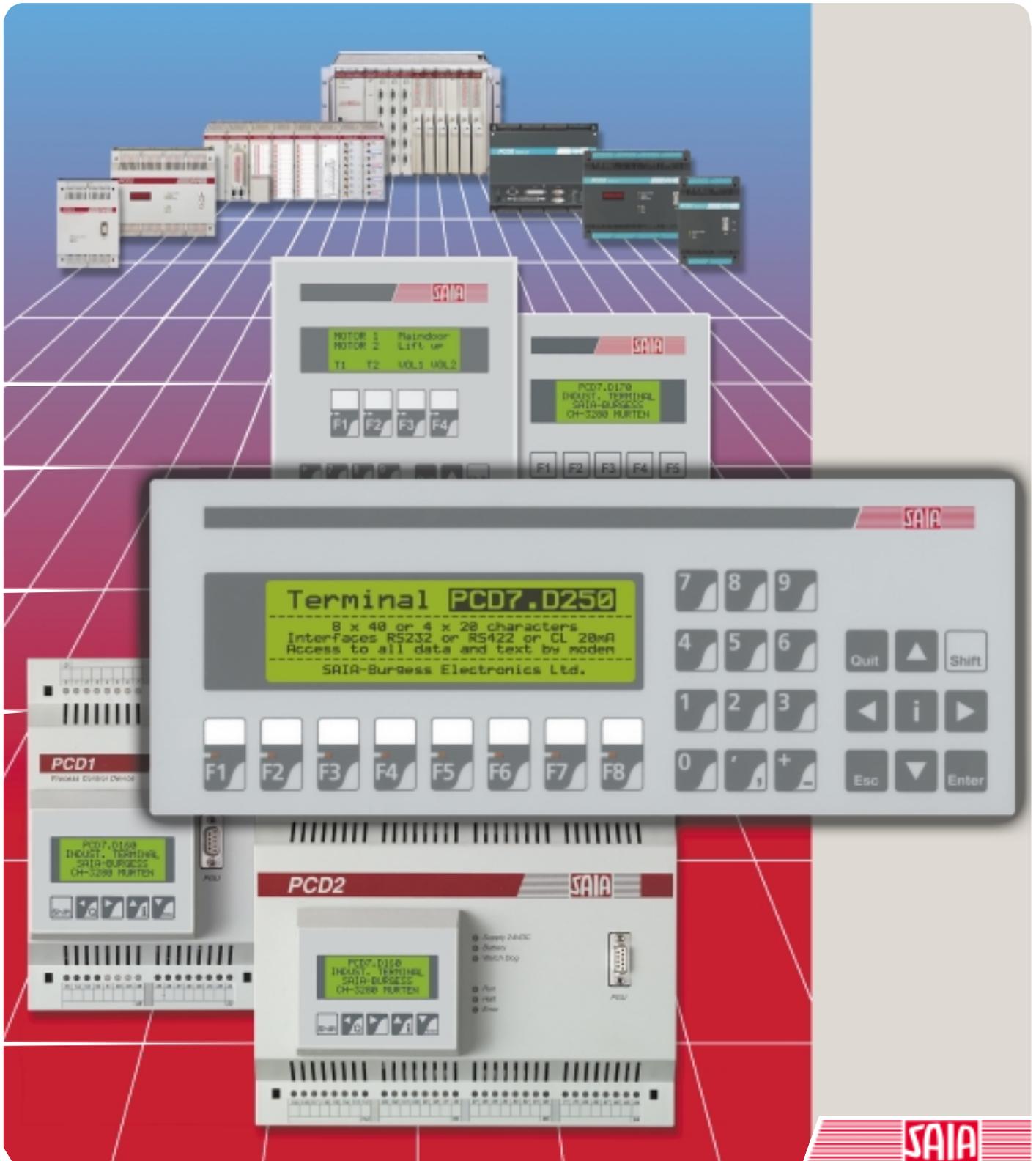


SAIA® PCD
Process Control Devices

PCD7.D250
Terminal industriel
Manuel



SAIA® Process Control Devices

Manuel

Terminal industriel

PCD7.D250

SAIA-Burgess Electronics SA 2000. Tous droits réservés
Edition 26/770 F1 - 06.2000

Sous réserve de modifications

Mise à jour

Manuel : Terminal industriel PCD7.D250 - édition F1

Date	Chapitre	Page	Description
01.12.2000	4.3	4-3	Interface série COM 1 : Modes opératoires

	Page
6. Commandes via l'interface série	
6.1 Configuration du terminal	6-2
6.2 Commandes du curseur	6-5
6.3 Commandes de l'affichage	6-8
6.4 Commandes des DEL	6-10
6.5 Autres commandes	6-11
6.6 Récapitulatif des commandes	6-13
7. Jeux de caractères	
7.1 Code ASCII standard (32 à 127 décimal, 20 à 7F hexa)	7-1
7.2 Code ASCII étendu (128 à 255 décimal, 80 à FF hexa)	7-1
7.2.1 CodePage 437 (défaut)	7-2
7.2.2 Jeux de caractères compatible ..D100	7-4
8. Exemples de programme utilisateur	
8.1 Configuration matérielle	8-1
8.2 Transmission d'un seul texte	8-2
8.2.1 Programme utilisateur en BLOCTEC	8-2
8.2.2 Programme utilisateur en GRAFTEC	8-4
8.2.3 Transmission d'un texte associant petits et gros caractères par fermeture de l'entrée 0	8-6
8.3 Transmission de plusieurs textes	8-7
8.3.1 Programme utilisateur en BLOCTEC	8-7
8.3.2 Programme utilisateur en GRAFTEC	8-9
8.4 Reconnaissance de la pression d'une touche	8-13
8.4.1 Programme utilisateur avec sauts	8-14
8.4.2 Programme utilisateur en BLOCTEC	8-17
8.4.3 Programme utilisateur en GRAFTEC	8-20
8.5 Saisie de paramètres numériques (ex. page 8-26)	8-25
8.6 Bloc de fonction : INPUT	8-33
9. Comparatif PCD7.D202 ↔ PCD7.D250	
10. Câbles de l'interface série RS 232	



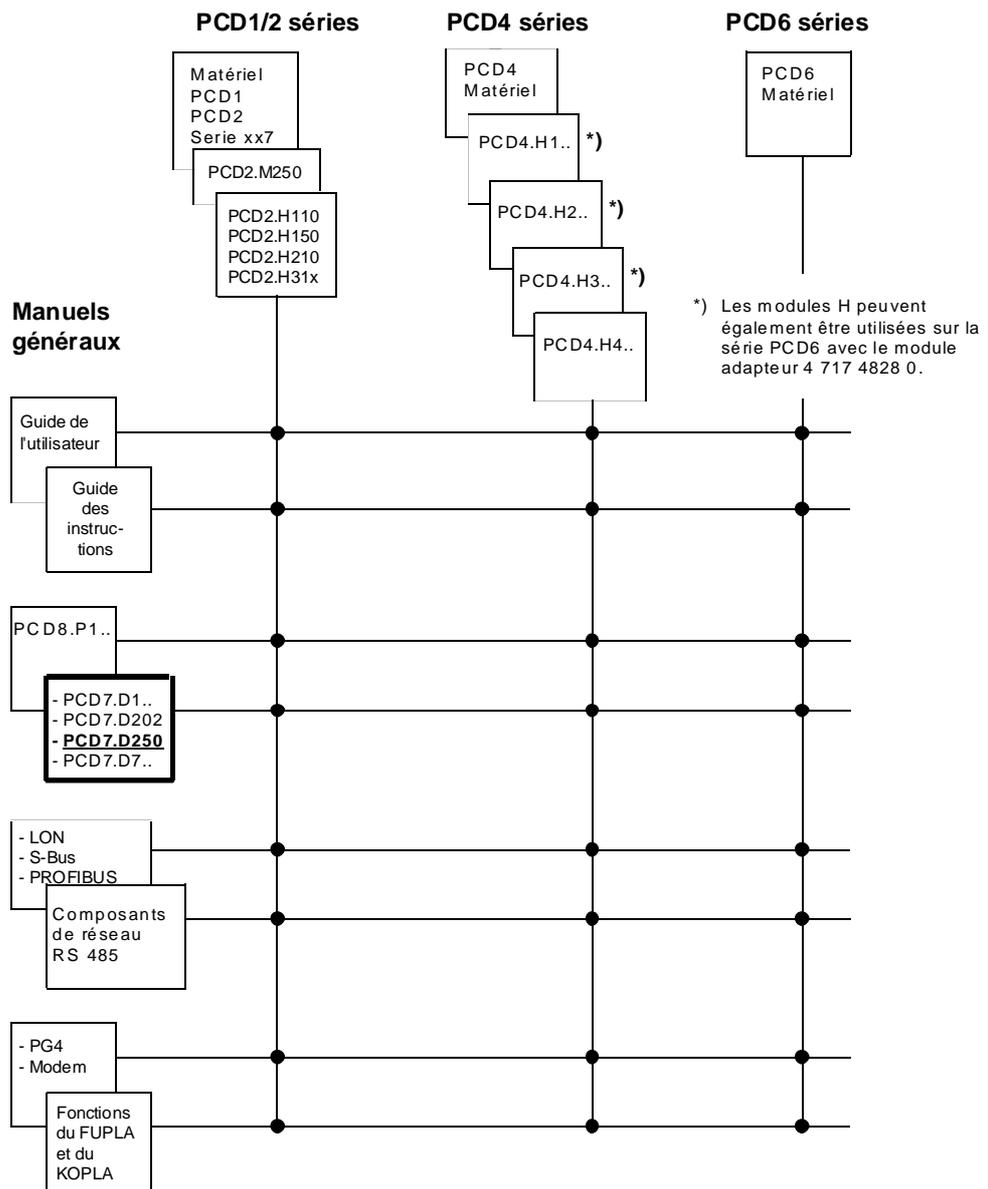
Avis aux lecteurs :

De nombreux manuels techniques précis et détaillés ont été élaborés par SAIA-Burgess Electronics SA afin de faciliter l’installation et l’exploitation de ses automates programmables ; ils s’adressent à un personnel qualifié ayant suivi au préalable nos stages de formation.

Pour optimiser les performances des appareils de commande de processus SAIA® PCD, nous vous conseillons de respecter scrupuleusement les consignes de montage, de câblage, de programmation et de mise en service figurant dans ces manuels. Cette démarche rigoureuse vous donnera l’assurance d’une satisfaction totale.

Toutefois, si vous souhaitez formuler des propositions ou des commentaires visant à améliorer la qualité et le contenu de nos documentations, nous vous serions reconnaissants de compléter le formulaire situé en dernière page de cette notice.

Vue d’ensemble de la gamme et de la documentation PCD



Fiabilité et sécurité des automates programmables

Soucieux d'offrir à sa clientèle des automates programmables fiables et sûrs, SAIA-Burgess Electronics SA apporte le plus grand soin à la conception, au développement et à la fabrication de ses produits.

Parmi ces mesures, citons :

- Technologie de pointe,
- Conformité aux normes,
- Certification ISO 9001,
- Agrément de nombreux organismes internationaux (Germanischer Lloyd, UL, Det Norske Veritas, marquage CE...),
- Choix de composants de haute qualité,
- Contrôles qualité aux différents stades de fabrication,
- Essais en conditions réelles de fonctionnement.

Malgré l'excellence et le grand soin apporté à sa production, SAIA-Burgess Electronics SA ne saurait être tenu responsable des défaillances naturelles d'un composant. A cet égard, les « Conditions générales de vente » exposent clairement les limites de garantie offertes par SAIA-Burgess Electronics SA.

Le responsable de production doit également s'assurer de la fiabilité de son installation ; il lui incombe en effet de se conformer aux spécifications techniques de l'automate sans jamais le soumettre à des conditions extrêmes d'utilisation (respect de la plage de températures, protection contre les surtensions, immunité aux parasites et tenue aux chocs).

Il lui faut en outre veiller à l'application de toutes les règles de sécurité en vigueur afin de garantir qu'aucun produit défectueux ne risque de porter atteinte à la sécurité des biens et des personnes. Tout défaut générateur de danger doit donner lieu à des mesures complémentaires visant à l'identifier et à en prévenir les conséquences. Ainsi les sorties directement liées à la sécurité de fonctionnement du matériel doivent être raccordées aux entrées et surveillées par logiciel. Il convient enfin de faire systématiquement appel aux fonctions de diagnostic du PCD (chien de garde, blocs d'organisation des exceptions « XOB », instructions de test ou de recherche d'erreurs).

Exploitée dans les règles de l'art, la gamme SAIA® PCD intègre des constituants d'automatismes modernes, alliant sécurité et haute fiabilité, et capables d'assurer pendant des années les fonctions de contrôle-commande, de régulation et de surveillance de votre équipement.

1. Présentation générale

Avec l'offre PCD7.D., SAIA propose une gamme de terminaux d'exploitation, spécialement conçue pour répondre aux milieux industriels les plus sévères et assurer une conduite optimale, au pied de la partie opérative. Grâce à l'édition de texte en clair, spécificité de la gamme SAIA® PCD, ces terminaux industriels facilitent le dialogue homme-machine guidé par menus.

Le ..D250 propose 2 différentes grandeurs de polices (permettant l'affichage de 4 x 20 ou 8 x 40 caractères). Ceci permet au ..D250 d'être utilisé aussi bien dans le domaine de la machine outil que dans la gestion de bâtiments.

Dotés d'un affichage de pointe (écran à cristaux liquides, à fort contraste et rétroéclairage par diodes électroluminescentes), les terminaux SAIA® PCD7.D250 permettent à l'opérateur de visualiser tout type d'information (température, pression, nombre d'unités, date, heure, etc.) ainsi que des messages d'état ou d'alarme.

Le clavier à habillage film polyester, pour une parfaite résistance à l'abrasion, compte 29 touches à effet tactile permettant une saisie ergonomique des données ou des fonctions de traitement indispensables à l'exploitation. Le dialogue, guidé par menus, s'établit directement avec l'automate par l'intermédiaire de l'interface série.

Les touches sont personnalisables selon les besoins de l'utilisateur grâce à des légendes amovibles se glissant facilement sous le panneau avant à membrane transparente. Disposées directement sous l'écran pour un plus grand confort d'exploitation, les touches de fonction peuvent également servir de touches programmables.



Pour une exploitation optimale et rapide du terminal PCD7.D250

Ce manuel a pour vocation de vous présenter l'ensemble des fonctionnalités du PCD7.D250 ; il se peut toutefois que votre application n'en exploite que quelques-unes.

Pour faciliter et accélérer votre apprentissage, nous vous proposons d'exécuter l'un des programmes du chapitre 8 avant d'approfondir l'étude de chaque test et commande du ..D250.

Cet exercice pratique vous permettra de vous familiariser avec l'édition de textes simples. Vous constaterez alors que le dialogue entre l'automate PCD et le terminal ..D250 est d'une simplicité extrême !

2. Caractéristiques techniques

Généralités

Affichage	<p>Ecran à cristaux liquides en technologie « supertwist »</p> <p>Rétroéclairé par LED</p> <p>8 lignes de 40 caractères (hauteur : 3.7 mm) et 4 lignes de 20 caractères (hauteur : 7.5 mm)</p> <p>avec curseur</p> <p>Jeu de caractères : ASCII (32 à 127), fonctions de contrôle et caractères spéciaux (IBM Extended Character Set 437)</p>
Clavier	<p>Clavier à membrane de 29 touches à effet tactile</p> <p>Pavé numérique de 12 touches, espacement 18 mm</p> <p>Pavé de commande de 9 touches, dont 4 fléchées, espacement 18 mm</p> <p>8 touches de fonction, espacement 20 mm, avec LED rouges intégrées et légendes amovi- bles pour personnalisation du marquage</p>
Interface	<p>Interface de communication (raccordement au SAIA®PCD) pour l'envoi de texte ou d'instruction</p> <p>COM 1 : RS 232 (fixe)</p> <p>COM 0 : pour un module de communi- cation PCD7.F2.. (RS 422 ou boucle de courant 20 mA)</p> <p>Vitesse de transmission : 110 à 9600 bit/s</p>

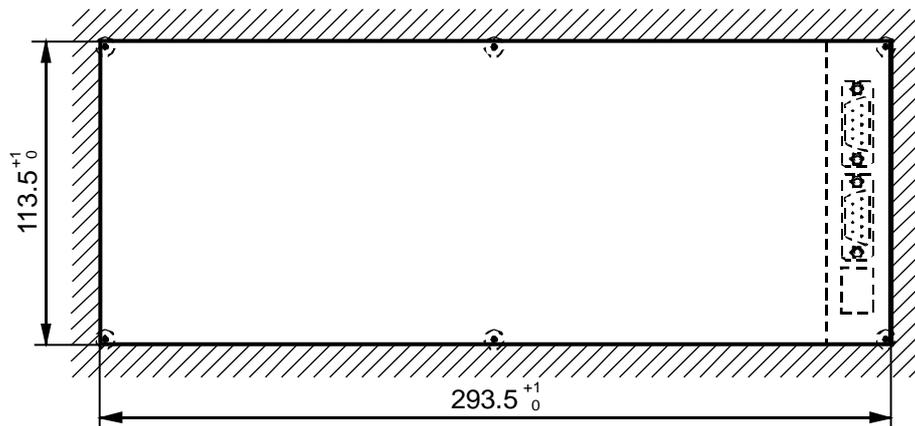
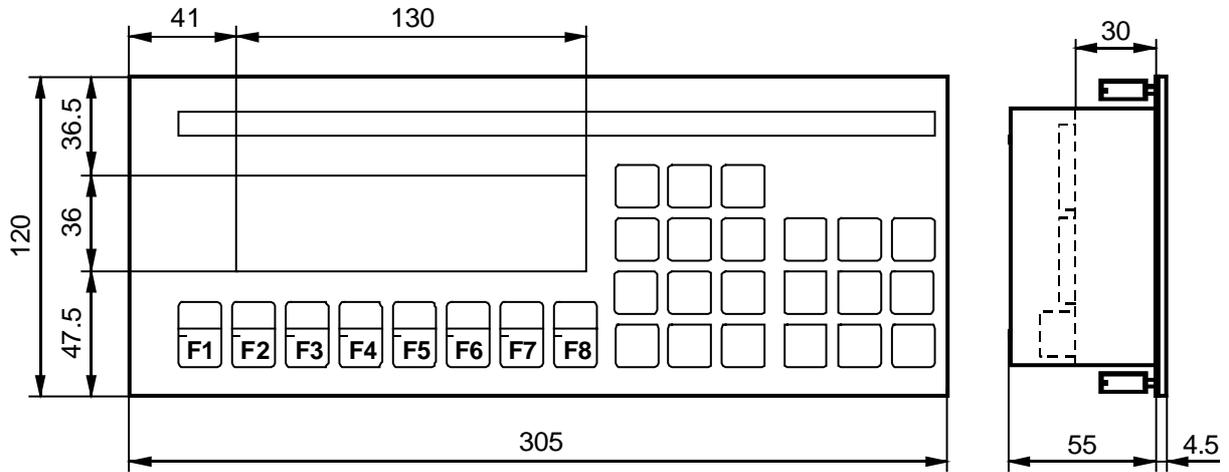
Caractéristiques électriques

Tension	24 VCC +30 % / -20 %, lissée, avec protection contre les inversions de polarité, ou 19 VCA ±15 %, redressée à doubles alternances, avec protection contre les inversions de polarité
Consommation	320 mA maxi sous 24 VCC
Raccordement	Alimentation : Bornier à vis débrochable, section de fil maxi : 2.5 mm ² Communication : Connecteur Sub-D 9 points
CEM	Emission : marquage CE selon EN 50 081-1 Immunité : marquage CE selon EN 50 082-2

Caractéristiques mécaniques

Boîtier	Plaque frontale : en aluminium recouvert d'un film polyester, classe de protection IP 65 Capot arrière : en aluminium, classe de protection IP 30
Encombrement	Voir chapitre 3 (cotes et découpe du panneau)
Montage	6 boulons filetés
Température ambiante	Fonctionnement : 0°C à +50°C Stockage : -25°C à +70°C
Hygrométrie	5 à 95 %, sans condensation selon CEI 1131-2 et DIN 40 040 classe F
Tenue mécanique	Vibrations de 10 à 57 Hz, 0.075 mm ou de 57 à 150 Hz, 1 g selon CEI 68-2-6

3. Encombrement



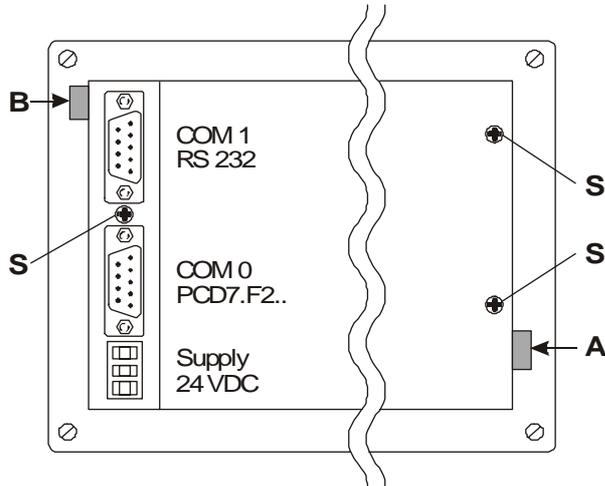
Découpe du panneau pour encastrément

Montage par 6 boulons filetés

Notes personnelles :

4. Constitution

4.1 Alimentation



Le raccordement de l'alimentation s'effectue sur bornier à vis débrochable pour fils de 2.5 mm² maxi (ou fils souples de 1.5 mm² maxi, équipés d'un embout de câblage).

24 VCC +30 % / -20 %, lissée, ou
19 VCA ±15 % redressée à doubles alternances, avec protection contre les inversions de polarité.

Le marquage personnalisé des 8 touches de fonction se glisse par la fente A, ou par la fente B pour un nouveau label.

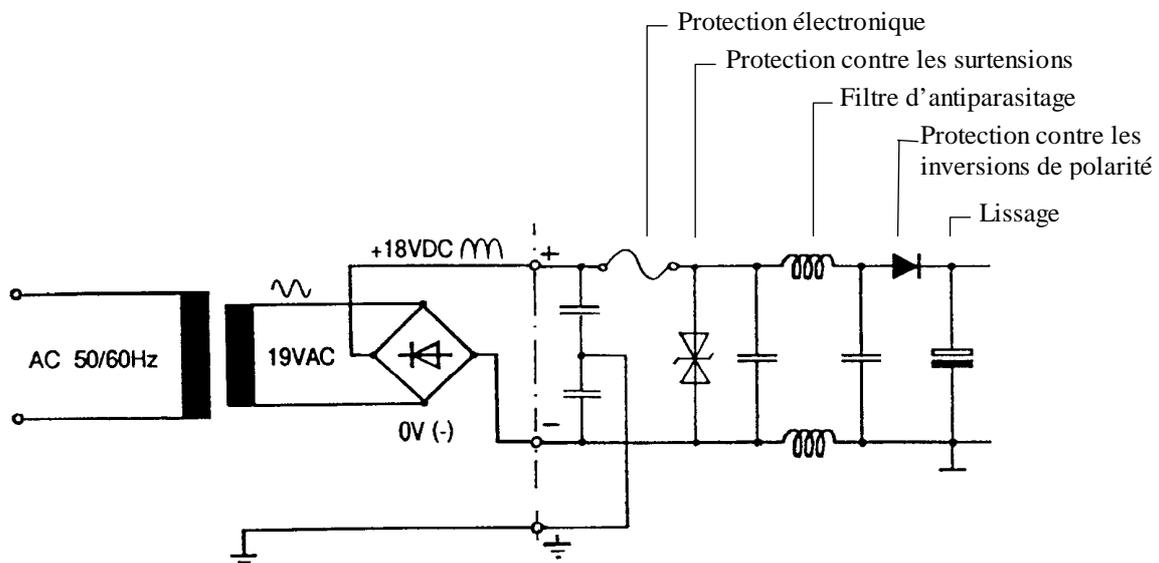


Pour une mise en service et une exploitation du terminal ..D250 conformes aux règles de l'art, la mise à la terre du boîtier est impérative. De plus, en cas de dépose du capot arrière, veuillez resserrer à fond la vis de fixation S afin de rétablir un parfait raccordement à la masse mécanique.

Alimentation alternative redressée à doubles alternances

Alimentation externe

PCD7.D250



4.2 Programme - système (firmware)

Pour mettre à jour le firmware, stocké en mémoire EPROM, il convient de retirer le capot arrière en appuyant sur les deux loquets représentés sur le schéma du § 4.1.

4.3 Interface série COM 1 : RS 232 (équipement de base)

L'interface série RS 232 est matérialisée par un connecteur Sub-D 9 points (port COM 1).

Modes opératoires

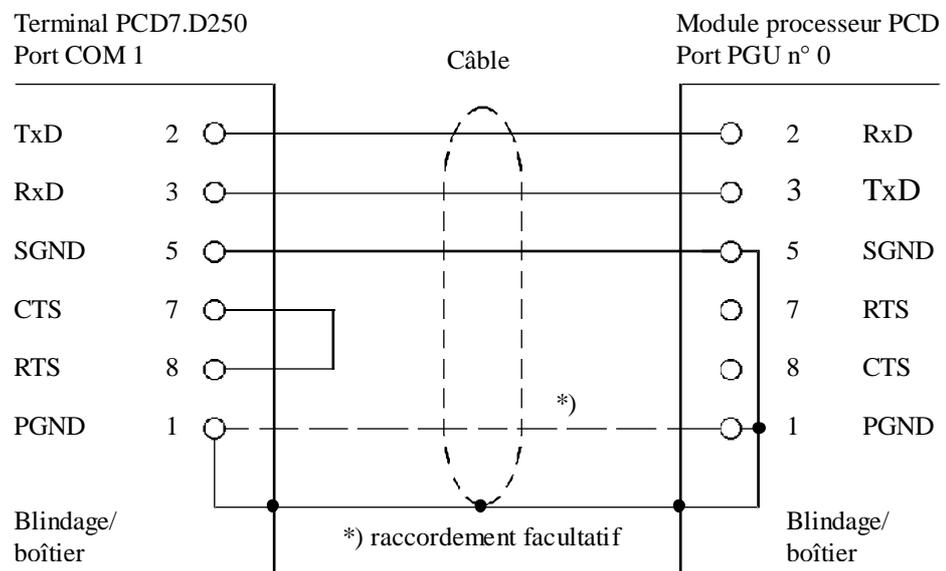
1) Avec ou sans HANDSHAKING			
Vitesse de transmission	Types	Handshaking	Contrôle
jusqu'à 9600 bauds	MC0	sans	---
jusqu'à 9600 bauds	MC1	avec	RTS/CTS
jusqu'à 9600 bauds	MC2	avec	XON/XOFF
2) Avec HANDSHAKING uniquement			
Vitesse de transmission	Types	Handshaking	Contrôle
19200 bauds	MC1	avec	RTS/CTS
19200 bauds	MC2	avec	XON/XOFF

4.3.1 Liaisons sans protocole de contrôle RTS/CTS (ou avec protocole de contrôle XON/XOFF)

Il importe de respecter les trois consignes suivantes, quel que soit le port de communication PCD mis en œuvre :

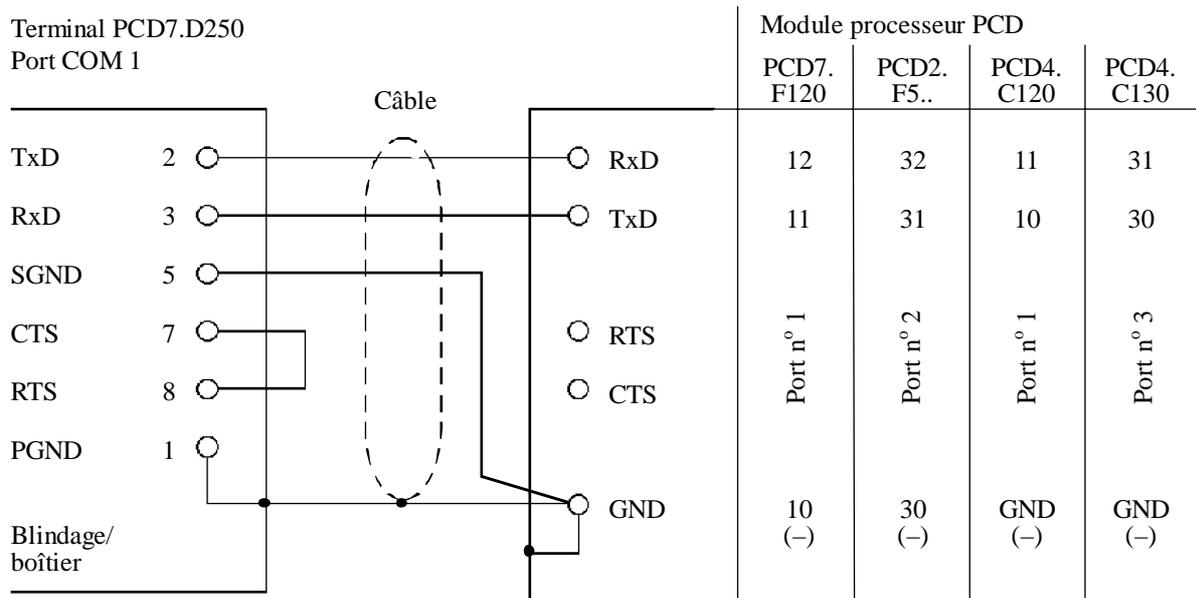
- Relier la broche RTS (Demande pour émettre) à la broche CTS (Prêt à émettre), côté ..D250.
- Si le temps de rafraîchissement est faible (300 à 500 ms), il est possible de transmettre en mode MC0 jusqu'à 9600 bauds.
- Toutefois, pour éviter une saturation du tampon d'entrée, il est conseillé d'opter pour le mode MC2 (XON/XOFF).

a) Liaison Terminal PCD7.D250 – Port PGU du PCD



Câble de liaison préconisé : PCD7.K412 (voir chapitre 10).

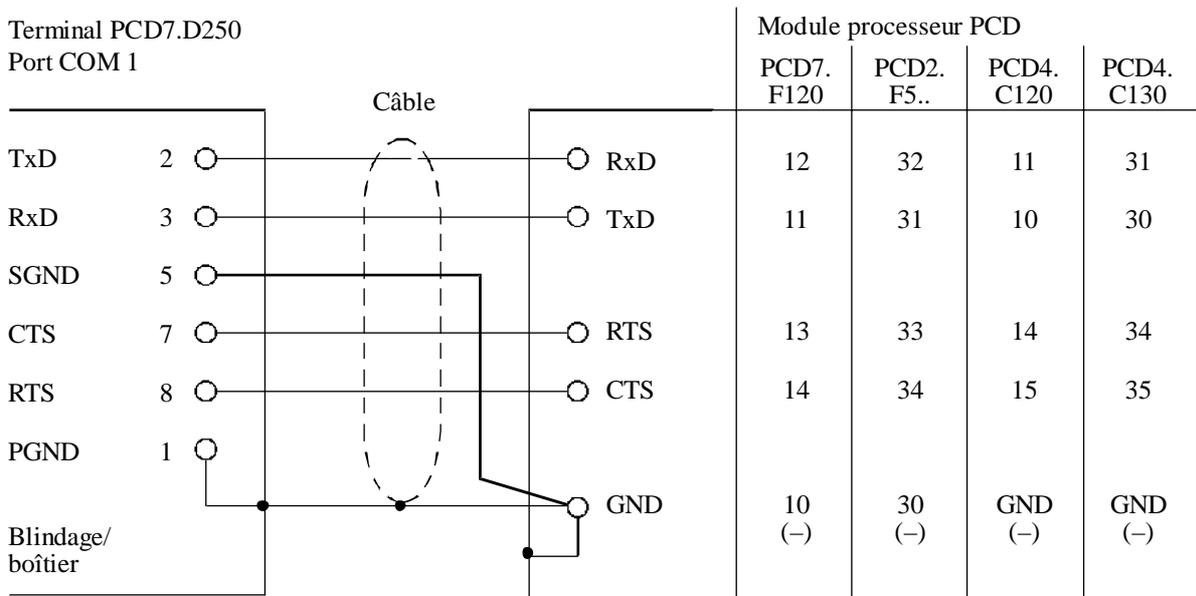
b) Liaison Terminal PCD7.D250 – Processeurs PCD, ports 1 à 3



Câble de liaison préconisé : PCD7.K422 (voir chapitre 10).

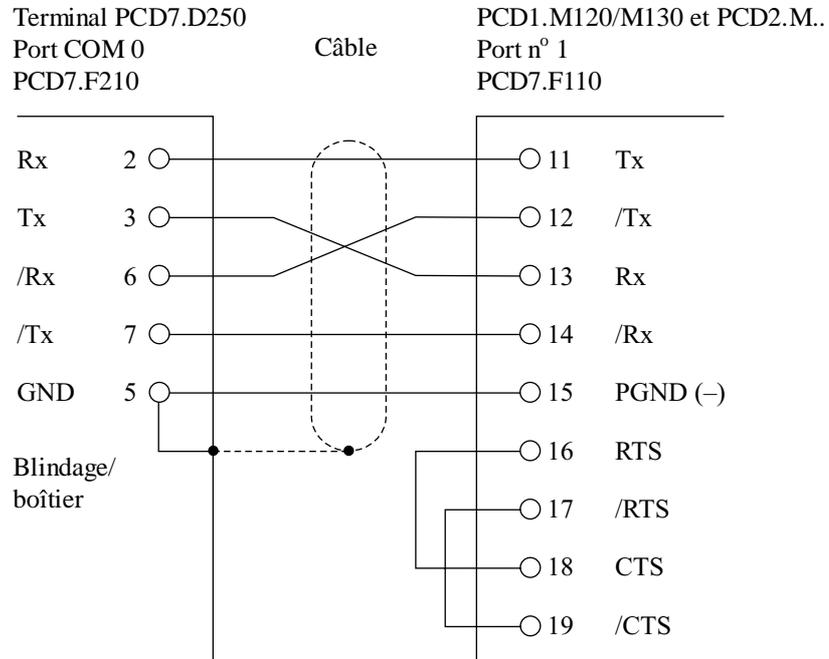
4.3.2 Liaison avec protocole de contrôle RTS/CTS

Le port correspondant du PCD doit être configuré en mode MC1.



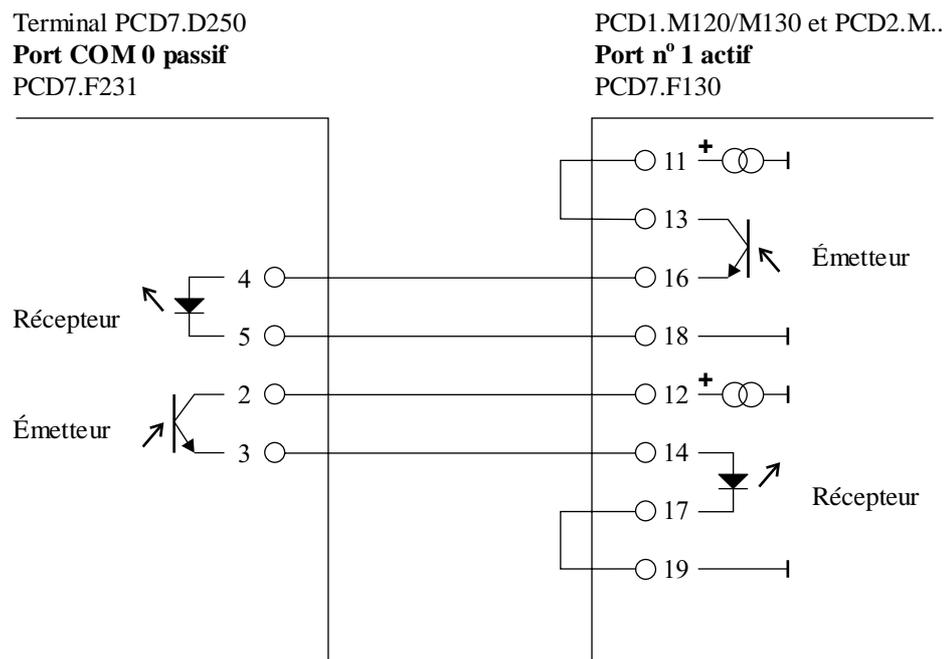
4.4 Interface série COM 0 : équipé avec module de communication PCD7.F2..

4.4.1 RS 422 avec module de communication PCD7.F210



Cavalier J1 ouvert : mode de transmission MC0 ou MC2

4.4.2 Boucle de courant 20 mA (TTY) avec module de communication PCD7.F231



Mode de transmission MC2 (XON/XOFF) : 9600 bauds maxi

5. Principe de fonctionnement

5.1 Tests à la mise sous tension

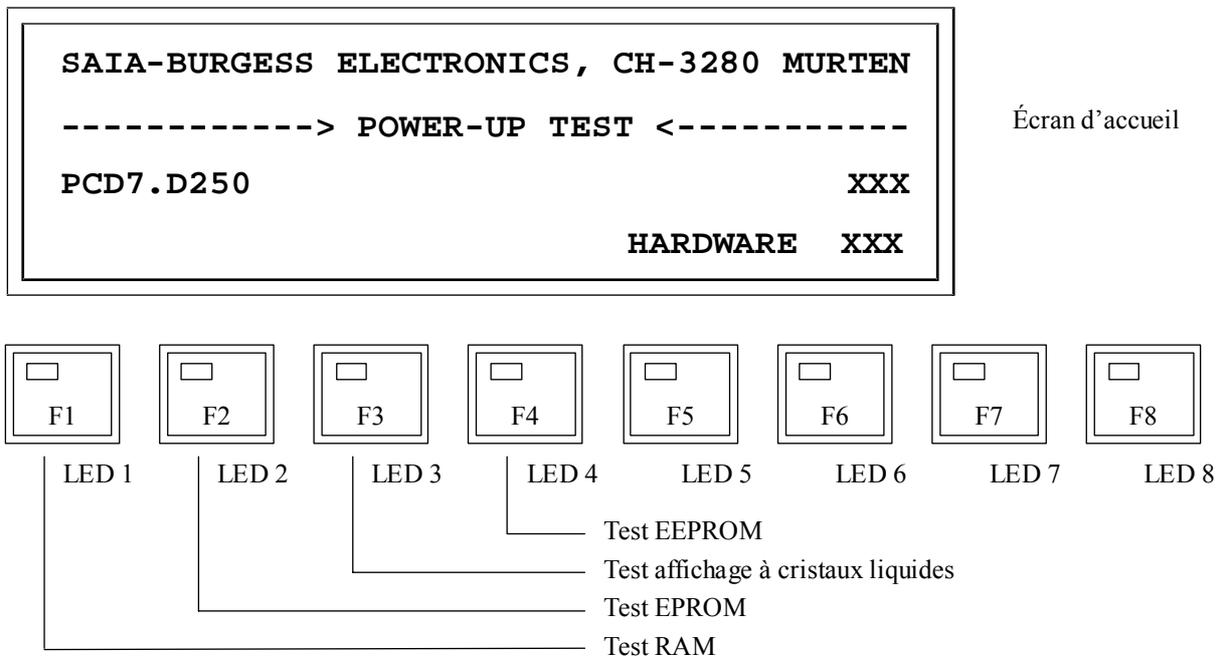
Au démarrage, le PCD7.D250 affiche :

```

SAIA-BURGESS ELECTRONICS, CH-3280 MURTEN
-----> POWER-UP TEST <-----
PCD7.D250                                XXX
                                           HARDWARE XXX
  
```

Il exécute ensuite une séquence d'autotests d'environ 3 secondes, pendant laquelle aucune commande ne doit lui être envoyée par le programme utilisateur sous peine d'être ignorée. Le programme utilisateur peut utiliser la commande « POLL » (voir § 6.5) pour sonder le ..D250 et savoir à quel moment il est prêt, ou se contenter de différer légèrement la prise en compte des commandes utilisateur.

La visualisation du déroulement des tests s'effectue avec les LED. Chaque début de test est repéré par l'allumage de l'une des LED :



La réussite du test initial, qui est celui du microprocesseur, se manifeste par le bref enclenchement des huit LED.

Les huit LED allumées = Test microprocesseur. Si toutes les LED restent allumées à feu fixe, le microprocesseur est en défaut.

5.2 Le Clavier

Le PCD7.D250 est doté d'un clavier à membrane, compatible avec celui du ..D202.



Table de correspondances des codes clavier :

Touche	Décimal	Hexa	ASCII	Signification
F1	65	41	'A'	
F2	66	42	'B'	
F3	67	43	'C'	
F4	68	44	'D'	
F5	69	45	'E'	
F6	70	46	'F'	
F7	71	47	'G'	
F8	72	48	'H'	
0	48	30	'0'	
1	49	31	'1'	
2	50	32	'2'	
3	51	33	'3'	
4	52	34	'4'	
5	53	35	'5'	
6	54	36	'6'	
7	55	37	'7'	
8	56	38	'8'	
9	57	39	'9'	
+	43	2B	'+'	
-	45	2D	'-'	Shift + '+'
.	46	2E	'.'	
,	44	2C	','	Shift + '.'

Touche	Décimal	Hexa	ASCII	Signification
i	105	69	'i'	Appel de l'aide en ligne
Quit	113	71	'q'	Sortie
Shift	-	-	-	Invalide si utilisée seule
Esc	27	1B	ESC	Echappement
Enter	13	0D	CR	Retour-chariot
↑	11	0B	VT	Flèche vers le haut
↓	5	05	ENQ	Flèche vers le bas
←	8	08	BS	Flèche vers la gauche
→	6	06	ACK	Flèche vers la droite
Shift + F1	119	77	'w'	} Modification de l'état des touches de fonction
Shift + F2	120	78	'x'	
Shift + F3	121	79	'y'	
Shift + F4	122	7A	'z'	
Shift + F5	115	73	's'	
Shift + F6	116	74	't'	
Shift + F7	117	75	'u'	
Shift + F8	118	76	'v'	
Shift + 0	97	61	'a'	} Modification de l'état des touches du pavé numérique
Shift + 1	98	62	'b'	
Shift + 2	99	63	'c'	
Shift + 3	100	64	'd'	
Shift + 4	101	65	'e'	
Shift + 5	102	66	'f'	
Shift + 6	103	67	'g'	
Shift + 7	104	68	'h'	
Shift + 8	106	6A	'j'	
Shift + 9	107	6B	'k'	
Shift + i	-	-	-	Validation du mode Paramétrage/Test
Shift + Quit à Shift + →				} Codes identiques sans touche « Shift »

5.3 Mode Paramétrage / Test (Setup/Test mode)

Pour valider le mode Paramétrage/Test → **Shift + i**. Le mode Paramétrage/Test est accessible, quel que soit l'état du ..D250, en ligne ou hors ligne. Il importe toutefois de préciser que sa validation interdit toute prise en compte des données reçues du système hôte.



- ① Faites défiler le menu Paramétrage/Test vers le haut/bas à l'aide des flèches ↑ ↓ pour choisir une option.
- ② Appuyez sur '**Enter**' pour valider votre choix et lancer la fonction correspondante.
- ③ Appuyez sur '**Quit**' ou '**Esc**' pour quitter Paramétrage/Test.Setup mode

Option	Fonction	Voir
Setup mode	Paramétrage du ..D250	§ 5.3.1
Default setup	Rappel de la configuration par défaut	§ 5.3.2
Demo display	Affichage de l'écran de démonstration	§ 5.3.3
Hardware tests	Déroulement en continu des tests matériel	§ 5.3.4
Display test	Test de l'affichage	§ 5.3.5
Keyboard test	Test des fonctionnalités du clavier	§ 5.3.6
LED test	Test des LED	§ 5.3.7

Durant le Setup/Test mode, la LED n° 8 clignote.



Important ! Les données envoyées par le système hôte au ..D250 durant le passage au mode Paramétrage/Test validé risquent d'être perdues et d'entraîner des erreurs d'affichage lors du retour à la normal.

5.3.1 Paramétrage (Setup mode)

Le mode Paramétrage permet de consulter et de modifier les paramètres de configuration du ..D250, stockés en mémoire permanente EEPROM. Un écran d'aide apparaît :

```

SETUP MODE
↑or↓ scrolls menu
←or→ changes data
Ent accepts, Esc aborts
  
```

Appuyez sur une touche pour afficher le premier paramètre à modifier :

```

SETUP MODE
Baudrate:
9600
  
```

Utilisez les touches de défilement vertical ↑ ↓ pour passer d'un paramètre à l'autre (vitesse de transmission baudrate, nombre de bits par caractère data bits, parité parity, nombre de bits d'arrêt stop bits, protocole de contrôle handshaking, etc...).

Utilisez les touches de défilement horizontal ← → pour modifier le paramétrage.

Exemple :	Appuyez sur → pour sélectionner une vitesse de transmission de 110 à 19 200 bit/s.
Baudrate	110, 150, 300, 600, 1200, 2400, 4800, [9600], 19200
Data bits	[8], 7
Parity	[Even], Odd, None, Low
Stop bits	[1], 2
Handshaking	[None], RTS/CTS, XON/XOFF
Echo key to display	[No], Yes
Page/scroll mode	[Page], Scroll
Auto line feed	[No], Yes
Key auto-repeat	[No], All keys, All keys 2 speed, Arrow keys, Arrow keys 2 speed.
Character set	[CodePage 437], D100 compatible
Backlight	[On], Off
Contrast	0...15 [7]
Display mode	[8 x 40], 4 x 20
Serial port	[COM 1 (RS 232)], COM 0 (PCD7.F2..)

[] La configuration par défaut du menu Setup, est défini par les paramètres entre crochets.

Appuyez sur **'Enter'** pour valider vos choix: les nouveaux paramètres sont alors mémorisés en EEPROM. En cas d'erreur, appuyez sur **'Quit'** ou **'Esc'** pour annuler la saisie et rétablir la configuration par défaut. Tous ces paramètres (à l'exception de la vitesse de transmission, du nombre de bits par caractère, de la parité, du nombre de bits d'arrêt et du protocole de contrôle) peuvent également être configurés par le système hôte, sous forme de séquence d'échappement envoyée au ..D250 (voir § 6.1).

Format de transmission

Il se définit par la vitesse de transmission, le nombre de bits par caractère, la parité et le nombre de bits d'arrêt :

- 1 bit de start
- 7 ou 8 bits par caractère
- 1 bit de parité (ou parité nulle)
- 1 ou 2 bits d'arrêt

Le mode Paramétrage ne propose pas d'option « Parité à l'état haut » (Parity = High). Il est néanmoins possible de forcer ce paramètre en choisissant une parité nulle (Parity = None) et 2 bits d'arrêt (Stop bits = 2), ce qui équivaut à une parité à l'état haut avec 1 bit d'arrêt.

Protocole de contrôle RTS/CTS ou XON/XOFF (Handshaking) [None]

Le handshaking est une séquence de « prise de contact » entre le système hôte et le ..D250, qui permet à ce dernier d'indiquer s'il est prêt à recevoir et à traiter les commandes et données de la ligne série. Précisons que le ..D250 est normalement capable de traiter en entrée des données à très haut débit et dispose, en outre, d'un tampon de réception de 512 caractères, ce qui le dispense de protocole de contrôle.

Celui-ci est toutefois obligatoire pour des transmissions à 19 200 bit/s et par le boucle de courant 20 mA.

On distingue deux protocoles :

- Contrôle matériel RTS/CTS

La séquence de prise de contact hôte-terminal repose sur deux signaux : « Demande pour émettre » (RTS) et « Prêt à émettre » (CTS). Le système hôte s'abstient de transmettre au ..D250 tant que le signal CTS du ..D250 est à 0. Inversement, le ..D250 suspend la transmission des données clavier tant que le CTS de l'hôte est à 0; les données sont alors placées en mémoire tampon jusqu'à ce que CTS passe à 1.

Pour mettre en œuvre ce protocole :

- Le brochage des signaux RTS/CTS de l'hôte et du terminal doit être conforme au schéma du § 4.3.2;
- Le SAIA[®] PCD doit être configuré en mode MC1.

– Contrôle logiciel XON/XOFF

La séquence de prise de contact hôte-terminal repose sur l'envoi des caractères XOFF (17 décimal, 11 hexa) et XON (19 décimal, 13 hexa) pour interrompre et reprendre respectivement la transmission.

Pour mettre en oeuvre ce protocole :

- Les broches CTS et RTS (n° 7 et 8), côté ..D250, doivent être reliées (voir § 4.3.1) ;
- Le SAIA® PCD doit être configuré en mode MC2.

Echo (Echo key to display) [No]

Lorsque le mode « Echo » est inhibé (Echo key to display = No), le code ASCII des caractères saisis au clavier est directement envoyé au système hôte, sans affichage. Il faut que le programme utilisateur de l'hôte renvoie en écho les caractères reçus au ..D250 pour permettre l'affichage. Lorsque le mode «Echo» est validé (Echo key to display = Yes), les caractères saisis au clavier sont transmis à l'hôte et automatiquement affichés à l'écran, à la position actuelle du curseur. Le connecteur 9 broches doit être branché sur le port de COM défini par le Setup.

Mode Page/Ligne (Page/scroll mode) [Page]

Sur réception d'un caractère de saut de ligne <LF> :

- En mode Page (Page), le curseur saute de la dernière ligne à la première ligne de l'écran, sans modification de l'affichage.
- En mode Ligne (Scroll), si le curseur est sur la dernière ligne de l'écran, l'affichage remonte d'une ligne et le curseur passe sur la dernière ligne (devenue vide) en restant à la même position. Si le saut de ligne est dû à un retour-chariot <CR>, en mode « Saut de ligne automatique » (Auto line feed = Yes), le curseur revient également au début de la ligne.

Saut de ligne automatique (Auto line feed) [No]

Sur réception d'un caractère de retour-chariot (13 décimal, 0D hexa) :

- Le curseur se place en début de ligne;
- La validation du mode « Saut de ligne automatique » (Auto line feed = Yes) positionne le curseur automatiquement sur la ligne suivante. Si le curseur est sur la dernière ligne de l'écran, il repasse sur la première ligne (mode Page) ou l'affichage se décale d'une ligne vers le haut (mode Ligne). Pour obtenir le même résultat lorsque le mode «Saut de ligne automatique» est inhibé (Auto line feed = No), il faut envoyer un caractère de saut de ligne (10 décimal, 0A hexa) après le retour-chariot.

Répétition automatique (Key auto-repeat) [No]

Les touches du clavier ..D250 peuvent être à « répétition automatique » : il suffit de maintenir l'une d'elles enfoncée pendant plus de 0.7 s pour répéter le caractère saisi, à la vitesse de 8 caractères par seconde.

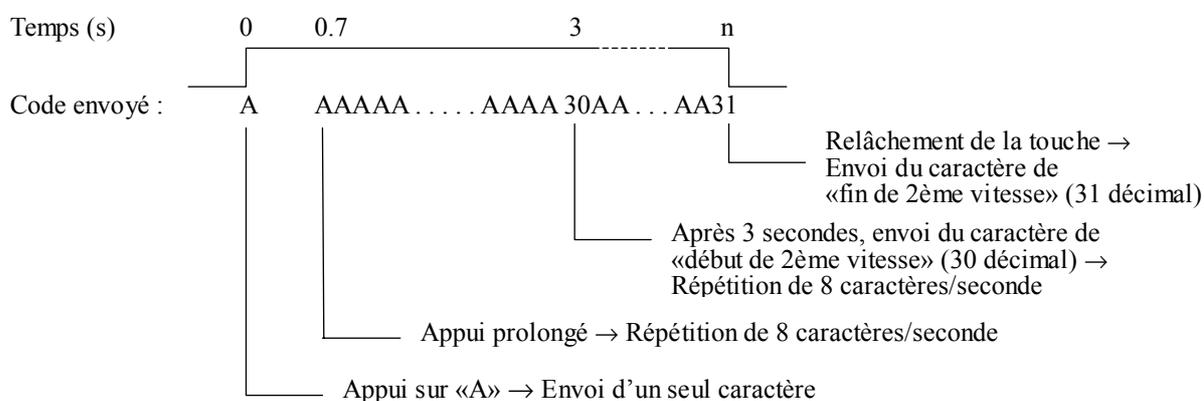
No	Pas de répétition (par défaut)
All keys	Répétition de toutes les touches
All keys "2 speed"	Répétition de toutes les touches, 2ème vitesse (voir ci-dessous)
Arrow keys	Répétition des touches de déplacement du curseur
Arrow keys "2 speed"	Répétition des touches de déplacement du curseur, 2ème vitesse (voir ci-dessous)

Rappelons qu'en « répétition automatique », le code correspondant à la touche enfoncée durant 0.7 seconde est répété 8 fois par seconde.

Le principe du mode double vitesse permet au système-hôte de changer de mode d'acquisition si une touche ou seulement les touches de déplacement curseur sont pressées pendant longtemps. Après avoir maintenu la touche enfoncée durant 3 secondes, un caractère spécial (30 décimal) est envoyé à l'hôte pour lui signaler le « début de 2ème vitesse »; ce caractère est suivi d'autres codes de touche, répétés au même rythme de 8 caractères par seconde. Un caractère de « fin de 2ème vitesse » (31 décimal) est envoyé à l'hôte dès que l'on relâche la touche.

Par exemple, sur réception du code correspondant aux touches ↑ ↓, le programme de l'hôte incrémente ou décrémente de 1 une valeur. S'il reçoit le caractère de « début de 2ème vitesse », il incrémente ou décrémente de 10 la même valeur, à chaque code ↑ ↓ reçu supplémentaire; puis il s'arrête à la réception du caractère de « fin de 2ème vitesse » (ou de tout caractère autre que le code clavier ↑ ↓). L'exemple suivant illustre également ce principe.

Appui sur la touche « A », en mode Répétition automatique, Option « All keys, 2 speed » :



Mode caractère 8 x 40 (4 x 20 off) ou 4 x 20 [8 x 40]

Dans le Setup vous pouvez choisir soit 8 x 40 ou 4 x 20 caractères. La superposition des deux modes est possible en envoyant la commande « transparence » par la ligne sérielle (voir § 6.2).

Jeu de caractères (Character set) [CodePage 437]

En mode 8 x 40 ou 4 x 20, deux jeux de caractères sont disponibles. Chaque jeu possède les mêmes caractères de 32 à 127 déc. par contre les caractères spéciaux 128 à 255 seront différents (voir tableaux dans chapitre 7).

Rétroéclairage (Backlight) [On]

Le rétroéclairage par LED, activé par défaut pour garantir une qualité d'affichage optimale, peut être désactivé, selon le besoin, ou désactivé puis réactivé (par le biais de séquences d'échappement reçues du système hôte) pour obtenir, par exemple, un affichage clignotant de signalisation d'alarme ..D250 (voir § 6.3).

Contraste (Contrast) [7]

Le contraste est réglable sur une échelle de 0 à 15 (0 à F hexa), du plus clair au plus foncé.

Canaux de communications [COM1]

COM 1 est installé de manière fixe en RS 232. Alternativement, COM 0 peut être activé en l'équipant d'un module PCD7.F2.. Il n'est pas possible d'utiliser les deux canaux en même temps.

5.3.2 Configuration par défaut

La configuration par défaut, stockée en EEPROM, regroupe tous les réglages usine du ..D250 :

Vitesse de transmission	9600
Nombre de bits par caractère	8
Parité	Paire
Nombre de bits d'arrêt	1
Echo	Non
Protocole de contrôle	Aucun
Mode Page/Ligne	Page
Saut de ligne automatique	Non
Répétition automatique	Non
Jeu de caractères	CodePage 437
Rétroéclairage	Activé
Réglage du contraste	7 (moyen)
Mode d'affichage	8 x 40
Port sériel	COM 1 (RS 232)

5.3.3 Ecran de démonstration

(PCD7.D250 non connecté au système hôte)

Il est utilisé lors de foire, pour pouvoir présenter le terminal s'il n'est pas connecté au PCD. Cet affichage montre quelques informations. Pour quitter la démonstration → **Shift + F4**.

```
*** PCD7.D250 ***  
INDUSTRIAL TERMINAL  
SAIA-BURGESS Ltd  
CH-3280 MURTEN
```

5.3.4 Tests matériel

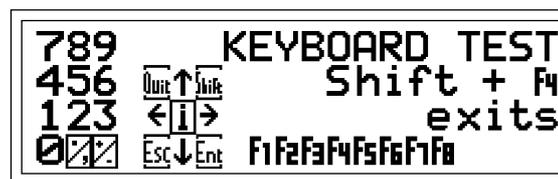
Il s'agit d'exécuter en continu les tests de mise sous tension pour détecter tout défaut intermittent du ..D250 en exploitation. Ces tests se déroulent jusqu'à l'apparition d'une erreur et l'affichage du message correspondant; le ..D250 est ensuite relancé par le chien de garde, et les tests reprennent. Pour arrêter les tests → Mettre le ..D250 **hors tension**, puis de nouveau **sous tension**.

5.3.5 Test de l'affichage

Ce test permet de tester l'ensemble des fonctionnalités de l'écran à cristaux liquides, ainsi que le jeu de caractères et la RAM interne du contrôleur LCD. Pour arrêter le test → Appuyer **sur une touche**.

5.3.6 Test du clavier

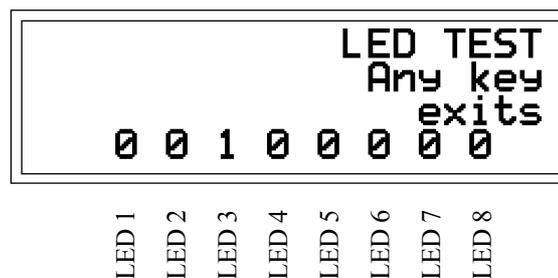
L'affichage montre sur sa gauche et en bas chaque touche (disposé comme sur le clavier). Lorsqu'une touche est activée, le champ correspondant vire au noir.



Pour sortir du test → **Shift + F4**.

5.3.7 Test des LED

Chaque LED s'allume durant 500 ms et provoque l'apparition d'un 1 à l'écran.



Pour sortir du test et revenir au mode Paramétrage → **Appuyer sur une touche**.

Notes personnelles :

6. Commandes via l'interface série

Des caractères de contrôle ou des séquences d'échappement constituées de deux, trois ou quatre caractères sont transmis au ..D250 à l'aide des instructions STXT (transmission série de texte) ou STXD (transmission série de caractère) du SAIA® PCD.



Important !

Certaines séquences d'échappement contiennent le caractère '@'. Ou, si le port de communication du PCD fonctionne en mode C, l'automate interprète ce caractère comme le début d'une chaîne de commande d'adressage indirect.

Pour lever toute ambiguïté, il faut doubler le caractère @ ('@@'), à chaque fois que vous souhaitez l'inclure dans une séquence d'échappement.

6.1 Configuration du terminal

La configuration du ..D250 peut être modifiée par l'envoi de commandes spéciales via l'interface série. Cette configuration reste active jusqu'à la mise hors tension du ..D250; la remise sous tension rétablit alors la configuration définie par le mode Paramétrage (voir § 5.3.1).

Toutes les commandes nécessaires au ..D250 sont regroupées dans un seul texte PCD et lui sont transmises en une seule fois.

Echo key to display

Fonction : Voir § 5.3.1

Commande	ASCII	Décimal	Hexa
Inhibition de l'écho	ESC @ 0	27 64 48	1B 40 30
Validation de l'écho	ESC @ 1	27 64 49	1B 40 31

Mode Page/Ligne

Fonction : Voir § 5.3.1

Commande	ASCII	Décimal	Hexa
Mode Ligne	ESC @ 4	27 64 52	1B 40 34
Mode Page	ESC @ 5	27 64 53	1B 40 35

Saut de ligne automatique après retour-chariot

Fonction : Voir 5.3.1

Commande	ASCII	Décimal	Hexa
Validation du saut de ligne automat.	ESC @ 2	27 64 50	1B 40 32
Inhibition du saut de ligne automat.	ESC @ 3	27 64 51	1B 40 33

Répétition automatique

Fonction : Voir 5.3.1

Commande	ASCII	Décimal	Hexa
Inhibition de la répétition automatique	ESC A	27 65	1B 41
Validation de la répétition automatique			
• Toutes les touches	ESC B	27 66	1B 42
• Touches de déplacement du curseur	ESC C	27 67	1B 43
• Toutes les touches, 2ème vitesse	ESC D	27 68	1B 44
• Touches de déplacement du curseur, 2ème vitesse	ESC E	27 69	1B 45

Modes caractères

Le mode choisi par défaut est 8 x 40 caractères. La commande indiquée dans le tableau ci-dessous vous permet de passer en 4 x 20 caractères ; toutefois, cela affiche une nouvelle page écran.

Les deux pages restent sauvegardées dans le terminal, dans les deux modes. Elles peuvent se superposer à l'écran avec la commande « Mode transparence, activation » (Cf. exemple de la page suivante).

Commande	ASCII	Décimal	Hexa
Mode caractères 8 x 40 (défaut)	ESC @ M0	27 64 77 48	1B 40 4D 30
Mode caractères 4 x 20	ESC @ M1	27 64 77 49	1B 40 4D 31

Jeux de caractères

Deux jeux sont disponibles (Cf. chapitre 7).

Commande	ASCII	Décimal	Hexa
CodePage 437 (défaut)	ESC @ J	27 64 74	1B 40 4A
Compatible D100	ESC @ F	27 64 70	1B 40 46

Mode transparence

Ce mode permet de superposer deux pages écran et d'afficher en même temps les deux tailles de caractères (8 x 40 et 4 x 20).

4 x 20 characters
8 x 40 characters

Commande	ASCII	Décimal	Hexa
Mode transparence			
• activation	ESC @ M5	27 64 77 53	1B 40 4D 35
• désactivation	ESC @ M4	27 64 77 52	1B 40 4D 34

Pour plus d'information sur l'inversion de l'affichage, reportez-vous au paragraphe 6.3.

Rétroéclairage et contraste

Fonction : Voir 6.3.

6.2 Commandes du curseur

Déplacement du curseur vers le haut/bas, à gauche/droite

Fonction : Commande à un seul caractère (commune au mode Page et au mode Ligne) permettant de déplacer le curseur d'une position vers le haut, vers le bas, à gauche ou à droite.

La sortie du curseur de l'écran provoque un « rebouclage automatique » :

Commande	ASCII	Décimal	Hexa
Curseur vers le haut	CTRL+K	11	0B
Curseur vers le bas	CTRL+E	5	05
Curseur à gauche	CTRL+H	8	08
Curseur à droite	CTRL+F	6	06

Positionnement du curseur

Fonction : Le positionnement du curseur nécessite deux paramètres :

- L'adresse du curseur proprement dit (16 décimal ou 10 hexa);
- L'adresse des coordonnées X et Y qui définissent sa position à l'écran. Précisons qu'un décalage de 32 décimal (20 hexa) est effectué sur ces deux adresses. En cas d'erreur d'adressage, le curseur reste sur place.
- La plage d'adresse pour le mode 8 x 40 ou 4 x 20 se présente différemment :

Code Y Code	Code X										Code X										...		Code X					
	<32>	<33>	<34>	<35>	<36>	<37>	<38>	<39>	<40>	<41>	<42>	<43>	<44>	<45>	<46>	<47>	<48>	<49>	<50>	<51>	<52>	<53>	<54>	<55>	<56>	...	<70>	<71>
1	X																									...	39	40
<32>	1	X																								...		
<33>	2			X																						...		
<34>	3																									...		
<35>	4														X											...		
<36>	5																									...		
<37>	6																									...	X	
<38>	7																									...		
<39>	8																									...		

Rappel du curseur en position d'origine

Fonction : Commande à un seul caractère permettant de ramener le curseur à sa position d'origine, c'est-à-dire au début de la première ligne de l'écran.

Commande	ASCII	Décimal	Hexa
Position d'origine	CTRL+Z	26	1A

Validation/Inhibition du curseur

Fonction : Commande de deux caractères permettant d'activer ou de désactiver le curseur.

Commande	ASCII	Décimal	Hexa
Activation curseur	ESC W	27 87	1B 57
Cursor off	ESC T	27 84	1B 54

Saut de ligne

Fonction : Décale le curseur d'une ligne vers le bas. Si le curseur est sur la dernière ligne :

- en mode Ligne, l'écran défile vers le haut;
- en mode Page, le curseur se place sur la première ligne de l'écran, sans changer de position.

Commande	ASCII	Décimal	Hexa
Saut de ligne	LF	10	0A

Retour-chariot

Fonction : Renvoie le curseur en début de ligne. Si le « Saut de ligne automatique après retour-chariot » est validé, un saut de ligne est également effectué comme indiqué au paragraphe précédent.

Commande	ASCII	Décimal	Hexa
Retour-chariot	CR	13	0D

Suppression de caractère

Fonction : Efface le caractère situé à gauche du curseur (remplacé par un espace) et déplace le curseur d'une position sur la gauche. Si le curseur est en début de ligne, il repasse en fin de ligne précédente. Cette commande est sans effet si le curseur est en position d'origine.

Commande	ASCII	Décimal	Hexa
Suppression de caractère	DEL	127	7F

6.3 Commandes de l'affichage

Effacement

Fonction : Efface la totalité de l'affichage et ramène le curseur en position d'origine.

Commande	ASCII	Décimal	Hexa
Effacement de l'écran	CTRL+L	12	0C

Sauvegarde/Restauration

Fonction : Sauvegarde, puis restaure l'affichage, la position et l'état du curseur (activé/désactivé). On distingue 10 zones de sauvegarde/restauration, numérotées 0 à 9.

Notons que la restauration d'une zone non sauvegardée au préalable donne un affichage vide avec le curseur enclenché en haut à gauche. Le contenu d'un affichage sauvegardé est perdu lorsque le ..D250 est éteint.

Commande	ASCII	Décimal	Hexa
Sauvegarde affichage 'n'	ESC @ S n	27 64 83 n	1B 40 53 n
Restauration affichage 'n'	ESC @ R n	27 64 82 n	1B 40 52 n
pour 'n' =	0...9	48...57	30...39

Activation/Désactivation du rétroéclairage

Fonction : Voir § 5.3.1

Commande	ASCII	Décimal	Hexa
Désactivation du rétroéclairage	ESC O	27 79	1B 4F
Activation du rétroéclairage	ESC L	27 76	1B 4C

Réglage du contraste

Fonction : Voir § 5.3.1

Commande	ASCII	Décimal	Hexa
Contraste maxi	ESC @ D 0	27 64 68 48	1B 40 44 30
Contraste moyen	ESC @ D 7	27 64 68 55	1B 40 44 37
Contraste mini	ESC @ D F	27 64 68 70	1B 40 44 46

Inversion de l'affichage des caractères

Par défaut, les caractères apparaissent en noir sur fond clair. Les commandes suivantes permettent d'inverser ce mode d'affichage (sur le texte en cours), puis d'annuler l'inversion.



Autre possibilité :

Affichage de gros caractères (4 x 20) en vidéo normale et de gros caractères en vidéo inverse superposés à de petits caractères (8 x 40) en vidéo normale, par la commande « Mode transparence », décrite au paragraphe 6.1.



Commande	ASCII	Décimal	Hexa
Caractères normaux	ESC @ N 0	27 64 78 48	1B 40 4E 30
Caractères inversés	ESC @ N 1	27 64 78 49	1B 40 4E 31

Limitation ! Si vous êtes en « Mode transparence », la commande d'inversion de l'affichage des caractères 8 x 40 ne peut être activée.

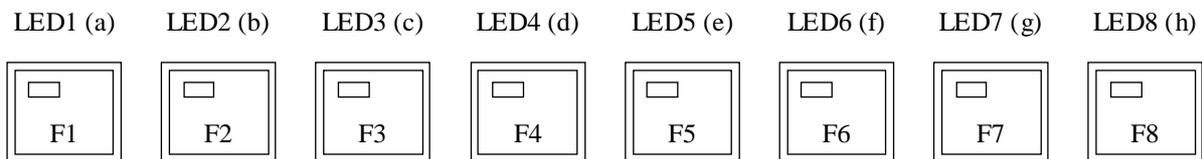
6.4 Commandes des LED

Le ..D250 est équipé de 8 LED de visualisation d'état, numérotées 1 à 8. Chaque LED peut être allumée ou éteinte par l'envoi d'une séquence d'échappement de 4 caractères.

Les minuscules 'a' à 'h' permettent de différencier les LED :

- 'a' = LED n° 1,
- 'b' = LED n° 2,
- et ainsi de suite jusqu'à
- 'h' = LED n° 8.

Un '1' ou un '0', en fin de séquence, commande l'allumage ou l'extinction de la LED correspondante.



Commande	ASCII	Décimal	Hexa
Allumage LED n° 1	ESC @ a 1	27 64 97 49	1B 40 61 31
Extinction LED n° 1	ESC @ a 0	27 64 97 48	1B 40 61 30
Allumage LED n° 2	ESC @ b 1	27 64 98 49	1B 40 62 31
Extinction LED n° 2	ESC @ b 0	27 64 98 48	1B 40 62 30
Allumage LED n° 3	ESC @ c 1	27 64 99 49	1B 40 63 31
Extinction LED n° 3	ESC @ c 0	27 64 99 48	1B 40 63 30
Allumage LED n° 4	ESC @ d 1	27 64 100 49	1B 40 64 31
Extinction LED n° 4	ESC @ d 0	27 64 100 48	1B 40 64 30
Allumage LED n° 5	ESC @ e 1	27 64 101 49	1B 40 65 31
Extinction LED n° 5	ESC @ e 0	27 64 101 48	1B 40 65 30
Allumage LED n° 6	ESC @ f 1	27 64 102 49	1B 40 66 31
Extinction LED n° 6	ESC @ f 0	27 64 102 48	1B 40 66 30
Allumage LED n° 7	ESC @ g 1	27 64 103 49	1B 40 67 31
Extinction LED n° 7	ESC @ g 0	27 64 103 48	1B 40 67 30
Allumage LED n° 8	ESC @ h 1	27 64 104 49	1B 40 68 31
Extinction LED n° 8	ESC @ h 0	27 64 104 48	1B 40 68 30

6.5 Autres commandes

Verrouillage/déverrouillage du clavier

Fonction : Interdit ou autorise la prise en compte de la frappe au clavier.

Commande	ASCII	Décimal	Hexa
Verrouillage du clavier	ESC N	27 78	1B 4E
Déverrouillage du clavier	ESC Q	27 81	1B 51

Démarrage à chaud/froid

Fonction : « Démarrage à chaud »: réinitialise le ..D250 et rappelle la configuration utilisateur. Cette commande a le même effet qu'une l'initialisation du système à la mise sous tension.

« Démarrage à froid »: rappelle la configuration usine par défaut (voir § 5.3.2).

Commande	ASCII	Décimal	Hexa
Démarrage à chaud	ESC H	27 72	1B 48
Démarrage à froid	ESC @ G	27 64 71	1B 40 47

Validation/Inhibition du mode Paramétrage/Test

Fonction : Interdit toute modification non autorisée de la configuration du ..D250 par inhibition du mode Paramétrage/Test via la séquence d'échappement 'ESC @ H', qui invalide la combinaison de touches 'Shift+i'.

Outre la séquence d'échappement 'ESC @ I', une mise hors tension, puis sous tension du ..D250, ou encore l'envoi d'une commande de « Démarrage à chaud/froid » permet de revalider le mode Paramétrage/Test.

Commande	ASCII	Décimal	Hexa
Inhibition du mode Paramétrage/Test	ESC @ H	27 64 72	1B 40 48
Validation du mode Paramétrage/Test	ESC @ I	27 64 73	1B 40 49

Ecran de démonstration et tests matériel

Fonction : Voir § 5.3.3 à 5.3.7. La commande de scrutation « POLL » permet au système hôte de reconnaître la fin des tests.

Commande	ASCII	Décimal	Hexa
Ecran de démonstration	ESC J	27 74	1B 4A
Test de l'affichage	ESC @ A	27 64 65	1B 40 41
Test du clavier	ESC @ 9	27 64 57	1B 40 39
Test des LED	ESC @ L	27 64 76	1B 40 4C
Tests matériel	ESC @ C	27 64 67	1B 40 43

Commande de scrutation (POLL)

Le programme utilisateur du PCD utilise la commande « Poll » pour savoir si le ..D250 est connecté et prêt à recevoir des commandes. Dans l'affirmative, le ..D250 renvoie le message '**SOH**' (1 décimal, 01 hexa) ; dans la négative, aucune réponse n'est envoyée.

La commande « Poll » sert principalement à reconnaître la fin des auto-tests exécutés à la mise sous tension. Elle est, en fait, la seule à générer une réponse du ..D250. Elle permet également de savoir si le ..D250 est toujours opérationnel. En l'absence de réponse du ..D250, le programme utilisateur en informe automatiquement l'opérateur.

Commande	ASCII	Décimal	Hexa
POLL	ESC @ B	27 64 66	1B 40 42
Réponse du ..D250 si opérationnel	SOH	1	01

6.6 Récapitulatif des commandes

Commande	ASCII	Décimal	Hexa
Configuration :			
Inhibition de l'écho	ESC @ 0	27 64 48	1B 40 30
Validation de l'écho	ESC @ 1	27 64 49	1B 40 31
Saut de ligne automatique			
• validation	ESC @ 2	27 64 50	1B 40 32
• inhibition	ESC @ 3	27 64 51	1B 40 33
Mode Ligne	ESC @ 4	27 64 52	1B 40 34
Mode Page	ESC @ 5	27 64 53	1B 40 35
Mode caractères 8 x 40	ESC @ M0	27 64 77 48	1B 40 4D 30
Mode caractères 4 x 20	ESC @ M1	27 64 77 49	1B 40 4D 31
Mode transparence			
• désactivation	ESC @ M4	27 64 77 52	1B 40 4D 34
• activation	ESC @ M5	27 64 77 53	1B 40 4D 35
CodePage 437	ESC @ J	27 64 74	1B 40 4A
Compatible D100	ESC @ F	27 64 70	1B 40 46
Inhibition de la répétition automatique	ESC A	27 65	1B 41
Validation de la répétition automatique			
• toutes les touches	ESC B	27 66	1B 42
• touches de déplacement du curseur	ESC C	27 67	1B 43
• toutes les touches, 2 vitesses	ESC D	27 68	1B 44
• touches de déplacement du curseur, 2 vitesses	ESC E	27 69	1B 45
Curseur :			
Curseur vers le haut	CTRL+K	11	0B
Curseur vers le bas	CTRL+E	5	05
Curseur à gauche	CTRL+H	8	08
Curseur à droite	CTRL+F	6	06
Curseur en position d'origine	CTRL+Z	26	1A
Activation du curseur	ESC W	27 87	1B 57
Désactivation du curseur	ESC T	27 84	1B 54
Saut de ligne	LF	10	0A
Retour-chariot	CR	13	0D
Suppression de caractère	DEL	127	7F
Positionnement du curseur	ASCII	CTRL+P ` `+X ` `+Y	
	Décimal	16 32+X 32+Y	
	Hexa	10 20+X 20+Y	

Rappel : Pour obtenir le caractère '@' dans chaque texte PCD, saisir obligatoirement '@@' !

Commande	ASCII	Décimal	Hexa
Affichage :			
Effacement	CTRL+L	12	0C
Sauvegarde de la zone 'n'	ESC @ S n	27 64 83 n	1B 40 53 n
Restauration de la zone 'n'	ESC @ R n	27 64 82 n	1B 40 52 n
Désactivation du rétroéclairage	ESC O	27 79	1B 4F
Activation du rétroéclairage	ESC L	27 76	1B 4C
Réglage du contraste			
• maxi	ESC @ D 0	27 64 68 48	1B 40 44 30
• moyen	ESC @ D 7	27 64 68 55	1B 40 44 37
• mini	ESC @ D F	27 64 68 70	1B 40 44 46
Caractères normaux	ESC @ N 0	27 64 78 48	1B 40 4E 30
Caractères inversés	ESC @ N 1	27 64 78 49	1B 40 4E 31
LED :			
Allumage LED n° 1	ESC @ a 1	27 64 97 49	1B 40 61 31
Extinction LED n° 1	ESC @ a 0	27 64 97 48	1B 40 61 30
(Autres LED :	'b'=2, 'c'=3, 'd'=4, 'e'=5, 'f'=6, 'g'=7, 'h'=8)		
Divers :			
Verrouillage du clavier	ESC N	27 78	1B 4E
Déverrouillage du clavier	ESC Q	27 81	1B 51
Démarrage à chaud	ESC H	27 72	1B 48
Démarrage à froid	ESC @ G	27 64 71	1B 40 47
Inhibition du mode Paramétrage/Test	ESC @ H	27 64 72	1B 40 48
Validation du mode Paramétrage/Test	ESC @ I	27 64 73	1B 40 49
Ecran de démonstration	ESC J	27 74	1B 4A
Test de l'affichage	ESC @ A	27 64 65	1B 40 41
Test du clavier	ESC @ 9	27 64 57	1B 40 39
Test des LED	ESC @ L	27 64 76	1B 40 4C
Tests matériel	ESC @ C	27 64 67	1B 40 43
Commande de scrutation POLL	ESC @ B	27 64 66	1B 40 42
Réponse D250 avec	SOH	1	10

Rappel : Pour obtenir le caractère '@' dans chaque texte PCD, saisir obligatoirement '@@' !

7. Jeux de caractères

Pour conserver la compatibilité avec les terminaux moins récents, le ..D250 accepte le jeu de caractères du ..D100.

7.1 Code ASCII standard (32 à 127 décimal, 20 à 7F hexa)

Les caractères de cette table sont les mêmes pour les deux jeux.

| Déc Hex ASC |
|-------------|-------------|-------------|-------------|-------------|-------------|
| 32 20 SP | 48 30 0 | 64 40 @ | 80 50 P | 96 60 ` | 112 70 p |
| 33 21 ! | 49 31 1 | 65 41 A | 81 51 Q | 97 61 a | 113 71 q |
| 34 22 " | 50 32 2 | 66 42 B | 82 52 R | 98 62 b | 114 72 r |
| 35 23 # | 51 33 3 | 67 43 C | 83 53 S | 99 63 c | 115 73 s |
| 36 24 \$ | 52 34 4 | 68 44 D | 84 54 T | 100 64 d | 116 74 t |
| 37 25 % | 53 35 5 | 69 45 E | 85 55 U | 101 65 e | 117 75 u |
| 38 26 & | 54 36 6 | 70 46 F | 86 56 V | 102 66 f | 118 76 v |
| 39 27 ' | 55 37 7 | 71 47 G | 87 57 W | 103 67 g | 119 77 w |
| 40 28 (| 56 38 8 | 72 48 H | 88 58 X | 104 68 h | 120 78 x |
| 41 29) | 57 39 9 | 73 49 I | 89 59 Y | 105 69 i | 121 79 y |
| 42 2A * | 58 3A : | 74 4A J | 90 5A Z | 106 6A j | 122 7A z |
| 43 2B + | 59 3B ; | 75 4B K | 91 5B [| 107 6B k | 123 7B { |
| 44 2C , | 60 3C < | 76 4C L | 92 5C \ | 108 6C l | 124 7C |
| 45 2D - | 61 3D = | 77 4D M | 93 5D] | 109 6D m | 125 7D } |
| 46 2E . | 62 3E > | 78 4E N | 94 5E ^ | 110 6E n | 126 7E → |
| 47 2F / | 63 3F ? | 79 4F O | 95 5F _ | 111 6F o | 127 7F DEL |

7.2 Code ASCII étendu (128 à 255 décimal, 80 à FF hexa)

Pour la table ASCII étendue, deux jeux de caractères sont disponibles :

- « CodePage 437 », jeu de caractères universel.
- « D100-compatible », compatible avec les terminaux moins récents.

Remarque : Il se peut que la représentation des caractères ASCII étendu sur le ..D250 diffère légèrement de celle sur IBM PC.

7.2.1 CodePage 437 (défaut)

Ce jeu de caractère universel contient les caractères dépendant d'une des langue définit dans les terminaux ..D160/..D170 et ..D202, c. à d. allemand, anglais, français et scandinave.

Déc	Hex	ASC									
128	80	Ç	144	90	É	160	A0	á	176	B0	☐
129	81	ü	145	91	æ	161	A1	í	177	B1	☐
130	82	é	146	92	Æ	162	A2	ó	178	B2	☐
131	83	â	147	93	ô	163	A3	ú	179	B3	
132	84	ä	148	94	ö	164	A4	ñ	180	B4	└
133	85	à	149	95	ò	165	A5	Ñ	181	B5	┘
134	86	â	150	96	û	166	A6	ª	182	B6	┌
135	87	ç	151	97	ù	167	A7	º	183	B7	┐
136	88	ê	152	98	ÿ	168	A8	¿	184	B8	┘
137	89	ë	153	99	Ö	169	A9	←	185	B9	┌
138	8A	è	154	9A	Ü	170	AA	→	186	BA	┐
139	8B	ï	155	9B	φ	171	AB	½	187	BB	┘
140	8C	î	156	9C	£	172	AC	¼	188	BC	┌
141	8D	ì	157	9D	¥	173	AD	¡	189	BD	┐
142	8E	Ä	158	9E	Pts	174	AE	«	190	BE	┘
143	8F	Å	159	9F	f	175	AF	»	191	BF	┌

Déc	Hex	ASC									
192	C0	Ł	208	D0	⌌	224	E0	α	240	F0	≡
193	C1	⊥	209	D1	⌍	225	E1	β	241	F1	±
194	C2	⌒	210	D2	⌎	226	E2	Γ	242	F2	≥
195	C3	└	211	D3	⌏	227	E3	π	243	F3	≤
196	C4	—	212	D4	⌐	228	E4	Σ	244	F4	┌
197	C5	┘	213	D5	⌑	229	E5	σ	245	F5	┘
198	C6	┌	214	D6	⌒	230	E6	μ	246	F6	÷
199	C7	┐	215	D7	⌓	231	E7	τ	247	F7	≈
200	C8	⌌	216	D8	⌔	232	E8	Φ	248	F8	°
201	C9	⌍	217	D9	⌕	233	E9	Θ	249	F9	·
202	CA	⌎	218	DA	⌖	234	EA	Ω	250	FA	·
203	CB	⌏	219	DB	■	235	EB	δ	251	FB	√
204	CC	⌐	220	DC	■	236	EC	∞	252	FC	n
205	CD	=	221	DD	■	237	ED	φ	253	FD	²
206	CE	⌑	222	DE	■	238	EE	ε	254	FE	■
207	CF	⌒	223	DF	■	239	EF	∩	255	FF	

7.2.2 Jeux de caractères compatible ..D100

Déc Hex ASC	Déc Hex ASC	Déc Hex ASC	Déc Hex ASC
128 80	144 90	160 A0	176 B0
129 81	145 91	161 A1 □	177 B1
130 82	146 92	162 A2	178 B2
131 83	147 93	163 A3	179 B3
132 84	148 94	164 A4	180 B4
133 85	149 95	165 A5	181 B5
134 86	150 96	166 A6	182 B6
135 87	151 97	167 A7	183 B7
136 88	152 98	168 A8	184 B8
137 89	153 99	169 A9	185 B9
138 8A	154 9A	170 AA	186 BA
139 8B	155 9B	171 AB	187 BB
140 8C	156 9C	172 AC	188 BC
141 8D	157 9D	173 AD	189 BD
142 8E	158 9E	174 AE Σ	190 BE
143 8F	159 9F	175 AF	191 BF

Déc Hex ASC	Déc Hex ASC	Déc Hex ASC	Déc Hex ASC
192 C0	208 D0	224 E0 α	240 F0
193 C1	209 D1	225 E1 ä	241 F1
194 C2	210 D2	226 E2 ß	242 F2
195 C3	211 D3	227 E3	243 F3
196 C4 Ä	212 D4	228 E4 ä	244 F4 Ω
197 C5 Å	213 D5	229 E5 å	245 F5
198 C6 Æ	214 D6 Ö	230 E6 æ	246 F6 ö
199 C7	215 D7	231 E7	247 F7 π
200 C8	216 D8 Ø	232 E8	248 F8 Ø
201 C9	217 D9	233 E9	249 F9
202 CA	218 DA	234 EA	250 FA
203 CB	219 DB □	235 EB x	251 FB
204 CC	220 DC Ü	236 EC φ	252 FC ü
205 CD	221 DD	237 ED	253 FD
206 CE	222 DE	238 EE	254 FE
207 CF	223 DF □	239 EF Ö	255 FF ■

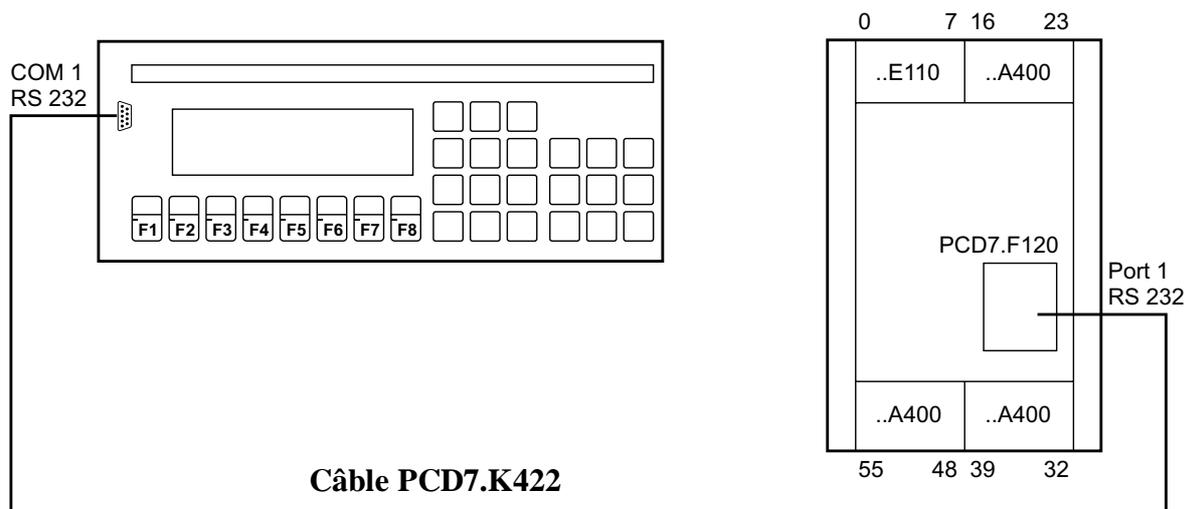
8. Exemples de programme utilisateur

8.1 Configuration matérielle

Les exemples suivants mettent en œuvre la configuration matérielle ci-dessous :

PCD7.D250

PCD1.M120/..M130 ou PCD2.M1..



Port série n° 1	:	RS 232 (PCD7.F120)
Câblage (PCD7.K422)	:	pour mode MC0 (sans contrôle RTS/CTS)
Paramétrage ..D250 (stocké en EPROM)	:	configuration par défaut (voir § 5.3.2)

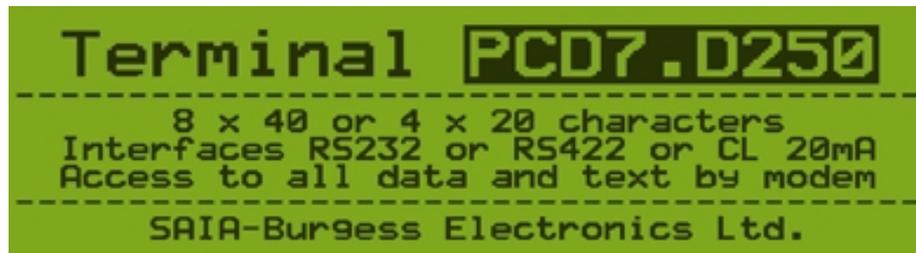
Consignes :

- Les indicateurs de diagnostic servent au contrôle et au diagnostic de la transmission série. Pour les rendre facilement visibles, on utilise des sorties dans ces exemples. Il est donc important que les trois modules de sorties ..A400 (ou ..A410) figurant sur le schéma soient présents et enfichés aux emplacements adéquats.
- Pour transmettre plusieurs pages écran en même temps, il est conseillé de travailler en mode MC1 (RTS/CTS) ou MC2 (XON/XOFF) de façon à éviter tout débordement du tampon d'entrée du terminal. Précisons que ce choix doit aussi être déclaré au paramétrage du terminal.

8.2 Transmission d'un seul texte

Il s'agit de transmettre un texte simple en gros caractères (4 x 20) en fermant le commutateur raccordé à l'entrée 0 et un texte en petits caractères (4 x 80) en fermant l'entrée 1.

- 8.2.1 Programme utilisateur en BLOCTEC
- 8.2.2 Programme utilisateur en GRAFTEC
- 8.2.3 Transmission d'un texte associant petits et gros caractères par fermeture de l'entrée 0



8.2.1 Transmission d'un seul texte en BLOCTEC

```

;-----
;
; User program example 8.2.1 for the industrial terminal PCD7.D250
; =====
; The program is structured in BLOCTEC
;
; File:          NDEMO21.SRC
;
; Creation:      16.01.97          U.Jäggi
; Modified:      03.02.00          C. Bruegger
;
;-----

```

```

TEXT      1      "<12>"                ; Clear display
            "<27><64><77><49>"          ; Display mode:4 x 20
            "<27><84>"                ; Cursor off
            "      INDUSTRIAL      "
            "  CONTROL-TERMINAL  "
            "      PCD7.D250      "
            "Display mode:4 x 20 "

```

```

TEXT      10     "<12>"                ; Clear display
            "<27><64><77><48>"          ; Display mode:8 x 40
            "<27><84>"                ; Cursor off
            "<10>"                    ; LF
            "#####"
            "#      INDUSTRIAL      #"
            "#      CONTROL-TERMINAL  #"
            "#      PCD7.D250      #"
            "#      Display mode:8 x 40  #"
            "#####"

```

```

TEXT      100    "UART:9600,8,E,1;MODE:MC0;DIAG:O16,R100"
;

```

```

;-----
; Coldstart
;-----
XOB      16
SASI     1           ; Assignation RS232 interface n°1
          100        ; Text 100
EXOB

;-----
; Mainprogram
;-----
COB      0
          0
STH      I          0           ; for display 4 x 20
DYN      F          0
ANL      O          22        ; Text busy flag
CPB      H          0           ; Send text
STH      I          1           ; for display 8 x 40
DYN      F          1
ANL      O          22        ; Text busy flag
CPB      H          1           ; Send text
ECOB

;=====
PB       0           ; Send text
STXT    1           ; Interface 1
          1           ; Text 1
EPB

PB       1           ; Send text
STXT    1           ; Interface 1
          10          ; Text 10
EPB

;-----

```

8.2.2 Transmission d'un seul texte en GRAFTEC

```

;+-----+
;
; User program example 8.2.2 for the industrial terminal PCD7.D250
; =====
; The program is structured in GRAFTEC
;
; File:          NDEMO22.SRC
;
; Creation:     29.01.97          U.Jäggi
; Modified:    03.02.00          C. Bruegger
;+-----+

TEXT      1          "<12>"          ; Clear display
              "<27><64><77><49>"      ; Display mode:4 x 20
              "<27><84>"          ; Cursor off
              "          INDUSTRIAL          "
              " CONTROL-TERMINAL          "
              "          PCD7.D250          "
              "Display mode:4 x 20 "

TEXT      10         "<12>"          ; Clear display
              "<27><64><77><48>"      ; Display mode:8 x 40
              "<27><84>"          ; Cursor off
              "<10>"            ; LF
              "#####"
              "#          INDUSTRIAL          #"
              "#          CONTROL-TERMINAL          #"
              "#          PCD7.D250          #"
              "#          Display mode:8 x 40          #"
              "#####"

TEXT      100        "UART:9600,8,E,1;MODE:MC0;DIAG:O16,R100"

;-----+

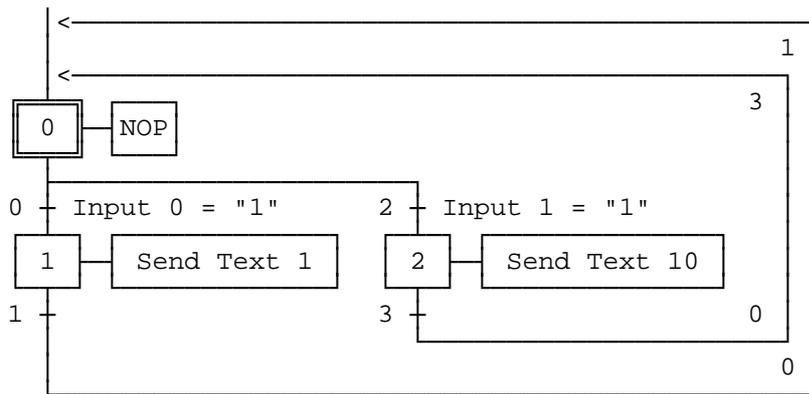
;-----+
; Coldstart
;-----+
XOB      16
SASI     1          ; Assigantion interface n°1
          100       ; Text 100
EXOB

;-----+

;-----+
; Mainprogram
;-----+
COB      0
          0
CSB      0
ECOB

;-----+

```



```

-----
SB      0
;-----
      IST      0          ; NOP
          I 1
          I 3
          O 0
          O 2
      Acc      1

      EST

;-----

      ST      1          ; Send Text 1
          I 0
          O 1
      STXT    1
          1
      EST      1          ; 1
;-----

      ST      2          ; Send Text 10
          I 2
          O 3
      STXT    1
          10
      EST      2          ; 2
;=====

      TR      0          ; Input 0 = "1"
          I 0          ; NOP
          O 1

      STH     I          0
      DYN     F          0
      ANL     O          22      ; Text busy
      ETR     ; 0
;-----

      TR      1
          I 1
          O 0          ; NOP
      ETR     ; 1
;-----

      TR      2          ; Input 1 = "1"
          I 0          ; NOP
          O 2

      STH     I          1
      DYN     F          1
      ANL     O          22      ; Text busy
      ETR     ; 2
;-----

      TR      3
          I 2
          O 0          ; NOP
      ETR     ; 3

      ESB     ; 0
;-----
    
```

8.2.3 Transmission d'un texte à petits et gros caractères en BLOCTEC

```

;-----
;
; User program example 8.2.3 for the industrial terminal PCD7.D250
; =====
; The program is structured in BLOCTEC
;
; File:          NDEMO23.SRC
;
; Creation:     15.02.00          C. Bruegger
;
;-----

```

```

TEXT    1          "<27><64><77><49>"      ; Display mode:4 x 20
          "<12>"                          ; Clear display
          "<27><84>"                        ; Cursor off
          "<27><64><78><48>"                ; Invert mode off
          " TERMINAL "
          "<27><64><78><49>"                ; Invert mode on
          "PCD7.D250"
          "<27><64><78><48>"                ; Invert mode off
          "<27><64><77><53>"                ; Transparent mode on
          "<27><64><77><48>"                ; Display mode:8 x 40
          "<12>"                          ; Clear display
          "<27><84>"                        ; Cursor off
          "<5><5>"                          ; Cursor down 2 x
          "-----"
          "      8 x 40 or 4 X 20 characters      "
          " Interfaces RS232 or RS422 or CL 20mA  "
          " Access to all data and text by modem  "
          "-----"
          "      SAIA -Burgess Electronics AG      "

```

```

TEXT    100       "UART:9600,8,E,1;MODE:MC0;DIAG:016,R100"

```

```

;-----
; Coldstart
;-----
XOB      16
SASI     1          ; Assignment RS232 interface n°1
          100       ; Text 100
EXOB
;-----
; Mainprogram
;-----
COB      0
          0
STH      I          0
DYN      F          0
ANL      O          22      ; Text busy flag
CPB      H          0      ; Send text
ECOB
;-----
PB       0          ; Send text
STXT     1          ; Interface 1
          1          ; Text 1
EPB
;-----

```

8.3 Transmission de plusieurs textes

Lorsque les commutateurs raccordés aux entrées 0, 1 et 2 sont activés, les textes suivants sont transmis :

Entrée 0 : Affichage d'un texte.

Entrée 1 : Affichage de l'état des entrées 4 et 5.

Entrée 2 : Affichage de l'état des entrées 6 et 7.

L'affichage s'effectue en mode 4 x 20.

- 8.3.1 Programme utilisateur en BLOCTEC
- 8.3.2 Programme utilisateur en GRAFTEC *)

*) Dans ce programme, l'état opérationnel du terminal est contrôlé par l'XOB 16 de démarrage à froid, à l'aide de la commande de scrutation POLL, qui a également pour effet de coordonner les mises sous tension de l'automate et du terminal.

8.3.1 Transmission de plusieurs textes en BLOCTEC

```

;+-----+
;
; User program example 8.3.1 for the industrial terminal PCD7.D250
; =====
; The program is structured in BLOCTEC
;
; File:          NDEMO31.SRC
;
; Creation:     16.01.97      U.Jäggi
; Modified:    03.02.00      C. Bruegger
;
;+-----+

```

```

TEXT    1      "<12>"                ; Clear display
          "<27><84>"                ; cursor off
          "  Main menue I0<10><13>  "
          "  Display status      "
          "  Input 4,5      : I1  "
          "  Input 6,7      : I2  "

TEXT    2      "<12>"
          "  Status <10><13>  "
          "  Input 4 : $i0004<10><13>  "
          "  Input 5 : $i0005<10><13>  "
          "  Main menue I0  "

TEXT    3      "<12>"
          "  Status <10><13>  "
          "  Input 6 : $i0006<10><13>  "
          "  Input 7 : $i0007<10><13>  "
          "  Main menue I0  "

TEXT    4      "<27><64><77><49>"                ; Display mode:4 x 20

TEXT    100    "UART:9600,8,E,1;MODE:MC0;DIAG:O16,R100"

;

```

```

;-----
; Coldstart
;-----
XOB      16
SASI     1           ; Assignation RS232 interface
          100        ; Text 100
STXT     1
          4           ; Text 4: mode 4 x 20
EXOB

;-----
; Mainprogram
;-----
COB      0
          0

;-----
STH      I          0
DYN      F          0
ANL      O          22      ; Text busy flag
CFB      H          0      ; Send text
          1           ; Text 1

;-----
STH      I          1
DYN      F          1
ANL      O          22      ; Text busy flag
CFB      H          0      ; Send text
          2           ; Text 2

;-----
STH      I          2
DYN      F          2
ANL      O          22      ; Text busy flag
CFB      H          0      ; Send text
          3           ; Text 3
ECOB

;=====
FB        0           ; Send text
STXT     1           ; Interface 1
          =          1      ; Textnumber
EFB

;-----

```

8.3.2 Transmission de plusieurs textes en GRAFTEC

```

;+-----+
;
; User program example 8.3.2 for the industrial terminal PCD7.D250
; =====
; The program is structured in GRAFTEC
;
; File:          NDEMO32.SRC
;
; Creation:     16.01.97          U.Jäggi
; Modified:    03.02.00          C. Bruegger
;
;+-----+

```

```

TEXT    1      "<12>"                ; Clear display
                "<27><84>"            ; Cursor off
                " Main menue I0<10><13> "
                " Display status      "
                " Input 4,5   : I1  "
                " Input 6,7   : I2  "

TEXT    2      "<12>"                " Status <10><13> "
                " Input 4  :$i0004<10><13> "
                " Input 5  :$i0005<10><13> "
                " Main menue I0  "

TEXT    3      "<12>"                " Status <10><13> "
                " Input 6  :$i0006<10><13> "
                " Input 7  :$i0007<10><13> "
                " Main menue I0  "

TEXT    4      "<27><64><77><49>"          ; Display mode:4 x 20

TEXT    10     "<ESC>@B"                ; Poll command

TEXT    100    "UART:9600,8,E,1;MODE:MC0;DIAG:O16,R100"
;
;-----;
; Coldstart
;-----;
XOB     16
SASI    1          ; Assignment interface n°1
        100       ; Text 100
;
;-----;

```

```

termpoll:  stxt  1          ; START OF THE POLL COMMAND
           sth   0 22
           jr    h -1

           acc   h          ; (ld T is accu dependent)
           ld   T 0         ; start short receive timeout
           sth   2          ; (must be min. 10 mS)
           sth   3

termwait:  sth   0 16       ; character received?
           jr    h termok   ; yes
           sth   T 0
           jr    h termwait ; loop for timeout period
           jr    termpoll   ; Terminal not ready, repeat the poll

termok:    srxd  1          ; read the character
           R 1
           cmp  R 1         ; SOH character ?
           sth  1
           jr   z termready ; yes, Terminal is ready
           jr   termpoll    ; no, repeat the poll

termready: ld   R 1         ; clear receive register
           sth  0          ; END OF THE POLL COMMAND

;

```

```

           STXT  1          ; Display mode : 4 x 20
           sth   4

           sth   0 22
           jr    h -1

           STXT  1          ; Text 1: Main menu
           sth   1
           sth   0 22
           jr    h -1

           EXOB

;

```

```

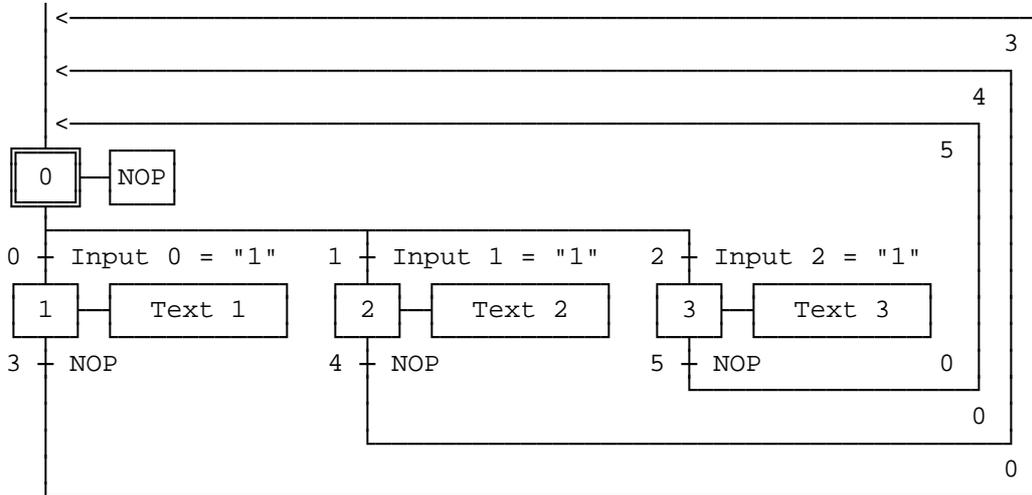
;-----
; Mainprogram
;-----

```

```

COB    0
      0
CSB    0
ECOB

```



```

SB    0
;-----

```

```

IST    0          ; NOP
      I    3      ; NOP
      I    4      ; NOP
      I    5      ; NOP
      O    0      ; Input 0 = "1"
      O    1      ; Input 1 = "1"
      O    2      ; Input 2 = "1"
EST
;-----

```

```

ST    1          ; Text 1
      I    0      ; Input 0 = "1"
      O    3      ; NOP
STXT  1
      1
EST
;-----

```

```

ST    2          ; Text 2
      I    1      ; Input 1 = "1"
      O    4      ; NOP
STXT  1
      2
EST
;-----

```

```

ST    3          ; Text 3
      I    2      ; Input 2 = "1"
      O    5      ; NOP
STXT  1
      3
EST
;-----

```

```

TR      0          ; Input 0 = "1"
        I          ; NOP
        O          ; Text 1
        1
STH     I          0
DYN     F          0
ANL     O          22      ; Text busy
ETR

;
-----
TR      1          ; Input 1 = "1"
        I          ; NOP
        O          ; Text 2
        2
STH     I          1
DYN     F          1
ANL     O          22      ; Text busy
ETR

;
-----
TR      2          ; Input 2 = "1"
        I          ; NOP
        O          ; Text 3
        3
STH     I          2
DYN     F          2
ANL     O          22      ; Text busy
ETR

;
-----
TR      3          ; NOP
        I          ; Text 1
        O          ; NOP
        0
ETR

;
-----
TR      4          ; NOP
        I          ; Text 2
        O          ; NOP
        0
ETR

;
-----
TR      5          ; NOP
        I          ; Text 3
        O          ; NOP
        0
ETR
ESB

;
-----

```

8.4 Reconnaissance de la pression d'une touche

L'appui sur une touche de fonction (F1, F2, F3 ou F4) provoque la transmission des textes suivants :

- F1 : Affichage d'un texte.
- F2 : Affichage de l'état des entrées 0 à 7 et des sorties 16 à 23.
- F3 : Affichage de la valeur des commutateurs BCD raccordés aux entrées 0 à 7.
- F4 : Affichage de la date, semaine et l'heure.

L'appui sur ces touches de fonction autorise une seule et unique transmission du texte. Si l'on souhaite rafraîchir périodiquement une valeur sur le terminal et garantir la stabilité de l'affichage, il faut :

- désactiver le curseur;
- omettre le code de commande 12 (effacement de l'affichage) en début de texte.

- 8.4.1 Programme utilisateur avec sauts *)
- 8.4.2 Programme utilisateur en BLOCTEC *)
- 8.4.3 Programme utilisateur en GRAFTEC *)

*) Dans ce programme, l'état opérationnel du terminal est contrôlé par l'XOB 16 de démarrage à froid, à l'aide de la commande de scrutation POLL, qui a également pour effet de coordonner les mises sous tension de l'automate et du terminal.

8.4.1 Reconnaissance de la pression d'une touche (avec sauts)

```

;+-----+
;
; User program example 8.4.1 for the industrial terminal PCD7.D250
; =====
; The program contains jumps
;
; File:          NDEMO41.SRC
;
; Creation:     29.01.97          U.Jäggi
; Modified:    07.02.00          C. Bruegger
;
;+-----+

TEXT    1      "<12>"                ; Clear display
                "<27><84>"            ; Cursor off
                "  Main menu  F1<10><13>"
                " I/O   0..23  F2<10><13>"
                " BCD-Switch F3<10><13>"
                " Date/Time  F4"

TEXT    2      "<12>"                ; Clear display
                "  Input Status<10><13>"
                " I0..7   :$I0000<10><13>"
                " O16..23:$O0016<10><13>"
                "  Main menu  F1"

TEXT    3      "<12>"                ; Clear display
                "  BCD-Value I0...7 <10><13>"
                " ----- <10><13>"
                " Value   : $R0010 <10><13>"
                "  Main menu  F1"

TEXT    4      "<12>"                ; Clear display
                "  Date   : $D<10><13>"
                "  Week  : $W<10><13>"
                "  Time  : $H<10><13>"
                "  Main menu  F1"

TEXT    5      "<27><64><77><49>"      ; Display mode:4 x 20

TEXT    10     "<ESC>@B"              ; Poll command

TEXT    100    "UART:9600,8,E,1;MODE:MC0;DIAG:O16,R100"

;-----+
; Symboldefinitions
;-----+
; Diagnostic outputs serial interface
;-----+
RBSY    EQU    O    16                ; Receiver Busy
RFUL    EQU    O    RBSY+1            ; Receive Buffer Full
RDIA    EQU    O    RBSY+2            ; Receiver Diagnostic
TBSY    EQU    O    RBSY+3            ; Transmitter Busy
TFUL    EQU    O    RBSY+4            ; Transmit Buffer Full
TDIA    EQU    O    RBSY+5            ; Transmitter Diagnostic
XBSY    EQU    O    RBSY+6            ; Text Busy
NEXE    EQU    O    RBSY+7            ; Not Executed
;-----+
; Function/Program blocks
;-----+
READ    EQU    FB    0                ; Read character
SEND    EQU    FB    1                ; Send text
COMPARE EQU    PB    0                ; Compare received character
;-----+
; Register
;-----+
RBUF_R  EQU    R    1000
;

```

```

;-----
; Coldstart
;-----
XOB      16
SASI     1          ; Assignment interface n°1
          100       ; Text 100

;-----
;
termpoll: stxt      1          ; START OF THE POLL COMMAND
           10
           sth      0 22
           jr       h -1

           acc      h          ; (ld T is accu dependent)
           ld       T 0        ; start short receive timeout
           4         ; (must be min. 10 mS)
           5

termwait: sth      0 16        ; character received?
           jr       h termok   ; yes
           sth      T 0
           jr       h termwait ; loop for timeout period
           jr       termpoll   ; Terminal not ready, repeat the poll

termok:   srxd     1          ; read the character
           R 1
           cmp      R 1        ; SOH character ?
           1
           jr       z termready ; yes, Terminal is ready
           jr       termpoll   ; no, repeat the poll

termready: ld      R 1         ; clear receive register
           0         ; END OF THE POLL COMMAND

;-----
;
STXT     1          ; Display mode : 4 x 20
           5

sth      0 22
jr       h -1

STXT     1          ; Main menu
           1
sth      0 22
jr       h -1

EXOB

;-----
;

```

```

;-----
; Main program
;-----
COB      0
         0
STH      0      RBSY      ; Receiver busy
ANL      0      XBSY      ; Text busy
JR       L      END       ; If RBSY = low then do nothing
SRXD     1      ; Interface 1
         R      RBUF_R    ; Receive buffer register
;
-----
; Compare received character
CMP      R      RBUF_R
         K      65      ; F1
ACC      Z
JR       L      F2
STXT     1      ; Interface 1
         1      ; Text 1
JR       END
;
-----
F2:      CMP      R      RBUF_R
         K      66      ; F2
ACC      Z
JR       L      F3
STXT     1      ; Interface 1
         2      ; Text 2
JR       END
;
-----
F3:      CMP      R      RBUF_R
         K      67      ; F3
ACC      Z
JR       L      F4
STXT     1      ; Interface 1
         3      ; Text 3
JR       END
;
-----
F4:      CMP      R      RBUF_R
         K      68      ; F4
ACC      Z
JR       L      END
STXT     1      ; Interface 1
         4      ; Text 4
JR       END
;
-----
; Read BCD-Switch
END:     DIGI     2
         I      0
         R      10
;
-----
ECOB

```

8.4.2 Reconnaissance de la pression d'une touche en BLOCTEC

```

;+-----+
;
; User program example 8.4.2 for the industrial terminal PCD7.D160/170
; =====
; The program is structured in BLOCTEC
;
; File:          NDEMO42.SRC
;
; Creation:      29.01.97          U.Jäggi
; Modified:      07.07.00          C.Bruegger
;+-----+

TEXT    1      "<12>"                ; Clear display
                "<27><84>"            ; Cursor off
                "  Main menu   F1<10><13>"
                " I/O   0..23  F2<10><13>"
                " BCD-Switch F3<10><13>"
                " Date/Time   F4"

TEXT    2      "<12>"                ; Clear display
                "  Input Status <10><13>"
                " I0..7   :$I0000 <10><13>"
                " O16..23:$O0016 <10><13>"
                "  Main menu   F1"

TEXT    3      "<12>"                ; Clear display
                "  BCD-Value I0...7 <10><13>"
                " ----- <10><13>"
                " Value   : $R0010 <10><13>"
                "  Main menu   F1"

TEXT    4      "<12>"                ; Clear display
                "  Date   : $D<10><13>"
                " Week   : $W<10><13>"
                " Time   : $H<10><13>"
                "  Main menu   F1"

TEXT    5      "<27><64><77><49>"      ; Display mode:4 x 20

TEXT    10     "<ESC>@B"              ; Poll command

TEXT    100    "UART:9600,8,E,1;MODE:MC0;DIAG:O16,R100"

;-----+
;                               ; Symboldefinitions
;-----+
;                               ; Diagnostic outputs serial interface
;-----+
RBSY    EQU    O    16           ; Receiver Busy
RFUL    EQU    O    RBSY+1       ; Receive Buffer Full
RDIA    EQU    O    RBSY+2       ; Receiver Diagnostic
TBSY    EQU    O    RBSY+3       ; Transmitter Busy
TFUL    EQU    O    RBSY+4       ; Transmit Buffer Full
TDIA    EQU    O    RBSY+5       ; Transmitter Diagnostic
XBSY    EQU    O    RBSY+6       ; Text Busy
NEXE    EQU    O    RBSY+7       ; Not Executed
;-----+
;                               ; Function/Program blocks
;-----+
READ    EQU    FB    0           ; Read character
SEND    EQU    FB    1           ; Send text
COMPARE EQU    PB    0           ; Compare received character
;-----+
;                               ; Register
;-----+
RBUF_R  EQU    R    1000
;

```

```

;-----
; Coldstart
;-----
XOB      16
SASI     1          ; Assignment interface n°1
          100       ; Text 100

;-----
;
termpoll: stxt      1          ; START OF THE POLL COMMAND
          10
          sth      0 22
          jr       h -1

          acc      h          ; (ld T is accu dependent)
          ld       T 0        ; start short receive timeout
          6         ; (must be min. 10 mS)
          7

termwait: sth      0 16        ; character received?
          jr       h termok    ; yes
          sth      T 0
          jr       h termwait  ; loop for timeout period
          jr       termpoll    ; Terminal not ready, repeat the poll

termok:   srxd     1          ; read the character
          R 1
          cmp      R 1        ; SOH character ?
          1
          jr       z termready ; yes, Terminal is ready
          jr       termpoll    ; no, repeat the poll

termready: ld      R 1        ; clear receive register
          0         ; END OF THE POLL COMMAND

;-----
;
          STXT     1          ; Display mode : 4 x 20
          5

          sth      0 22
          jr       h -1

          STXT     1          ; Main menu
          1
          sth      0 22
          jr       h -1

          CFB      SEND

          EXOB     1

;-----
;

```


8.4.3 Reconnaissance de la pression d'une touche en GRAFTEC

```

;+-----+
;
; User program example 8.4.3 for the industrial terminal PCD7.D250
; =====
; The program is structured in GRAFTEC
;
; File:          NDEMO43.SRC
;
; Creation:     29.01.97          U.Jäggi/T.Hofer
; Modified:    07.02.00          C. Bruegger
;+-----+

TEXT    1      "<12>"                ; Clear display
                "<27><84>"            ; Cursor off
                "  Main menu   F1<10><13>"
                " I/O   0..23  F2<10><13>"
                " BCD-Switch F3<10><13>"
                " Date/Time   F4"

TEXT    2      "<12>"                ; Clear display
                "  Input Status <10><13>"
                " I0..7   :$I0000 <10><13>"
                " O16..23:$O0016 <10><13>"
                "  Main menu  F1"

TEXT    3      "<12>"                ; Clear display
                "  BCD-Value I0...7 <10><13>"
                " ----- <10><13>"
                " Value   : $R0010 <10><13>"
                "  Main menu  F1"

TEXT    4      "<12>"                ; Clear display
                "  Date   : $D<10><13>"
                " Week   : $W<10><13>"
                " Time   : $H<10><13>"
                "  Main menu  F1"

TEXT    5      "<27><64><77><49>"      ; Display mode:4 x 20

TEXT    10     "<ESC>@B"              ; Poll command

TEXT    100    "UART:9600,8,E,1;MODE:MC0;DIAG:O16,R100"

;-----; Symboldefinitions
;-----; Diagnostic outputs serial interface
;-----;
RBSY    EQU    O    16                ; Receiver Busy
RFUL    EQU    O    RBSY+1            ; Receive Buffer Full
RDIA    EQU    O    RBSY+2            ; Receiver Diagnostic
TBSY    EQU    O    RBSY+3            ; Transmitter Busy
TFUL    EQU    O    RBSY+4            ; Transmit Buffer Full
TDIA    EQU    O    RBSY+5            ; Transmitter Diagnostic
XBSY    EQU    O    RBSY+6            ; Text Busy
NEXE    EQU    O    RBSY+7            ; Not Executed
;-----;
;-----; Function/Program blocks
;-----;
READ    EQU    FB    0                ; Read character
SEND    EQU    FB    1                ; Send text
COMPARE EQU    PB    0                ; Compare received character
;-----;
;-----; Register
;-----;
RBUF_R  EQU    R    1000
;

```

```

;-----
; Coldstart
;-----
XOB      16
SASI     1          ; Assignment interface n°1
          100       ; Text 100

;-----
;
termpoll: stxt      1          ; START OF THE POLL COMMAND
          10
          sth      0 22
          jr       h -1

          acc      h          ; (ld T is accu dependent)
          ld      T 0        ; start short receive timeout
          8        ; (must be min. 10 mS)
          9

termwait: sth      0 16      ; character received?
          jr      h termok   ; yes
          sth      T 0
          jr      h termwait ; loop for timeout period
          jr      termpoll   ; Terminal not ready, repeat the poll

termok:   srxd     1          ; read the character
          R 1
          cmp     R 1        ; SOH character ?
          1
          jr     z termready ; yes, Terminal is ready
          jr     termpoll   ; no, repeat the poll

termready: ld      R 1       ; clear receive register
          0                ; END OF THE POLL COMMAND

;-----
;
          STXT     1          ; Display mode : 4 x 20
          5

          sth     0 22
          jr     h -1

          STXT     1          ; Text 1: Main menu
          1
          sth     0 22
          jr     h -1

          EXOB

;-----
;

```

```

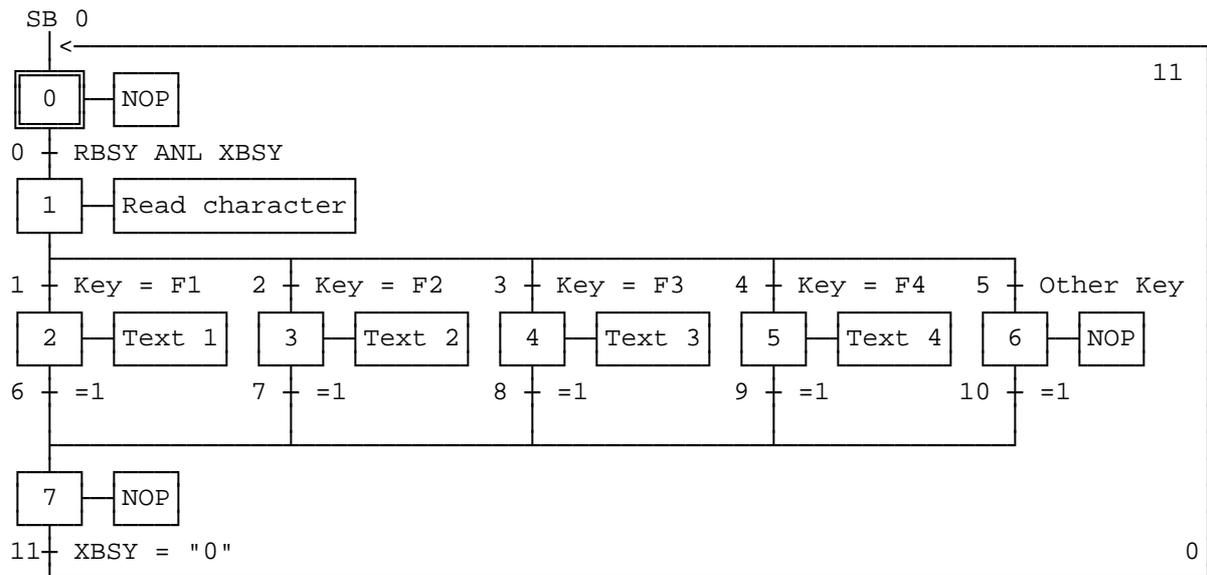
;-----
; Main program
;-----
COB      0
         0
CSB      0
;

```

```

; Read BCD-Switch
DIGI     2
         I      0
         R      10
ECOB
;

```



```

SB      0
;

```

```

IST      0          ; NOP
         I      11   ; XBSY = "0"
         O      0    ; RBSY ANL XBSY
EST
;

```

```

ST      1          ; Read character
         I      0    ; RBSY ANL XBSY
         O      1    ; Key = F1
         O      2    ; Key = F2
         O      3    ; Key = F3
         O      4    ; Key = F4
         O      5    ; Other key
SRXD    1
         R      RBUF_R
EST
;

```

```

ST      2          ; Text 1
         I      1    ; Key = F1
         O      6    ; =1
STXT    1          ; send
         1          ; text 1
EST
;

```

```

      ST      3          ; Text 2
      I      2          ; Key = F2
      O      7          ; =1
      STXT   1          ; send
      EST    2          ; text 2
;
-----
      ST      4          ; Text 3
      I      3          ; Key = F3
      O      8          ; =1
      STXT   1          ; send
      EST    3          ; text 3
;
-----
      ST      5          ; Text 4
      I      4          ; Key = F4
      O      9          ; =1
      STXT   1          ; send
      EST    4          ; text 4
;
-----
      ST      6          ; NOP
      I      5          ; Other key
      O     10          ; =1
      EST
;
-----
      ST      7          ; NOP
      I      6          ; =1
      I      7          ; =1
      I      8          ; =1
      I      9          ; =1
      I     10          ; =1
      O     11          ; XBSY = "0"
      EST
;
-----
      TR      0          ; RBSY ANL XBSY
      I      0          ; NOP
      O      1          ; Read character
      STH    0          16 ; Receiver busy
      ANL    0          22 ; Text busy
      ETR
;
-----
      TR      1          ; Key = F1
      I      1          ; Read character
      O      2          ; Text 1
      CMP    R          RBUF_R
      K      65         ; F1
      ACC    Z
      ETR
;
-----
      TR      2          ; Key = F2
      I      1          ; Read character
      O      3          ; Text 2
      CMP    R          RBUF_R
      K      66         ; F2
      ACC    Z
      ETR
;
-----

```

```

TR      3
I      1
O      4
CMP    R      RBUF_R
      K      67
ACC
ETR    Z
; Key = F3
; Read character
; Text 3

;
-----
TR      4
I      1
O      5
CMP    R      RBUF_R
      K      68
ACC
ETR    Z
; Key = F4
; Read character
; Text 4

;
-----
TR      5
I      1
O      6
ETR
; Other key
; Read character
; NOP

;
-----
TR      6
I      2
O      7
ETR
; =1
; Text 1
; NOP

;
-----
TR      7
I      3
O      7
ETR
; =1
; Text 2
; NOP

;
-----
TR      8
I      4
O      7
ETR
; =1
; Text 3
; NOP

;
-----
TR      9
I      5
O      7
ETR
; =1
; Text 4
; NOP

;
-----
TR      10
I      6
O      7
ETR
; =1
; NOP
; NOP

;
-----
TR      11
I      7
O      0
STL   O      22
ETR
; XBSY = "0"
; NOP
; NOP
; Text busy

;
-----
ESB

```

8.5 Saisie de paramètres numériques

Il est possible de modifier le contenu d'un registre et d'un compteur par le ..D250, sous réserve de respecter les conditions suivantes :

- Les valeurs du registre doivent toujours être positives ou négatives et marquées d'un point décimal fixe.
- Les valeurs du compteur peuvent n'être que positives, sans point décimal.
- Exemple programme utilisateur à la page suivante

Le bloc de fonction universel **INPUT** a été spécialement développé pour remplir cette fonction. Une description complète de ce bloc figure en annexe du programme de démonstration (voir chapitre 8.6).

Saisie de paramètres numériques

```

;-----
;
; User program example 8.5 for the industrial terminal PCD7.D250
; =====
; Input of numerical parameters
;
; File:          DEMO.SRC
; Version:       1.0
;
; Creation:      21.01.93          U.Jäggi
; Modified:      08.02.00          C. Bruegger
;-----

RBSY_F    EQU    0 32              ; Receiver Busy
          PUBL   RBSY_F
XBSY_F    EQU    0 38              ; Text Busy
          PUBL   XBSY_F
SIGN      EQU    0 48              ; Sign input
IN_BUSY   EQU    0 49              ; Input busy
          PUBL   IN_BUSY
DIGIT     EQU    R 0               ; Number of digits
X_POS     EQU    R 1               ; X-position
Y_POS     EQU    R 2               ; Y-position
DECIMAL   EQU    R 3               ; Number of decimal places
          DOC    R 500
DIAG_R    EQU    R 999             ; Diagnostic register
          DOC    R 1000
          DOC    C 100
MAIN      EQU    TEXT 0            ; Main menue
IN_TXT_R  EQU    TEXT 1            ; Input text register
IN_TXT_C  EQU    TEXT 2            ; Input text counter
DISP      EQU    TEXT 10

ASSIGN    EQU    TEXT 999          ; Assignment of the serial interfac
CHAN_N    EQU    1                 ; Number of serial channel
          PUBL   CHAN_N
          DOC    COB 0
          DOC    XOB 16
INPUT     EQU    FB 0              ; Functionblock input
          PUBL   INPUT

TEXT      ASSIGN    "UART:9600,8,E,1;"
          "MODE:MC0;"
          "DIAG: ",RBSY_F.T," ",DIAG_R.T," "

TEXT      DISP      "<27><64><77><49>"          ; Display mode:4 x 20

TEXT      MAIN      "<12>"                  ; Clear display
          "<27><84>"                        ; Cursor off
          "=="PARAMETER INPUT=="
          "-----"
          "MODIFY REGISTER [F1]"
          "MODIFY COUNTER [F2]"

TEXT      IN_TXT_R  "<12>"
          "R-Value : $%00.3d$R0500<10><13>"
          "ACCEPT VALUE [CR]"
          "MODIFY VALUE [F1]"
          "MAIN MENU [F2]"

TEXT      IN_TXT_C  "<12>"
          "C-Value : $C0100<10><13>"
          "ACCEPT VALUE [CR]"
          "MODIFY VALUE [F1]"
          "MAIN MENU [F2]"

```

```

;-----
; Coldstart
;-----

XOB      16

SASI     CHAN_N      ; Assignation RS232 interface
         ASSIGN      ; Text 100
ACC      H
RES      IN_BUSY     ; Reset input busy flag

STXT     CHAN_N
         DISP

EXOB

;=====

;-----
; Main program
;-----

COB      0
         0

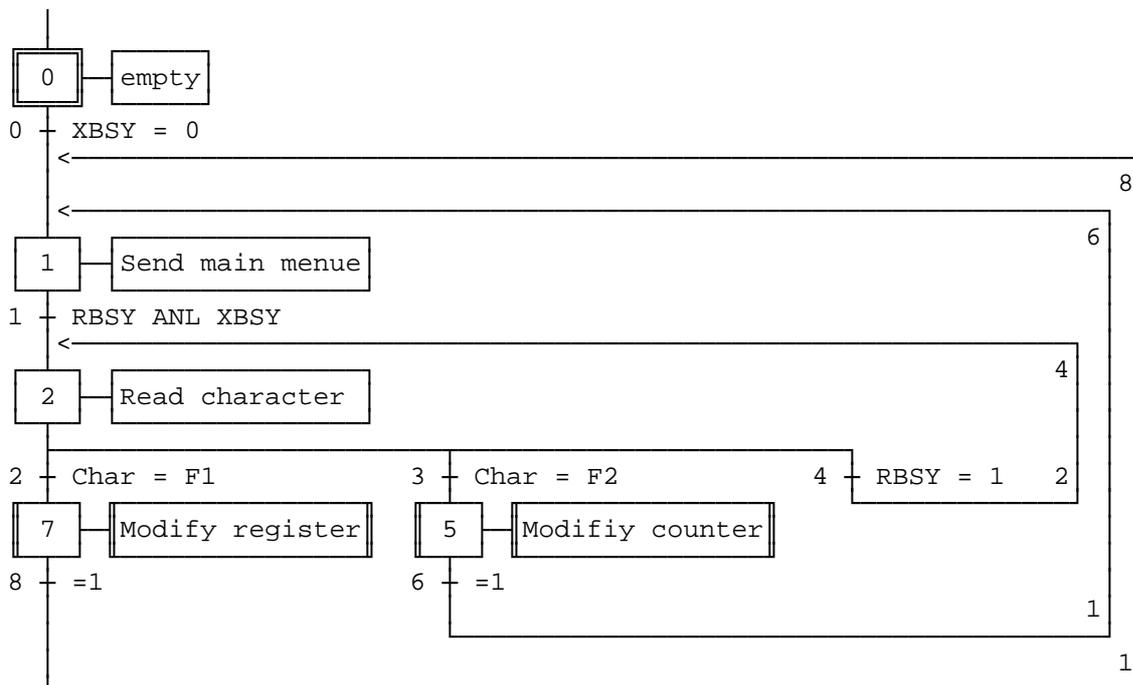
CSB      0           ; Call communication SB 0

ECOB

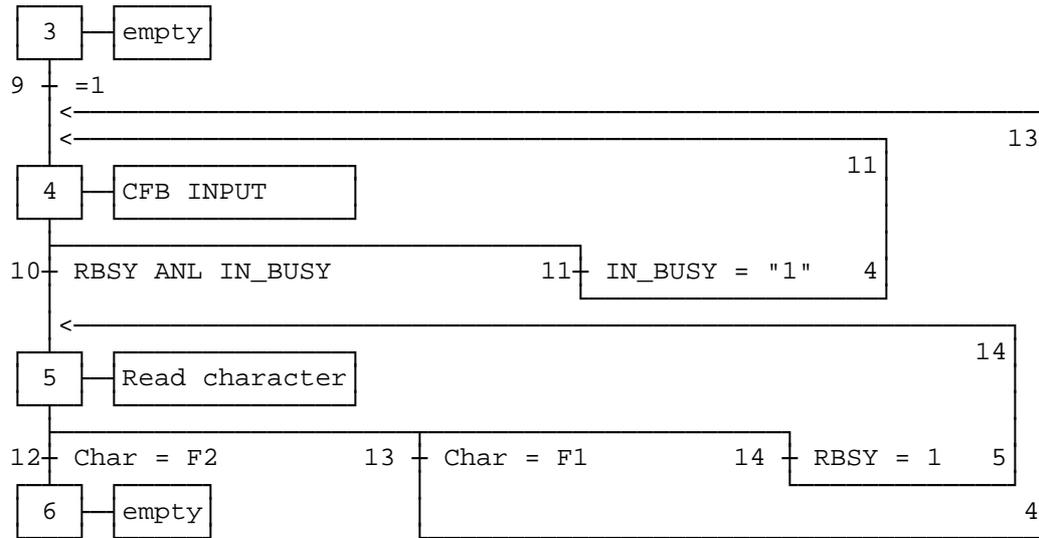
;-----

```

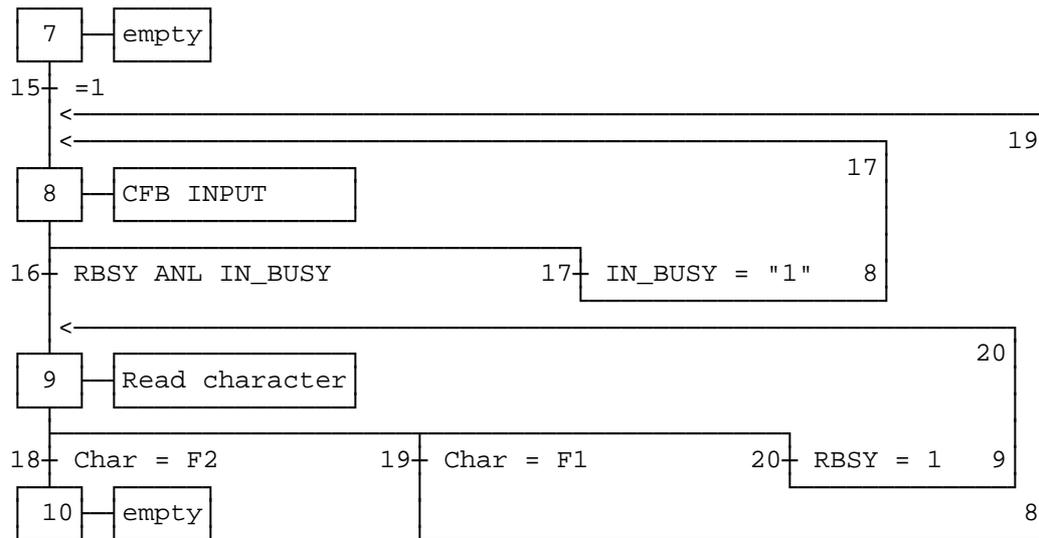
SB 0



Page Nb 5, Modify counter



Page Nb 7, Modify register



```

;
;          SB      0
;
;-----
;          IST      0          ; empty
;          O 0      ; XBSY = 0
;          EST
;
;-----
;          ST       1          ; Send main menue
;          I 0      ; XBSY = 0
;          I 8      ; =1
;          I 6      ; =1
;          O 1      ; RBSY ANL XBSY
;          STXT    CHAN_N      ; Send
;          MAIN     ; the main menue
;          EST
;
;-----

```

```

ST      2          ; Read character
        I 1        ; RBSY ANL XBSY
        I 4        ; RBSY = 1
        O 2        ; Char = F1
        O 3        ; Char = F2
        O 4        ; RBSY = 1
SRXD    CHAN_N    ; Read character
        R 1000    ; from the receive buffer
EST
;
-----
ST      3          ; empty
        I 3        ; Char = F2
        O 9        ; =1
EST
;
-----
ST      4          ; CFB INPUT
        I 9        ; =1
        I 11       ; IN_BUSY = "1"
        I 13       ; Char = F1
        O 10       ; RBSY ANL IN_BUSY
        O 11       ; IN_BUSY = "1"
RES     SIGN      ; Sign input not allowed
LD      X_POS     ; X-position
        42
LD      Y_POS     ; Y-position
        32
LD      DIGIT     ; Number of digits
        9
LD      DECIMAL   ; Number of decimal places
        0
CFB     INPUT     ; D100 input
        IN_TXT_C  ; Input text counter
        C 100    ; Counter to be modified
        DIGIT    ; Number of digits
        DECIMAL  ; Number of decimal places
        X_POS    ; X-position
        Y_POS    ; Y-position
        SIGN     ; Sign input yes/no (1/0)
EST
;
-----
ST      5          ; Read character
        I 10       ; RBSY ANL IN_BUSY
        I 14       ; RBSY = 1
        O 12       ; Char = F2
        O 13       ; Char = F1
        O 14       ; RBSY = 1
SRXD    CHAN_N    ; Read character
        R 1000    ; from the receive buffer
EST
;
-----
ST      6          ; empty
        I 12       ; Char = F2
        O 6        ; =1
EST
;
-----
ST      7          ; empty
        I 2        ; Char = F1
        O 15       ; =1
EST
;
-----

```

```

ST      8          ; CFB INPUT
        I 15      ; =1
        I 17      ; IN_BUSY = "1"
        I 19      ; Char = F1
        O 16      ; RBSY ANL IN_BUSY
        O 17      ; IN_BUSY = "1"
SET     SIGN      ; Sign input allowed
LD      X_POS     ; X-position
        42
LD      Y_POS     ; Y-position
        32
LD      DIGIT     ; Number of digits
        9
LD      DECIMAL   ; Number of decimal places
        3
CFB     INPUT     ; D100 input
        IN_TXT_R  ; Input text register
        R 500    ; Register to be modified
        DIGIT    ; Number of digits
        DECIMAL  ; Number of decimal places
        X_POS    ; X-position
        Y_POS    ; Y-position
        SIGN     ; Sign input yes/no (1/0)
EST
;
-----
ST      9          ; Read character
        I 16      ; RBSY ANL IN_BUSY
        I 20      ; RBSY = 1
        O 18      ; Char = F2
        O 19      ; Char = F1
        O 20      ; RBSY = 1
SRXD    CHAN_N    ; Read character
        R 1000   ; from the receive buffer
EST
;
-----
ST      10         ; empty
        I 18      ; Char = F2
        O 8       ; =1
EST
;
=====
TR      0          ; XBSY = 0
        I 0       ; empty
        O 1       ; Send main menu
STL     XBSY_F
ETR
;
-----
TR      1          ; RBSY ANL XBSY
        I 1       ; Send main menu
        O 2       ; Read character
STH     RBSY_F
ANL     XBSY_F
ETR
;
-----
TR      2          ; Char = F1
        I 2       ; Read character
        O 7       ; empty
CMP     R 1000
        K 65      ; F1
ACC     Z
ETR
;
-----

```

```

TR      3          ; Char = F2
        I 2        ; Read character
        O 3        ; empty
CMP     R 1000
        K 66      ; F2
ACC     Z
ETR

;
-----
TR      4          ; RBSY = 1
        I 2        ; Read character
        O 2        ; Read character
STH     RBSY_F
ETR

;
-----
TR      5          ; Modify counter
        I 3        ; empty
        O 6        ; empty
ETR

;
-----
TR      6          ; =1
        I 6        ; empty
        O 1        ; Send main menu
ETR

;
-----
TR      7          ; Modify register
        I 7        ; empty
        O 10       ; empty
ETR

;
-----
TR      8          ; =1
        I 10       ; empty
        O 1        ; Send main menu
ETR

;
-----
TR      9          ; =1
        I 3        ; empty
        O 4        ; CFB INPUT
ETR

;
-----
TR      10         ; RBSY ANL IN_BUSY
        I 4        ; CFB INPUT
        O 5        ; Read character
STH     RBSY_F
ANL     IN_BUSY
ETR

;
-----
TR      11         ; IN_BUSY = "1"
        I 4        ; CFB INPUT
        O 4        ; CFB INPUT
STH     IN_BUSY
ETR

;
-----
TR      12         ; Char = F2
        I 5        ; Read character
        O 6        ; empty
CMP     R 1000
        K 66      ; F2
ACC     Z
ANL     XBSY_F
ETR

;
-----

```

```

TR      13                ; Char = F1
        I 5                ; Read character
        O 4                ; CFB INPUT
CMP     R 1000
        K 65                ; F1
ACC     Z
ANL     XBSY_F
ETR

;
-----
TR      14                ; RBSY = 1
        I 5                ; Read character
        O 5                ; Read character
STH     RBSY_F
ETR

;
-----
TR      15                ; =1
        I 7                ; empty
        O 8                ; CFB INPUT
ETR

;
-----
TR      16                ; RBSY ANL IN_BUSY
        I 8                ; CFB INPUT
        O 9                ; Read character
STH     RBSY_F
ANL     IN_BUSY
ETR

;
-----
TR      17                ; IN_BUSY = "1"
        I 8                ; CFB INPUT
        O 8                ; CFB INPUT
STH     IN_BUSY
ETR

;
-----
TR      18                ; Char = F2
        I 9                ; Read character
        O 10               ; empty
CMP     R 1000
        K 66                ; F2
ACC     Z
ANL     XBSY_F
ETR

;
-----
TR      19                ; Char = F1
        I 9                ; Read character
        O 8                ; CFB INPUT
CMP     R 1000
        K 65                ; F1
ACC     Z
ANL     XBSY_F
ETR

;
-----
TR      20                ; RBSY = 1
        I 9                ; Read character
        O 9                ; Read character
STH     RBSY_F
ETR

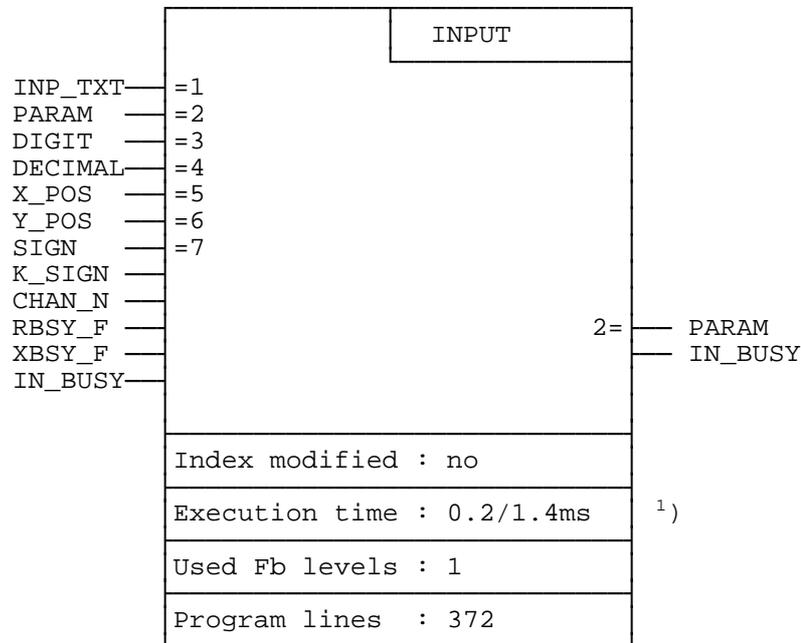
ESB

;
-----

```

8.6 Bloc de fonction : INPUT

Saisie de données par terminal industriel PCD7.D250 :



- 1) 0.2 ms : Indicateur RBSY_F à 0 (tampon de réception vide)
 1.4 ms : Indicateur RBSY_F à 1 (lecture des numéros 0 à 9 du tampon de réception et traitement)

Fonction

Permet la saisie du contenu d'un registre ou d'un compteur par le PCD7.D250. Le signe moins '-' et le point décimal sont acceptés.

Entrées-sorties

Symbole	Description	Para- mètre	Données			Adresse
			Type	Format	Valeur	
INP_TXT	Texte d'entrée	oui	X	Texte	Indifférente	0...3999
PARAM	Paramètre d'entrée (registres ou compteurs)	oui	R/C	Entier	- 2 147 483 648... + 2 147 483 647	0...4095
DIGIT	Nombre de chiffres	oui	R	Entier	1...11	0...4095
DECIMAL	Nombre de décimales	oui	R	Entier	0,1...10	0...4095
X_POS	Position du curseur en x	oui	R	Entier	32...51	0...4095
Y_POS	Position du curseur en Y	oui	R	Entier	32...35	0...4095
SIGN	Signe oui/non (1/0)	oui	F/I/O	Binaire	0/1	0...8191
K_SIGN	Touche de signe (ASCII code)	non	K	ASCII	0...255	-
CHAN_N	Port série	non	K	Numéro	0...3	-
RBSY_F	Récepteur occupé	non	F/O	Binaire	0/1	0...8191
XBSY_F	Interface occupée	non	F/O	Binaire	0/1	0...8191
IN_BUSY	Entrée occupée	non	F/O	Binaire	0/1	0...8191

Légende :

C	Compteur
F	Indicateur
I	Entrée
K	Fonction spéciale
O	Sortie
R	Registre

Éléments réservés à un usage interne :

Le bloc INPUT utilise en interne 7 registres et 6 indicateurs de travail. Ceux-ci contiennent des valeurs intermédiaires, acquises en cours de saisie, qui ne sont exploitables que par ce bloc de fonction. Seules les adresses de base de ces éléments doivent être définies dans le module.

Symbole	Description	Données		Adresse
		Type	Format	
WORK_R	Adresse de base des 7 registres de travail	R	Entier	0...4089 (+6)
WORK_F	Adresse de base des 6 indicateurs de travail	F	Binaire	0...8186 (+5)

Affectation des touches :

Les touches étant personnalisables pour répondre aux besoins de l'application, des symboles peuvent être utilisés pour leur affecter n'importe quel code ASCII.

Par défaut, les touches ont les fonctions et valeurs d'un clavier standard. Seules les touches du pavé numérique (0 à 9) ne doivent pas être modifiées pour éviter tout risque d'erreur lors de la conversion ASCII-décimal.

Symbole	Description	Données		Valeur
		Type	Format	
K_BS	Backspace key	K	ASCII	0...255
K_CR	Carriage return key	K	ASCII	0...255
K_DP	Decimal point key	K	ASCII	0...255
K_SIGN	Negative sign key	K	ASCII	0...255
K_0	0 key	K	ASCII	48
K_1	1 key	K	ASCII	49
K_2	2 key	K	ASCII	50
K_3	3 key	K	ASCII	51
K_4	4 key	K	ASCII	52
K_5	5 key	K	ASCII	53
K_6	6 key	K	ASCII	54
K_7	7 key	K	ASCII	55
K_8	8 key	K	ASCII	56
K_9	9 key	K	ASCII	57

Appel du bloc de fonction

```

CFB      INPUT      ; Input
          INP_TXT    ; Input text
R        PARAM      ; Parameter
R        DIGIT      ; Number of digits
R        DECIMAL    ; Number of decimal places
R        X_POS      ; X-position
R        Y_POS      ; Y-position
F        Sign       ; Sign input yes/no (1/0)

```

Détail des entrées-sorties

- Texte d'entrée INP_TXT :

Ce texte est transmis au premier appel du bloc de fonction. La valeur de l'élément à éditer (registre ou compteur) n'est affichée que lors de la sortie du texte à l'aide du bloc de fonction ; elle doit obligatoirement figurer dans le texte. Hormis cette contrainte, le texte peut être de longueur variable et contenir tout type de caractères. La valeur actuelle de l'élément à éditer peut apparaître sous n'importe quel format. Il est toutefois conseillé d'utiliser le même format pour la saisie et pour l'affichage.

Exemple :

```
TEXT    INP_TXT "<12>"                ; Clear display
        "PARAMETER INPUT<10><13>"
        "=====<10><13>"
        "Value : $%00.3d$",PARAM.04T,"<10><13>"
        "Accept value [CR]"
```

- Paramètre d'entrée PARAM :

Indique le registre ou le compteur à modifier.

- Nombre de chiffres DIGIT :

Indique le nombre de chiffres, signe '-' et point décimal compris, constituant le champ d'entrée. Ce nombre est contrôlé et limité en cours de saisie.

Important : le dernier chiffre de la ligne d'affichage ne doit pas être utilisé pour le champ d'entrée.

- Nombre de décimales DECIMAL :

Au format « point décimal fixe », ce registre définit le nombre de décimales. Sans point décimal, il est à 0. Ce nombre est contrôlé et limité en cours de saisie.

Exemple:

```

                                DIGIT (nombre de chiffres)= 8
                                |
                                |
Input field  1 2 3 4 . 5 6 7
                                |
                                |
                                DECIMAL (nombre de décimales) = 3
```

- Position du curseur en x/y X_POS et Y_POS :

Définit la position du premier caractère du champ d'entrée.

- **Signe SIGN :**
Donne le signe du premier caractère du champ d'entrée.
SIGN = 0 —> Saisie du signe '–' inhibée
SIGN = 1 —> Saisie du signe '–' validée
- **Touche de signe K_SIGN :**
Définit le code ASCII correspondant à la touche '–'.
Le signe '+' ne peut pas être saisi en début de texte.
- **Port série CHAN_N :**
Indique le numéro du port série devant être configuré en mode C avant d'appeler le bloc de fonction. (Précisons que le PCD7.D250 est configurable en MC0, MC1 et MC2.)
- **Indicateurs RBSY_F et XBSY_F :**
Les adresses des indicateurs de diagnostic RBSY_F (récepteur occupé) et XBSY_F (interface occupée) du port série doivent correspondre à celles des indicateurs de diagnostic définis par l'instruction SASI.
- **Indicateur IN_BUSY :**
IN_BUSY (entrée occupée) doit au départ être à 0 pour permettre le bon fonctionnement du bloc de fonction —> Remettre IN_BUSY à 0 dans XOB16. Il passe à 1 au premier appel du bloc de fonction, puis repasse à 0 sur réception d'un retour-chariot <CR>.

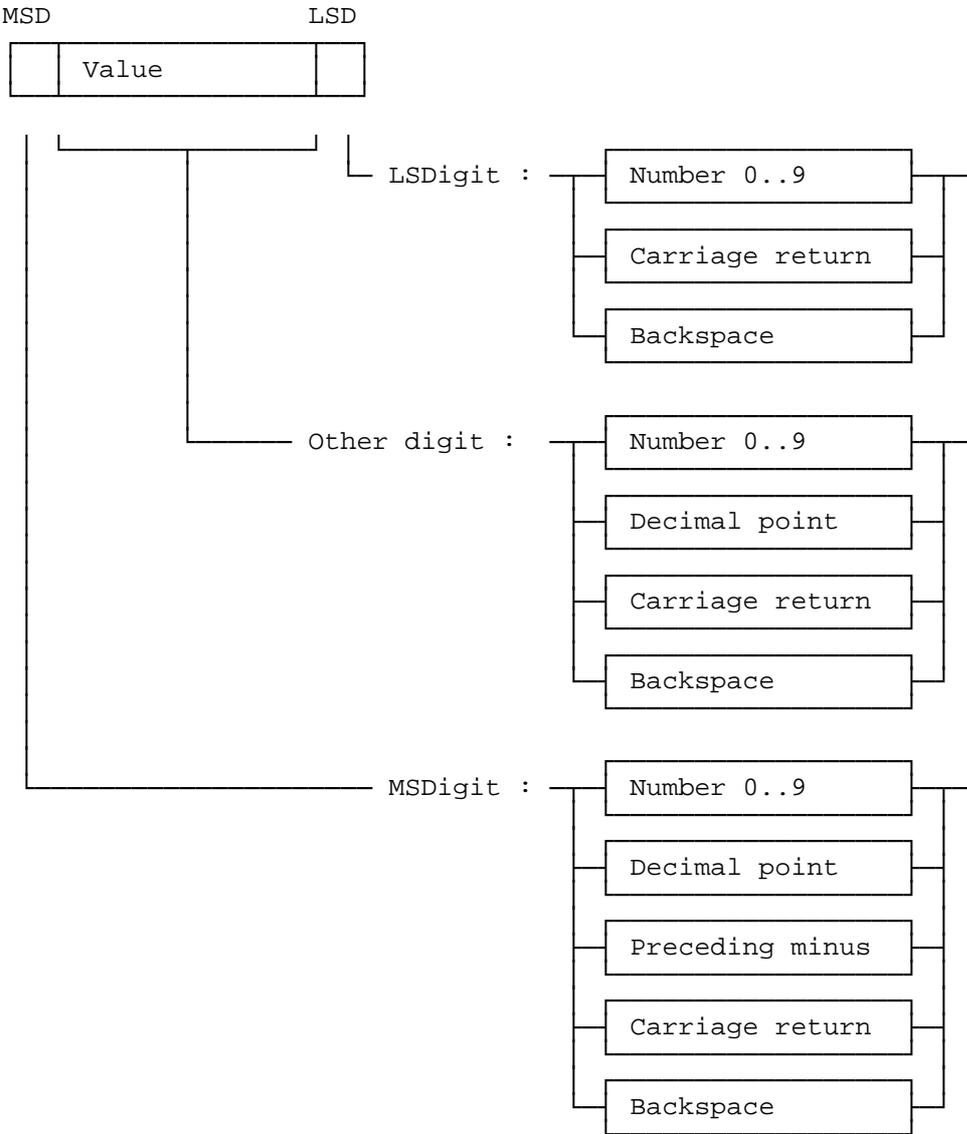
Saisie et modification de paramètre

Lorsque le bloc de fonction est appelé pour la première fois, le texte d'entrée est édité, l'indicateur IN_BUSY passe à 1 et le curseur se positionne conformément aux paramètres X_POS et Y_POS du bloc de fonction. On peut alors saisir un nombre au clavier (précédé éventuellement du signe '–'). Si la première frappe est un chiffre, le signe '–' ou le point décimal, le champ d'entrée du paramètre est effacé.

La taille du champ d'entrée est limitée par le nombre maximal de chiffres (voir DIGIT), lui-même contrôlé par le bloc de fonction au cours de la saisie. L'appui sur la touche de retour-chariot entraîne la mémorisation du nombre dans le registre ou le compteur PARAM, la remise à 0 de l'indicateur IN_BUSY et la fin de la saisie.

Pendant la saisie d'une valeur (IN_BUSY à 1), le bloc de fonction doit être appelé cycliquement par le programme utilisateur.

Format d'entrée d'une valeur numérique vers le terminal PCD7.D250 :



L'exemple ci-dessous illustre ces principes.

Les paramètres du bloc de fonction contiennent les valeurs suivantes :

```

Input text  "INP_TXT"  : "<12>"  ; Clear display
                    "PARAMETER INPUT<10><13>"
                    "=====<10><13>"
                    "Value : $%00.3d$",PARAM.04T,"<10><13>"
                    "Accept value [CR]"

Register  "PARAM"      : 567890
Register  "DIGIT"      : 8
Register  "DECIMAL"    : 3
Register  "X_POS"      : 40
Register  "Y_POS"      : 34

```

Le signe '-' se définit comme suit :

K_SIGNEQUK 45 ; Negative sign key (touche signe '-')

Le premier appel du bloc de fonction provoque l'affichage du texte suivant :

```

PARAMETER INPUT
=====
Value : 567.890
Accept value [CR]

```

La saisie des données s'effectue ensuite conformément au tableau de la page suivante. Seul le champ d'entrée est modifié ; le reste de l'affichage ne change pas au cours de la saisie.

Touche enfoncée	Code ASCII décimal	Affichage du champ d'entrée (max. 8 digits)	Registre/Compteur PARAM	Indicateur IN_BUSY (entrée occupée)
1. appel FB		567.890	567890	0
3	51	3_	567890	1
5	53	35_	567890	1
7	55	357_	567890	1
<-	8	35_	567890	1
<-	8	3_	567890	1
<-	8	_	567890	1
<-	8	567.890	567890	1
1	49	1_	567890	1
2	50	12_	567890	1
3	51	123_	567890	1
4	52	1234_	567890	1
.	54	1234._	567890	1
7	55	1234.7_	567890	1
8	56	1234.78_	567890	1
9	57	1234.789	567890	1
4	52	1234.784	567890	1
<-	8	1234.78_	567890	1
<-	8	1234.7_	567890	1
CR	13	1234.7	1234700	0
1. appel FB		1234.700	1234700	0
-	45	_	1234700	1
8	56	-8_	1234700	1
4	52	-84_	1234700	1
6	54	-846_	1234700	1
CR	13	-846	-846000	0

Utilisation du bloc de fonction dans les programmes utilisateurs

Le bloc de fonction figure dans le fichier D1_INP.SRC. Celui-ci contient également la définition de tous les symboles indispensables au bloc de fonction.

Tous les symboles utilisés globalement par le programme utilisateur sont interprétés dans ce fichier comme des symboles externes (EXTN), qui doivent être définis dans un autre fichier utilisateur. Cela signifie que le fichier D1_INP.SRC ne doit être assemblé qu'une seule fois, puis être relié aux autres fichiers utilisateur.

Symboles globaux : INPUT, CHAN_N, IN_BUSY, RBSY_F, XBSY_F

Si le bloc de fonction est inclus dans le programme utilisateur à l'aide de la directive assembleur \$INCLUDE, les définitions EXTN doivent être effacées ou remplacées par des définitions de symboles locaux.

9. Comparatif PCD7.D202 ↔ PCD7.D250

Fonctions	..D202	..D250
Face frontale		
Affichage Dimensions Touches de fonction Marquage personnalisé Carte de base Interfaces série	4 x 20 caractères 141 x 181 mm 4 avec LED touches de fonction = N° 1 : RS 232	8 x 40 / 4 x 20 caractères (commutables) 305 x 120 mm 8 avec LED touches de fonction et label SAIA = N° 1 : RS 232 ou N° 0 : pour module ..F2.. (RS 422 ou TTY)
Jeu de commandes / Paramétrage	..D202	..D202 plus : – commutation de 8 x 40 à 4 x 20 – positionnement du curseur amélioré (8 x 40) – commutation du port série n° 1 au n° 0 – jeu de caractères complet IBM CodePage 437 – début et fin de caractère inverse – mode transparence
FBs Editeur (bibliothèque de dialogue)	..D202 Ancienne et nouvelle versions	..D202 modifié Editeur de dialogue opérateur (HMI) en préparation

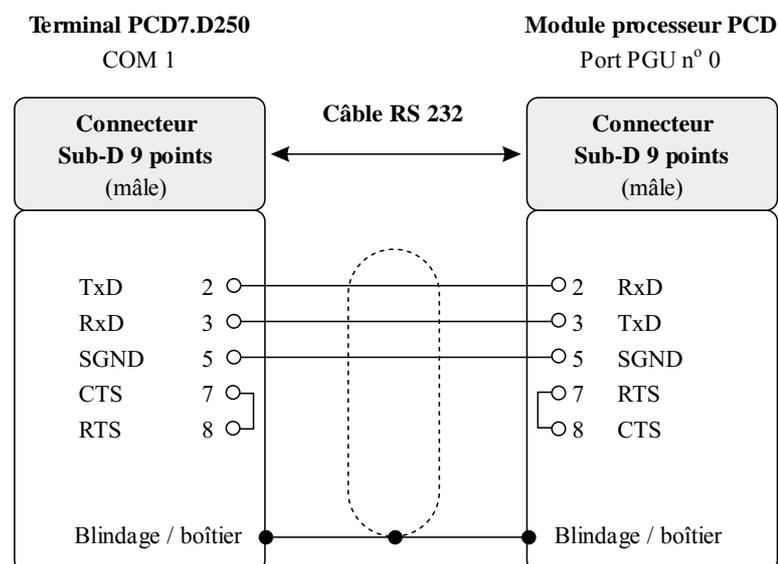
Notes personnelles :

10. Câbles de l'interface série RS 232

Les câbles, de longueur standard 2,5 m, sont dotés d'un double blindage et de connecteurs métallisés.

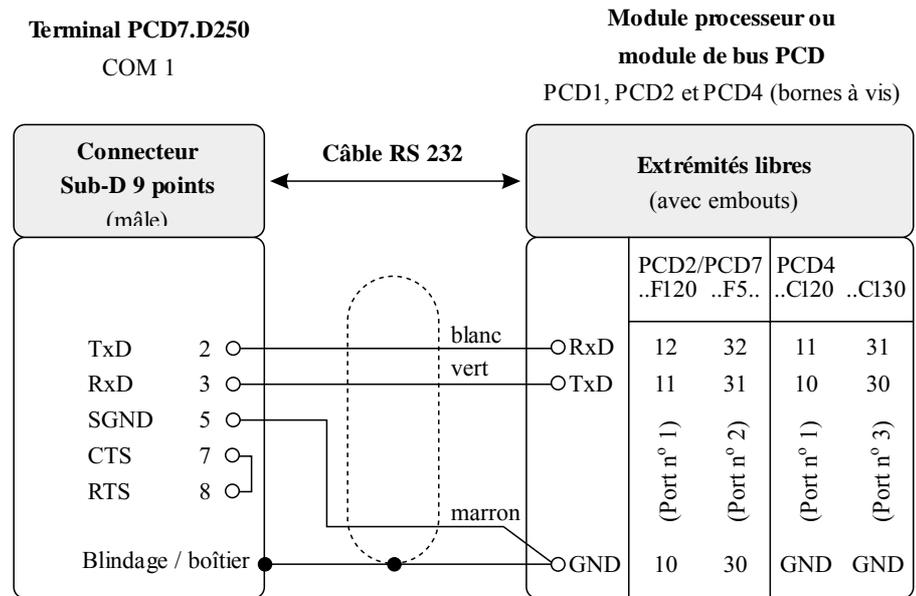
Type PCD7.K412 : Câble de liaison série RS 232
(sans contrôle RTS/CTS)

Raccordement ..D250 → port PGU (n°0) de tous les modules proces-
seur PCD.



Type PCD7.K422 : Câble de liaison série RS 232
(sans contrôle RTS/CTS)

Raccordement ..D250 → module processeur ou module de bus du PCD1, PCD2 ou PCD4. Extrémités de câble libres (avec embouts).



Vos coordonnées :

Société :

Service :

Nom :

Adresse :

Téléphone :

Date :

A renvoyer à :

SAIA-Burgess Electronics SA

18, rue de la Gare

CH-3280 Morat (Suisse)

<http://www.saia-burgess.com>

DIV : Electronic Controllers

Terminal industriel PCD7.D250

Vos commentaires seront les bienvenus pour améliorer la qualité et le contenu de cette documentation SAIA[®] PCD. Nous vous remercions par avance de votre collaboration.

Vos commentaires :