 000 déc |          | 1*)     | 1        | 4 300 000 000 déc |          |

\*) pour les blocs logiciels créés à partir de fichiers MAK des versions du FST200 postérieures à 3.2 - Révision 8 (8/97), le dépassement est indiqué dans FU32.

Dans les versions antérieures l'erreur de dépassement en addition n'est pas indiquée par FU32 <> 0, dans ce cas, consulter FU35 (voir remarques) !

Exemple 2 :  
Multiplication de valeurs 32 bits dans les paires de registres R1:R0 et R3:R2 (fort:faible), stockage du résultat dans la paire de registres R11:R10 (fort:faible)

```
CMP 0                                'calcul '32-bits
    WITH R0
    WITH R1
    WITH V3    "Multiplication
    WITH R2
    WITH R3
IF FU32 "Overrun ?
    = V41
THEN JMP TO overfl
OTHRW LOAD FU33 "Result is valid
    TO R10
    LOAD FU34
    TO R11
```

| 1er terme   |          | 2e terme    |          | Etat   | Produit  |                   |          |
|-------------|----------|-------------|----------|--------|----------|-------------------|----------|
| R1          | R0       | R3          | R2       | FU32   | FU35     | FU34              | FU33     |
| 0000 hex    | A7F8 hex | 0000 hex    | 0064 hex | 0 hex  | 0000 hex | 0041 hex          | 9CE0 hex |
| 43 000 déc. |          | 100 déc     |          | 0      | 0        | 4 300 000 déc     |          |
| R1          | R0       | R3          | R2       | FU32   | FU35     | FU34              | FU33     |
| 0000 hex    | A7F8 hex | 0000 hex    | 03E8 hex | 0 hex  | 0000 hex | 0290 hex          | 20C0 hex |
| 43 000 déc  |          | 1000 déc    |          | 0      | 0        | 43 000 000 déc    |          |
| R1          | R0       | R3          | R2       | FU32   | FU35     | FU34              | FU33     |
| 0000 hex    | A7F8 hex | 0000 hex    | 2710 hex | 0 hex  | 0000 hex | 19A1 hex          | 4780 hex |
| 43 000 déc  |          | 10000 déc   |          |        | 0        | 4 30 000 000 déc  |          |
| R1          |          | R3          |          | FU32   | FU35     | FU34              | FU33     |
| 0000 hex    | A7F8 hex | 0001 hex    | 86A0 hex | 29 hex | 0001 hex | 004C hex          | CB00 hex |
| 43 000 déc  |          | 100 000 déc |          | 41 déc | 1        | 4 300 000 000 déc |          |

### Remarques

En créant un bloc pour le calcul 32 bits à partir d'un fichier MAK provenant d'une version du logiciel FST200 antérieure à la version 3.2, révision 8 (8/97), le dépassement en addition n'est pas indiqué par FU32 (<> 0). Dans ce cas, il est nécessaire de consulter FU35 !

### Workaround :

Lorsque le dépassement n'est pas indiqué, il peut toutefois être décelé en exploitant le paramètre de réponse FU35 :

FU35 = 0 : pas de dépassement

FU35 <> 0 : dépassement



# **Annexe B**

## **Annexe technique**



**Sommaire**

|                                                                                       |      |
|---------------------------------------------------------------------------------------|------|
| <b>Réglages des consoles d'affichage et de commande</b> .....                         | B-3  |
| <b>Câbles de liaison</b> .....                                                        | B-4  |
| Câbles de connexion SF 3 - console ABG.                                               | B-4  |
| Câble de liaison entre PC et console ABG.                                             | B-5  |
| Câble de liaison entre l'ordinateur et l'afficheur avec interface série 9 pôles : ... | B-6  |
| Câble de liaison pour la programmation de l'EPROM. ....                               | B-6  |
| Câbles de connexion PCS - SF3 .....                                                   | B-7  |
| <b>Messages d'erreur.</b> .....                                                       | B-8  |
| <b>Définition des concepts</b> .....                                                  | B-15 |

## Réglages des consoles d'affichage et de commande

S'assurer que pour le fonctionnement des consoles ABG les réglages de l'interface correspondent à ceux du SF 3.

Interface :

- RS232 (V.24),
- vitesse de transmission 9600 baud,
- fonctionnement Full Duplex
- 1 bit de départ, 8 bits de données, pas de parité, 1 bit d'arrêt  
(9600, N, 8, 1)

Recommandation : Régler également la console ABG sur "Text input!output (saisie/affichage de textes)" (mode interactif).

Des informations sur le réglage des consoles ABG figurent dans le manuel d'utilisation joint à la console.

## Câbles de liaison

Des câbles de liaison sont nécessaires pour transmettre les données entre les différents appareils.

Des câbles de liaison spéciaux non compris dans le programme de livraison de Festo doivent être confectionnés. La figure correspondante indique l'affectation des broches.

## Câbles de connexion SF 3 - console ABG

| Type de console ABG                  | Câbles de connexion                                        | Référence          |
|--------------------------------------|------------------------------------------------------------|--------------------|
| E.ABG-1,<br>E.ABG-80-LED             | E.KA1S2-K (9 pôles) ou<br>E.KABG-SB202-ST9                 | 269 112<br>150 288 |
| E.ABG2,<br>E.FD-2/40-S               | E.KA2S2-K (25 pôles)                                       | 269 113            |
| PCS                                  | voir paragraphe "Câbles de connexion PCS - SF3"            | –                  |
| Afficheur ou terminal ASCII au choix | Raccordement au PC en fonction de la console ABG utilisée. | –                  |

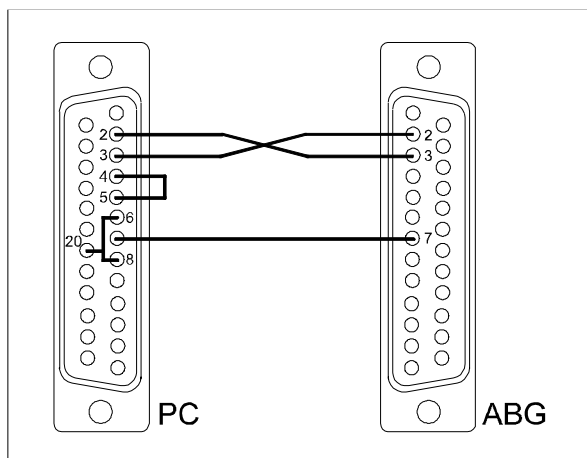
**Câble de liaison entre PC et console ABG**

Fig. B/1 : Câble de connexion PC (9 pôles) - console ABG

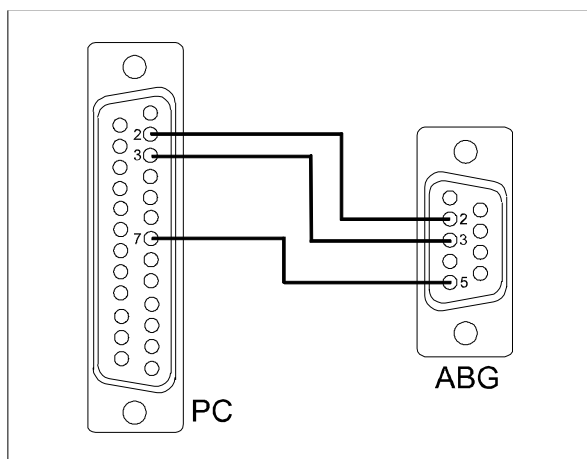


Fig. B/2 : Câble de connexion PC (25 pôles) - console ABG

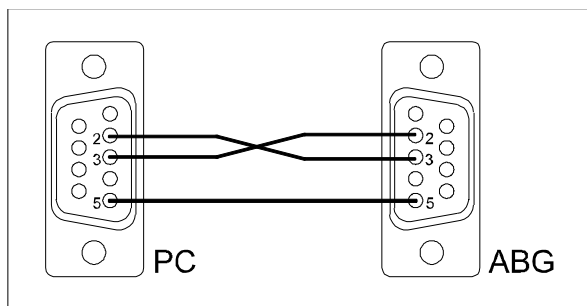


Fig. B/3 : Câble de connexion PC (25 pôles) - console ABG (25 pôles)

### **Câble de liaison pour la programmation de l'EPROM**

Utiliser un câble répondant aux caractéristiques du programmeur EPROM.

## Câbles de connexion PCS - SF3

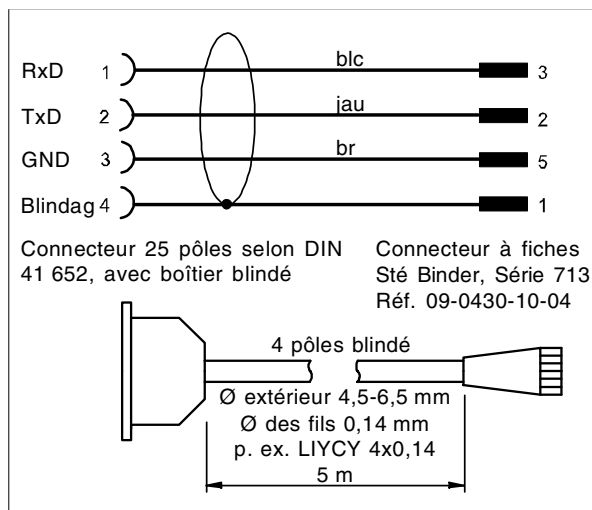


Fig. B/4 : Câble de connexion PCS-SF3

## Messages d'erreur

Comme ils sont courts, les messages d'erreur de l'éditeur d'affichage ne peuvent pas donner des indications détaillées sur les causes d'erreurs et leur élimination. Ce paragraphe récapitule donc les messages d'erreur par ordre alphabétique et donne des explications pour chaque message. Il s'agit d'indications sur l'apparition et l'élimination de l'erreur.

### D

#### **Display file is empty! (fichier d'affichage vide)**

CAUSE : La fonction activée (p. ex. Burn EPROM (graver EPROM)) n'a aucune signification pour le fichier sélectionné car ce dernier est vide. Le fichier a sans doute été sélectionné par erreur.

SOLUTION : Sélectionner un autre fichier.

### F

#### **Filename may only be used as a function module (impossible de créer un bloc de programme pour cet affichage)**

CAUSE : On désire créer un bloc de programme pour le mode standard (créer CMP). Le fichier sélectionné a cependant été créé pour un afficheur qui ne permet pas ce mode de fonctionnement.

SOLUTION : Après activation de la fonction Create CMP (créer CMP), sélectionner un fichier créé pour un afficheur qui permet ce mode de fonctionnement.

#### **File too long. Delete something (trop de données : effacer quelque chose)**

CAUSE : Le fichier d'affichage créé nécessite un espace mémoire supérieur à la capacité de l'afficheur pour lequel il a été créé.

SOLUTION : Effacer des caractères de texte ou de commande superflus.

**I**

**Invalid field bus address. Value must be 2 or 4 (Adresse bus non valide, seules les valeurs 2 et 4 sont possibles) !**

CAUSE : L'adresse indiquée pour le coupleur de bus est incorrecte.

SOLUTION : Seules les adresses 2 et 4 sont possibles. Entrer 2 ou 4.

**Invalid field bus participant address. Range is 01 to 99 (Adresse abonné non valide. Entrer une valeur entre 01 et 99) !**

CAUSE : Le numéro d'abonné entré pour l'afficheur est incorrect.

SOLUTION : Seul un numéro d'abonné compris entre 1 et 99 est autorisé.

**L**

**Load display RAM via field bus not possible with this display (chargement RAM d'affichage via bus de terrain impossible avec ce type d'afficheur)**

CAUSE : Après activation de la fonction Load display RAM (field bus) (chargement RAM d'affichage (bus de terrain)), un mauvais fichier a été sélectionné. Ce fichier a été créé pour un type d'afficheur pour lequel il est impossible de charger via un bus de terrain.

SOLUTION : Sélectionner un fichier pour lequel le fonctionnement avec bus de terrain est possible.



**Load display RAM via V24 not possible with this display (chargement RAM d'affichage via V24 impossible avec ce type d'afficheur) !**

CAUSE : Le fichier sélectionné après activation de la fonction Load display RAM (charger RAM d'affichage) (V.24), a été créé pour un type d'afficheur pour lequel il est impossible de charger via l'interface V.24.

SOLUTION : Sélectionner un fichier créé pour un afficheur avec interface V-24.

## **M**

**MAK file not installed (fichier MAK non installé) !**

CAUSE : On désire créer un bloc pour le mode standard. Mais le driver d'affichage nécessaire (\*.MAK) ne se trouve pas dans le fichier de bibliothèque (ANZED.BIB). Soit le mode standard n'est pas possible pour le type d'afficheur sélectionné, soit le driver n'est pas encore disponible au moment de la création du logiciel.

SOLUTION : Vérifier si la version actuelle de l'éditeur d'affichage est installée sur l'ordinateur. Il est possible aussi qu'une simple erreur de sélection du fichier ait eu lieu lors de la création du bloc.

**Module ?? ! CPU ? already exists in the project directory (bloc ?? / UC ? existe déjà dans le répertoire des projets) ! Overwrite? (l'écraser ?)**

CAUSE : Lors de la création du bloc, un numéro de bloc ou un numéro d'UC a été sélectionné pour lequel un bloc existe déjà.

SOLUTION : Entrer Y pour écraser le bloc existant. OU : Entrer N pour non et sélectionner un autre numéro de bloc ou d'UC.

**N****No display files available (pas de fichier d'affichage trouvé)**

CAUSE : La fonction activée suppose la sélection d'un fichier (p. ex. Delete file (supprimer fichier)). Mais le projet activé ne contient pas de fichier d'affichage. Aucune sélection de fichier ne peut donc être proposée.

SOLUTION : Vérifier si le bon projet est activé.  
OU : Créer d'abord un fichier pour le projet actuel.

**P****Project does not exist (pas de projet) !**

CAUSE : L'éditeur d'affichage a été lancé sans qu'un projet n'ait été sélectionné dans le logiciel FST. L'éditeur d'affichage doit connaître le projet pour lequel on veut créer des fichiers d'affichage.

SOLUTION : Sélectionner un projet à l'aide du logiciel FST ou créer un nouveau projet.

**S****Specified file is not a FESTO display file (le fichier indiqué n'est pas un fichier d'affichage Festo) !**

CAUSE : Chaque fichier d'affichage contient une note. Le fichier d'affichage sélectionné (\*.ANZ) ne contient pas cette identification indispensable. Le fichier est sans doute endommagé.

SOLUTION : Seule une personne possédant des connaissances sur le système d'exploitation peut remédier à ce problème. Comme une copie de sauvegarde avec l'extension ".BAK" est créée à chaque nouvel enregistrement d'un fichier d'affichage, il existe probablement aussi une copie de sauvegarde pour ce fichier endommagé. Effacer d'abord le fichier endommagé à l'aide du système d'exploitation.

**REMARQUE :**

Si le fichier est effacé à l'aide de l'éditeur d'affichage, la copie de sauvegarde est également effacée. Après avoir effacé le fichier, il est possible de renommer la copie de sauvegarde à l'aide du système d'exploitation. Donner au fichier le même nom mais avec l'extension .ANZ. Vérifier que la fonction désirée est possible avec ce fichier.

**T****The last text overwritten (overrun)!  
Insert anyway? (le texte va être écrasé  
(dépassement). Insérer quand même ?)**

**CAUSE :** L'utilisateur veut insérer un nouveau numéro de texte dans le fichier d'affichage. Pour cela, tous les numéros de texte suivants doivent être incrémentés de 1. Comme du texte a déjà été saisi pour le numéro de texte maximal et que le numéro ne peut pas être incrémenté, ce texte sera perdu en cas d'insertion.

**SOLUTION :** Entrer Y pour Yes (oui) pour insérer un numéro de texte et le texte attribué au numéro maximal est effacé. OU : Entrer N pour non et effacer un numéro de texte inutile avant d'insérer un nouveau numéro de texte.

**Text ??? already exists. Overwrite?  
(Le texte ??? existe déjà : l'écraser ?)**

**CAUSE :** On désire copier un texte sélectionné dans un numéro de texte pour lequel du texte est déjà saisi.

**SOLUTION :** Entrer Y pour Yes (oui) pour écraser le texte existant. OU : Entrer N pour non pour annuler la copie.

**There exists no MAK file (fichier MAK introuvable)**

CAUSE : On désire créer un bloc. Le driver d'affichage nécessaire (\*.MAK) ne se trouve pas dans le répertoire de la bibliothèque (LIB).

SOLUTION : Importer le driver d'affichage nécessaire dans le répertoire LIB.

**There is no project selected (aucun projet sélectionné) !**

CAUSE : L'éditeur d'affichage a été lancé sans qu'un projet n'ait été sélectionné dans le logiciel FST. L'éditeur d'affichage doit connaître le projet pour lequel on veut créer des fichiers d'affichage.

SOLUTION : Sélectionner le projet désiré avant de lancer l'éditeur d'affichage.

**Too many files (trop de fichiers)**

CAUSE : La création d'un nouveau fichier d'affichage provoque le dépassement du nombre maximal autorisé de fichiers d'affichage pour le projet actuel. Chaque projet peut contenir au maximum 127 fichiers d'affichage.

SOLUTION : Effacer d'abord du projet actuel un fichier d'affichage devenu inutile avant d'en créer un nouveau.

**U****Unknown display type (type d'afficheur inconnu) !**

CAUSE : Le fichier d'affichage avec lequel on désire travailler a été créé pour un type d'afficheur qui n'est pas répertorié dans le fichier de bibliothèque (ANZED. BIB). Le fichier a probablement été créé sur un autre ordinateur sur lequel se trouve une version plus récente du fichier de bibliothèque.

SOLUTION : Installer la version de l'éditeur d'affichage avec laquelle le fichier d'affichage a été créé.

## **Y**

**You cannot burn an EPROM for this display  
(impossible de créer une mémoire EPROM  
pour cet affichage)**

CAUSE : Le fichier sélectionné après activation de la fonction Burn EPROM (graver EPROM) a été créé pour un afficheur qui ne permet pas des opérations sur l'EPROM.

SOLUTION : Sélectionner un fichier créé pour un afficheur qui permet des opérations sur l'EPROM.

## **Définition des concepts**

L'utilisateur rencontrera dans les différents paragraphes des concepts qui sont propres au produit et donc ne lui sont pas forcément familiers. Ces concepts sont expliqués dans ce paragraphe.

### **Driver d'affichage :**

Les drivers d'affichage sont des blocs bruts. Des blocs de programmes sont produits à partir de ces blocs bruts qui permettent l'affichage de texte. Ils sont spécialement réglés sur le mode de fonctionnement et la configuration matérielle utilisée.

### **Vitesse de transmission (bits par seconde) :**

Vitesse de transmission lors du transfert des données entre les systèmes communicants. Elle indique le nombre de bits qui peuvent être transmis par seconde. Les vitesses de transmission les plus courantes sont : 75, 150, 300, 600, 1200, 2400, 4800, 9600, 19200 bauds.

### **Editeur :**

L'éditeur désigne le programme en arrière-plan qui permet de créer p. ex. des programmes d'automatisation ou de saisir du texte. Un éditeur fournit les fonctions indispensables et est conçu pour remplir parfaitement sa tâche.

**EPROM (erasable programmable read only memory) :**

Mémoire effaçable uniquement par rayons UV et insensible aux pannes. Elle peut être reprogrammée avec un programmeur spécial. Il est possible d'utiliser de tels composants pour certains afficheurs Festo.

**Bus de terrain :**

Système de bus sériel destiné à l'échange d'informations entre des parties éloignées d'un processus ou d'un processus de fabrication. Des capteurs et des actionneurs peuvent être raccordés à ces stations décentralisées du processus.

**Coupleur de bus :**

Permet la liaison entre les abonnés du bus et le système de commande.

**Fenêtre :**

Une fenêtre est une zone de l'écran clairement délimitée dans laquelle sont affichés des remarques ou des messages particuliers.

**Analyse du front :**

Contrôle effectué avant l'affichage d'un texte sans affichage de variables. Les numéros de texte et de blocs sont comparés pour vérifier si le texte à afficher est le même que celui affiché auparavant.

**Configuration :**

Assemblage de différents appareils pour former un système. En cas d'utilisation d'un logiciel, celui-ci doit être accordé (configuré) sur le système pour garantir un transfert de données correct.

**Programme :**

Dans le logiciel FST, un programme est un projet d'automatisation indépendant. Plusieurs programmes peuvent former un projet.

**Projet :**

Dans le logiciel FST, un projet est le regroupement de plusieurs programmes et versions de programmes avec une documentation. Lors de la création d'un nouveau projet, le logiciel FST crée un répertoire. Tous les programmes et fichiers qui font partie de ce projet sont enregistrés dans ce répertoire.

**Interface RS-232 :**

Interface série universelle standard. Permet de connecter un ordinateur ou un automate à des périphériques comme un afficheur de textes.



**RAM (random access memory) :**

Mémoire à accès libre. Les données enregistrées dans une mémoire de ce type peuvent être lues, modifiées et effacées par l'ordinateur. S'il n'est pas prévu de piles de sauvegarde, tout le contenu de la mémoire et donc toutes les données enregistrées sont perdus définitivement en cas de coupure de l'alimentation électrique.

**Analyse du signe :**

Définit le type de représentation ou la plage de valeurs des variables lors de l'affichage. Si l'analyse du signe est activée, la valeur de la variable est comprise entre -32768 et +32767. Si l'analyse du signe n'est pas activée, la valeur de la variable est comprise entre 0 et 65535