

PS/CO/Note 83-16
15.11.1983

Manuel d'utilisation du
Module CAMAC 3IDI 2092

F. Perriollat

Le module CAMAC 3IDI 2092 est une version reconfigurée (hardware et software) du module SEN 3CI 2092 pour les besoins de la connexion "IDI" du PS. Ainsi reconfiguré il réalise, en outre, les fonctions d'un PAD (Packet Assembly Deassembly) pour le protocole IDI-PS.

TABLE DES MATIERES

1. INTRODUCTION
2. DESCRIPTION FONCTIONNELLE SOMMAIRE
3. SPECIFICATION DE L'INTERFACE CAMAC
 - 3.1 Ressources CAMAC du module
 - 3.2 Fonctions CAMAC
 - 3.3 Format du LAM register et du LAM mask register
4. FONCTIONNEMENT DU MODULE
 - 4.1 Généralités
 - 4.2 Communication avec CAMAC
 - 4.2.1 Output Mailbox
 - 4.2.2 Input Mailbox
 - 4.2.3 Status register
 - 4.2.4 LAM register
 - 4.2.5 LAM mask register et LAM CAMAC du module
 - 4.3 Requête générale au module
 - 4.3.1 Requête d'information générale
 - 4.3.1.1 Lecture de l'erreur générale du module
 - 4.3.1.2 Lecture d'une adresse du microprocesseur
 - 4.3.1.3 Lecture du nom de la version software du module
 - 4.3.2 Requête de définition des paramètres de fonctionnement
 - 4.3.2.1 Définition du time-out de lecture de CAMAC input mailbox
 - 4.3.2.2 Libération immédiate de CAMAC input mailbox

- 4.4 Requêtes spécifiques pour un canal
 - 4.4.1 Dépose pour transmission d'un message
 - 4.4.2 Demande de message reçu pour lecture
 - 4.4.3 Demandes d'information
 - 4.4.3.1 Lecture de l'erreur du canal
 - 4.4.3.2 Etat de l'autre extrémité de la ligne du canal
 - 4.4.3.3 Nombre de messages prêts à être lus dans le buffer circulaire de réception
 - 4.4.3.4 Lecture de nombre de caractères du prochain message à lire
 - 4.4.3.5 Lecture du dernier caractère reçu sur la ligne
 - 4.4.3.6 Lecture des indicateurs d'erreurs de ligne
 - 4.4.3.7 Lecture de l'état de la ligne "Data set ready"
 - 4.4.3.8 Lecture du mode de fonctionnement du canal
 - 4.4.4 Définition des conditions de fonctionnement du canal
 - 4.4.4.1 Initialisation du canal
 - 4.4.4.2 Désactivation du canal
 - 4.4.4.3 Définition des paramètres de fonctionnement d'un canal
 - 4.4.4.4 Purge du buffer circulaire de réception
- 4.5 Modes de fonctionnement d'un canal
 - 4.5.1 Mode IDI natif
 - 4.5.2 Mode Emulation IDI par terminal
 - 4.5.3 Mode terminal pur
- 4.6 Etat après Reset du module
- 4.7 Temps d'exécution
 - 4.7.1 Exécution d'un Reset
 - 4.7.2 Charge de base
 - 4.7.3 Réception d'un caractère
 - 4.7.4 Transmission d'un caractère
 - 4.7.5 Demande de status
 - 4.7.6 Setting du module ou des canaux
 - 4.7.7 Transfert d'un message de l'output mailbox dans le buffer de transmission
 - 4.7.8 Mise à disposition d'un message dans l'input mailbox

5. INDICATEURS DU PANNEAU AVANT DU MODULE

5.1 Diodes des canaux

5.2 Diodes du module

ANNEXE 1	Codes d'erreur
ANNEXE 2	Valeur de défaut des paramètres de fonctionnement
ANNEXE 3	Software du module
ANNEXE 4	Connexion des lignes de communication
ANNEXE 5	Protocole de ligne pour le mode de fonctionnement IDI émulation
ANNEXE 6	Protocole de ligne pour le mode de fonctionnement terminal pur
ANNEXE 7	Fonction CAMAC du module
ANNEXE 8	Fomat du LAM Register
ANNEXE 9	Codes des requêtes au module et aux canaux
ANNEXE 10	Jeu de caractères ASCII

1. INTRODUCTION

Le module de SEN 3CI 2092 est construit autour d'un microprocesseur Motorola MC 6802.

Nous avons mis à profit cette facilité pour faire une version entièrement nouvelle du software de ce module pour lui permettre de supporter le protocole de communication des "Intelligent Devices" proposé comme standard pour les contrôles du PS (cf. PS/CO/Note 82-26).

Nous avons adjoint aussi d'autres modes de fonctionnement des canaux de ce module pour lui permettre aussi de :

- faire l'émulation d'une liaison IDI par un terminal simple.
- permettre des liaisons à un terminal interactif ou à une imprimante.

Une large reconfiguration et des modifications hardware ont été nécessaire pour :

- corriger des fautes de conceptions hardware du module original.
- permettre l'utilisation de circuits intégrés mémoire de plus grande capacité.

Le module ainsi modifié est baptisé 3IDI-2092. La note ci-contre décrit ses facilités et son fonctionnement.

La note PS/CO/Note 83-18: Reconfiguration hardware pour réaliser le module CAMAC 3IDI 2092 décrit sommairement la motivation des diverses modifications hardware réalisées au module SEN 3CI-2092, pour le transformer en 3IDI-2092.

La note PS/CO/Note 83-17: Protocole de ligne pour la liaison des Intelligent Devices "IDI-PS protocole", décrit le protocole de ligne pour les liaisons IDI.

Il faut noter qu'une mauvaise conception du registre LAM-status de ce module n'a pas pu être corrigée par des modifications.

Nous proposons donc qu'un module, compatible au point de vue usage, et utilisant un microprocesseur plus performant et plus rapide (M 6809B) soit développé pour l'avenir.

2. DESCRIPTION FONCTIONNELLE SOMMAIRE

Le module fournit 3 portes indépendantes qui peuvent fonctionner en mode RS232/V24 ou boucle de courant (20mA).

Du côté CAMAC ce module dispose de :

- i) 2 boîtes à lettres (une dans le sens CAMAC-dataway → module : sens "output"; l'autre dans le sens module → CAMAC-dataway : sens "input").
- ii) Un registre de LAM de 16 bits, avec un registre masque (mask register) de 16 bits.
- iii) Un registre d'état (status register) de 8 bits.

Le fonctionnement de ce module est déterminé par la programmation du microprocesseur Motorola MC 6802.

La programmation fait que chaque porte RS 232 fonctionne de façon indépendante. Chaque canal associé à une porte supporte le protocole de ligne pour le mode de fonctionnement choisi. Il dispose de mémoires tampons (buffers) propres, une pour la transmission, une pour la réception, à l'aide desquelles se fait l'assemblage et le désassemblage des messages échangés sur la ligne.

Trois modes de fonctionnement sont implémentés pour chacun des canaux:

- i) Mode IDI ou IDI-natif : connexion opérationnel d'un ID.
- ii) Mode d'émulation d'IDI par un terminal. Ce mode permet de faire des tests en émulant par un terminal l'autre bout d'une ligne de connexion d'ID.
- iii) Mode terminal : ce mode permet disposer par exemple d'un terminal comme outil d'interaction à travers CAMAC.

Dans la version 2 du software un mode de "trace d'un autre canal" sera implémenté pour permettre de disposer d'un outil de surveillance du trafic sur une ligne de connexion d'un ID.

Le fonctionnement de ce module a été conçu de sorte que la connexion de 2 systèmes CAMAC puisse se faire aux deux bouts par celui-ci. Dans ce cas le protocole de ligne, et l'assemblage/désassemblage des messages sont complètement transparents pour les utilisateurs. De plus, dans une telle connexion les deux extrémités ont un fonctionnement parfaitement symétrique, ce qui veut dire qu'aucune relation maître/esclave n'est imposée ni par le module, ni par le protocole.

3. SPECIFICATION DE L'INTERFACE CAMAC

Les fonctions CAMAC du module sont décrites ci-contre. Le fonctionnement détaillé des diverses actions est décrite dans le chapitre 4.

3.1 Ressources CAMAC du module

L'interface du module avec le CAMAC dataway se fait au moyen de ses 4 ressources;

- i) Output mailbox
- ii) Input mailbox
- iii) Status register
- iv) LAM status register

De plus, le module génère un LAM; qui est le résultat du filtrage du LAM status register par le LAM mask register.

3.2 Fonctions CAMAC

F0 A0	lecture d'un byte (dans R1...R8) de l'Input Mailbox. Si Q = 0, il n'y a pas, ou il n'y a plus, de byte dans l'Input Mailbox.
F1 A0	lecture du status register (dans R1...R8) Le contenu du status register dépend de la requête déposée. Q = 1
F1 A12	lecture du LAM register (dans R1....R16), voir § 3.3 pour le format du LAM register Q = 1
F1 A14	lecture des LAM source après masquage, voir § 3.3. Q = 1
F8 A0	test de l'état du LAM Q = 1 si le LAM est actif
F9 A0	Reset complet du module. Même effet que Z ou que la mise sous tension. Q = 0
F16 A0	dépose en séquence d'un caractère dans l'Output Mailbox pour le canal 0 réponse Q, voir ci-dessous
F16 A1	idem F16 A0 pour canal 1
F16 A2	idem F16 A0 pour canal 2
F16 A3	demande de mise à disposition du prochain message pour le canal spécifié par R1...R8 (0, 1 ou 2) réponse Q, voir ci-dessous

- F16 A4 dépose en séquence d'un paramètre pour une requête au canal 0
réponse Q, voir ci-dessous
- F16 A5 idem F16 A4 pour canal 1
- F16 A6 idem F16 A4 pour canal 2
- F16 A7 dépose en séquence d'un paramètre pour une requête globale au module

Etat de la réponse Q pour toutes les fonctions F16

Q = 1 si les données de R1...R8 ont bien été déposées dans l'Output Mailbox.

Q = 0 si l'output Mailbox n'est pas disponible ou encore si elle est pleine (capacité 255 bytes). Le byte n'a pas été déposé.

- F17 A13 écriture du Lam mask register (cf. § 3.3 pour le format)
1 pour l'un des bits R1...R16 fait que le LAM correspondant est masqué. Il ne peut donc pas généré de LAM request.
Q = 1.
- F23 A12 Effacement sélectif du LAM register. Les LAM correspondant aux lignes W1...W16 à 1 sont effacés (0 logique).
Q = 1.
- F25 A0 Fin d'une séquence de fonctions F16. Demande au module l'exécution de l'action requise.
Q = 0.

Z ou mise sous tension: Reset complet du module. Même action que F9 A0.

3.3 Format du LAM register et du LAM mask register

Les 16 bits du LAM register sont partitionnés en 4 zones de 4 bits. 3 zones de 4 bits pour les 3 canaux, une zone de 4 bits pour le module dans son ensemble. Le format est le suivant :

- R1 = 1 : canal 0 buffer de transmission disponible
- R2 = 1 : canal 0 message(s) disponible(s) dans le buffer de réception
- R3 = 1 : canal 0 indications d'erreur(s)
- R4 = 1 : canal 0 actif

R5 idem R1 pour canal 1
R6 idem R2 pour canal 1
R7 idem R3 pour canal 1
R8 idem R4 pour canal 1
R9 idem R1 pour canal 2
R10 idem R2 pour canal 2
R11 idem R3 pour canal 2
R12 idem R4 pour canal 2

R13 = 1 : indicateur d'erreur(s) générale(s) du module

R14 = 1 : données disponibles dans le status register

R15 = 1 : message disponible dans input mailbox

R16 = 1 : output mailbox disponible

Le LAM mask register a exactement le même format. Toute source de LAM dont le bit correspondant est à 1 dans le LAM mask register est bloqué, et donc ne peut pas générer de LAM request.

La lecture des LAM après masquage (F1 A14) retourne donc le résultat de l'opération logique

$$(LAM\ register)\ AND\ \overline{(LAM\ mask)}$$

Les conditions d'apparition et disparition des différents bits de LAM sont décrites au chapitre 4.

4. FONCTIONNEMENT DU MODULE

4.1 Généralités

Le fonctionnement de chacun des canaux du module est déterminé par

- l'état du canal (actif ou passif).
- le mode de fonctionnement choisi (connexion IDI, émulation IDI par terminal, terminal pur).

Le module dans son ensemble fonctionne comme un esclave du CAMAC (donc en fait du calculateur qui contrôle le CAMAC).

Le fonctionnement se fait toujours sous la forme demande du calculateur, réponse du module.

Le module signale au calculateur ses changements d'états par les bits du LAM register.

Toute demande se fait par dépose des paramètres de la requête dans l'output Mailbox (série de F16, A0...7 terminée par F25 A0). A l'action requise le module peut répondre par une indication d'erreur (pour le module en entier, ou pour le canal considéré). Si des données sont retournées elles peuvent être retournées dans l'Input Mailbox pour les transferts de messages vers le calculateur ou dans le status register.

La disponibilité de l'output Mailbox est représentée par l'état du bit R16 du LAM status register. Rien n'est effectivement déposé par F16 dans l'Output Mailbox quand cette boîte à lettres n'est pas disponible pour le CAMAC (réponse Q = 0 à F16).

La seule requête qui ne suit pas le chemin décrit ci-dessus est le reset du module (F9 A0 ou Z) qui est toujours et immédiatement exécuté.

4.2 Communication avec CAMAC

4.2.1 Output Mailbox

Comme décrit ci-dessus (4.1) cette ressource est utilisée pour toute requête au module (sauf pour le Reset général).

Pour pouvoir déposer des données dans cette boîte à lettre, il faut qu'elle soit disponible pour le CAMAC. Cet état est signalé par le bit R16 du LAM status register égal à 1 (voir § 3.3).

La dépose des données se fait par une série de fonction F16. La valeur de la sous-adresse (A0...A7) détermine le type de requête. Une réponse Q = 1 à une fonction F16 indique que la donnée a correctement été déposée dans la boîte à lettre. Une réponse Q = 0 signifie que la donnée n'a pas été déposée parce que :

- soit la boîte à lettre n'est pas disponible pour le CAMAC (R16 de LAM register = 0)
- soit la boîte à lettre est déjà pleine.

La capacité est de 255 bytes. Dans ce cas il y a eu une demande implicite d'exécution de la requête.

Il faut noter qu'aucune des requêtes au module ne doit entraîner un dépassement de la capacité de la boîte à lettres.

Lorsque toutes les données d'une requête sont déposées dans la boîte à lettres, la demande de prise en compte et d'exécution se fait par la fonction F25 A0 (voir § 3.3). Dès cet instant, la boîte à lettres n'est plus disponible pour le CAMAC jusqu'à ce que le module ait pris en compte toutes les données de la requête.

Il est interdit de mélanger divers types de requête (valeurs de A différentes). Une telle erreur n'est signalée que lors de l'exécution par le module de la requête. Le module signale cette erreur par le code 7 (voir Annexe 1). Cette erreur est signalée selon la valeur de A associé à la 1ère donnée soit à un canal (A = 0, 1, 2, 4, 5 ou 6) soit au module (A = 7).

La dépose des données d'une requête dans la boîte à lettres peut se faire à la vitesse maximum du CAMAC, donc en particulier par des transferts DMA entre le calculateur et le CAMAC.

4.2.2 Input Mailbox

Cette ressource est utilisée par le module pour transmettre un message (chaîne de caractères) depuis le buffer de réception d'un canal au CAMAC. Cette transmission est le résultat d'une requête déposée par F16 A3.

La disponibilité pour le CAMAC du message dans cette boîte à lettres est signalée par une valeur 1 du bit R15 du LAM register (voir § 3.3).

Le 1er byte d'un message, qui est lu par la 1ère fonction F0 A0, donne le nombre de caractères dans le message (qui peut être 0 pour un message vide).

Pour vider cette boîte à lettres il faut faire n+1 fonction de lecture F0 A0. Le module surveille par un time-out la vidage de cette boîte à lettre (voir 4.3.2.1 pour définir la valeur de ce time-out). Dans le cas où ce time-out vient à être dépassé, le module met une erreur 10 dans le canal auquel appartient le message et libère la boîte à lettres. Le message qui n'a pas été lu par le CAMAC reste dans le buffer de réception du canal. Il peut donc être lu ultérieurement.

La lecture de cette boîte à lettres (F0 A0) peut se faire à la vitesse maximum du CAMAC, donc en particulier par des transferts DMA entre le calculateur et le CAMAC.

4.2.3 Status register

Ce registre de 1 byte est utilisé par le module pour répondre à des demandes d'information. La disponibilité des données dans ce registre est signalée par une valeur 1 du bit R14 du LAM register.

La signification des données dans ce registre dépend des requêtes au module ou à un canal (voir § 4.3 et 4.4).

Après une initialisation du module ce registre contient l'identification de version software du module (voir § 4.6).

4.2.4 LAM register

Le format de ce registre de 16 bits est donné au § 3.3. Nous décrivons ci-dessous les conditions qui font apparaître et disparaître les divers bits de ce registre. Les diagrammes de temps, pour certains de ces événements, sont donnés au § 4.7. Le cas particulier du Reset général de module est décrit au § 4.6

a) LAM pour les états globaux du module

R16 : définit la disponibilité pour CAMAC de l'output Mailbox

(1 = disponible pour les écritures par F16;
0 = non disponible)

* Mis à zéro dès l'exécution de F25 A0

* Remis à 1 dès que le module a traité toutes les données déposées dans la boîte à lettre.

Il faut noter que des erreurs détectées au moment de l'exécution d'une requête font que le bit d'erreur correspondant (R13, ou R11, ou R7, ou R3) est mis à 1 avant que R16 ne soit mis à 1.

R15 : Etat de l'Input Mailbox

(1 = message disponible pour être lu par le CAMAC avec F0 A0)

(0 = boîte à lettres vide; pas de lecture possible: Q = 0 pour F0 A0)

* Mis à 1 dès que le module a déposé un message dans cette boîte à lettres et que l'accès par le CAMAC est autorisé

* Remis à zéro dès que cette boîte à lettres n'est plus disponible pour la lecture par CAMAC, à savoir :

- dès la lecture du dernier caractère du message, ou la lecture du nombre de caractères pour un message vide;
- par le dépassement du temps alloué pour lire par le CAMAC cette boîte à lettres;
- par une requête de désactivation d'un canal pour lequel un message est disponible dans la boîte à lettres d'entrée;
- par une requête de libération explicite de cette boîte à lettres.

R14 : Etat des données dans le status register

(1 = données valides dans le status register
0 = données non valides pour la requête en cours d'exécution)

* Mis à 1 dès que les données correspondantes à la requête sont disponibles dans ce registre

* Mis à 0 dès l'exécution d'une requête d'état à un canal (F16 A4, A5 ou A6) ou au module (F16 A7).

Il faut noter que R14 est mis à 1 avant que R16 ne le soit, dans le cas d'une demande d'information (F16, A4, 5, 6 ou 7).

R13 : Etat d'erreur globale du module

(1 = erreur globale du module non encore acquitée
0 = pas d'erreur globale en suspens)

- * Mis à 1 dès la détection par le module d'une erreur globale au module.
- * Remis à 0 par la lecture de l'erreur du canal

b) LAM spécifiques pour un canal

Le jeu des 4 LAMs d'un canal a un fonctionnement strictement identique pour les 3 canaux. Seuls les temps de certains phénomènes peuvent varier de quelques micro-secondes d'un canal à l'autre par le fait du traitement séquentiel des canaux par le software du MC 6802.

R4, R8, R12 : Etat du canal

(1 : canal actif
0 : canal passif ou en mode de trace)

- * Mis à 1 à la fin de l'exécution d'une requête d'initialisation du canal.
- * Mis à 0 à la fin de l'exécution d'une requête de désactivation du canal.

R3, R7, R11 : Etat d'erreur du canal

(1 : erreur pour le canal non encore acquitté
0 : pas d'erreur en suspens pour le canal)

- * Mis à 1 dès la détection par le module d'une erreur spécifique de ce canal.
- * Remis à 0 par la lecture de l'erreur du canal.

R2, R6, R10 : Etat des messages en réception

(1 = s'il y a au moins un message disponible dans le buffer circulaire de réception, sans qu'il y ait de message dans la boîte à lettres d'entrée pour le canal considéré.

0 = pas de message disponible dans le buffer de réception, ou un message pour le canal considéré est disponible dans la boîte à lettres d'entrée (R15=1)).

- * Mis à 1
 - dès qu'un message en réception est complet, (réception de CR) sans qu'il y ait de message pour ce canal dans la boîte à lettres d'entrée.
 - dès qu'un message, pour le canal considéré, disparaît de la boîte à lettres d'entrée (soit par fin de lecture, time out, nettoyage de cette boîte à lettres, ou demande d'initialisation de la voie de réception par le protocole de ligne) et qu'il y a encore des messages disponibles dans le buffer circulaire de réception du canal considéré.

* Mis à 0

- dès qu'un message pour le canal considéré est déposé dans boîte à lettres d'entrée,
- par la requête de nettoyage du buffer de réception de ce canal (en mode terminal-pure) cf. § 4.4.4.4.
- par une demande d'initialisation de la voie de réception par le protocole de ligne (en mode IDI-natif ou IDI-émulation)
- par une requête de désactivation du canal.

R1, R5, R9 : Etat du buffer de transmission du canal

- (1 = buffer de transmission libre pour un nouveau message
- 0 = buffer de transmission utilisé par un message en cours de transmission).

* Mis à 0

- dès que le module a fini le transfert d'un message depuis la boîte à lettres de sortie dans le buffer de transmission, ou bien par la désactivation du canal.
- Par une requête de désactivation du canal.

* Mis à 1 par la fin de la transmission du message ou par l'activation du canal.

c) Etat après l'exécution d'un reset global du module

Pendant l'exécution d'un reset général du module (F9 A0, Z ou power-up) le LAM register est remis à zéro.

A la fin de la procédure d'initialisation du module, le LAM register vient dans l'état : R16 = 1 (output mailbox disponible) et R14 = 1 (data disponible dans le status register + identificateur de version software). Tous les autres bits sont à zéro.

4.2.5 LAM mask register et LAM CAMAC du module

La manipulation du LAM mask n'est possible que par la fonction CAMAC: d'écriture : F17 A12. Le module n'a aucun moyen pour manipuler ce registre, ni même de le lire.

Chaque bit écrit à 1 dans ce registre masque le bit correspondant pour la génération du LAM CAMAC du module, ainsi que pour la lecture des LAM après le mask (F1 A14).

Le module active la ligne CAMAC LAM si, après filtrage par le LAM mask register, le résultat est non zéro.

Les fonctions booléennes sont :

LAM after mask = (LAM register) and ($\overline{\text{LAM mask}}$)

Module LAM CAMAC actif si (LAM register) and ($\overline{\text{LAM mask}}$) ≠ 0.

4.3 Requêtes générales au module

Ce sont les requêtes qui sont déposées par la fonction F16 A7 dans la boîte à lettres de transmission. Ces requêtes se divisent en deux classes :

- i) demande d'information
- ii) requête d'action ou de définition de paramètres

Le format général de ces requêtes est la suivante :

	W8	W1	Fonction CAMAC
1er byte	code de la requête		F16 A7
2ème byte	n paramètres (suivant le cas)		.
			.
			.
			.
n + 1 byte	fin de la requête et exécution		F16 A7 F25 A0

Pour tous les paramètres la règle suivante s'applique :

i) Une valeur négative (W8 = 1 dans F16 A7) entraîne la non modification de la valeur courante de ce paramètre.

ii) Une valeur nulle (W1...W8 = 0). Le paramètre correspondant est mis à sa valeur de défaut (cf. Annexe 1).

iii) Une valeur positive (W8 = 0) définit une nouvelle valeur. Les valeurs autorisées dépendent de la requête et du paramètre considérés:

4.3.1 Requête d'information générale

4.3.1.1 Lecture de l'erreur générale du module

L'apparition d'une telle erreur est signalée par R13 = 1 dans le LAM register.

Format :

code 0
pas de paramètre

Action : - retourne dans status register le code d'erreur (0 s'il n'y avait pas d'erreur)
- efface cette erreur et le LAM correspondant.

4.3.1.2 Lecture d'une adresse du microprocesseur

L'objet de cette commande est de disposer de moyen pour réaliser la mise au point du software du module. Elle ne devrait pas être utilisée de façon opérationnelle.

Format :

code : 0
paramètres : byte de haut poids de l'adresse à lire
byte de bas poids de l'adresse à lire

Action : retourne dans le status register le contenu de l'adresse désignée.

Note : 2 cas de fonctionnement sont donnés suivant le contenu de l'EPR0M. Un mode opérationnel où il n'y a pas d'accès aux ACIA, VIA et CAMAC Mailbox, un mode de debugging où il y a l'accès total. Dans ce dernier cas certains accès peuvent détruire les automates de fonctionnement software. Donc pour des modules en opération il faut s'assurer qu'ils soient de type fonctionnement opérationnel (cf. listing étiquette DEBUGM).

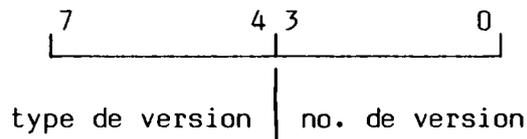
4.3.1.3 Lecture du nom de la version software du module

Format :

code : 2
pas de paramètres

Action : retourne dans le status register l'identificateur de la version software du module.

Le codage de cette version est le suivante :



Le type de version (0 à 15) détermine le type de module hardware et les facilités incluses.

Le numéro de version définit le niveau software pour un type donné.

Le type pour le module SEN 3CI-2092 modifié en 3IDI-2092 est 0.

4.3.2 Requête de définition de paramètres de fonctionnement

4.3.2.1 Définition du time-out de lecture de CAMAC-input-mailbox

Ce paramètre définit le temps maximum alloué pour lire par le CAMAC le message déposé dans cette boîte à lettres.

Format :

code : 255
paramètres : valeur du time-out

Valeur du paramètre:

- 1 (ou négatif) : pas de modification de la valeur courante
- 0 : mettre la valeur de défaut (voir annexe 1)
- 2....127 : nouvelle valeur (en secondes).

Toute autre valeur donne une erreur 16 pour le module.

Note de fonctionnement : le temps alloué est défini à une seconde près. Il est compris entre la valeur du paramètre moins 1 et la valeur définie.

4.3.2.2 Libération immédiate de CAMAC input mailbox

L'objet de cette requête est de rendre disponible la boîte à lettres, sans lecture du message qui y est déposé pour permettre de demander un message d'un canal qui doit être acquis en priorité.

Format :

code 254
pas de paramètre

Action : i) aucun effet si la boîte aux lettres est déjà libre

ii) si la boîte aux lettres est occupée par un message d'un canal (même si ce message est partiellement lu): vide cette boîte à lettres, rend cette boîte à lettres non lisible par CAMAC (signalé par bit R15 LAM status, et Q = 0 pour FO A0, signale ce phénomène par une erreur 23 pour le canal dont le message a été nettoyé.

4.4 Requêtes spécifiques pour un canal

Ce sont les requêtes qui sont déposées par

F16 A0, A1, A2	pour transmission d'un message pour les canaux 0, 1, 2
F16 A3	pour demande de transfert dans CAMAC input mailbox d'un message d'un canal
F16 A4, A5, A6	pour lecture ou définition de paramètres de fonctionnement pour les canaux 0, 1 ou 2.

La même règle générale que pour les paramètres généraux du module s'applique pour les paramètres de fonctionnement spécifiques à un canal (cf. § 4.3) à savoir :

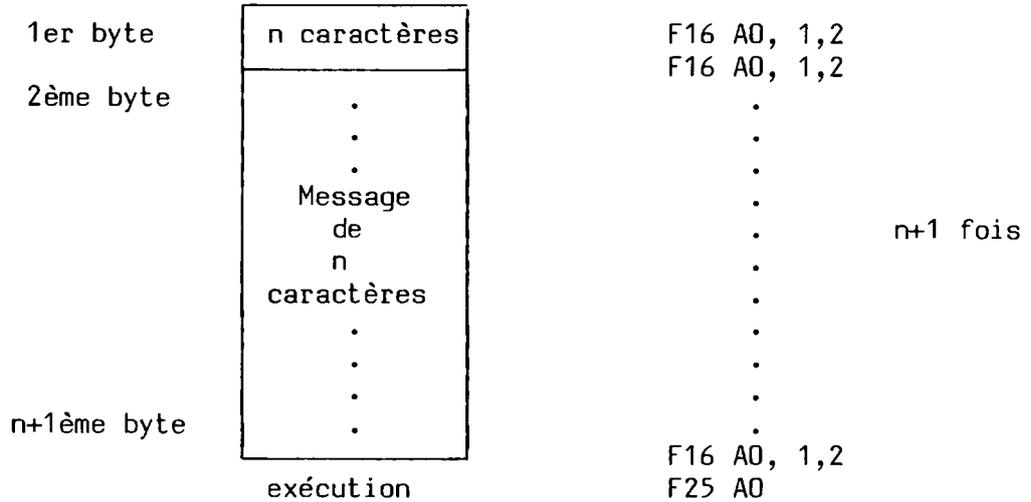
i) toute valeur négative (R8 = 1) n'entraîne aucune modification de la valeur courante de ce paramètre.

ii) toute valeur nulle (R1....R8 = 0) entraîne l'utilisation de la valeur de défaut de ce paramètre.

iii) toute valeur positive (strictement) définit une nouvelle valeur du paramètre.

4.4.1 Dépose pour transmission d'un message

Format :



Note : - les classes de caractères qui peuvent composer un message dépendent du mode de fonctionnement du canal.

- la signification du 1er byte est légèrement modifiée pour le mode terminal pur

Action :

transporte le message depuis la boîte à lettres dans le buffer de transmission du canal.

Pour que cette action puisse être entreprise il faut que:

- i) le canal soit actif
- ii) le buffer de transmission soit libre.

Si non, les erreurs 12 ou 2 sont respectivement mises pour le canal.

Tout message plus grand que la taille maximum autorisée (80 caractères) est coupé au maximum, et l'erreur 3 est mise pour le canal.

Rappel : contenu possible d'un message :
Le contenu dépend du mode de fonctionnement.

i) Mode IDI-natif ou IDI-émulation par terminal

Un message ne doit contenir que des caractères ASCII imprimables (alphabet étendu) (code octal 40 à 177 inclus). Le message peut être terminé par un "carriage-return" (code octal 15) ou une paire "carriage return, line feed" (codes octal 15, 12). Ces 2 caractères de contrôle ne peuvent en aucun cas se trouver dans le corps du message. Aucun autre caractère de contrôle (code octal 0 à 37 inclus) n'est admis où que ce soit dans un message.

ii) Mode terminal pur

Tout caractère ASCII, de contrôle ou imprimable, (code 0 à 177) est admis dans le message et transmis sans altération d'ordre ou de code.

Le 1er byte de la boîte à lettres détermine la taille maximum du message. Donc la taille effective du message est déterminée par l'indicateur d'exécution (F25 A0) ou par la taille maximum, si l'indicateur d'exécution n'a pas été trouvé avant la lecture de n caractères dans la boîte à lettres.

Aucun test logique n'est exécuté sur la structure du message.

Il faut noter toutefois que le protocole de ligne utilise des caractères de contrôle (bell, back space, DC1, DC3) ce qui pourrait éventuellement interférer avec certains caractères de contrôle du corps de message.

Il faut rappeler que le nombre maximum de caractères d'un message ne peut pas être plus grand que 80.

4.4.2 Demande de message reçu pour lecture

Format :

1er byte	<table border="1"><tr><td>no de canal</td></tr></table>	no de canal	F16 A4
no de canal			
	excécution	F25 A0	

Action: Si, pour le canal désigné, un message au moins est disponible dans le buffer circulaire de réception, transfère le 1er message dans la boîte à lettres de lecture et autorise la lecture de celle-ci par CAMAC.

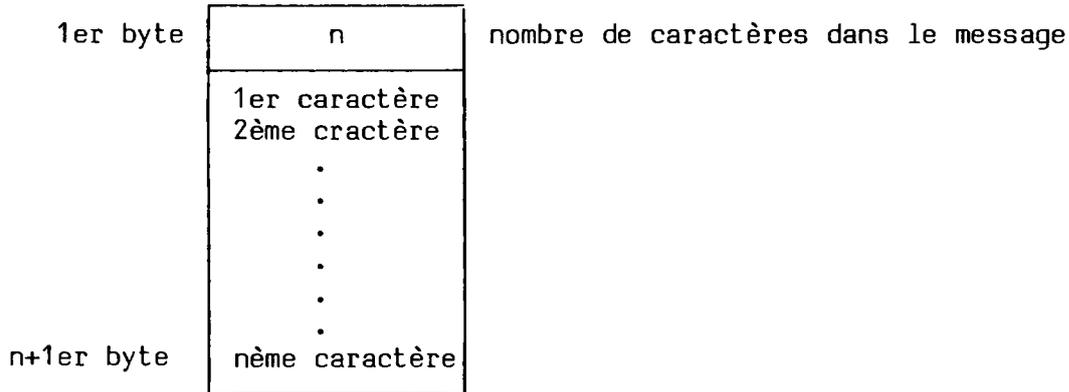
Pour que cette action puisse s'accomplir il faut que :

- le canal soit actif (si non erreur 12)
- au moins un message soit disponible (si non erreur 8)
- la boîte à lettre de lecture soit disponible (si non erreur 9)

Les erreurs éventuelles qui ont été détectées pendant la réception du message sont aussi signalées par des erreurs pour le canal pendant l'exécution de cette requête. Dans ce cas le code d'erreur transmis est égal à 200(10) (= erreur différée) plus le code d'erreur natif. Il peut s'agir de :

- dépassement de capacité du message (code d'erreur 200+1)
- erreur de ligne lors de la réception d'un caractère du message (code 200+22). Cette erreur a pu être une erreur de parité, de structure ou encore une surécriture. Elle a déjà été signalée au moment de son apparition.

Le format du message déposé dans la boîte à lettres est :



La lecture du message doit se faire par n+1 fonctions FO A0 (n = donnée lue au premier FO A0).

Lorsque le message est complètement lu (n+1 FO A0 avant le temps maximum alloué) la boîte aux lettres est libérée, et le message est enlevé du buffer circulaire de réception.

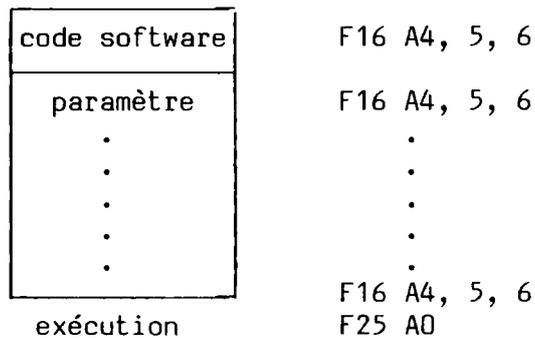
Si le message n'a pas été complètement lu à cause d'un dépassement de temps (code d'erreur 10) ou par une demande de libération immédiate de la boîte aux lettres (code d'erreur 23) le message reste disponible dans le buffer circulaire de réception. Il peut être relu par une demande de lecture ultérieure.

Un message ne comporte jamais les caractères de fin de message qui transitent sur la ligne. Il n'y a donc pas de "carriage return" ni de "line feed" en fin de message.

4.4.3 Demandes d'information

Ces demandes sont déposées par les fonctions F16 A4, A5 et A6 respectivement pour les canaux 0, 1, 2.

le format général d'une telle demande est le suivant :



4.4.3.1 Lecture de l'erreur du canal

Format :

code 0
pas de paramètre

Action : Retourne dans le status register le code d'erreur du canal (0 s'il n'y avait pas d'erreur) et efface l'indicateur d'erreur pour ce canal dans le registre d'état des LAMs)

Note : une seule erreur est mémorisée pour un canal. C'est la dernière qui a été détectée.

4.4.3.2 Etat de l'autre extrémité de la ligne du canal

Cette requête permet de connaître quel est l'état supposé par le module pour l'élément connecté à l'autre extrémité de la ligne du canal.

Format :

code 1
pas de paramètres

Action : retourne dans status register : 0 si passif, 1 si actif.

Notes :

- Si le canal n'est pas actif, l'état de l'autre bout de la ligne est toujours supposé non actif.
- Pour les modes terminaux, dès que le canal est activé, l'autre bout de la ligne est supposé être actif.
- Pour un mode IDI natif, la déclaration de l'état actif est faite à la réception du premier caractère DC1 (control Q) en réponse à la demande d'initialisation (control G) de la voie d'entrée de l'autre bout de la ligne.

4.4.3.3 Nombre de messages prêts à être lus dans le buffer circulaire réception

Format :

code 2
pas de paramètre

Action : retourne dans le status register le nombre de messages prêts à être lus et disponibles dans le buffer circulaire de réception.

Note : Tout message déposé dans la boîte à lettres de réception, même partiellement lu, est compté parmi les messages disponibles. C'est la lecture complète du message par CAMAC (FO A0) qui fait disparaître ce message (cf. 4.4.2).

Action: retourne dans status register l'état du signal DSR du canal (pine 6 du connecteur Cannon 25P)

Format: valeur = 1 : DSR vrai
valeur = 0 : DSR faux

Notes : Dans le cas où la pine 6 du connecteur Cannon 25P n'est pas connecté DSR est toujours vrai.

Quand DSR est faux toute transmission de caractères est suspendue.

4.4.3.8 Lecture du mode de fonctionnement du canal

Format :
code 7
pas de paramètre

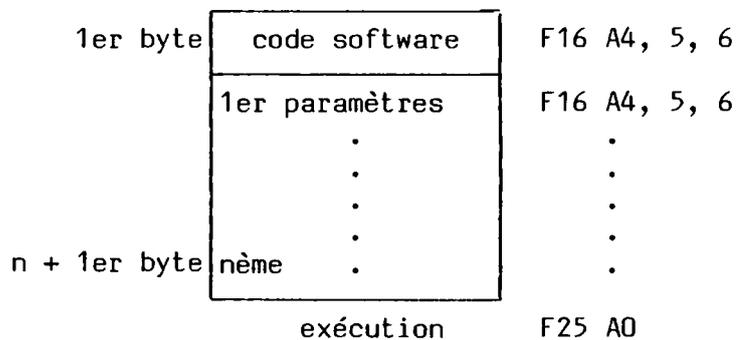
Action: Retourne dans le status register le mode de fonctionnement du canal dans le même format que celui utilisé pour définir le mode de fonctionnement (cf. § 4.4.4.2) soit

- 0 = mode IDI natif
- 1 = IDI émulation par terminal
- 2 = terminal pur

4.4.4 Définition des conditions de fonctionnement du canal

Ces demandes sont déposées par les fonctions F16 A4, A5, A6 respectivement pour les canaux 0, 1, 2.

Le format général d'une telle demande est le suivant :



La conversion pour les paramètres reste la même que pour le cas du module en entier, à savoir :

valeur d'un paramètre	action
-1 ou toute valeur < 0	pas de modification de la valeur courante
0	mettre la valeur de défaut
>0	nouvelle valeur du paramètre. Les valeurs acceptées dépendent du cas.

4.4.4.1 Initialisation du canal

Format:

code 255
pas de paramètre

Action: rend le canal actif dans un état déterminé par la valeur courante des paramètres du canal.

Si le canal est déjà actif retourne l'erreur 13 pour le canal.

4.4.4.2 Désactivation du canal

Format:

code 254
pas de paramètres

Action: arrête l'activité du canal. Tout message en cours de transmission continue à être envoyé. Tous les messages reçus ou en cours de réception sont complètement effacés.

4.4.4.3 Définition des paramètres de fonctionnement d'un canal

Format:

code 253
paramètres : - Vitesse de transmission
- Stop-bit
- Contrôle de la parité
- Time-out de transmission
- Mode de fonctionnement.

Action: si le canal est non actif, définit les conditions de fonctionnement du canal suivant les valeurs des paramètres. Si le canal est actif, aucune action n'est entreprise et le code d'erreur 13 est retourné au canal.

Valeur des paramètres :

i) vitesse de transmission

1	:	50	bauds
2	:	75	"
3	:	110	"
4	:	135	"
5	:	150	"
6	:	300	"
7	:	600	"
8	:	1200	"
9	:	1800	"
10	:	2400	"
11	:	3600	"
12	:	4800	"
13	:	7200	"
14	:	9600	"
15	:	19200	"

ii) Stop bits

1	:	1	stop bit
2	:	2	stop bits

iii) Contrôle de la parité

1	:	parité impaire en réception et transmission
2	:	parité paire en réception et transmission
3	:	"mark" transmis, pas de contrôle en réception
4	:	"space" transmis, pas de contrôle en réception

iv) Time out de transmission

2...127 valeur du temps maximum en seconde
- pour le cas terminal pur ou IDI émulation:
127 = time-out infini

v) Mode de fonctionnement

0	=	mode IDI natif (ou normal) (défaut)
1	=	IDI émulation par terminal
2	=	terminal pur

Notes : i) Le temps effectif maximum du time-out de transmission est déterminé à une seconde près. Ce temps est compris entre une seconde de moins que le paramètre et le paramètre. Ce fait explique que la valeur 1, qui donne en fait un temps entre 0 et 1 seconde, ne soit pas admis.

ii) Le nombre de bits dans un caractère est toujours 7. Ce nombre n'est pas programmable.

4.4.4.4 Purge du buffer circulaire de réception

N'est autorisé que pour le mode terminal pur

Format :

code 252
pas de paramètre

Action : efface tout message du buffer circulaire de réception, que ces messages soient complets ou en cours de réception.

Met à zéro le code de dernier caractère reçu.

Si le mode de fonctionnement n'est pas terminal pur, l'erreur 17 est retournée.

4.5 Modes de fonctionnement d'un canal

Un canal peut fonctionner dans l'un des 3 modes de fonctionnement prévu suivant la valeur du paramètre associé.

Ces 3 modes de fonctionnement sont :

- Mode IDI natif
- Mode IDI en émulation par un terminal
- Mode terminal pur

4.5.1 Mode IDI natif

Encore appelé mode IDI normal.

Dans ce mode de fonctionnement le canal support complètement le protocole de ligne IDI-PS (ref. PS/CO/Note 83-17 "Protocole de ligne IDI-PS").

Nous rappelons que dans ce mode la ligne fonctionne en mode "Start Stop"; donc elle ne nécessite strictement que les 2 paires de fils transmit/receive. Le standard électrique peut être soit current loop (20mA) soit RS232-C (\equiv V24).

4.5.2 Mode Emulation IDI par terminal

Ce mode est fait pour émuler l'autre bout d'une connexion IDI par un simple terminal (jeu de caractère ASCII). Ceci doit permettre :

i) de développer et de tester le software des 2 bouts d'une connexion IDI sans que ces 2 extrémités soient connectées. (Il faut toutefois que les messages à échanger soient définis).

ii) de simuler facilement les cas d'exception qui peuvent se produire.

Protocole de ligne IDI-émulation

C'est le protocole de ligne IDI modifié pour supporter les particularités d'un terminal: à savoir :

- il est toujours prêt (il n'y a aucun traitement dans le terminal du message reçu autre que son affichage immédiat).
- il permet d'éditer de façon interactive les messages à envoyer.

Le protocole IDI est modifié comme suit:

i) il n'y a pas de suspension implicite de la transmission en fin de message transmis. Ceci veut dire qu'il n'est pas nécessaire de répondre par "Control Q" (= XON) à la réception d'une fin de message.

ii) une demande d'initialisation de la voie de réception du terminal (réception du caractère Control G (= Bell)) ne demande pas de réponse par "Control Q" (= XON).

Il faut noter que Control S (=XOFF) peut à tout moment être utilisé pour suspendre la transmission de message afin de pouvoir examiner la situation. Toutefois cette suspension n'arrête pas le décomptage du temps pour l'envoi par le module des messages. Il est donc souhaitable dans bien des cas d'adapter le paramètre de time-out d'envoi pour permettre cette examination sans difficulté.

Nous rappelons que dans ce mode de fonctionnement une valeur de 127(10) du time-out de transmission donne un temps infini. Cette valeur donne donc toute liberté d'utiliser Control S pour examiner la situation. La reprise de transmission se fait par Control Q (= XON) ou par la frappe de tout caractère imprimable (=XON implicite comme dans SINTRAN III).

En transmission depuis le terminal au module c'est la frappe de la touche "carriage return" qui termine le message en cours de transmission.

Facilité d'édition

Pour permettre l'émulation par le terminal des messages envoyés au module un minimum de facilités interactives d'édition sont fournies à savoir :

- i) écho sur le terminal de tout caractère imprimable frappé.
- ii) une fin de message (clef carriage return) produit en écho un retour en début de ligne et un saut à la ligne suivante.
- iii) le caractère "Back space" permet d'effacer du message et de l'écran le dernier caractère du message en cours de transmission.

Il faut noter qu'un message ne comporte jamais de caractère de contrôle (= caractère non imprimable de code 0 à 37 octal). Les seuls caractères de contrôle reconnus sont :

- carriage return (CR code = 15 octal) : fin de message
- Control G (Bell, code = 7 octal) : demande d'initialisation de la voie d'entrée du canal
- Control Q (XON, code 21 octal) = continuer la transmission
- Control S (XOFF, code 23 octal) = suspendre la transmission
- Back space (BS, code 10 octal) = effacer le dernier caractère du message en cours de transmission.

4.5.3 Mode terminal pur

Ce mode de fonctionnement du canal est fait pour permettre de disposer d'un moyen de connexion de terminal pour des programmes interactifs ou encore de disposer d'imprimantes locales.

Ce mode se caractérise par la transmission du module au terminal de de tous caractères du message sans aucun filtrage des caractères de contrôles.

Depuis le clavier vers le module le fonctionnement est semblable au mode émulation IDI. A savoir que:

- la fin d'un message en cours de transmission est signifiée par "carriage return".
- Seul les caractères imprimables sont inclus dans le message en cours de transmission qui a une taille maximum de 80 caractères.
- Le caractère "back space" efface le dernier caractère du message en cours de transmission et de l'écran.
- L'écho de tout action concernant le message en cours de transmission depuis le terminal se fait sur l'écran.

Par le fait que seul les caractères imprimables peuvent être inclus dans un message et que leur taille est limitée à 80 caractères, ceux-ci se trouvent pratiquement limités à une ligne de l'écran.

Du module vers le terminal aucun test sur la structure du message déposé dans la boîte à lettre de transmission n'est fait. Seul le nombre de caractères (1er byte dans la boîte à lettres ou marque d'exécution F25 A0) détermine le nombre de caractères à transmettre (maximum 80 caractères par message).

Depuis le terminal (ou une imprimante), il est possible de contrôler le flot des caractères par le mécanisme XON-XOFF (Control Q, Control S). Mais il faut noter que la surveillance de temps n'est pas suspendue quand la transmission vers le terminal est en état XOFF. Il convient donc d'adapter le paramètre de time-out pour le canal.

De plus tout caractère imprimable frappé au clavier est un Control Q implicite, et par conséquent redémarre la transmission vers le terminal, si elle est suspendue.

Ce mode devrait permettre de réaliser des programmes interactifs travaillant soit en plein écran, soit en style menu.

4.6 Etat après Reset du module

Un Reset du module est crée par

- la mise sous tension du module
- Z du chassis CAMAC
- la fonction F9 A0.

L'effet du Reset est de forcer un redémarrage du microprocesseur MC-6802 du module, qui réalise alors la séquence complète d'initialisation du module.

La séquence des évènements visibles par CAMAC est la suivante:

- masquage de tous les LAMs: mise à 1 du LAM mask register (retard 0)
- verrouillage des accès aux 2 boîtes à lettres (entrée et sortie) (retard 0)
- effacement de tous les LAM : mise à zéro du LAM status register (retard maxi 200µs).
- mise à disposition de la boîte à lettres de sortie avec LAM status bit R16 actif (= output mailbox disponible) (retard 29ms).
- Identificateur de version software déposé dans status register avec LAM status bit R14 actif (= data disponible dans status register) (retard 29ms).

Ce dernier évènement signale que la séquence d'initialisation du module est terminée, et qu'il est en état de recevoir la 1ère requête.

L'état du module et des canaux est alors le suivant :

1. Module

i) ressources

- Output mailbox disponible (bit R16 du LAM status à 1)
- Status register contient l'identification de la version software (bit R14 du LAM status à 1)
- Input mailbox vide, et non lisible par CAMAC
- LAM mask register ne laisse passer aucun LAM
- LAM status contient 1010 000 000 000 (binaire)
- Tous les canaux sont non actifs.

ii) paramètres de fonctionnement

- time-out de lecture de l'Input mailbox : 20 secondes.

2. Canaux

i) ressources

- Buffer de transmission vide
- Buffer de réception vide
- Circuit CA (Request to Send) pin 4 du connecteur CANON 25P OFF (voltage négatif)
- tout caractère reçu est ignoré.
- L'état de l'autre bout de la ligne est supposé non actif.

- ii) paramètres de fonctionnement présélectionnés (= de défaut) :
- Baud rate : 1200 baud
 - Stop bit : 2 stop bits
 - contrôle de parité : paire en transmission et réception
 - time-out de transmission: 20 secondes
 - mode de fonctionnement : IDI natif

Nous rappelons que le nombre de bits par caractère est de 7.

4.7 Temps d'exécution

Les temps d'exécution des divers activités du module donné ci-contre sont valables pour la version software 01. Dans bien des cas ces temps sont des valeurs minima pour l'activité.

La connaissance de ces temps devrait permettre d'évaluer à priori le degré de charge du module, et donc de savoir s'il lui reste un degré de sécurité suffisant pour le fonctionnement.

Ce module utilise un microprocesseur fonctionnant en mode d'interruption. Nous pensons donc qu'une charge globale de 50% est le maximum acceptable pour des canaux fonctionnant de façon purement alléatoire.

Il est possible d'admettre une charge plus grande si le fonctionnement des canaux est parfaitement disjoint dans le temps.

4.7.1 Exécution d'un Reset

29 ms pour l'exécution complète du Reset. Les bits du LAM status sont mis à zéro après un délai de 200 µs après le début de l'exécution.

4.7.2 Charge de base (aucun canal en activité)

Toutes les 20 millisecondes : 120 µs

Toutes les secondes : 450 µs

4.7.3 Réception d'un caractère

caractère normal : 260 µs

fin de message (CR) : 1,25 ms

en mode terminal:

caractère imprimable : 770 µs

back space : 1,50 ms

Note : Il est très important que l'activité qui se déroule lors de la réception d'une fin de message puisse s'exécuter complètement avant que le deuxième caractère du message suivant soit transmis au module. Ceci assure qu'il n'y ait pas de perte d'information. (Le 1er caractère peut être mémorisé dans le hardware du module avant d'être reçu par le software).

4.7.4 Transmission d'un caractère

pour tout caractère : 270 μ s

4.7.5 Demande de status

450 μ s

Le bit R14 (data OK dans status register) est mis à zéro après 180 μ s il repasse à 1 après 280 μ s.

4.7.6 Setting du module ou des canaux

- définition des paramètres de fonctionnement: 960 μ s
- activation/désactivation d'un canal : 4,8 ms / 500 μ s respectivement

4.7.7 Transfert d'un message de l'output mailbox dans le buffer de transmission

nch = nombre de caractères dans le message

$$\text{temps (ms)} = 4.4 + \text{nch} * 0.092$$

4.7.8 Mise à disposition d'un message dans l'input mailbox

nch = nombre de caractères dans le message

$$\text{temps (ms)} = 0.6 + \text{nch} * 0.041$$

5. INDICATEURS DU PANNEAU AVANT DU MODULE

5.1 Diodes des canaux

Sur la face avant du module chaque canal dispose de 2 diodes qui donnent purement l'activité des lignes de transmission et de réception du canal.

5.2 Diodes du module

Le module dans son ensemble dispose de 2 diodes : "ov.run" et "time-out".

Le diode "ov.run" signale les erreurs dans leur ensemble. Elle est éclairée si un canal ou le module a détecté une erreur. Elle est éteinte lorsque aucune erreur n'est en attente, où si toutes les erreurs ont été effacées par suite de leur lecture.

Si une erreur de ligne (framing error, parity, overrun) est détectée alors que le canal est non actif, cette diode s'éclaire, mais elle est éteinte automatiquement dans la seconde qui suit. Ceci est vrai sauf si d'autres erreurs sont aussi en attente.

La diode "time-out" signale l'apparition de tout phénomène de dépassement de temps. Elle est automatiquement éteinte dans la seconde qui suit.

Ce type d'erreurs est mémorisé pour le canal concerné ou pour le module. Ces erreurs sont signalées de façon permanente par la diode d'indication d'erreur "ov.run".

ANNEXE 1 Codes d'erreurs

Les codes d'erreurs retournés par le module se répartissent en 3 classes:

- Erreurs immédiates d'usage
- Erreurs différées: se sont des erreurs qui ont été détectées lors de la réception d'un message et qui sont signalées à l'utilisateur lors de la demande de lecture de ce message.
- Erreurs du software du module. Cette classe d'erreurs n'est pas le fait de l'utilisateur. Elle devrait être signalée sans tarder aux auteurs.

Codes d'erreurs

0 : pas d'erreur

Erreurs immédiates

- 1 : message tronqué (receive ring buffer overflow)
- 2 : buffer de transmission non libre
- 3 : message à transmettre trop grand (> 80 caractères), message tronqué au maximum (80 caractères)
- 4 : caractère de contrôle non autorisé dans le message
- 5 : caractère de fin de message qui n'est pas à la fin
- 6 : ce n'est pas un code de caractère (code > 177 octal)
- 7 : commandes CAMAC mélangées (valeur de A différente) n'est pas autorisée
- 8 : pas de message disponible pour lecture
- 9 : boîte à lettres d'entrée en cours d'utilisation
- 10 : time-out dans la lecture du message dans la boîte à lettres d'entrée
- 11 : time out dans la transmission d'un message
- 12 : canal non actif
- 13 : pas autorisé pour un canal actif
- 14 : pas assez de données pour la requête (liste de paramètres incomplètes)
- 15 : non autorisé pour le software du module quand il n'est pas en état de debugging
- 16 : paramètre hors limite
- 17 : pas autorisé en dehors du mode terminal pur
- 18 : erreur de cadencement dans la ligne (framing error)

- 19 : surécriture du caractère reçu (receiver overrun)
- 20 : erreur de parité à réception d'un caractère
- 21 : code software d'une requête hors limite
- 22 : erreur de réception d'un caractère du message
- 23 : boîte aux lettres d'entrée libérée par requête

Erreurs différées

Les codes d'erreurs sont égales à 200 (décimal) plus le code d'erreur immédiate correspondant.

Erreurs du software du module

- 254 : erreur de stack
- 255 : incohérence dans les variables d'états

ANNEXE 2 Valeur de défaut des paramètres de fonctionnement

Les valeurs de défaut des paramètres de fonctionnement sont mises :

- par le Reset du module
- par un setting de ces paramètres avec une valeur 0

Les valeurs de défauts sont :

- i) pour le module dans son ensemble :
 - time out de lecture de l'Input mailbox : 20 secondes
- ii) pour les canaux
 - Baud rate : 1200 baud
 - Stop bits : 2
 - Contrôle de parité : paire pour transmission et réception
 - time out de transmission du message : 20 secondes
 - mode de fonctionnement du canal : IDI natif

Remarque :

le nombre de bit par caractère dans les transmissions (transmission et réception) est toujours de 7 bits plus parité. Ce nombre de bits n'est pas programmable.

ANNEXE 3 Software du module

Le software que nous avons produit pour ce module est un système en temps réel piloté par interruptions. Les sources d'interruption sont :

- un timer (20 millisecondes)
- l'activité des boîtes à lettres d'entrée et de sortie
- les transmissions et réceptions de caractère sur les lignes
- les interruptions software pour les requêtes au moniteur

Les modules software principaux sont :

- les handlers d'interruptions et leur bibliothèque de routines associées
- un mini-moniteur temps réel
- les processeurs des requêtes au moniteur
- les serveurs des requêtes au module
- une bibliothèque (non réentrante) pour les serveurs
- une bibliothèque (réentrante) pour les handlers et les serveurs
- le processeur de démarrage du module.

Les divers modules software ont été découpés pour faire en sorte qu'il soit aisé de rajouter de nouveaux serveurs de requêtes.

Le software est assemblé à l'aide du cross software pour les microprocesseurs Motorola 6800 de la division DD sur l'IBM/SIEMENS.

La liste du software du module (plus que 100 pages) peut être consultée (ou une copie obtenue) auprès des auteurs.

Deux signaux internes au module ont été programmés pour faciliter l'évaluation du fonctionnement du module:

- timer 20 ms (signal carré de 40ms de période), patte 17 du circuit intégré 15 (SY 6522)
- indicateur d'activité globale: pine 7 du circuit intégré 16 (MC 6802)
: potentiel haut = processeur inactif; potentiel bas = processeur actif.

ANNEXE 4 Connexion des lignes de communication

Les lignes de communication de chaque canal peuvent fonctionner soit en standard RS 232 C (= 24V) soit en mode boucle de courant 20mA.

Mode RS 232 C (ref. EIA standad RS 232 C - August 1969)

Strap 3 mis, Straps 1 et 2 enlevés

Le protocole de ligne fonctionnant en mode start/stop, la connexion minimum intéresse donc seulement les circuits de terre et de données, soit :

circuit (des connecteurs CANON 25p)

AA (pin 1) ground (ou AB pin 7)

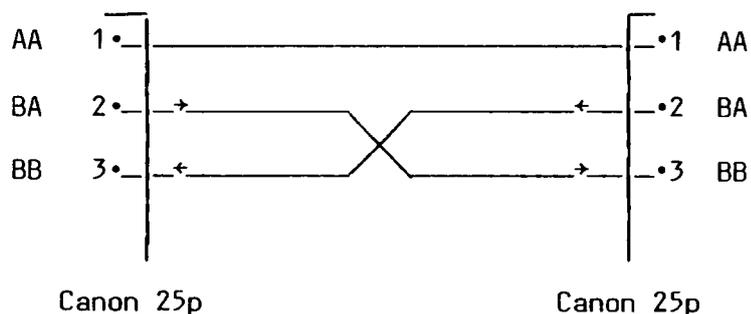
BA (pin 2) transmitted data

BB (pin 3) received data

De plus un équipement peut utiliser le circuit CB (pin 5) Clear to Send pour contrôler le flot de données vers lui et le circuit CC (pin 6) Data Set Ready pour signaler son état.

Le circuit CA (pin 4) Request to Send est aussi disponible.

Connexion typique minimum (strap 3 mis, straps 1 et 2 enlevés)



Mode boucle de courant (20mA)

Dans ce mode chaque ligne (transmission et réception) d'une liaison peut être soit source de courant soit passive suivant le cablage. Il n'y a pas toutefois dans ce module d'isolation galvanique des lignes.

Circuit de transmission : strap 2 mis, strap 3 enlevé

La source de courant est disponible sur la pine 11 du connecteur Canon 25P. Les données (Data) sont sur la pine 2.

Circuit de réception : données sur la pine 3.

strap 1 mis : source de courant

strap 1 enlevé: passif

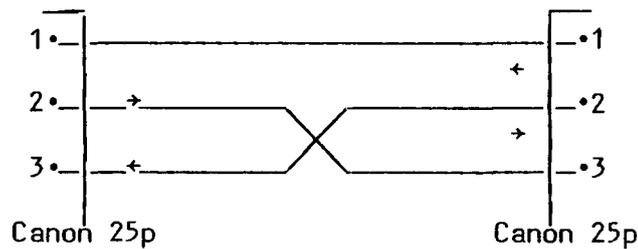
Les terres (pines 1 ou 7) doivent toujours être raccordées.

Du fait qu'il n'y a pas d'isolation galvanique il peut être nécessaire d'interposer sur la ligne un tel dispositif.

Les interconnexions les plus classiques sont donc schématisées ci-contre.

i) connexion symétrique

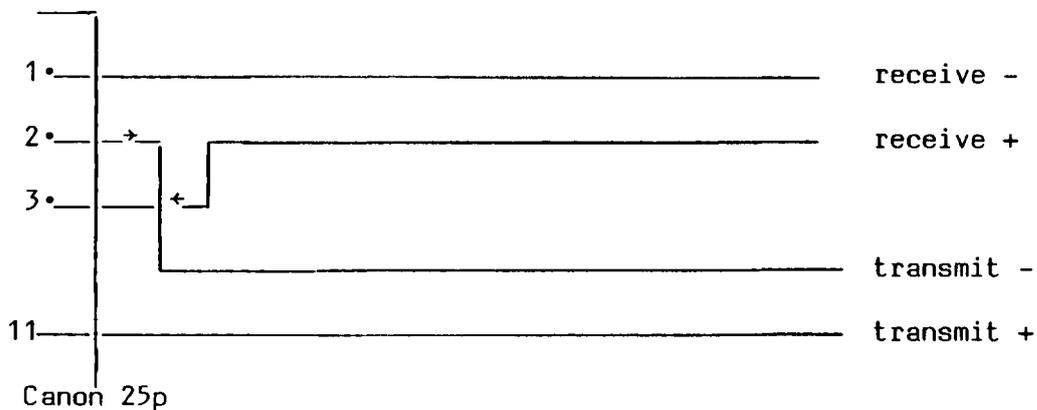
Récepteur source de courant: (straps 1 et 2 mis, strap 3 enlevé)
transmettre passif



Strap 1 mis aux 2 extrémités

ii) connexion source de courant

transmetteur et récepteur sont source de courant (straps 1 et 2 mis, strap 3 enlevé)



source de courant

ANNEXE 5 Protocole de ligne pour le mode de fonctionnement IDI émulation

Le protocole de ligne IDI natif (cf. PS/CO/Note 83/17) est modifié comme suit pour permettre une émulation aisée par terminal.

1. La liaison se fait toujours en mode start/stop (start = control Q, stop = control S) mais le module ne considère jamais de stop implicite ce qui veut dire que:

i) en fin d'activation d'un canal (après l'envoi de control G) le terminal est immédiatement supposé être en état actif.

ii) en fin de message (après l'envoi de CR) le terminal est capable immédiatement de recevoir les caractères suivants, à savoir entre autre le line-feed (LF).

Control S/Control Q (ou tout autre caractère imprimable), peut être utilisé pour contrôler la transmission vers le terminal.

2. Tout caractère imprimable frappé au clavier, et qui est déposé dans le message en cours de réception est retransmis pour écho vers le terminal.

3. La fin de message est déclenchée par la clef "carriage return" (CR). L'écho correspondant est CR, LF.

4. L'effacement du dernier caractère du message couramment ouvert se fait par la clef back space (BS). Si un caractère a été effectivement effacé, l'écho est la suite BS, espace, BS pour effacer le caractère de l'écran.

5. Tout caractère de contrôle autre que Control Q, Control S, Back space et Carriage return est complètement ignoré en entrée.

ANNEXE 6 Protocole de ligne pour le mode de fonctionnement terminal pur

Ce mode est utilisable soit pour disposer de terminal simple en mode interactif (utilisation en style menu) soit pour des imprimantes avec connexions en série.

Ce protocole est de type start-stop.

Transmission Tous les caractères (de contrôles ou imprimables) d'un message sont transmis sans modification.

Aucun caractère n'est transmis en fin de message (pas de auto line feed).

Il n'y a pas de stop implicite en fin de message, donc le flux de transmission n'est pas suspendu par la fin d'un message.

Le flux de caractères transmis peut être contrôlé par l'envoi au module du Control S (stop), Control Q (start) ou tout autre caractère imprimable (=start).

En fin d'activation du canal, le module envoie Control G (Bell).

Réception L'unité de réception est le message composé d'une chaîne de caractère imprimable (qui peut être vide).

Le seul caractère reconnu de fin de message est "carriage return" (CR) dont l'écho est CR, LF.

Tout caractère imprimable déposé dans le message est transmis en écho (pour l'écran).

Aucun caractère de contrôle ne peut être déposé dans un message.

L'édition du message en cours se fait par la clef Back space (BS) qui supprime le dernier caractère du message. Si cette suppression a eu lieu l'écho est la suite BS, espace, BS, pour effacer et faire reculer le curseur d'une position sur l'écran.

ANNEXE 7 Fonctions CAMAC du module 3IDI 2092

F0 A0	Q=1/0	lecture d'un byte de l'input mailbox
F1 A0	Q=1	lecture du status register
F1 A12	Q=1	lecture du LAM register
F1 A14	Q=1	lecture du LAM registre après masquage
F8 A0	Q=1/0	test de l'état du LAM
F9 A0	Q=0	reset complet du module
F16 A0	Q=1/0	dépose d'un byte dans l'output mailbox pour le canal 0
F16 A1	Q=1/0	idem F16 A0 pour canal 1
F16 A2	Q=1/0	idem F16 A0 pour canal 2
F16 A3	Q=1/0	demande de message pour le canal spécifié par la donnée
F16 A4	Q=1/0	dépose d'un paramètre pour un requête au canal 0
F16 A5	Q=1/0	idem F16 A4 pour canal 1
F16 A6	Q=1/0	idem F16 A4 pour canal 2
F16 A7	Q=1/0	dépose d'un paramètre pour une requête globale au module
F17 A13	Q=1	écriture du LAM mask register
F23 A12	Q=1	effacement sélectif du LAM register
F25 A0	Q =0	demande d'exécution de la requête

Z ou mise sous tensions action identique à F9 A0

ANNEXE 8 Format du LAM register

canal 0	R1=1	buffer de transmission disponible
	R2=1	message(s) disponible(s) dans le buffer de recetpion
	R3=1	indicateur d'erreurs du canal
	R4=1	canal actif
canal 1	R5	idem R1 pour canal 1
	R6	idem R2 pour canal 1
	R7	idem R3 pour canal 1
	R8	idem R4 pour canal 1
canal 2	R9	idem R1 pour canal 2
	R10	idem R2 pour canal 2
	R11	idem R3 pour canal 2
	R12	idem R4 pour canal 2
module	R13=1	indicateur d'erreurs du module
	R14=1	données dans status register
	R15=1	message dans input mailbox
	R16=1	output mailbox disponible

ANNEXE 9 Codes de requêtes pour le module et les canaux

i) Requêtes au module (déposées par F16 A7)

0	lecture de l'erreur générale du module
1	lecture d'un byte de la mémoire du microprocesseur
2	lecture de la version software
254	libération immédiate de la boîte à lettres d'entrée
255	définition du time out de lecture de la boîte à lettres d'entrée

ii) Requêtes au canaux (déposées par F16 A4,5,6)

0	lecture de l'erreur pour le canal
1	lecture de l'état de l'autre extrémité de la ligne
2	lecture du nombre de messages prêts à être lus
3	nombres de caractères dans le prochain message disponible
4	lecture du dernier caractère reçu sur la ligne
5	lecture des indicateurs d'erreur de ligne pour le dernier caractère reçu
6	lecture de l'état du signal "Data set ready"
7	lecture du mode de fonctionnement du canal
252	purge du buffer circulaire de reception du canal
253	paramètres de fonctionnement du canal
254	désactivation du canal
255	initialisation du canal

ANNEXE 10

ASCII Character Set																					
dec	octal	hex	ASCII	dec	octal	hex	ASCII	dec	octal	hex	ASCII	dec	octal	hex	ASCII						
0	000	000	00 00	NUL	null	32	040	240	20 A0	space	64	100	300	40 C0	@	96	140	140	60 60		
1	001	201	01 81	SOH	start of heading	33	041	041	21 21	!	65	101	101	41 41	A	97	141	341	61 E1	a	
2	002	202	02 82	STX	start of text	34	042	042	22 22	"	66	102	102	42 42	B	98	142	342	62 E2	b	
3	003	003	03 03	ETX	end of text	35	043	243	23 A3	#	67	103	303	43 C3	C	99	143	143	63 63	c	
4	004	204	04 84	EOT	end of transmission	36	044	044	24 24	\$	68	104	104	44 44	D	100	144	344	64 E4	d	
5	005	005	05 05	ENQ	enquiry	37	045	245	25 A5	%	69	105	305	45 C5	E	101	145	145	65 65	e	
6	006	006	06 06	ACK	acknowledge	38	046	246	26 A6	&	70	106	306	46 C6	F	102	146	146	66 66	f	
7	007	207	07 87	BEL	bell	39	047	047	27 27	'	71	107	107	47 47	G	103	147	347	67 E7	g	
8	010	210	08 88	BS	back space	40	050	050	28 28	(72	110	110	48 48	H	104	150	350	68 E8	h	
9	011	011	09 09	HT	horizontal tab	41	051	251	29 A9)	73	111	311	49 C9	I	105	151	151	69 69	i	
10	012	012	0A 0A	LF	line feed	42	052	252	2A AA	*	74	112	312	4A CA	J	106	152	152	6A 6A	j	
11	013	213	0B 8B	VT	vertical tab	43	053	053	2B 2B	+	75	113	113	4B 4B	K	107	153	353	6B EB	k	
12	014	014	0C 0C	FF	form feed	44	054	254	2C AC	,	76	114	314	4C CC	L	108	154	154	6C 6C	l	
13	015	215	0D 8D	CR	carriage return	45	055	055	2D 2D	-	77	115	115	4D 4D	M	109	155	355	6D ED	m	
14	016	216	0E 8E	SO	shift out	46	056	056	2E 2E	.	78	116	116	4E 4E	N	110	156	356	6E EE	n	
15	017	017	0F 0F	SI	shift in	47	057	257	2F AF	/	79	117	317	4F CF	O	111	157	157	6F 6F	o	
16	020	220	10 90	DLE	data link escape	48	060	060	30 30	0	80	120	120	50 50	P	112	160	360	70 F0	p	
17	021	021	11 11	DC1	device control 1	49	061	261	31 B1	1	81	121	321	51 D1	Q	113	161	161	71 71	q	
18	022	022	12 12	DC2	device control 2	50	062	262	32 B2	2	82	122	322	52 D2	R	114	162	162	72 72	r	
19	023	223	13 93	DC3	device control 3	51	063	063	33 33	3	83	123	123	53 53	S	115	163	363	73 F3	s	
20	024	024	14 14	DC4	device control 4	52	064	264	34 B4	4	84	124	324	54 D4	T	116	164	164	74 74	t	
21	025	225	15 95	NAK	negative acknowledge	53	065	065	35 35	5	85	125	125	55 55	U	117	165	365	75 F5	u	
22	026	226	16 96	SYN	synchronous idle	54	066	066	36 36	6	86	126	126	56 56	V	118	166	366	76 F6	v	
23	027	027	17 17	ETB	end of transmission block	55	067	267	37 B7	7	87	127	327	57 D7	W	119	167	167	77 77	w	
24	030	030	18 18	CAN	cancel	56	070	270	38 B8	8	88	130	330	58 D8	X	120	170	170	78 78	x	
25	031	231	19 99	EM	end of medium	57	071	071	39 39	9	89	131	131	59 59	Y	121	171	371	79 F9	y	
26	032	232	1A 9A	SUB	substitute	58	072	072	3A 3A	:	90	132	132	5A 5A	Z	122	172	372	7A FA	z	
27	033	033	1B 1B	ESC	escape	59	073	273	3B BB	;	91	133	333	5B DB	{	123	173	173	7B 7B	{	
28	034	234	1C 9C	FS	file separator	60	074	074	3C 3C	<	92	134	134	5C 5C	}	124	174	374	7C FC	}	
29	035	035	1D 1D	GS	group separator	61	075	275	3D BD	=	93	135	335	5D DD	~	125	175	175	7D 7D	~	
30	036	036	1E 1E	RS	record separator	62	076	276	3E BE	>	94	136	336	5E DE	^	126	176	176	7E 7E	^	
31	037	237	1F 9F	US	unit separator	63	077	077	3F 3F	?	95	137	137	5F 5F	_	127	177	377	7F FF	^	RUB rubout

Columns: 1: decimal; 2: octal, no parity; 3: octal, odd parity; 4: hex, no parity; 5: hex, odd parity.