

CARTES PASSIVES
DOCUMENTATION TECHNIQUE

DOCUMENTATION TECHNIQUE DES CARTES PASSIVES

©COPYRIGHT ACKSYS 1998

Ce document contient des informations qui sont protégées par Copyright.

Tout ou partie du présent document ne pourra être reproduit, transcrit, stocké dans n'importe quel système informatique ou autre, traduit dans n'importe quelle langue et n'importe quel langage informatique sans le consentement préalable et écrit de **ACKSYS**, 3 & 5 rue du Stade, BP 80, 78302 POISSY CEDEX.

MARQUES DEPOSEES ®

- **ACKSYS** est une marque déposée de **ACKSYS**.
- IBM P.C., AT sont des marques déposées de International Business Machines Corporation.
- MS-DOS, Windows 95, Windows NT sont des marques déposées de Microsoft Inc.
- UNIX est une marque déposée de OSF.
- SCO est une marque déposée de Santa Cruz Operation.
- INTERACTIVE et 386/IX sont des marques déposées de INTERACTIVE.

NOTICE

ACKSYS ne garantit en aucune façon le contenu du présent document et dégage son entière responsabilité quant à la rentabilité et la conformité du matériel aux besoins de l'utilisateur.

ACKSYS ne pourra en aucun cas être tenu pour responsable des erreurs éventuellement contenues dans ce document, ni des dommages quelle qu'en soit leur importance, du fait de la fourniture, du fonctionnement ou de l'utilisation du matériel.

ACKSYS se réserve le droit de réviser périodiquement ce document, ou d'en changer le contenu, sans aucune obligation pour **ACKSYS** d'en aviser qui que ce soit.

Société **ACKSYS**
3 & 5 rue du Stade
BP 4580
78302 POISSY CEDEX

Téléphone : 33 (0)1-39-11-62-81
Télécopie : 33 (0)1-39-11-47-96
e-mail : acksys@compuserve.com

SOMMAIRE

I. INTRODUCTION	I-1
I.1 TABLEAU DE SÉLECTION DES CARTES PASSIVES	I-2
I.2 AVERTISSEMENT	I-3
I.3 GARANTIE.....	I-3
II. LA CARTE COMFAST.....	II-1
II.1 INTRODUCTION	II-1
II.2 CARACTÉRISTIQUES ÉLECTRIQUES ET MÉCANIQUES	II-1
II.3 INSTALLATION DE LA CARTE	II-2
II.3.1 Emplacement des interrupteurs.....	II-2
II.3.2 Sélection des adresses et interruption	II-2
II.4 RÉPARTITION DES SIGNAUX SUR LES CONNECTEURS	II-4
II.4.1 Connecteur 25 points (COM1 ou COM3).....	II-4
II.4.2 Connecteur 9 points (COM2 ou COM4).....	II-4
II.5 MARQUAGE CE ET CEM	II-4
III. LA CARTE FAST 400.....	III-1
III.1 INTRODUCTION	III-1
III.2 CARACTÉRISTIQUES ÉLECTRIQUES ET MÉCANIQUES	III-1
III.3 INSTALLATION DE LA CARTE	III-3
III.3.1 Emplacement des interrupteurs.....	III-3
III.3.2 Configuration usine.....	III-4
III.3.3 L'interrupteur SW1 - Interruption et adresse des ports.....	III-5
III.3.4 Les interrupteurs SW2-SW3 - Mode RS232, RS422, RS485.....	III-7
III.3.5 Les cavaliers ST1, ST2, ST3, ST4 et le « strap » ST5.....	III-10
III.4 RÉPARTITION DES SIGNAUX SUR LES CONNECTEURS	III-11
III.5 MARQUAGE CE & CEM.....	III-13
IV. LA CARTE 2RS232.....	IV-1
IV.1 INTRODUCTION	IV-1
IV.2 CARACTÉRISTIQUES MÉCANIQUES	IV-1
IV.3 INSTALLATION DE LA CARTE	IV-2
IV.3.1 Emplacement des interrupteurs.....	IV-2
IV.3.2 Sélection des adresses et interruptions.....	IV-3
IV.4 CONFIGURATIONS PARTICULIÈRES.....	IV-3
IV.5 RÉPARTITION DES SIGNAUX SUR LES CONNECTEURS SUBD 9	IV-4
IV.6 MARQUAGE CE ET CEM	IV-4
V. LA CARTE 4RS232.....	V-1
V.1 INTRODUCTION	V-1
V.2 CARACTÉRISTIQUES MÉCANIQUES	V-1
V.3 INSTALLATION DE LA CARTE	V-2
V.3.1 Emplacement des interrupteurs.....	V-2
V.3.2 Sélection des adresses et interruptions.....	V-3
V.3.3 Configurations particulières.....	V-3
V.4 RÉPARTITION DES SIGNAUX SUR LES CONNECTEURS SUBD 9	V-4
V.5 MARQUAGE CE ET CEM	V-4

VI. LA CARTE 8RS232.....	VI-1
VI.1 INTRODUCTION	VI-1
VI.2 INSTALLATION DE LA CARTE	VI-1
VI.2.1 Emplacement des interrupteurs.....	VI-2
VI.2.2 Sélection de l'adresse de base de la carte	VI-3
VI.2.3 Sélection du niveau d'interruption	VI-5
VI.2.4 Sélection des signaux RI ou DSR.....	VI-5
VI.3 RÉPARTITION DES SIGNAUX SUR LES CONNECTEURS SUBD 9	VI-6
VI.4 MARQUAGE CE ET CEM	VI-6
VII. LA CARTE UNX232	VII-1
VII.1 INTRODUCTION	VII-1
VII.2 INSTALLATION DE LA CARTE	VII-1
VII.2.1 Emplacement des interrupteurs.....	VII-2
VII.2.2 Adresse de base de la carte	VII-3
VII.2.3 Sélection 4 ou 8 ports	VII-6
VII.2.4 Sélection du niveau d'interruption	VII-7
VII.2.5 Registres de "masque" et de "polling"	VII-7
VII.2.6 Partage de la ligne d'interruption	VII-9
VII.3 BOÎTIER DE CONNEXION RS232D	VII-10
VII.3.1 Attribution des lignes sur les connecteurs.....	VII-10
VII.3.2 Caractéristiques ESD	VII-11
VII.3.3 Extraits du compte rendu des essais de décharges électrostatiques	VII-11
VII.3.4 L'interface RS422A intégrée.....	VII-12
<i>Configuration de l'interface RS422A :</i>	<i>VII-12</i>
<i>Caractéristiques de l'interface RS422A :</i>	<i>VII-12</i>
VII.4 BOÎTIER DE CONNEXION RS422A	VII-14
VII.4.1 Attribution des lignes sur les connecteurs.....	VII-14
VII.4.2 Raccordement des signaux entre 2 ports série RS422A.....	VII-15
VII.5 OPTIONS POUR CARTES UNX.....	VII-16
VII.6 MARQUAGE CE & CEM	VII-17
VIII. CONFIGURATIONS PARTICULIÈRES UNX232.....	VIII-1
VIII.1 CONFIGURATION SOUS SCO XENIX 2.3.x ET SCO UNIX	VIII-1
VIII.2 CONFIGURATION SOUS 386/IX INTERACTIVE SYSTEMS	VIII-3
VIII.3 CONFIGURATION SOUS PICK	VIII-4
VIII.4 CONFIGURATION SOUS PROLOGUE	VIII-5
VIII.5 CONFIGURATION SOUS QNX DE QUANTUM SOFTWARE	VIII-6
IX. L'INTERFACE RS422A-RS485-V11	IX-1
IX.1 NOTATION DE L'ÉTAT DES SIGNAUX.....	IX-1
IX.2 PARTICULARITÉS DES LIGNES DE TRANSMISSION RS422A ET RS485	IX-3
IX.2.1 Résistance de terminaison :	IX-3
IX.2.2 Polarisation :	IX-3
IX.2.3 Type de câble de raccordement :	IX-3
IX.2.4 Raccordement des masses électriques (GND) :	IX-3
IX.2.5 Blindage :	IX-3
IX.2.6 Masse et blindage :	IX-4
IX.2.7 Longueur de câble pour une liaison RS422A OU RS485.....	IX-4
X. EXTRAITS DE LA NORME EIA	X-1
XI. PERTURBATIONS RADIOELECTRIQUES (FCC).....	XI-2
XII. CARACTÉRISTIQUES DU 16C550	XII-1

XII.1	GÉNÉRALITÉS	XII-1
XII.2	ADRESSAGE DES REGISTRES	XII-2
XII.2.1	Registre de contrôle de ligne.....	XII-4
XII.2.2	Le générateur de Bauds.....	XII-5
XII.2.3	Le registre d'état ligne	XII-6
XII.2.4	Le registre de contrôle du FIFO.....	XII-7
XII.2.5	Le registre d'identification des interruptions	XII-8
XII.2.6	Registre de validation des interruptions.....	XII-9
XII.2.7	Le registre de contrôle modem	XII-11
XII.2.8	Le registre d'état modem	XII-12
XII.3	PARTICULARITÉS DES CARTES 8RS232 ET UNX232.....	XII-13
XIII.	INSTALLATION DES CARTES SOUS WINDOWS NT	XIII-1
XIII.1	INSTALLATION	XIII-1
XIII.2	TEST D'UNE CARTE	XIII-2
XIII.3	RETRAIT D'UNE CARTE.....	XIII-2
XIV.	INSTALLATION DES CARTES SOUS WINDOWS 95.....	XIV-1
XIV.1	CARTES 4 ET 8 VOIES	XIV-1
XIV.2	CARTES 2 VOIES.....	XIV-9
XIV.3	PROBLÈMES D'INSTALLATION.....	XIV-9
XIV.4	SERVICES FICHIERS APPLICABLES AUX PORTS SÉRIE	XIV-10
XIV.5	SERVICES DE COMMUNICATION.....	XIV-10
XV.	PILOTE POUR OS/2.....	XV-1
XV.1	CARACTÉRISTIQUES GÉNÉRALES.....	XV-1
XV.2	INSTALLATION	XV-2
XV.3	SUPPORT DE L'API OS/2	XV-3
XVI.	PILOTE POUR 386/IX ET UNIX SCO.....	XVI-1
XVI.1	CARTES SUPPORTÉES ET CONFIGURATION	XVI-1
XVI.2	INSTALLATION	XVI-1
XVI.3	UTILISATION	XVI-2
XVI.3.1	Détection par génération de caractères.....	XVI-4
XVI.3.2	Influence de l'environnement d'utilisation ; Précautions d'usage	XVI-4
XVI.3.3	Exemple d'utilisation.....	XVI-5
XVII.	PILOTE POUR MS-DOS.....	XVII-1
XVII.1	INTRODUCTION	XVII-1
XVII.2	CONFIGURATION DES CARTES	XVII-3
XVII.2.1	La Carte 2RS232.....	XVII-3
XVII.2.2	La Carte 4RS232.....	XVII-3
XVII.2.3	La Carte 8RS232.....	XVII-3
XVII.2.4	La Carte UNX232	XVII-3
XVII.2.5	La Carte FAST400.....	XVII-4
XVII.2.6	La Carte COMFAST.....	XVII-4
XVII.2.7	L'extension <i>Lite-485</i> pour <i>MCX-Lite</i>	XVII-4
XVII.2.8	L'extension <i>Lite-UNIX</i> pour <i>MCX-Lite</i>	XVII-4
XVII.2.9	L'extension <i>Lite-104</i> pour <i>MCX-Lite</i>	XVII-4
XVII.3	INSTALLATION	XVII-5
XVII.3.1	Paramètres de lancement.....	XVII-5
XVII.3.2	Compatibilité avec UNXDRV.SYS	XVII-7
XVII.3.3	Initialisation du programme de commande.....	XVII-8
XVII.4	UTILISATION DU PROGRAMME DE COMMANDE.....	XVII-9
XVII.4.1	Depuis le processeur de commandes.....	XVII-9

XVII.4.2	Utilisation depuis l'assembleur.....	XVII-10
XVII.4.3	Utilisation depuis le basic	XVII-16
XVII.4.4	Utilisation sous quick basic 4.0 microsoft	XVII-18
XVII.5	LES COMMANDES IOCTL	XVII-19
XVII.5.1	Généralités	XVII-19
XVII.5.2	Description des commandes IOCTL.....	XVII-21
	<i>Commande VINIT</i>	<i>XVII-21</i>
	<i>Commande DIMBF</i>	<i>XVII-23</i>
	<i>Commande VMODE</i>	<i>XVII-24</i>
	<i>Commande STMOD</i>	<i>XVII-25</i>
	<i>Commande TIMEO</i>	<i>XVII-25</i>
	<i>Commande RXENB</i>	<i>XVII-26</i>
	<i>Commande CLRRX</i>	<i>XVII-26</i>
	<i>Commande FLUSH</i>	<i>XVII-26</i>
	<i>Commande TFREE</i>	<i>XVII-27</i>
	<i>Commande CNTRX</i>	<i>XVII-27</i>
	<i>Commande ERLOC</i>	<i>XVII-27</i>
	<i>Commande LSTAT et ISTAT</i>	<i>XVII-28</i>
	<i>Commande BSTAT</i>	<i>XVII-29</i>
	<i>Commande BSIZE</i>	<i>XVII-30</i>
	<i>Commande STSIG</i>	<i>XVII-31</i>
	<i>Commande BREAK</i>	<i>XVII-31</i>
	<i>Commande RELRP</i>	<i>XVII-31</i>
	<i>Commande CFGRP</i>	<i>XVII-32</i>
	<i>Commande EXFMT</i>	<i>XVII-32</i>
	<i>Commande RDREG</i>	<i>XVII-33</i>
	<i>Commande WRREG</i>	<i>XVII-34</i>
XVII.5.3	Exemple de programmation des fonctions ioctl du driver.....	XVII-35
XVII.6	LE PROGRAMME COMCTL.COM.....	XVII-37
XVII.7	INTERRUPTIONS 60H ET 14H.....	XVII-39
XVII.7.1	Description des fonctions.....	XVII-39
	<i>Fonction 00H - Initialisation des paramètres de communication</i>	<i>XVII-39</i>
	<i>Fonction 01H - Envoi d'un caractère</i>	<i>XVII-40</i>
	<i>Fonction 02H - Lecture d'un caractère</i>	<i>XVII-40</i>
	<i>Fonction 03H - Lecture de l'état du port de communication</i>	<i>XVII-40</i>
	<i>Fonction 04H - Initialisation étendue</i>	<i>XVII-41</i>
	<i>Fonction 05H - Contrôle du port de communication</i>	<i>XVII-42</i>
XVII.8	CONFIGURATION DU PROGRAMME DE COMMANDE.....	XVII-43
XVIII.	NOTES.....	XVIII-1
XIX.	FICHE ERREUR.....	XIX-1

I. INTRODUCTION

Vous venez de faire l'acquisition d'une carte de communication ACKSYS et nous vous en remercions. L'objet de cette documentation est de décrire la procédure d'installation des cartes de communication passives de notre gamme et de vous donner tous les éléments indispensables à leur utilisation.

Dans la première partie de ce manuel, vous trouverez les spécifications techniques des cartes 2RS232, 4RS232, ComFast, Fast400, 8RS232 et UNX232. Le chapitre concernant votre carte vous donnera toutes les informations utiles de configuration des adresses et interruptions.

Des informations générales sur les normes en vigueur vous guideront ensuite dans l'utilisation des transmissions différentielles. Un chapitre est consacré au composant 16C550 qui est au cœur de toutes les cartes décrites dans ce manuel. Il vous sera utile pour le développement de vos propres drivers et applications 'pointues' nécessitant une programmation au niveau du composant.

La section suivante vous guidera dans l'installation de votre carte sous Windows 95 et Windows NT. Les cartes passives ACKSYS ne nécessitent pas de pilote spécifique pour fonctionner dans ces environnements, néanmoins, le fichier d'installation des cartes 4 et 8 ports pour Windows 95 vous permettra une installation plus rapide.

Les pilotes de périphériques pour UNIX (fourni en option) et OS/2 sont décrits avec leur installation, leurs caractéristiques et leur programmation.

Vient ensuite la description du pilote pour MS-DOS. Ce dernier peut vous faire gagner un temps précieux lors de vos développements sous DOS. Sa simplicité de programmation sous Quick Basic par exemple sera particulièrement appréciée pour le développement rapide de petites applications.

I.1 TABLEAU DE SELECTION DES CARTES PASSIVES

CARTE	COMFAST	FAST400	2RS232	4RS232	8RS232	UNIX232
2 Ports	♦	♦	♦	-	-	-
4 Ports	-	-	-	♦	-	♦
8 Ports	-	-	-	-	♦	♦
RS232	♦	♦	♦	♦	♦	♦
RS422	-	♦	-	-	-	♦
RS485	-	♦	-	-	-	◊
Isolement galvanique	-	♦	-	-	-	◊
COM1, COM2	♦	♦	♦	♦	-	-
COM3, COM4	♦	♦	♦	♦	-	-
Adresses consécutives	-	-	-	♦	♦	♦
IRQ Individuelles	3 et 4	3 et 4	3, 4, 5, 7, 9, 10, 11, 15	3, 4, 5, 7, 9, 10, 11, 15	-	-
IRQ Communes	5	9, 10, 11, 12	3, 4, 5, 7, 9, 10, 11, 15	3, 4, 5, 7, 9, 10, 11, 15	3, 4, 5, 7, 10, 11, 12, 15	3, 4, 5, 6, 7, 9, 10, 11, 12, 15
Registre de masque	-	-	-	-	-	♦
Registre de polling	-	-	-	-	♦	♦
Compatible 16C550	♦	♦	♦	♦	♦	♦
Horloge 1.8432 MHz	♦	♦	♦	♦	♦	♦
Horloge 6 MHz	-	♦	-	-	-	-
Horloge 8 MHz	-	◊	-	-	-	-
Compatible Windows 95	♦	♦	♦	♦	♦	♦
Compatible Windows NT	♦	♦	♦	♦	♦	♦
Compatible UNIX	♦	♦	♦	♦	♦	♦
Pilote Dos	♦	♦	♦	♦	♦	♦
Pilote OS/2	♦	♦	♦	♦	♦	♦
Pilote industriel UNIX	-	-	-	◊	◊	◊

◊ Option

I.2 AVERTISSEMENT

Le courant électrique provenant de l'alimentation, du téléphone et des câbles de transmission, peut présenter un danger. Connectez et déconnectez les câbles uniquement lorsque la machine dans laquelle est installée la carte est hors tension. Ne touchez pas aux câbles pendant un orage.

DANGER

**NE JAMAIS BRANCHER NI DEBRANCHER
LES CONNECTEURS SUB D LORSQUE
LA MACHINE EST SOUS TENSION**

**NE JAMAIS TOUCHER AUX
CONNECTEURS NI AUX CABLES PENDANT
UN ORAGE**

Les pannes engendrées par une manipulation répétée des connecteurs SUB D25 et SUB D9 avec la machine sous tension sont fréquentes et détruisent le plus souvent les amplificateurs de ligne contenus dans le dispositif de connexion.

La plupart des pannes sont facilement évitables à condition de respecter les règles énoncées ci-dessus.

RESPECTEZ LES !

I.3 GARANTIE

La période de garantie est définie par nos conditions générales de garantie c'est-à-dire :

Garantie de 5 ans pièces et main-d'œuvre contre tout vice de fabrication ou de fonctionnement à l'exception des pannes engendrées par une utilisation non conforme ou bien par l'action excessive d'un agent ou d'une circonstance naturelle.

Attention, la garantie ne couvre donc pas la destruction des amplificateurs de ligne due à une surtension (voir l'avertissement ci-dessus).

Les réparations sous garantie sont effectuées en nos locaux dans un délai moyen de deux jours ouvrés.

II. LA CARTE COMFAST

II.1 INTRODUCTION

La carte COMFAST est une carte de communication asynchrone 2 lignes RS232D haute vitesse. Elle fonctionne avec la famille des PC IBM et compatibles et occupe un emplacement 8 bits du bus ISA dans la machine.

La carte COMFAST est bâtie autour de 2 éléments de communication asynchrone avec FIFOS de type 16C550, répandu sur toutes les machines P.C., dont les caractéristiques seront décrites plus loin dans ce manuel.

Son utilisation principale est la connexion de modems haute vitesse (NUMERIS & V42) qui ne peuvent pas fonctionner correctement à 57600 bauds avec des unités de communications classiques.

Cette carte est supportée directement par la majorité des systèmes d'exploitation tels que Windows 95®, Windows NT®, 386/IX...

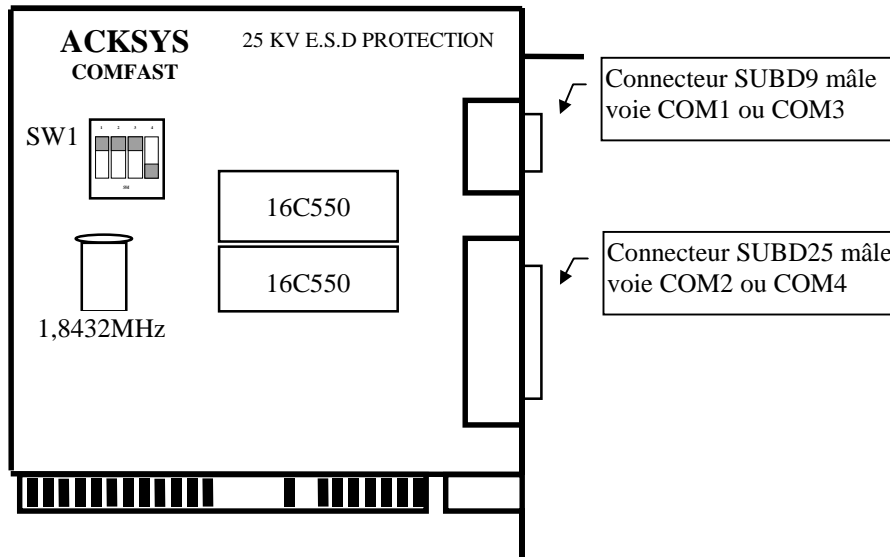
D'autre part, comme la plupart des cartes ACKSYS, la carte COMFAST bénéficie de systèmes de protection lui permettant de résister à des décharges électrostatiques de 25 KV sur les lignes RS232D.

II.2 CARACTERISTIQUES ELECTRIQUES ET MECANIQUES

Dimensions	107×107 mm
Consommation max	+5V - 170 mA +12V - 35 mA -12V - 35 mA total 1,7 W max.
Température	en fonctionnement : 0 à +60°C de stockage : -25 à +70°C
Connecteurs	1 SUBD9 et 1 SUBD25 mâles

II.3 INSTALLATION DE LA CARTE

II.3.1 Emplacement des interrupteurs



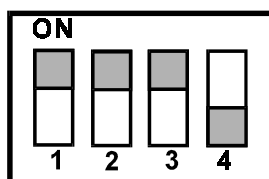
II.3.2 Sélection des adresses et interruption

La carte COMFAST peut utiliser pour chacun de ses 2 ports les adresses et niveaux d'interruptions des ports COM1 et COM2 ou COM3 et COM4

Il est aussi possible d'utiliser les 2 ports de la carte sur le niveau d'interruption commun IRQ5.

Ces configurations d'adresses et d'interruptions sont programmables à l'aide du bloc de quatre mini interrupteurs SW1

SW1-1	OFF	: Mode COM1 et COM2
	ON	: Mode COM3 et COM4
SW1-2	OFF	: Interruption IRQ4 (COM1 et COM3) inactive
	ON	: Interruption IRQ4 (COM1 et COM3) active
SW1-3	OFF	: Interruption IRQ3 (COM2 et COM4) inactive
	ON	: Interruption IRQ3 (COM2 et COM4) active
SW1-4	OFF	: Interruption IRQ5 inactive
	ON	: Interruption IRQ5 active pour les 2 ports

CONFIGURATION A LA LIVRAISON DE SW1

Mode COM3/COM4,
IRQ 3 et 4 actives, IRQ5 inactive

Rappel :

Les adresses de base et niveaux d'interruptions des ports COM1, COM2, COM3 et COM4 selon les spécifications IBM sont les suivants :

COM1.....	3F8h / IRQ4
COM2.....	2F8h / IRQ3
COM3.....	3E8h / IRQ4
COM4.....	2E8h / IRQ3

Les spécifications concernant les ports COM3 et COM4 est intervenue assez tardivement (lors de la sortie de MS-DOS version 4), ce qui nous conduit à un signaler un certain nombre de problèmes :

En effet, sur un bus ISA, il n'est pas possible de partager une même ligne d'interruption entre deux périphériques (ce qu'il est possible de faire sur les machines équipées des bus EISA ou MCA).

Par conséquent, si votre machine dispose déjà en standard des ports COM1 et COM2 vous pouvez avoir des difficultés à utiliser les ports COM1 et COM3 ou COM2 et COM4 simultanément. Vous pouvez dans ce cas utiliser l'interruption IRQ5 pour les deux ports supplémentaires.

En revanche, si votre machine ne dispose d'aucun port de COM ou que vous pouvez les désactiver, vous aurez la possibilité d'installer une ou deux cartes COMFAST sans aucun problème, car ces dernières supportent le partage des interruptions entre elles.

II.4 REPARTITION DES SIGNAUX SUR LES CONNECTEURS

II.4.1 Connecteur 25 points (COM1 ou COM3)

No	FONCTION
1	PG
2	TXD
3	RXD
4	RTS
5	CTS
6	DSR
7	MASSE 0V
8	CD
9	N.C
10	N.C
11	N.C
12	N.C
13	N.C

No	FONCTION
14	N.C
15	N.C
16	N.C
17	N.C
18	N.C
19	N.C
20	DTR
21	N.C
22	RI
23	N.C
24	N.C
25	N.C

II.4.2 Connecteur 9 points (COM2 ou COM4)

No	FONCTION
1	CD
2	RXD
3	TXD
4	DTR
5	MASSE 0V
6	DSR
7	RTS
8	CTS
9	RI

II.5 MARQUAGE CE ET CEM

Nous consulter.

III. LA CARTE FAST 400

III.1 INTRODUCTION

La carte FAST400 est une carte de communication conçue autour d'une UART 16C552 intégrant 2 ports série de type 16550, avec FIFOS, et un port parallèle. Elle est équipée d'un oscillateur à 6 MHz (8 MHz en option) autorisant des vitesses de communication en RS422 ou RS485 pouvant atteindre 375 Kbps (500 Kbps avec l'oscillateur à 8 MHz). Elle dispose également d'un oscillateur classique à 1,8432 MHz pour une parfaite compatibilité avec les ports COM standard.

La carte FAST400 ajoute donc à votre PC deux ports série sécurisés asynchrones à hauts débit, compatibles COM1/COM2 ou COM3/COM4, et un port parallèle bidirectionnel compatible LPT.

Elle vous permet de piloter des lignes de communication industrielles en RS232, RS422 ou RS485 en toute sécurité. Comme la plupart des cartes ACKSYS, la carte FAST400 est équipée de systèmes de protection contre les décharges électrostatiques et les surtensions transitoires. De plus, elle assure un isolement galvanique entre les circuits d'émission réception et l'interface P.C., jusqu'à 1200 V_{eff}.

Cette carte est supportée directement par la majorité des systèmes d'exploitation tels que Windows 95®, Windows NT®, 386/IX ...

Elle offre différentes possibilités de connectique au niveau des ports de communication :

- 2 connecteurs SUBD 25 points femelles pour le port parallèle.
- 3 connecteurs (1 SUBD 9 mâle non isolable, et 2 SUBD 25 mâles isolables) pour les deux ports série.

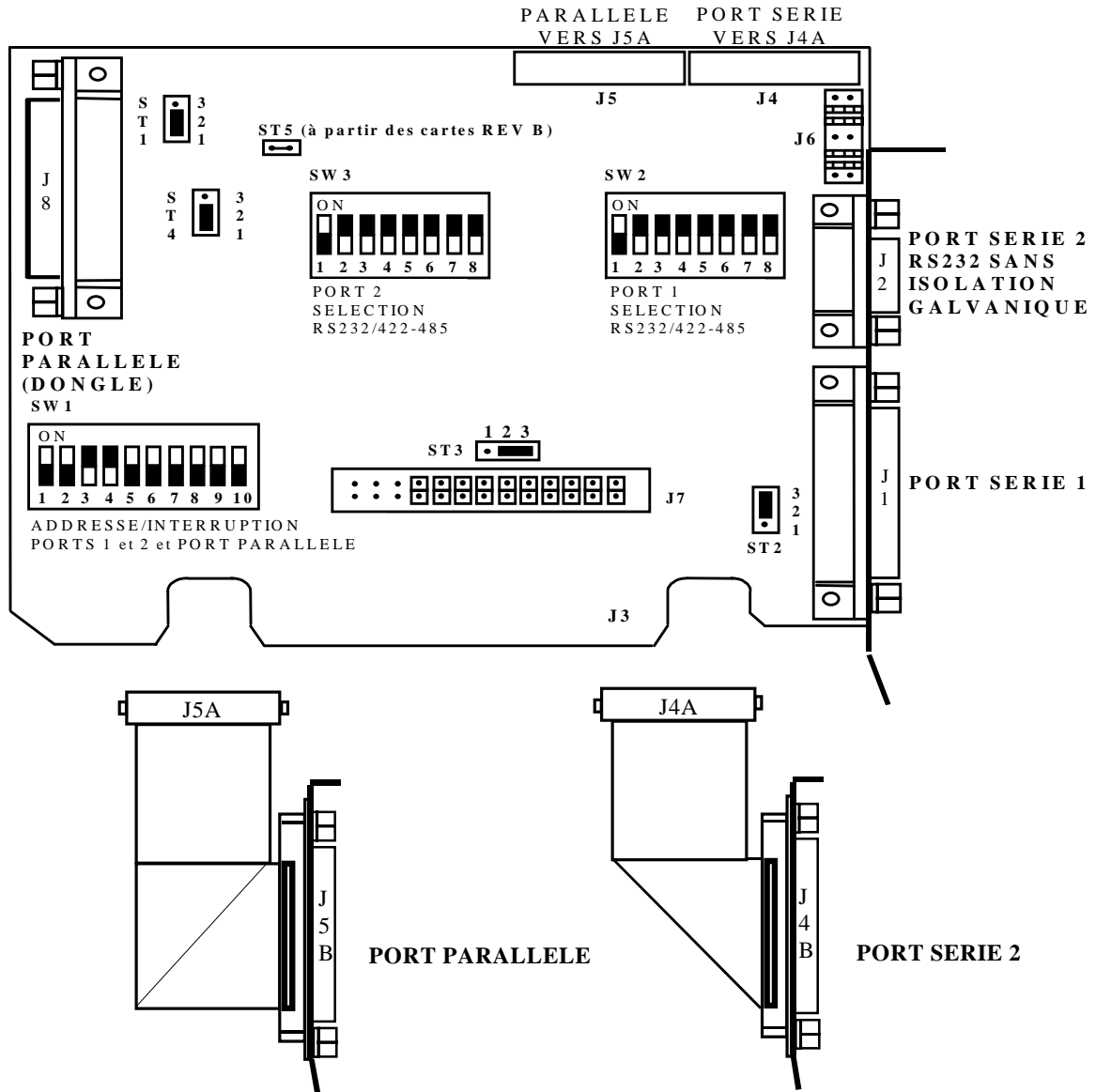
III.2 CARACTERISTIQUES ELECTRIQUES ET MECANIQUES

Type de transmission	Asynchrone, full duplex, half duplex et simplex.
Type de l'interface	- EIA RS232D/CCITT V24, EIA 574 - EIA RS422A - EIA RS485/CCITT V11
Signaux avec isolement galvanique	RX et TX, ±RX et ±TX
Signal de contrôle	RTS permettant de valider la transmission. - circuit RTS fermé (actif) : transmetteur activé. - circuit RTS ouvert (inactif) : transmetteur inhibé.
Signaux sans isolement galvanique	RTS, DTR, CTS, DCD, RI et DSR
Débit maximum	115.2 Kbps (Horloge 1.8432 MHz) 375 Kbps (Horloge 6 MHz) en RS422-RS485 512 Kbps (Horloge 8 MHz) en RS422-RS485

Tension maximale en mode commun (RS422A-RS485)	$\pm 7V$
Charge maximale en RS422A	10 récepteurs (limité par la norme EIA)
Charge maximale en RS485	32 émetteurs/récepteurs (limité par la norme EIA)
Distance maximale de transmission, interfaçage ligne (RS422A-RS485)	1200 m pour un câble de jauge 24 (0,22 mm ²) et d'une capacité de 50 pF/m entre conducteurs (limité par la norme EIA, dépendante de la vitesse et du type de câble utilisé)
Type de câble à utiliser (RS422A-RS485)	Paire(s) torsadée(s) jauge 24, 50 pF/m, impédance nominale 120 Ω . En milieu industriel très perturbé, l'utilisation d'un écran de masse est obligatoire. Capacité entre écran de masse et conducteurs : 75pF/m
Distance de transmission, interfaçage RS232D	15 m minimum
Protection contre les surtensions transitoires (RS422A-RS485)	Par transils, tension de claquage : - $\pm 7V$ en mode commun. - $\pm 14V$ en mode différentiel. - Capacité d'absorption : 0,3 kW pendant 1 mS.
Isolement galvanique (tension d'essai 1 minute) RS422A-RS485 et RS232D	Optocoupleurs et convertisseur dc/dc isolement 2.5 kV _{eff} & 3 kV _{dc} 1200V _{eff} mini.
Immunité aux transitoires (RS422A-RS485 et RS232D avec isolement)	10000 V/ μ s à V _{cm} = 1000 V
Consommation	160mA/5V et 15mA/ $\pm 12V$, soit 1,25 W max.
Poids	230 g max
Dimensions hors tout	H x L = 121mm x 158mm
Plage de température	-5°C à +65°C en fonctionnement
Humidité	0 à 95 % RH, sans condensation

III.3 INSTALLATION DE LA CARTE

III.3.1 Emplacement des interrupteurs



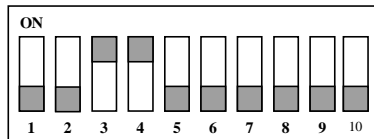
J1	= SUB D 25 MALE - PORT SERIE 1 - RS232/422-485	ST1	= MODE INTERRUPTION PORT PARALLELE
J2	= SUB D 9 MALE - PORT SERIE 2 - RS232 UNIQUEMENT SANS ISOLATION GALVANIQUE	ST2	= ALIMENTATION AUXILIAIRE (+12VOUT)
J3	= CONNECTEUR BUS ISA	ST3	= SELECTION ISOLATION GALVANIQUE
J4	= HE10 26 MALE - PORT SERIE 2 - LIAISON VERS J4A PUIS J4B	ST4	= SELECTION HORLOGE UART
J4B	= SUB D 25 MALE - PORT SERIE 2 - RS232/422-485	ST5	= SUPPRESSION PORT PARALLELE
J5	= HE10 26 MALE - PORT PARALLELE - LIAISON VERS J5A PUIS J5B	SW1	= SELECTION ADRESSE & INTERRUPTION
J5B	= SUD D 25 FEMELLE - PORT PARALLELE	SW2	= PORT 1 - SELECTION MODE RS232 OU RS422-485 & CONFIGURATION RS422-485
J6 & J7	= CONNECTEURS CARTE OPTION	SW3	= PORT 2 - SELECTION MODE RS232 OU RS422-485 & CONFIGURATION RS422-485
J8	= SUD D 25 FEMELLE - PORT PARALLELE (CLEF DE PROTECTION LOGICIELS)		

III.3.2 Configuration usine

ST1 :	1-2	Configuration de l'interruption du port parallèle en mode « latched »
ST2 :	2-3	Alimentation +12V sur la broche n°9 des connecteurs J1 et J4B
ST3 :	2-3	Isolation galvanique des ports série 1 (connecteur J1) et 2 (connecteur J4B uniquement)
ST4 :	1-2	Sélection horloge 1.8432 MHz

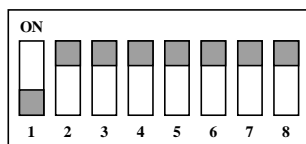
Un strap « ST5 » permet de supprimer le port parallèle de la carte FAST400 en cas de conflit.

SW1 (ports 1&2, port parallèle)



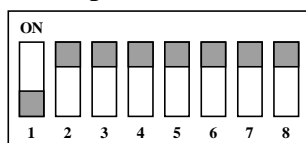
- SW1-1 : Ports série 1 et 2 en mode COM1/COM2
- SW1-2 : Port parallèle à l'adresse 3BCh
- SW1-3 : IRQ 3 port série 2 active
- SW1-4 : IRQ 4 port série 1 active
- SW1-5 : IRQ 9 port série 1 et 2 inactive
- SW1-6 : IRQ10 port série 1 et 2 inactive
- SW1-7 : IRQ11 port série 1 et 2 inactive
- SW1-8 : IRQ12 port série 1 et 2 inactive
- SW1-9 : IRQ7 port parallèle inactive
- SW1-10 : IRQ5 port parallèle inactive

SW2 (port 1)



- SW2-1 : Mode RS422-RS485
- SW2-2 : Transmission contrôlée par RTS
- SW2-3,4,5: Mode RS485 sans écho
- SW2-6 : Avec résistance de terminaison
- SW2-7,8 : Avec polarisation de ligne

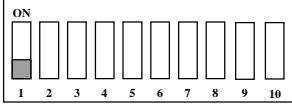
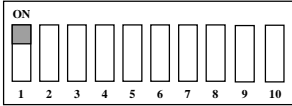
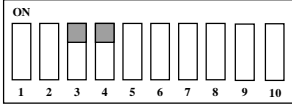

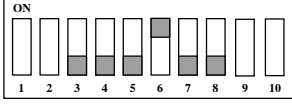
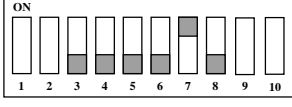
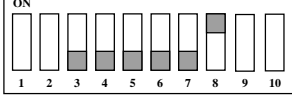

SW3 (port 2)



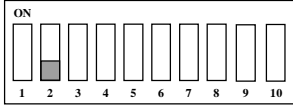
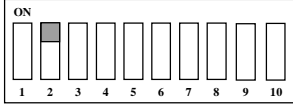
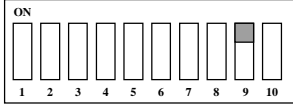
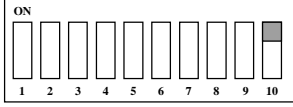

- SW2-1 : Mode RS422-RS485
- SW2-2 : Transmission contrôlée par RTS
- SW2-3,4,5: Mode RS485 sans écho
- SW2-6 : Avec résistance de terminaison
- SW2-7,8 : Avec polarisation de ligne

III.3.3 L'interrupteur SW1 - Interruption et adresse des ports

Interruptions et adresses des ports série :

	<p>MODE COM1 & COM2 (SW1-1 OFF) PORT 1 ADRESSE 3F8h PORT 2 ADRESSE 2F8h</p>
	<p>MODE COM3 & COM4 (SW1-1 ON) PORT 1 ADRESSE 3E8h PORT 2 ADRESSE 2E8h</p>
	<p>INTERRUPTION IRQ 3 PORT 2 ACTIVE (SW1- 3 ON) INTERRUPTION IRQ 4 PORT 1 ACTIVE (SW1- 4 ON)</p>
	<p>INTERRUPTION IRQ 9 PORT 1 & PORT 2 ACTIVE (SW1- 5 ON)</p>
	<p>INTERRUPTION IRQ 10 PORT 1 & PORT 2 ACTIVE (SW1- 6 ON)</p>
	<p>INTERRUPTION IRQ 11 PORT 1 & PORT 2 ACTIVE (SW1- 7 ON)</p>
	<p>INTERRUPTION IRQ 12 PORT 1 & PORT 2 ACTIVE (SW1- 8 ON)</p>
	<p>INTERRUPTION INACTIVE SWITCH SUR OFF</p>

L'interrupteur SW1 - Interruption et adresse du port parallèle

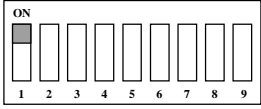
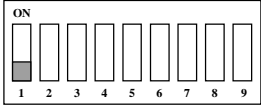
	PORT PARALLELE ADRESSE 3BCh (SW1-2 OFF)
	PORT PARALLELE ADRESSE 378h (SW1-2 ON)
	INTERRUPTION IRQ 7 PORT PARALLELE ACTIVE (SW1- 9 ON)
	INTERRUPTION IRQ 5 PORT PARALLELE ACTIVE (SW1- 10 ON)
	INTERRUPTION INACTIVE SWITCH SUR OFF

III.3.4 Les interrupteurs SW2-SW3 - Mode RS232, RS422, RS485

Le bloc SW2 configure le port N°1.

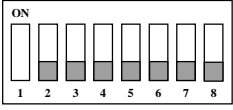
Le bloc SW3 configure le port N°2.

Sélection du mode des ports série (SWx désigne SW2 ou SW3) :

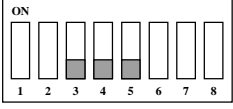
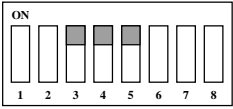

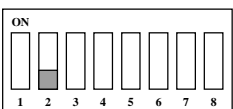
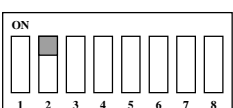
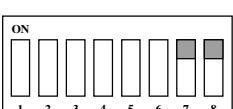
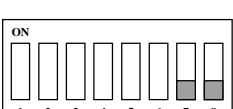
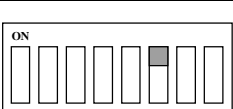
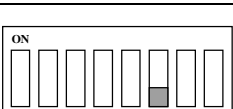
	MODE RS232 (SW _x -1 ON)
	MODE RS422-485 (SW _x -1 OFF)

Attention le connecteur SUB D 9 points J2 du port 2 ne peut être utilisé qu'en mode RS232 non isolé.

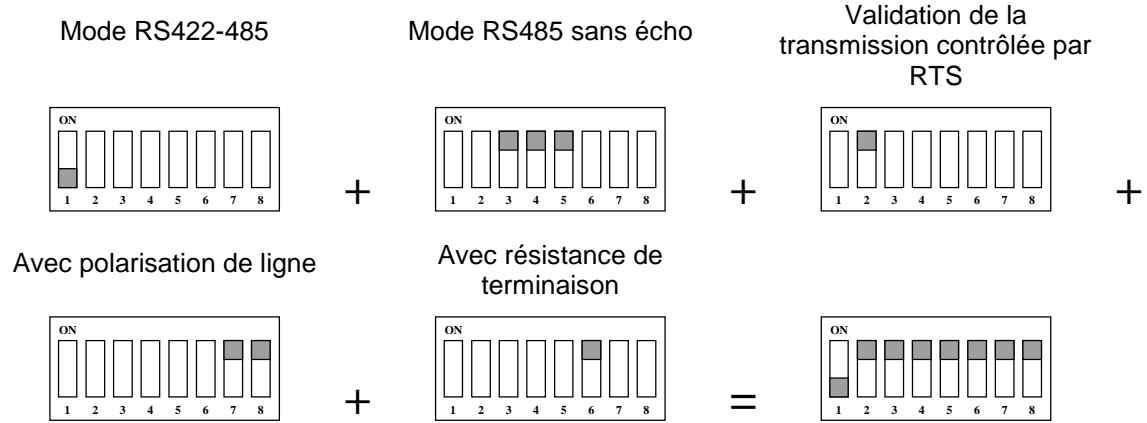
Configuration du mode RS232 des ports série (SWx désigne SW2 ou SW3) :

	PAS DE SIGNIFICATION PARTICULIERE (SWx-2, SWx-3, SWx-4, SWx-5, SWx-6, SWx-7, SWx-8 OFF)
---	---

Configuration du mode RS422 ou RS485 des ports série (SWx désigne SW2 ou SW3) :

	MODE RS422A (SWx-3,SWx-4,SWx-5 OFF)
	MODE RS485 SANS ECHO (SWx-3,SWx-4,SWx-5 ON)
	MODE RS485 AVEC ECHO (SWx-3 OFF, SWx-4 ET SWx-5 ON)
	VALIDATION DE LA TRANSMISSION EN PERMANENCE (SWx-2 OFF)
	VALIDATION DE LA TRANSMISSION CONTROLEE PAR RTS (SWx-2 ON)
	AVEC POLARISATION DE LIGNE (SWx-7 ET SWx-8 ON)
	SANS POLARISATION DE LIGNE (SWx-7 ET SWx-8 OFF)
	AVEC RESISTANCE DE TERMINAISON (SWx-6 ON)
	SANS RESISTANCE DE TERMINAISON (SWx-6 OFF)

Exemple de configuration de SW2 ou SW3



III.3.5 Les cavaliers ST1, ST2, ST3, ST4 et le « strap » ST5

- **Le cavalier ST1 - Configuration du mode d'interruption du port parallèle**

ST1 en position 1-2 : Latched mode

ST1 en position 2-3 : Ack. Interrupt mode

- **Le cavalier ST2 - Alimentation auxiliaire +12V broche N°9 des connecteurs J1 et J4B**

ST2 en position 1-2 : Pas d'alimentation

ST2 en position 2-3 : Alimentation présente

- **Le cavalier ST3 - Sélection isolation galvanique 1200 V_{eff} des ports série n°1 et n°2**

ST3 en position 1-2 : sans isolation galvanique

ST3 en position 2-3 : avec isolation galvanique

Attention, le connecteur J2 (SUBD 9) du port 2 ne peut pas être utilisé en mode isolé.

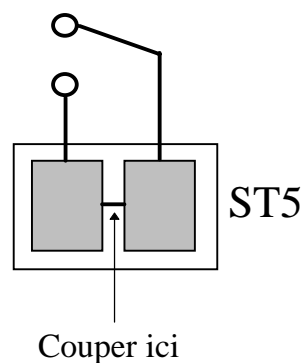
- **Le cavalier ST4 - Sélection horloge UART**

ST4 en position 1-2 : 1,8432 MHz, débit maximum : 115,2 Kb/s

ST4 en position 2-3 : 6 MHz (ou 8MHz) , débit maximum : 375 Kb/s (ou 512Kb/s)

- **Le strap ST5 - Suppression du port parallèle (à partir des cartes REV B)**

En cas de conflit avec le port parallèle du P.C., il est possible de supprimer le port parallèle de la carte en coupant le strap ST5 à l'aide d'un cutter.



III.4 REPARTITION DES SIGNAUX SUR LES CONNECTEURS

Les tableaux ci-après donnent la répartition des signaux sur chaque connecteur 9 points et 25 points de la carte FAST400 :

- **Connecteurs J1 et J4B**

Ces connecteurs sont des SUBD 25 points mâles. Ils sont respectivement connectés aux ports série n°1 et n°2.

CONNECTEURS J1 et J4B - PORT N°1 et N°2 - SUBD 25 mâle

Broche N°	Désignation des signaux		EIA RS232D	CCITT V 24	EIA 422A	EIA 485	CCITT V 11
1	PG	Protection Ground	AA	101			
2 ^(IG)	TXD	Transmitted Data	BA	103			
3 ^(IG)	RXD	Received Data	BB	104			
4	RTS	Request To Send	CA	105			
5	CTS	Clear to Send	CB	106			
6	DSR	Data Set Ready	CC	107			
7	GND	Signal Ground	AB	102			
8	DCD	Data Carrier Detect	CF	109			
9	+12	Vout 200mA					
12 ^(IG)	-TX (RS 422A)	-TX/-RX (RS 485)			B	B/B'	B
13 ^(IG)	+TX (RS 422A)	+TX/+RX (RS 485)			A	A/A'	A
20	DTR	Data Terminal Ready	CD	108/2			
21 ^(IG)	-RX (RS 422A)				B'		B'
22	RI	Ring Indicator	CE	125			
23 ^(IG)	+RX (RS 422A)				A'		A'
25 ^(IG)	GNDI	Isolated Ground					

+RX -RX = Réception de la ligne RS422A
+TX -TX = Transmission de la ligne RS422A
+TX/+RX -TX/-RX = Transmission/réception de la ligne RS485

(IG) Ces signaux sont isolés galvaniquement si le cavalier ST3 est en position 2-3. Dans ce cas la masse de référence est sur la broche 25 et non plus sur la broche 7.

- **Connecteur J2**

Ce connecteur est un SUBD 9 points mâle. Il est connecté au port série n°2.

CONNECTEUR J2 - PORT N°2 RS232 sans isolation galvanique- SUBD 9 mâle

Broche N°	Désignation des signaux		EIA/TIA 574
3	TXD	Transmitted Data	103
2	RXD	Received Data	104
7	RTS	Request To Send	105
8	CTS	Clear To Send	106
6	DSR	Data Set Ready	107
5	GND	Signal Ground	102
1	DCD	Data Carrier Detect	109
4	DTR	Data Terminal Ready	108/2
9	RI	Ring Indicator	125

III.5 MARQUAGE CE & CEM**DECLARATION  DE CONFORMITE**

NOUS : **ACKSYS** 3-5 rue du Stade 78302 Poissy - France

déclarons que le produit :

FAST400

est conforme aux dispositions des :

Directives du Conseil de l'Union Européenne n° **89/336/CEE** du **3 mai 1989** (concernant le rapprochement des législations des Etats membres relatives à la **Compatibilité ElectroMagnétique**) et n° **73/23/CEE** (**Directive Basse Tension**) du **19 février 1973**, modifiées par la Directive **93/68/CEE** (**Marquage CE de Conformité**) du **22 juillet 1993**.

Cette conformité est présumée par la référence aux spécifications suivantes :

- ⇒ Norme **NF EN 55022 (décembre 1994)**
Limites et méthodes de mesure des caractéristiques de perturbations radioélectriques produites par les Appareil de Traitement de l'Information - Norme produit **émission**
- ⇒ Norme **NF EN 50082-2 (juin 1995)**
Compatibilité électromagnétique - Norme générique **immunité**
Partie 2 : **Environnement industriel**
- ⇒ Norme **NF EN 60950 (janvier 1993) + Amendements A1 (mars 1993) et A2 (octobre 1993)**
Sécurité des matériels de traitement de l'information y compris les matériels de bureau électriques

Essais réalisés dans les laboratoires :

ACKSYS 78302 POISSY

HAZTEC 91962 COURTABOEUF

Fait à POISSY, le 30 juillet 1996

Luciano IASCHI
Directeur
Technique

Réf. : DECLARATION CE FAST400 JUILLET 1996

COMPATIBILITE ELECTROMAGNETIQUE Selon Directive **89/336 CEE** modifiée par la Directive **93/68 CEE**
CARTE DE COMMUNICATION

IMMUNITE: ENVIRONNEMENT INDUSTRIEL

/

EMISSION: ENVIRONNEMENT RESIDENTIEL

MARQUAGE CE ANNEE 1996		ESSAIS D'IMMUNITE								ESSAIS D'EMISSION		
		RAYONNEE	RAYONNEE	CONDUITE	CONDUITE	CONDUITE				CONDUITE	RAYONNEE	
<i>folio</i> __1/1__ <i>Rév.</i> _26/07_												
C.E.M. selon: NF EN 50082-2 & NF EN 55022		DECHARGES ELECTRO- STATIQUES	CHAMPS FREQ RADIO MODULE EN AMPLITUDE	CHAMPS FREQ RADIO MODULE EN IMPULSION	TRANSITOIRES RAPIDES EN SALVES	ONDES DE CHOC	CHAMPS FREQ RADIO INDUITS	TENSIONS, CREUX COUPURES ET VARIATIONS BREVES	TENSION, VARIATIONS PERMANENTES	classe B	classe B	
DESIGNATION DU PRODUIT ACKSYS	APPOSITION DU MARQUAGE CE	NF EN 61000-4-2 8 et 4 KV	ENV 50140 10 V/m 80 à 1000 MHz	ENV 50204 10 V/m 900 MHz	NF EN 61000-4-4 2 KV	NF EN 61000-4-5 1,2/50 (8/20) µs 0,5 à 4 KV <i>facultatif</i>	ENV 50141 10 V 0,15 à 80 MHz	NF EN 61000-4-11 -30 à -100% 10 ms à 5 s <i>facultatif</i>	GT6 SC77A ± 20% Vdc ± 10% Vac <i>facultatif</i>	NF EN 55022 56 à 46 dB µV/m 46 et 50 dB µV/m	NF EN 55022 30 et 37 dB µV/m	
FAST400	OUI	C. 16,5 KV air 9 KV cont.	C.	C.	C.	A.E.	C.	N.A.	N.A.	N.A.	4,5 dB µV/m en dessous de la limite B RETENUE CLASSE A	

Ordinateurs utilisés pour les essais :

- P.C. ISA-PCI PENTIUM 75 MHz réf. PCD-5H marque SIEMENS-NIXDORF, écran réf. MCM 1503 marque SIEMENS-NIXDORF.
- P.C. ISA-PCI PENTIUM 75 MHz réf. P 75 marque BRETT COMPUTERS, écran réf. MCM 1503 marque SIEMENS-NIXDORF.
- P.C. ISA-PCI PENTIUM 90 MHz réf. 8513P-250 marque TEXAS MICRO, écran réf. 1014R marque TEXAS MICRO.

Essais effectués avec le produit équipé de câbles et de connecteurs **blindés**, avec reprise de blindage à 360°.

Les valeurs des mesures en immunité indiquées dans le tableau, sont des valeurs d'essai et pas forcément des valeurs limites.

CLASSE A :

AVERTISSEMENT

Ce produit est un produit de Classe A. Dans un environnement résidentiel ce produit peut provoquer des brouillages radioélectriques. Dans ce cas, il est peut être demandé à l'utilisateur de prendre des mesures appropriées.

A.E.= A Effectuer ultérieurement (*facultatif*)

C.= Conforme

CO.= Composant pour maintenance ou a intégrer dans un système

N.A.= Non Applicable

N.C.= Non Conforme

IV. LA CARTE 2RS232

IV.1 INTRODUCTION

La carte 2RS232 est une carte de communication asynchrone 2 ports RS232D. Elle fonctionne avec la famille des PC IBM et compatibles (et PS 8530) et occupe un emplacement du bus ISA de la machine.

Cette carte est supportée directement par la majorité des systèmes d'exploitation tels que Windows 95®, Windows NT®, 386/IX ...

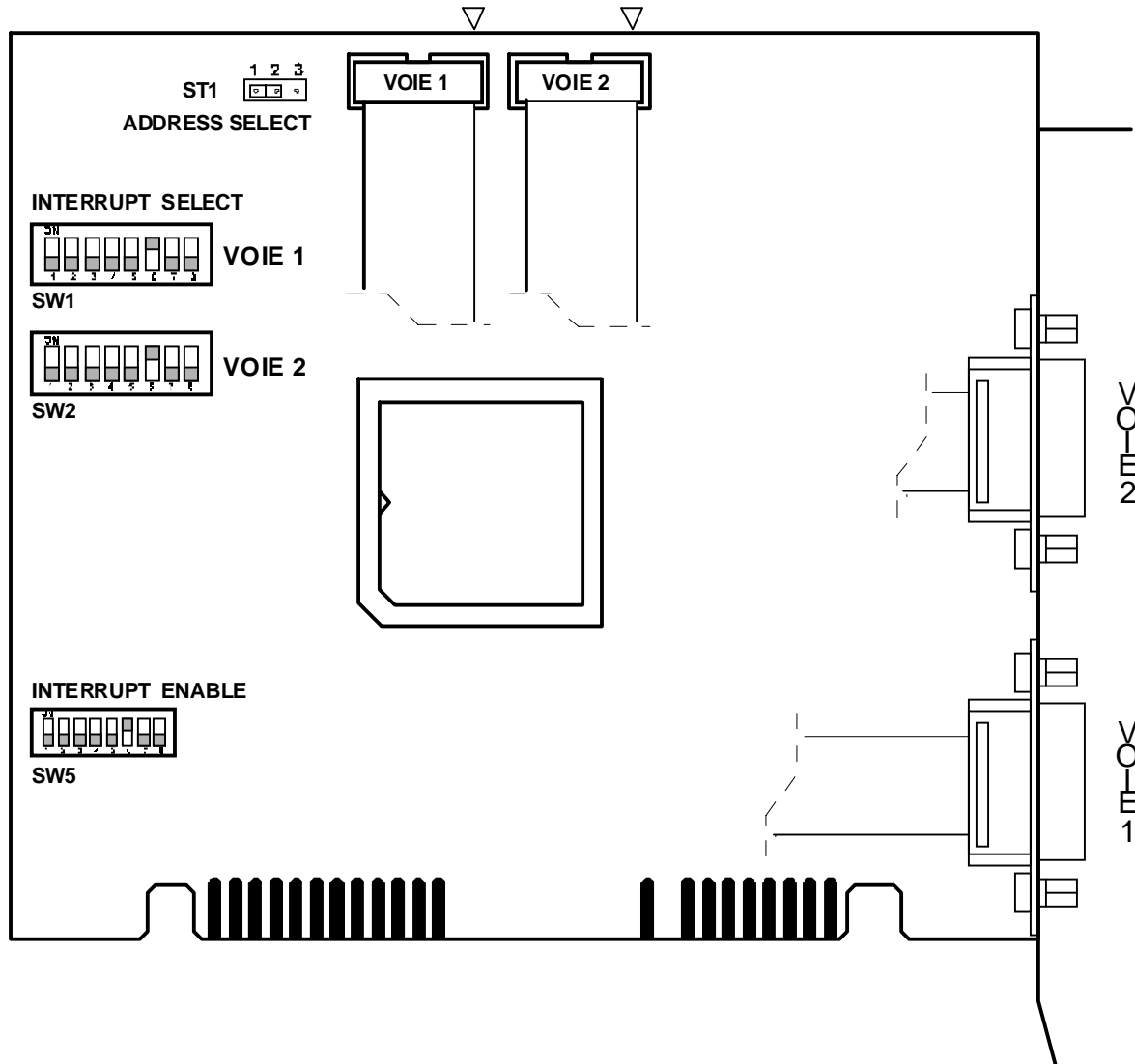
La carte 2RS232 est bâtie autour d'un composant 16C552 intégrant deux unités de communication asynchrone 16C550 dont les caractéristiques sont décrites plus loin dans ce manuel. Elle peut être configurée aux adresses conventionnelles de COM1 et COM2 (3F8h et 2F8h) ou COM3 et COM4 (3E8h et 2E8h).

IV.2 CARACTERISTIQUES MECANIQUES

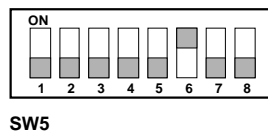
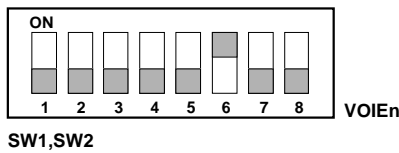
Dimensions	H×L 107×130 mm
Connecteurs	deux SUBD9 EIA574

IV.3 INSTALLATION DE LA CARTE

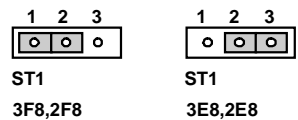
IV.3.1 Emplacement des interrupteurs



Sélection du niveau d'interruption



Sélection de l'adresse



Position des inters: ON OFF

Inter de SW1, SW2, SW5 sur ON :

- | | |
|----------|-----------|
| 1 = IRQ9 | 5 = IRQ7 |
| 2 = IRQ3 | 6 = IRQ10 |
| 3 = IRQ4 | 7 = IRQ11 |
| 4 = IRQ5 | 8 = IRQ15 |

ST1 en position 1-2 (COM1 – COM2)
Voie 1 = 3F8h , voie 2 = 2F8h

ST1 en position 2-3 (COM3 - COM4)
Voie 1 = 3E8h, voie 4 = 2E8h

IV.3.2 Sélection des adresses et interruptions

La configuration de la carte 2RS232 est définie par trois blocs d'interrupteurs DIL et un cavalier. Les blocs d'interrupteurs SW1 et SW2 permettent de sélectionner les IRQ utilisées par les voies 1 et 2 respectivement. Le bloc d'interrupteurs SW5 valide les IRQ sélectionnées.

Exemple :

Voie 1 sur IRQ4 :.....SW1-3 sur ON
 Voie 2 sur IRQ3 :.....SW2-2 sur ON
 Validation :.....SW5-3, SW5-2 sur ON

Le cavalier ST1 permet de choisir une des deux configurations d'adresses I/O possibles :

Lorsque ST1 est en position 1-2, les ports 1 et 2 sont aux adresses conventionnelles de COM1 et COM2, soit 3F8h-3FFh et 2F8h-2FFh. Avec ST1 placé en position 2-3, les ports 1 et 2 sont aux adresses conventionnelles de COM3 et COM4, soit 3E8h-3Efh et 2E8h-2Efh.

A la livraison :

SW1 = Sélection de l'IRQ 4 pour la voie 1 (SW1-3 ON)
 SW2 = Sélection de l'IRQ 3 pour la voie 2 (SW2-2 ON)
 SW5 = Validation des IRQ 3 et 4 (SW5-2 et SW5-3 ON)
 ST 1 = Adresses COM1 et COM2 (ST1 en 1-2)

IV.4 CONFIGURATIONS PARTICULIERES

Certains modèles de la carte 2RS232 supportent des adresses différentes :

Modèle	ST1	Adresse
2RS-APLIC	1-2	2F8h, 3E8h
	2-3	3E8h, 2E8h
2RS-ICL	1-2	3F8h, 2F8h
	2-3	3E8h, 330h

IV.5 REPARTITION DES SIGNAUX SUR LES CONNECTEURS SUBD 9

La carte 2RS232 dispose en sortie de 2 connecteurs SUBD 9 points situés sur la plaquette de fixation de la carte.

Le tableau ci-dessous donne la répartition des signaux sur chaque connecteur 9 points :

No	FONCTION	
1	DCD	Data Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal ready
5	GND	Ground
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI	Ring Indicator

IV.6 MARQUAGE CE ET CEM

Nous consulter.

V. LA CARTE 4RS232

V.1 INTRODUCTION

La carte 4RS232 est une carte de communication asynchrone 4 ports RS232D. Elle fonctionne avec la famille des PC IBM et compatibles (et PS 8530).

Cette carte est supportée directement par la majorité des systèmes d'exploitation tels que Windows 95®, Windows NT®, 386/IX ...

La carte 4RS232 est bâtie autour de deux composants 16C552 intégrant chacun deux unités de communication asynchrone 16C550 dont les caractéristiques sont décrites plus loin dans ce manuel. Elle peut être configurée aux adresses conventionnelles de COM1, COM2, COM3 et COM4 (respectivement 3F8h, 2F8h, 3E8h et 2E8h) ou à partir de l'adresse 280h.

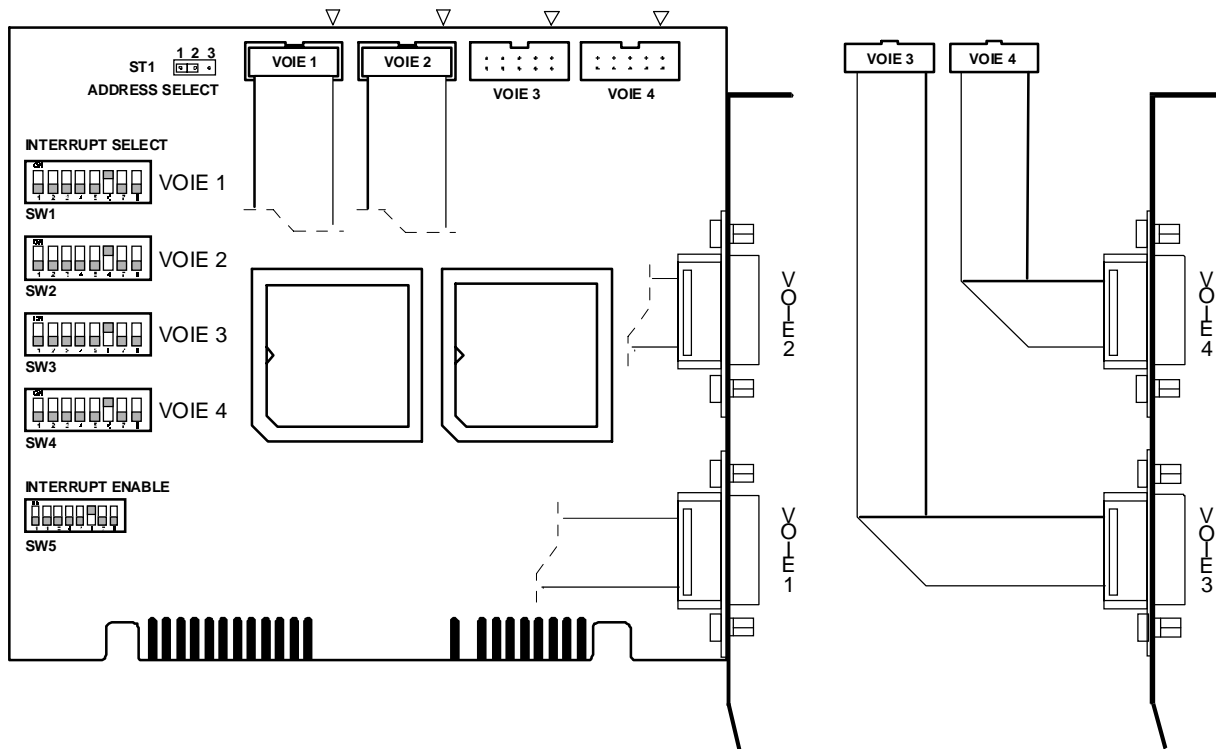
V.2 CARACTERISTIQUES MECANIQUES

Dimensions H×L 107×130 mm

Connecteurs quatre SUBD9 EIA574 mâles répartis sur deux barrettes.

V.3 INSTALLATION DE LA CARTE

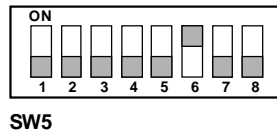
V.3.1 Emplacement des interrupteurs



Sélection du niveau d'interruption



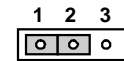
SW1, SW2, SW3, SW4



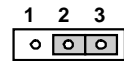
SW5

Position des inters :
 ON
 OFF

Sélection de l'adresse



ST1
 3F8, 2F8
 2E8, 2E8



ST1
 280, 288
 290, 298

Inters 1 à 8 de SW1, SW2, SW3, SW4, SW5 :

- 1 = IRQ9 3 = IRQ4 5 = IRQ7 7 = IRQ11
- 2 = IRQ3 4 = IRQ5 6 = IRQ10 8 = IRQ15

L'IRQ est sélectionnée lorsque l'inter est sur ON.

ST1 - Sélection de l'adresse des ports :

VOIE	ST1 en position 1-2	ST1 en position 2-3
1	3F8h-3FFh (COM1)	280h-287h
2	2F8h-2FFh (COM2)	288h-28Fh
3	3E8h-3EFh (COM3)	290h-297h
4	2E8h-2EFh (COM4)	298h-29Fh

V.3.2 Sélection des adresses et interruptions

La configuration de la carte 4RS232 est définie par cinq blocs d'interrupteurs DIL et un cavalier. Les blocs d'interrupteurs SW1, SW2, SW3 et SW4 permettent de sélectionner les IRQ utilisées par les voies 1, 2, 3 et 4 respectivement. Le bloc d'interrupteurs SW5 valide les IRQ sélectionnées.

Exemple :

Voie 1 sur IRQ4 :.....SW1-3 sur ON
 Voie 2 sur IRQ3 :.....SW2-2 sur ON
 Voie 3 sur IRQ10 :.....SW3-6 sur ON
 Voie 4 sur IRQ11 :.....SW4-7 sur ON
 Validation :.....SW5-2, SW5-3, SW5-6, SW5-7 sur ON

Le cavalier ST1 permet de choisir une des deux configurations d'adresses I/O possibles :

Lorsque ST1 est en position 1-2, les quatre ports de la carte sont aux adresses conventionnelles de COM1, COM2, COM3 et COM4, soit 3F8h-3FFh, 2F8h-2FFh, 3E8h-3EFh, 2E8h-2EFh. Si ST1 est placé en position 2-3, les ports 1 à 4 occupent respectivement les adresses 280h-287h, 288h-28Fh, 290h-297h, 298h-29Fh.

V.3.3 Configurations particulières

Certains modèles de la carte 4RS232 supportent des adresses différentes :

Carte 4RS-APLIC :

VOIE	ST1 en position 1-2	ST1 en position 2-3
1	2F8h-2FFh (COM2)	3E8h-3EFh (COM3)
2	3E8h-3EFh (COM3)	2E8h-2EFh (COM4)
3	2E8h-2EFh (COM4)	280h-287h
4	280h-287h	288h-28Fh

V.4 REPARTITION DES SIGNAUX SUR LES CONNECTEURS SUBD 9

La carte 4RS232 dispose en sortie de 2 connecteurs SUBD 9 points situés sur la plaquette de fixation de la carte (voies 1 et 2) et 2 connecteurs SUBD 9 points situés sur une plaquette additionnelle (voies 3 et 4).

Le tableau ci-dessous donne la répartition des signaux sur chaque connecteur 9 points :

No	FONCTION	
1	DCD	Data Carrier Detect
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal ready
5	GND	Ground
6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	RI	Ring Indicator

V.5 MARQUAGE CE ET CEM

Nous consulter.

VI. LA CARTE 8RS232

VI.1 INTRODUCTION

La carte 8RS232 est une carte de communication asynchrone 8 lignes RS232D. Elle fonctionne avec la famille des PC IBM et compatibles (et PS 8530) et n'occupe qu'un seul emplacement du bus ISA de la machine.

Cette carte est supportée directement par la majorité des systèmes d'exploitation tels que Windows 95®, Windows NT®, 386/IX ...

La carte 8RS232 est bâtie autour de quatre éléments de communication asynchrone 16C552 (équivalents à huit 16C550) dont les caractéristiques seront décrites plus loin dans ce manuel.

Plusieurs cartes 8RS232 peuvent cohabiter au sein d'une même machine pour réaliser des configurations 8, 16, 24 voies et plus, en fonction des ressources disponibles.

VI.2 INSTALLATION DE LA CARTE

La carte 8RS232 dispose en sortie d'un câble pieuvre terminé par 8 connecteurs SUB 9 points mâles.

Deux blocs d'interrupteurs DIL et une rangée de huit triples plots équipent la carte 8RS232 :

Le bloc SW1 permet de sélectionner l'adresse de base des quatre unités de communication.

Le bloc SW2 est dédié à la sélection de la ligne d'interruption utilisée.

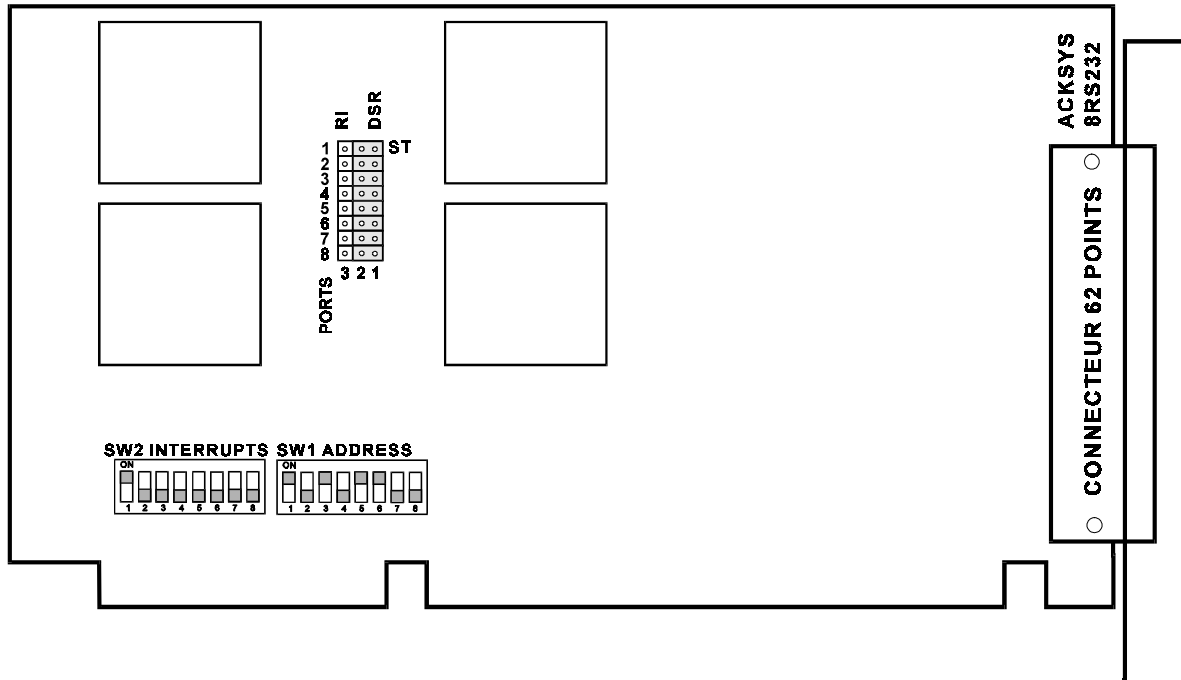
La rangée de 8 triples plots équipés de cavaliers permet de connecter la broche 6 de chaque connecteur SUB D 9 points à la broche RI ou DSR de l'unité de communication associée.

Il faudra en fonction de votre système, modifier la position des interrupteurs de manière à sélectionner les adresses et le niveau d'interruption correspondants à votre application.

A la livraison :

SW1	=	280H
SW2	=	IRQ3
ST	=	8 cavaliers en position 1-2
		Broche 6 connectée à la broche DSR de l'UART,
		Broche RI de l'UART à l'état de repos

VI.2.1 Emplacement des interrupteurs



VI.2.2 Sélection de l'adresse de base de la carte

La carte 8RS232 utilise un bloc de 64 adresses consécutives pour 8 ports. La première adresse du bloc (ou adresse de base) est déterminée par SW1, comme indiqué ci-dessous.

Les adresses de base possibles, en hexadécimal, vont de 000H à FC0H par pas de 40H.

Interrupteur	Position OFF	Position ON
SW1-1	40H	0
SW1-2	80H	0
SW1-3	100H	0
SW1-4	200H	0
SW1-5	400H	0
SW1-6	800H	0
Adresse de base	Somme des SW1-n	
SW1-7 SW1-8	Sans utilité (non connectés)	

Exemple :

SW1-1	SW1-2	SW1-3	SW1-4	SW1-5	SW1-6
OFF	OFF	ON	OFF	ON	ON

Adresse de base 2C0H = 200H + 80H + 40H

La carte occupe la plage d'adresse [adresse de base; adresse de base + 3FH]

Les adresses de chacun des ports sont les suivantes :

PORT 1 Base + 0 à Base + 7
 PORT 2 Base + 8 à Base + 0FH
 PORT 3 Base + 10H à Base + 17H
 PORT 4 Base + 18H à Base + 1FH
 PORT 5 Base + 20H à Base + 27H
 PORT 6 Base + 28H à Base + 2FH
 PORT 7 Base + 30H à Base + 37H
 PORT 8 Base + 38H à Base + 3FH
 Registre de polling Base + 7 (lecture seule)

Base + n : n est un nombre hexadécimal

Note importante :

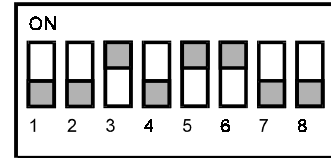
Une fois l'adresse de base sélectionnée, vérifiez que la plage d'adresse occupée par les huit ports de la carte [adresse de base; adresse de base + 3FH] ne rentre en conflit avec aucun périphérique (ports de communication COM1 (3F8H), COM2 (2F8H), COM3 (3E8h), COM4 (2E8H), carte réseau ...).

Certains PC ou cartes ne décodent pas les adresses au-delà de 400H ou 800H. Si vous utilisez une adresse au-delà de 400H, vérifiez qu'aucun équipement de ce type ne décote les adresses égales à (adresse de base - 400H ou 800H).

Les exemples suivants montrent le positionnement des interrupteurs de SW1 pour le décodage en 2C0H et 780H de la carte 8RS232 (les interrupteurs SW1-8 et SW1-7 ne sont pas utilisés) :

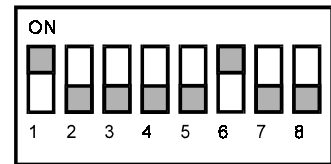
SW1 = 2C0H

2C0H = 200H SW1-4 « OFF »
 + 080H SW1-2 « OFF »
 + 040H SW1-1 « OFF »
 Autres interrupteurs « ON »
 SW-7 et SW-8 non utilisés



SW1 = 780H

780H = 400H SW1-5 « OFF »
 + 200H SW1-4 « OFF »
 + 100H SW1-3 « OFF »
 + 080H SW1-2 « OFF »
 Autres interrupteurs « ON »
 SW-7 et SW-8 non utilisés



Quelques exemples de configurations de SW1 :

<p>Adresse base 100h</p>	<p>Adresse base 180h</p>
<p>Adresse base 200h</p>	<p>Adresse base 240h</p>
<p>Adresse base 280h</p>	<p>Adresse base 300h</p>
<p>Adresse base 700h</p>	<p>Adresse base 740h</p>

VI.2.3 Sélection du niveau d'interruption

Chacune des unités de communication de la carte 8RS232 dispose de sa propre ligne d'interruption. Le signal envoyé sur le niveau d'IRQ sélectionné par SW2 est un OU logique de chacune de ces lignes d'interruption. Il revient donc au programmeur le soin d'autoriser chaque unité de communication à générer des interruptions, en activant les bits du registre de validation des interruptions (IER) des 16C550.

Le niveau d'interruption utilisé peut être sélectionné parmi les interruptions IRQ3, IRQ4, IRQ5, IRQ7, IRQ10, IRQ11, IRQ12 et IRQ15.

La sélection d'une ligne est réalisée par la mise sur ON de l'interrupteur de SW2 correspondant. On prendra soin de choisir un niveau d'IRQ disponible.

*** SW2 ***

SW2-1	IRQ3
SW2-2	IRQ4
SW2-3	IRQ5
SW2-4	IRQ7
SW2-5	IRQ10
SW2-6	IRQ11
SW2-7	IRQ12
SW2-8	IRQ15

VI.2.4 Sélection des signaux RI ou DSR

Le connecteur 62 broches de la carte ne permettant pas de disposer de tous les signaux RS232D, il est nécessaire d'effectuer un choix entre l'utilisation des signaux entrants RI et DSR pour chacun des huit ports de communication de la carte.

La carte dispose d'une rangée ST de huit triples plots permettant de configurer pour chaque port le signal utilisé (voir emplacement de ST dans l'appendice IV.1).

- ST n en position 1-2 :

Broche 6 du connecteur SUBD 9 du port n connectée à l'entrée DSR de l'unité de communication associée, entrée RI forcé à l'état de repos.

- ST n en position 2-3 :

Broche 6 du connecteur SUBD9 du port n connectée à l'entrée RI de l'unité de communication, entrée DSR forcé à l'état de repos.

Dans cette position, on ne peut pas utiliser un câble standard ; il est nécessaire de relier la broche RI de votre périphérique à la broche 6 du connecteur SUBD9 du port concerné.

VI.3 REPARTITION DES SIGNAUX SUR LES CONNECTEURS SUBD 9

Le tableau ci-dessous donne la répartition des signaux sur chaque connecteur 9 points :

No	FONCTION
1	DCD
2	RXD
3	TXD
4	DTR
5	GND
6	DSR-RI (*)
7	RTS
8	CTS
9	N.C

(*) DSR ou RI selon la position du cavalier ST

VI.4 MARQUAGE CE ET CEM

Nous consulter.

VII. LA CARTE UNX232

VII.1 INTRODUCTION

La carte UNX232 est une carte de communication asynchrone de 4 ou 8 ports série. Elle fonctionne avec la famille des PC IBM et compatibles (et PS 8530) et n'occupe qu'un seul emplacement du bus ISA dans la machine. Il en existe quatre modèles de base :

UNX232-4	4 ports RS232
UNX232-8	8 ports RS232
UNX232-4 RS422	4 ports RS422
UNX232-8 RS422	8 ports RS422

De plus, les ports 1 et 5 de chaque modèle sont mixtes et peuvent être utilisés soit en RS232 soit en RS422.

Les systèmes d'exploitation tels que Windows® 95, Windows NT®, 386/IX sont capables dans la plupart des cas de gérer directement cette carte.

La carte UNX232 est bâtie autour de 4 ou 8 éléments de communication asynchrone 16C550 dont les caractéristiques seront décrites plus loin dans ce manuel.

Des interfaces quadruples RS422A, RS485 et boucle de courant 20 mA sont aussi disponibles afin de satisfaire aux environnements industriels. Les caractéristiques de l'interface RS422A intégrée sont données à la fin de ce manuel tandis que celles des interfaces quadruples RS422A, RS485 et boucle de courant font l'objet de documentations séparées.

Plusieurs cartes UNX232 peuvent cohabiter au sein d'une même machine pour réaliser des configurations 8, 16, 32 voies et plus.

VII.2 INSTALLATION DE LA CARTE

La carte UNX232 est reliée par l'intermédiaire d'un câble rond blindé de 50 points (version 4 voies) ou 100 points (version 8 voies) au dispositif de connexion supportant les connecteurs SUB D 25 points.

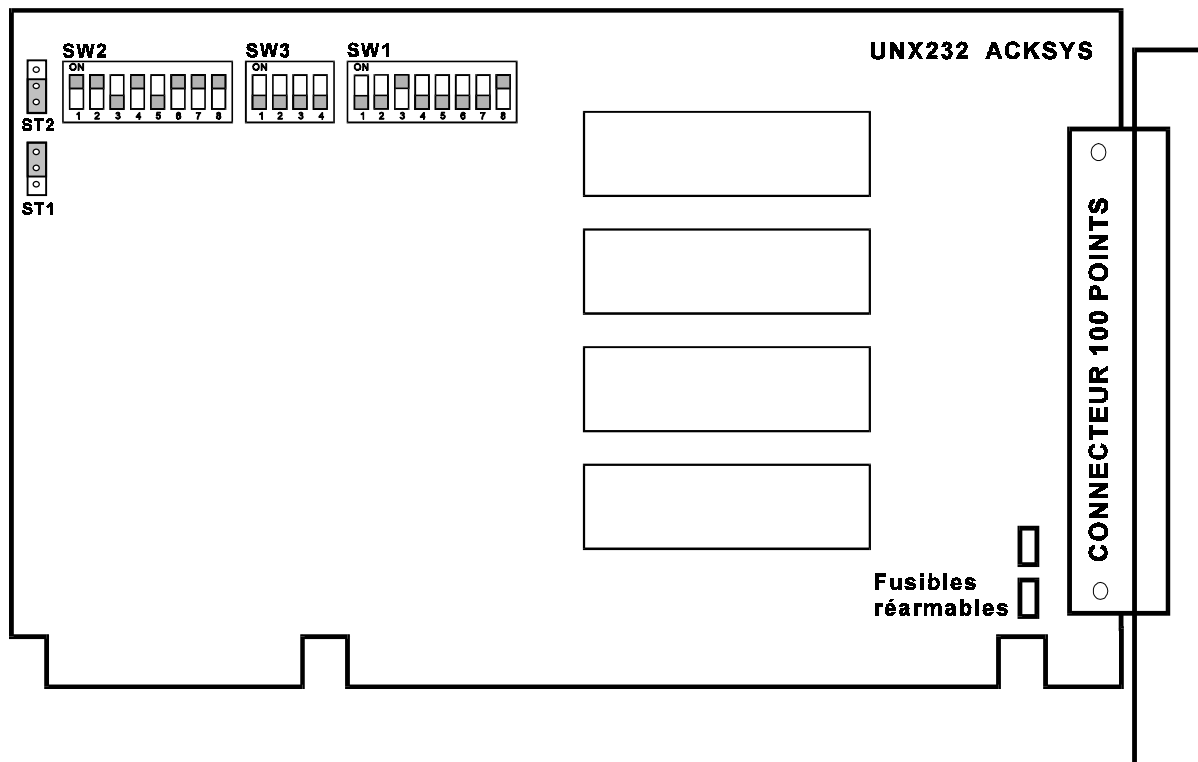
Il faudra en fonction de votre système, modifier la position des Inters de manière à sélectionner les adresses correspondantes à votre application. Les différentes combinaisons d'adressage seront étudiées au chapitre VII.2.2.

Il est recommandé de bien vérifier le branchement du dispositif de connexion. En effet, il est actif et un mauvais branchement le détruirait immanquablement.

VII.2.1 Emplacement des interrupteurs

Trois blocs d'interrupteurs DIL équipent la carte UNX232. Les blocs SW1 et SW3 sont dédiés à la sélection de la ligne d'interruption utilisée. SW2 permet de sélectionner l'adresse de départ des 8 unités de communication.

Les inters 1 de SW2 et 8 de SW1 ainsi que le cavalier ST1 déterminent si la carte UNX232 est configurée en 4 ou en 8 ports.



SW1 : Sélection du niveau d'interruption / activation du registre de masque (voir page VII-7)

SW2 : Sélection de l'adresse de base (voir page VII-3)

SW3 : Sélection du niveau d'interruption (voir page VII-7)

ST1 : Sélection 4 ou 8 ports (voir page VII-6)

ST2 : Autorisation du partage de l'interruption (voir page VII-9)

VII.2.2 Adresse de base de la carte

La carte UNX232 utilise un bloc de 64 adresses séquentielles pour 8 ports et 32 pour 4 ports. La première adresse du bloc est déterminée par SW2.

Attention : L'inter 1 de SW2 doit toujours être sur OFF pour l'utilisation en 8 ports alors qu'il représente un bit d'adresse (0 ou 1) pour l'utilisation en 4 ports.

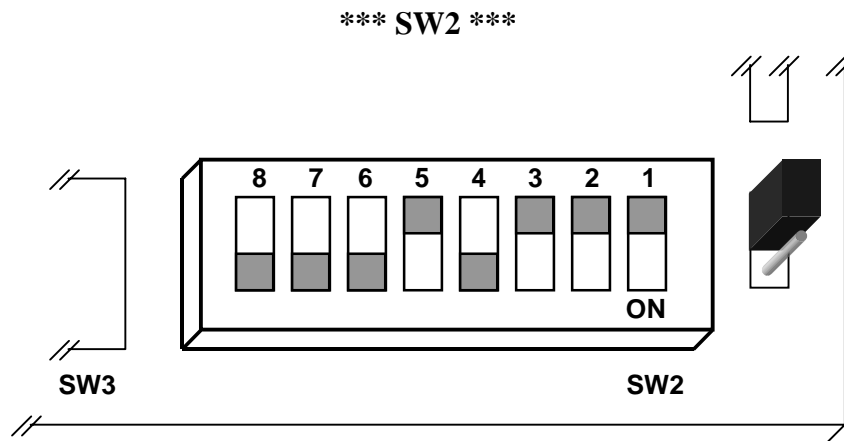
Bloc d'interrupteurs DIL SW2		
Inter	ON	OFF
1	8 ports : Interdit 4 ports : A5 = 0	8 ports : A5 = 0 4 ports : A5 = 1
2	A6 = 0	A6 = 1
3	A7 = 0	A7 = 1
4	A8 = 0	A8 = 1
5	A9 = 0	A9 = 1
6	A10 = 0	A10 = 1
7	A11 = 0	A11 = 1
8	A12 = 0	A12 = 1

L'adresse de base 280H est la plus fréquemment utilisée par les systèmes d'exploitation (sauf XENIX SCO, PICK et UNIX V de MICROPORT). Par contre, l'adresse de départ 2C0H supportera COM2 sur le 8^{ème} port de la carte UNX232.

ATTENTION

Les inters de SW2 doivent être positionnés sur OFF pour un bit d'adresse à 1, et sur ON pour un bit d'adresse à 0

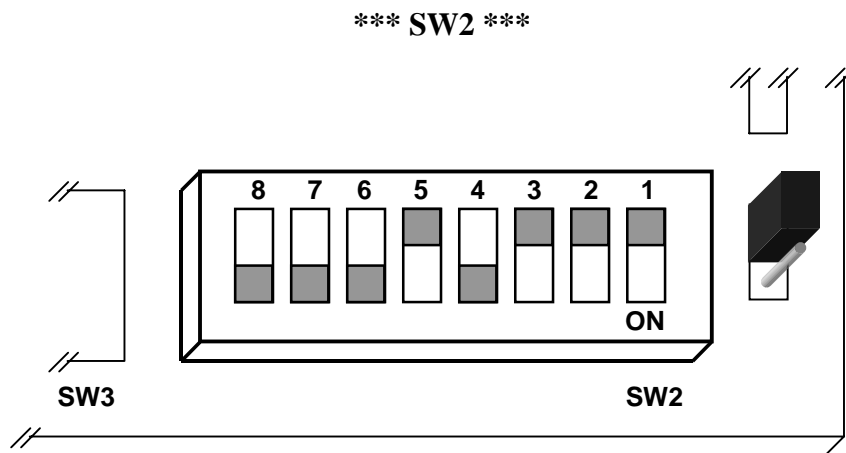
L'exemple suivant montre le positionnement des inters pour le décodage en 2C0H de la carte UNX232 en 8 ports:



Ce qui nous donne la suite binaire :

Inter	-	8	7	6	5	4	3	2	-	
Adresse	A15-13	A12	A11	A10	A9	A8	A7	A6	A5-4	A3-0
	000	0	0	0	1	0	1	1	00	0000
	0	2				C			0	

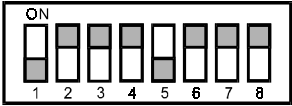
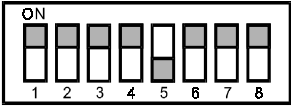
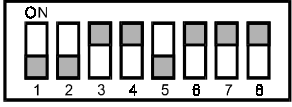
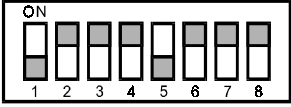
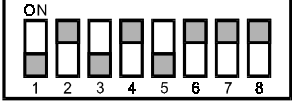
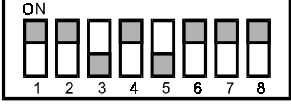
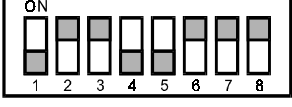
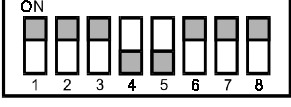
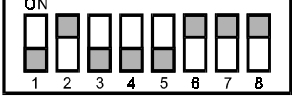
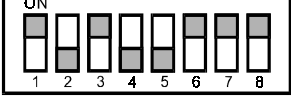
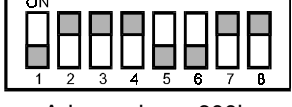
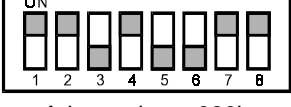
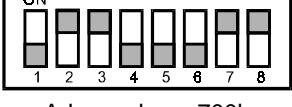
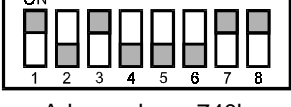
Positionnement des inters pour le décodage en 280H de la carte UNX232 en 4 ports:



Ce qui nous donne la suite binaire :

Inter	-	8	7	6	5	4	3	2	1	-	
Adresse	A15-13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3-0
	000	0	0	0	1	0	1	0	0	0	0000
	0	2				8			0		

Quelques exemples de configuration SW2

Carte UNX 8 Ports	Carte UNX 4 Ports
 <p>Adresse base 200h</p>	 <p>Adresse base 200h</p>
 <p>Adresse base 240h</p>	 <p>Adresse base 220h</p>
 <p>Adresse base 280h</p>	 <p>Adresse base 280h</p>
 <p>Adresse base 300h</p>	 <p>Adresse base 300h</p>
 <p>Adresse base 380h</p>	 <p>Adresse base 340h</p>
 <p>Adresse base 600h</p>	 <p>Adresse base 680h</p>
 <p>Adresse base 700h</p>	 <p>Adresse base 740h</p>

Une fois l'adresse de base sélectionnée, les unités de communication auront les adresses suivantes :

- PORT 1..... Base + 0
- PORT 2..... Base + 8
- PORT 3..... Base + 16
- PORT 4..... Base + 24
- PORT 5..... Base + 32
- PORT 6..... Base + 40
- PORT 7..... Base + 48
- PORT 8..... Base + 56
- Registre de masque/polling Base + 7

Base + n : n est un nombre décimal

VII.2.3 Sélection 4 ou 8 ports

La carte UNX232 4 ports est une carte simple au format demi-slot. En version 8 ports, une seconde carte est superposée à la première.

La sélection du nombre de ports montés sur la carte s'effectue par l'intermédiaire de l'inter 1 de SW2, de l'inter 8 de SW1 et enfin du cavalier ST1.

Carte UNX232 en 4 ports

Inter 1 de SW2..... bit d'adresse
Inter 8 de SW1..... ON
Cavalier ST1 Position 2-3

Carte UNX232 en 8 ports

Inter 1 de SW2..... OFF
Inter 8 de SW1..... OFF
Cavalier ST1 Position 1-2

VII.2.4 Sélection du niveau d'interruption

Les interruptions de chacune des voies sont individuellement contrôlées mais elles utilisent toutes la même ligne d'interruption. Le niveau d'interruption utilisé peut être sélectionné parmi les interruptions IRQ2, IRQ3, IRQ4, IRQ5, IRQ6, IRQ7, IRQ10, IRQ11, IRQ12 et IRQ15.

La sélection du niveau d'interruption s'effectue de la manière suivante :

INTER DIL SW1		
Inter 1	ON	Registre de masque validé
Inter 2	ON	IRQ9 (ex IRQ2)
Inter 3	ON	IRQ3 (COM2)
Inter 4	ON	IRQ4 (COM1)
Inter 5	ON	IRQ5 (LPT2)
Inter 6	ON	IRQ6 (FDC)
Inter 7	ON	IRQ7 (LPT1)
Inter 8	ON	4 ports (OFF = 8 Ports)

INTER DIL SW3		
Inter 1	ON	IRQ15 (Port IDE secondaire)
Inter 2	ON	IRQ12 (Port Souris Bus)
Inter 3	ON	IRQ11
Inter 4	ON	IRQ10

L'activation d'une ligne est déclenchée par la mise sur ON d'un inter de SW1 ou bien SW3. On prendra soin de choisir un niveau d'IRQ disponible. Les attributions conventionnelles des IRQ indiquées ci-dessus entre parenthèses vous aideront à résoudre les conflits possibles avec les autres périphériques du PC (notez également que les cartes réseau utilisent fréquemment les IRQ5 et 10)

VII.2.5 Registres de "masque" et de "polling"

Chacune des unités de communication de la carte UNX dispose de sa propre ligne d'interruption. Le signal envoyé sur le niveau d'IRQ sélectionné par SW1 ou SW3 est un OU logique de chacune de ces lignes d'interruption. Il revient donc au programmeur le soin d'autoriser chaque unité de communication à générer des interruptions, en activant les bits du registre de validation des interruptions (IER) des 16C550. La ligne d'interruption de la carte vers le PC étant unique, il revient également au programmeur le soin de déterminer, dans son programme d'interruption, quelles unités de communication sont à l'origine de l'activation de cette ligne (plusieurs unités peuvent générer une interruption simultanément). Les registres de "MASQUE" et de "POLLING" permettent de faciliter ces tâches.

Le registre de "MASQUE" est un registre 8 bits qui permet à l'utilisateur d'inhiber individuellement ou en combinaisons les interruptions de chaque port, et ce sans changer le registre de validation des interruptions des 16C550: lorsque la ligne d'interruption d'un port est masquée, elle n'est plus prise en compte pour la génération du signal d'interruption envoyé sur le bus du PC. Le masquage de l'interruption d'un port s'effectue par la mise à 0 du bit correspondant à la ligne voulue. De la même manière le démasquage s'effectue par la mise à 1 de ce bit.

ATTENTION: Le registre d'identification des interruptions (IIR) d'une unité de communication dont la ligne d'interruption est masquée indiquera toujours la présence des conditions d'interruption programmées, bien que le signal d'interruption ne soit pas envoyé vers le PC.

Pour déterminer quels sont les ports qui ont déclenché une interruption, les registres d'identification des interruptions (IIR) du 16C550 ainsi que le registre de "POLLING" peuvent être consultés:

Le registre de "POLLING" est un registre dont chacun des 8 bits représente la ligne d'interruption d'un 16C550. Un bit à 1 indique que la ligne d'interruption de l'unité de communication correspondante est activée, et par conséquent qu'une condition d'interruption est présente sur ce port.

L'accès aux registres de "MASQUE" et de "POLLING" s'effectue à l'adresse Base+7, c'est à dire à l'adresse du "scratch register" de la voie 1. Le registre de "MASQUE" est sélectionné en écriture tandis que le registre de "POLLING" fonctionne uniquement en lecture.

Le registres de "MASQUE" peut être totalement désactivé par la mise à OFF de l'inter 1 de SW1. Par contre le registre de polling est toujours actif.

ECRITURE BASE+7**REGISTRE DE MASQUE**

- Bit 0 = 0 : Ligne d'interruption port 1 masquée (inhibée)
- Bit 1 = 0 : Ligne d'interruption port 2 masquée (inhibée)
- Bit 2 = 0 : Ligne d'interruption port 3 masquée (inhibée)
- Bit 3 = 0 : Ligne d'interruption port 4 masquée (inhibée)
- Bit 4 = 0 : Ligne d'interruption port 5 masquée (inhibée)
- Bit 5 = 0 : Ligne d'interruption port 6 masquée (inhibée)
- Bit 6 = 0 : Ligne d'interruption port 7 masquée (inhibée)
- Bit 7 = 0 : Ligne d'interruption port 8 masquée (inhibée)

L'inter 1 de SW1 doit être en position "ON".

LECTURE BASE+7**REGISTRE DE POLLING**

Bit 0 = 1 : Ligne d'interruption port 1 active
Bit 1 = 1 : Ligne d'interruption port 2 active
Bit 2 = 1 : Ligne d'interruption port 3 active
Bit 3 = 1 : Ligne d'interruption port 4 active
Bit 4 = 1 : Ligne d'interruption port 5 active
Bit 5 = 1 : Ligne d'interruption port 6 active
Bit 6 = 1 : Ligne d'interruption port 7 active
Bit 7 = 1 : Ligne d'interruption port 8 active

NOTE : Si l'inter 1 de SW1 est sur "ON", alors toutes les interruptions venant de la carte UNX232 seront conditionnées par le registre de masque.

VII.2.6 Partage de la ligne d'interruption

Les cartes UNX232 révision C et suivantes offrent la possibilité de monter deux cartes 4 ou 8 lignes sur un même niveau d'interruption. Cette fonctionnalité est rendue possible grâce à un mécanisme de OU câblé; cependant, certaines machines peu compatibles au niveau du contrôleur d'interruptions peuvent être perturbées par ce mécanisme. Dans ce cas, il est possible de le désactiver en plaçant le cavalier ST2 en position 2-3.

VII.3 BOITIER DE CONNEXION RS232D

VII.3.1 Attribution des lignes sur les connecteurs

Le tableau ci-dessous donne la répartition des signaux sur chaque connecteur 25 points :

n°	Fonction
1	PG
2	TXD
3	RXD
4	RTS
5	CTS
6	DSR
7	MASSE 0V
8	CD
9	+12V
10	-12V
11	Non connecté
12	- TX ⁽¹⁾
13	+ TX ⁽¹⁾

n°	Fonction
14	Non connecté
15	Non connecté
16	Non connecté
17	Non connecté
18	Non connecté
19	Non connecté
20	DTR
21	- RX ⁽¹⁾
22	RI
23	+ RX ⁽¹⁾
24	Non connecté
25	Non connecté

(1) Les signaux, +TX, -TX, +RX, -RX en RS422A ne sont disponibles que sur les voies 1 et 5

Les sorties + 12 VDC et - 12 VDC sont protégées par des fusibles thermiques; l'intensité maximale totale admissible pour un dispositif de 8 lignes est de 750 mA sur chaque tension (à répartir sur le nombre de voies chargées).

AVERTISSEMENT

Il est vivement recommandé de vérifier l'attribution des broches 9 et 10 sur le connecteur des périphériques que vous raccordez au dispositif de connexion de la carte UNX232; en effet, certains Modems utilisent ces broches pour alimenter leur électronique. Si c'est le cas, nous vous conseillons de ne pas câbler ces broches ou bien de vérifier que la consommation du Modem est compatible avec le courant disponible (750 mA MAX). Un voyant lumineux indique lorsqu'il est allumé, que le dispositif de connexion est correctement alimenté par le système. Ce voyant peut s'éteindre pour les raisons suivantes :

- Mauvaise connexion du câble de raccordement
- Panne sur les tensions +12V et -12V du système
- Court-circuit sur un connecteur SUB D 25 points
- Surcharge des alimentations sur les SUB D 25 points
- Panne du voyant

En cas de court-circuit accidentel sur les connecteurs SUB D 25 points il est recommandé d'arrêter le système et de détecter la cause du court-circuit; attention, la tension ne peut être rétablie dans le dispositif de connexion qu'après arrêt du système et attente d'un délai de 20 secondes, ceci à cause des fusibles de protection réarmables.

VII.3.2 Caractéristiques ESD

La carte UNX232 bénéficie d'une protection exceptionnelle contre les surtensions et les DECHARGES ELECTROSTATIQUES (ESD). Cette protection est assurée sur chaque signal RS232D par des dispositifs absorbants les pointes de tension dangereuses pour l'électronique.

Associé au câble blindé, ce dispositif de protection assure à la carte UNX232 une grande fiabilité, une grande longévité, une émission très réduite de rayonnements électromagnétiques ainsi qu'une grande immunité aux parasites extérieurs.

Tous ces éléments font de la carte UNX232 l'outil idéal pour les applications de communication en milieu industriel perturbé.

En cas de surcharge importante sur les lignes de communication (ex : foudre, connexion directe à des alimentations etc.), les dispositifs de protection s'autodétruisent afin de protéger efficacement la carte UNX232 et le système hôte.

VII.3.3 Extraits du compte rendu des essais de décharges électrostatiques

Ces essais ont été réalisés selon les normes CEI 801-2, CEI 801-4 et MIL STD 883C méthode 3015-X; le degré de sévérité des essais a dépassé le dernier niveau de ces normes.

Le tableau ci-dessous résume les résultats obtenus par application de décharges électrostatiques directement sur les signaux RS232D; ces essais ont été effectués lors du déroulement d'un programme de test et en connectant les signaux sortants sur les signaux entrants :

Tensions minimales d'e.s.d. perturbant les signaux de sortie			
Décharges ponctuelles	12 décharges coup par coup	Rafales 20Hz 1 seconde	Rafales 20Hz 5 secondes
9kV	7kV	7kV	7kV
Tensions minimales d'e.s.d. appliquées sur les signaux de sortie et provoquant des perturbations semi-permanentes ou permanentes			
14kV	8kV	8kV	8kV

Aucune perturbation fonctionnelle ni aucun dommage n'ont été constatés au terme des essais d'E.S.D supérieures à 22kV, lorsque celles-ci sont appliquées sur l'armature métallique des boîtiers de connexion.

Le rapport complet des essais de décharges électrostatiques peut être communiqué sur simple demande.

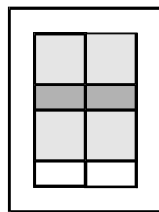
VII.3.4 L'interface RS422A intégrée

Lorsqu'il est nécessaire d'effectuer des communications rapides et sur des longues distances, la RS232D n'est plus suffisante. La norme RS422A est une norme d'interface électrique qui permet la transmission des informations en différentiel. Cette méthode permet d'obtenir une meilleure immunité aux bruits et en fait donc le moyen idéal pour la communication série sur longue distance.

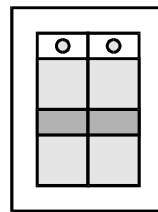
L'interface RS422A de la carte UNX232 est disponible uniquement sur les voies 1 et 5 du boîtier de connexion standard (il existe également des boîtiers de connexion 4 voies RS422A)

Configuration de l'interface RS422A :

Dans la configuration standard de la carte, les lignes 1 et 5 sont positionnées en mode RS232D. La sélection du mode RS422A pour ces lignes est réalisée par des commutateurs (2 par voie) situés au dos du boîtier de connexion :



MODE RS232D
Points cachés



MODE RS422A
Points apparents

Le dos du boîtier de connexion est ajouré de façon à autoriser le changement de mode à l'aide, par exemple, d'un petit tournevis plat et d'une source lumineuse. Cette manipulation doit être effectuée quand l'ordinateur est hors tension.

Le démontage du boîtier est absolument interdit, sous peine de perte de la garantie.

Caractéristiques de l'interface RS422A :

Débit maximum : 115200 bits/s

Gamme de tension maximale admissible en sortie et en mode commun : $\pm 7V$.

Gamme de tension maximale admissible en entrée :

En mode commun : $\pm 10V$.

En mode différentiel : $\pm 12V$.

Une résistance de terminaison de 121 Ohms est connectée entre les signaux -RX et +RX.

Charge maximale en RS422A :

10 récepteurs (limitée par la norme EIA)

Câble conseillé :

Paire torsadée gauge 24 AWG (0,22 mm²), 50 pF/m, impédance nominale 120 Ohms.

En milieu industriel très perturbé, l'utilisation d'un écran de masse est obligatoire: capacité entre écran de masse et conducteur : 75 pF/m. (BELDEN 8102).

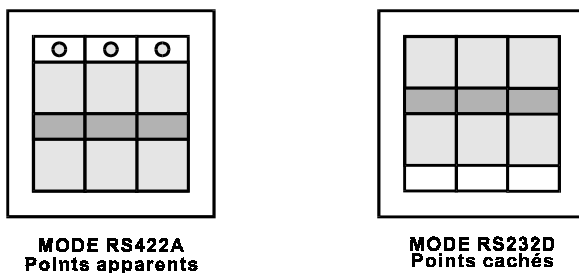
Distance maximale de transmission :

La distance maximale de transmission est limitée par la norme EIA et dépend de la vitesse et du type de câble utilisé.

Elle est de 1200 m pour un câble de gauge 24 et d'une capacité de 50 pF/m entre conducteurs.

VII.4 BOITIER DE CONNEXION RS422A

Le boîtier de connexion de la carte UNX232 existe également en version 4 ou 8 voies RS422A. Les interfaces RS422 de ce boîtier présentent les mêmes caractéristiques que celles du boîtier standard décrites plus haut, mais sont équipées de dispositifs de protections supplémentaires contre les surtensions et disposent de deux broches permettant de polariser la ligne. Les voies 1 et 5 sont également modulables en RS232D par des commutateurs (3 par voie) situés au dos du boîtier de connexion :



Le dos du boîtier de connexion est ajouré de façon à autoriser le changement de mode à l'aide, par exemple, d'un petit tournevis plat et d'une source lumineuse. Cette manipulation doit être effectuée quand l'ordinateur est hors tension.

Le démontage du boîtier est absolument interdit, sous peine de perte de la garantie.

VII.4.1 Attribution des lignes sur les connecteurs

Le tableau ci-dessous donne la répartition des signaux sur chaque connecteur 25 points :

No	Fonction	
	RS422	RS232 voies 1 et 5
1	PGND	PGND
2	NC	TXD
3	NC	RXD
4	NC ⁽¹⁾	RTS
5	NC ⁽¹⁾	CTS
6	NC ⁽¹⁾	DSR
7	NC ⁽¹⁾	0V
8	NC ⁽¹⁾	CD
9	+ POL	+12V
10	- POL	- 12V
11	NC	NC
12	- TX	Réservé
13	+ TX	Réservé

No	Fonction	
	RS422	RS232 voies 1 et 5
14	NC	NC
15	NC	NC
16	NC	NC
17	NC	NC
18	NC	NC
19	NC	NC
20	NC ⁽¹⁾	DTR
21	- RX	Réservé
22	NC ⁽¹⁾	RI
23	+ RX	Réservé
24	NC	NC
25	NC	NC

⁽¹⁾ Ne rien relier à ces signaux sur les voies 1 et 5 utilisées en mode RS422.

Protection contre les surtensions de ligne

Par transils, tension de claquage +/- 7V en mode commun, +/- 14V en mode différentiel.
capacité d'absorption: 0,4 KW pendant 1ms.

Polarisation de ligne

Elle est nécessaire afin d'avoir un état stable en mode RS422A lorsque plusieurs transmetteurs sont en bus et par conséquent que l'utilisation de l'état haute impédance est requis. Une seule polarisation par ligne est nécessaire.

Connexion de la polarisation :

- Connecter la broche n° 9 à la broche n° 23**
- Connecter la broche n° 10 à la broche n° 21**

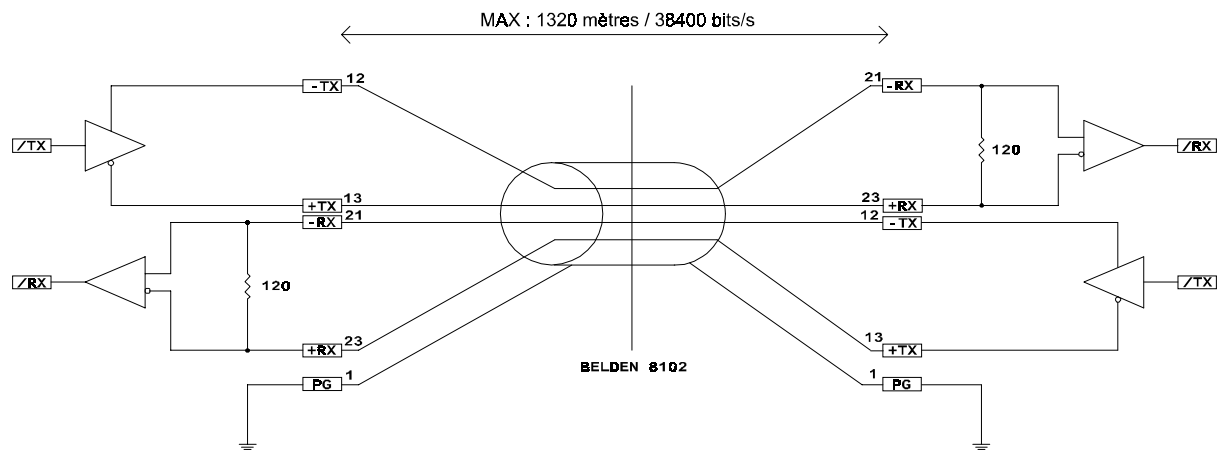
VII.4.2 Raccordement des signaux entre 2 ports série RS422A

- TX (12)	- RX (21)
+TX (13)	+ RX (23)
- RX (21)	- TX (12)
+RX (23)	+ TX (13)

Les numéros entre parenthèses indiquent les numéros de broche des SUBD 25 points.

La liaison entre deux ports série RS422A de la carte UNX232 est une liaison croisée.

Raccordement de deux ports série en RS422A



VII.5 OPTIONS POUR CARTES UNX

ADAPTATEURS

UNX-BP232 : Boîtier de connexion 4 ports RS232.

UNX-BP422 : Boîtier de connexion 4 ports RS422.

AD422/485-UNX : Adaptateur 4 canaux RS422/485 à isolement galvanique pour cartes UNX232.

BC20MA-UNX : Adaptateur 4 canaux boucle de courant 20mA à isolement galvanique pour cartes UNX232

LOGICIELS

SRLPORTS: Pilotes pour système d'exploitation MS-DOS, OS/2, Windows 95

UNX-UNIX : "driver" UNIX (SCO, 386/IX) pour applications industrielles.

VII.6 MARQUAGE CE & CEM

DECLARATION  DE CONFORMITE

NOUS : **ACKSYS** 3-5 rue du Stade 78302 Poissy - France

déclarons que les produits :

UNX232-4	UNX232-8	UNX232-40	UNX232-80
UNX-BP232-4	UNX-BP232-8	UNX-BP422-4	UNX-BP422-8
UNX-EXT	AD422/485-UNX	PWS-5	

sont conformes aux dispositions des :

Directives du Conseil de l'Union Européenne n° **89/336/CEE** du **3 mai 1989** (concernant le rapprochement des législations des Etats membres relatives à la Compatibilité ElectroMagnétique) et n° **73/23/CEE** (Directive Basse Tension) du **19 février 1973**, modifiées par la Directive **93/68/CEE** (Marquage **CE** de Conformité) du **22 juillet 1993**.

Cette conformité est présumée par la référence aux spécifications suivantes :

- ⇒ Norme **NF EN 55022 (décembre 1994)**
Limites et méthodes de mesure des caractéristiques de perturbations radioélectriques produites par les Appareil de Traitement de l'Information - Norme produit **émission**.
- ⇒ Norme **NF EN 50082-2 (juin 1995)**
Compatibilité électromagnétique - Norme générique **immunité**
Partie 2 : **Environnement industriel**
- ⇒ Norme **NF EN 60950 (janvier 1993) + Amendements A1 (mars 1993) et A2 (octobre 1993)**
Sécurité des matériels de traitement de l'information y compris les matériels de bureau électriques

Essais réalisés dans les laboratoires :

ACKSYS 78302 POISSY

HAZTEC 91962 COURTABOEUF

Fait à POISSY, le 30 juillet 1996

Luciano IASCHI
Directeur
Technique

Réf. : DECLARATION CE GAMME UNX JUILLET 1996

VIII. CONFIGURATIONS PARTICULIERES UNX232

VIII.1 CONFIGURATION SOUS SCO XENIX 2.3.x ET SCO UNIX

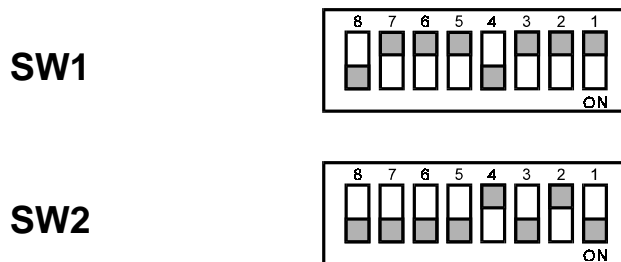
Les versions de XENIX SCO 2.3.1, 2.3.2 ainsi que UNIX SCO V.3.2 intègrent déjà un certain nombre de "Handlers" pour les cartes de communication.

Le système inclut la notion de remplacement COM1 ou remplacement COM2.

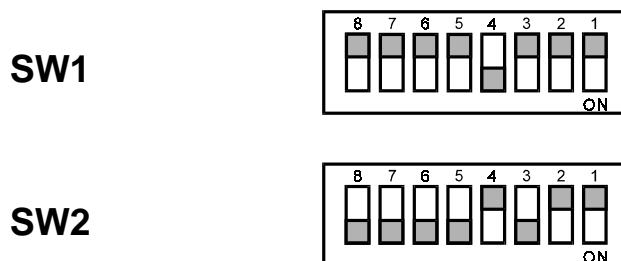
Avant d'installer la carte, assurez-vous qu'il ne peut pas y avoir de conflit avec les périphériques déjà installés dans votre machine.

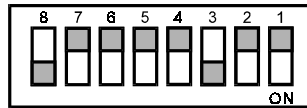
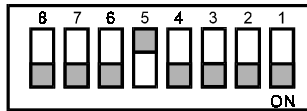
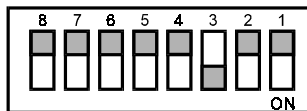
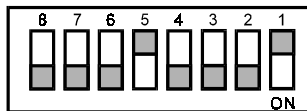
Pour toutes les combinaisons données ci-dessous, les interrupteurs de SW3 doivent être en position OFF.

Configuration : Remplacement de COM1, 140h, IRQ4, 4 ports



Configuration : Remplacement de COM1, 140h, IRQ4, 8 ports



Configuration : Remplacement de COM2, 200h, IRQ3, 4 ports**SW1****SW2****Configuration : Remplacement de COM2, 200h, IRQ3, 8 ports****SW1****SW2**

La carte UNX232 doit être reconnue par le système dès lors que le système d'exploitation est chargé.

Il faut ensuite créer les "devices" par l'utilisation de la fonction "mkdev" ou mieux encore par l'utilitaire de configuration "sysadm" sous XENIX et "sysadmsh" sous UNIX V.

VIII.2 CONFIGURATION SOUS 386/IX INTERACTIVE SYSTEMS

Pour une utilisation sous 386/IX, l'adresse de base de la carte UNX232 doit être 280H et le niveau d'interruption IRQ3.

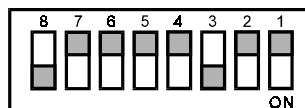
Il est à noter que sous 386/IX, la carte UNX n'est pas utilisée avec le registre de "polling" et est donc dans un mode totalement compatible avec la carte ADRS232.

Avant d'installer la carte, assurez-vous qu'il ne peut pas y avoir de conflit avec les périphériques déjà installés dans votre machine et notamment en ce qui concerne COM2 qui utilise IRQ3.

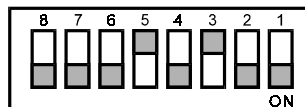
Pour toutes les combinaisons données ci-dessous, les interrupteurs de SW3 doivent être en position OFF.

Configuration : 280H, IRQ3, 4 ports, pas de masque/polling

SW1

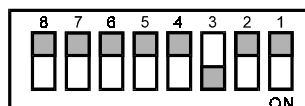


SW2

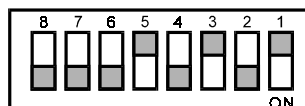


Configuration : 280H, IRQ3, 8 ports, pas de masque/polling

SW1



SW2



VIII.3 CONFIGURATION SOUS PICK

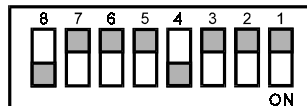
Les adresses de base possibles pour une utilisation de la carte sous PICK sont 240H et 140H, la première fonctionne en conjonction avec IRQ4, la deuxième avec IRQ3.

Les anciennes versions de PICK ne peuvent supporter que quatre ports asynchrones, en ce qui concerne les nouvelles versions, nous n'avons pas encore à ce jour tous les éléments nécessaires pour vous communiquer les adresses et IRQ utilisables.

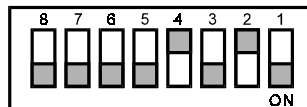
Avant d'installer la carte, assurez-vous qu'il ne peut pas y avoir de conflit avec les périphériques déjà installés dans votre machine notamment en ce qui concerne COM1 qui utilise IRQ4 et COM2 qui utilise IRQ3. Pour toutes les combinaisons données ci-dessous, les interrupteurs de SW3 doivent être en position OFF.

Configuration : 140H, IRQ4, 4 ports, pas de masque/polling

SW1

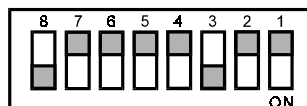


SW2

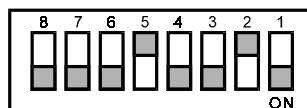


Configuration : 240H, IRQ3, 4 ports, pas de masque/polling

SW1



SW2



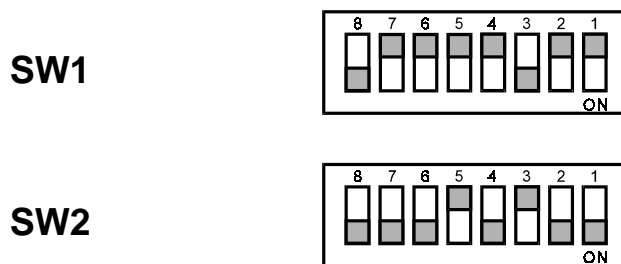
VIII.4 CONFIGURATION SOUS PROLOGUE

Pour une utilisation sous PROLOGUE, l'adresse de base de la carte UNX232 doit être 280H et le niveau d'interruption IRQ3.

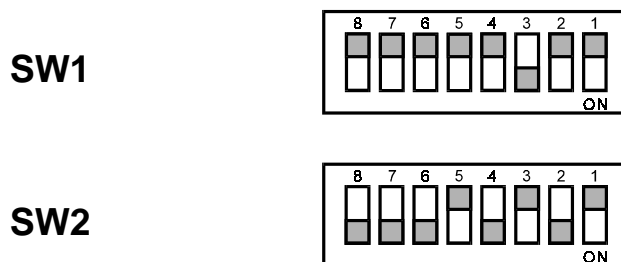
Il est à noter que sous PROLOGUE, la carte UNX n'est pas utilisée avec le registre de "polling" et est donc dans un mode totalement compatible avec la carte ADRS232.

Avant d'installer la carte, assurez-vous qu'il ne peut pas y avoir de conflit avec les périphériques déjà installés dans votre machine et notamment en ce qui concerne COM2 qui utilise IRQ3. Pour utiliser la carte UNX232, vous devez sélectionner le "Coupleur standard 4/8 ports" dans SYSCONF. Pour toutes les combinaisons données ci-dessous, les interrupteurs de SW3 doivent être en position OFF.

Configuration : 280H, IRQ3, 4 ports, pas de masque/polling



Configuration : 280H, IRQ3, 8 ports, pas de masque/polling



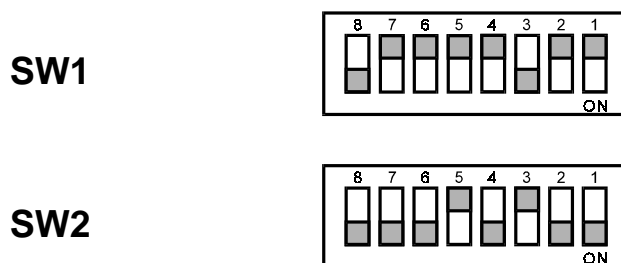
VIII.5 CONFIGURATION SOUS QNX DE QUANTUM SOFTWARE

Pour une utilisation sous QNX, l'adresse de base de la carte UNX232 doit être 280H et le niveau d'interruption IRQ3 ou bien IRQ4.

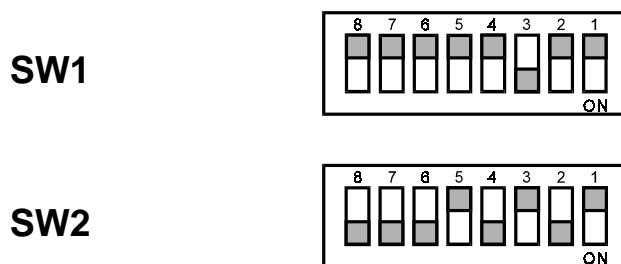
Il est à noter que sous QNX, la carte UNX n'est pas utilisée avec le registre de "polling" et est donc dans un mode totalement compatible avec la carte ADRS232.

Avant d'installer la carte, assurez-vous qu'il ne peut pas y avoir de conflit avec les périphériques déjà installés dans votre machine et notamment en ce qui concerne COM1 si vous avez choisi IRQ4 et COM2 si vous avez choisi IRQ3. Pour toutes les combinaisons données ci-dessous, les interrupteurs de SW3 doivent être en position OFF.

Configuration : 280H, IRQ3, 4 ports, pas de masque/polling



Configuration : 280H, IRQ3, 8 ports, pas de masque/polling



IX. L'INTERFACE RS422A-RS485-V11

Les normes RS422-RS485 sont des normes d'interface électrique qui permettent la transmission des informations en différentiel. Cette méthode permet d'obtenir une meilleure immunité aux bruits et des transmissions sur de longues distances.

IX.1 NOTATION DE L'ETAT DES SIGNAUX

TENSION	NEGATIVE +TX < -TX A < B	POSITIVE +TX > -TX A > B
ETAT BINAIRE (signaux de données)	« Bits de stop » et bits à 1, état MARK	« Bits de start » et bits à 0, état SPACE
FONCTION (signaux de contrôle)	FERME, OFF (inactif)	OUVERT, ON (actif)

EIA RS422 EIA-RS485	CCITT V11	
+ TX	A	A est à un potentiel inférieur à B à l'état de repos
- TX	B	
+ RX	A'	A' est à un potentiel inférieur à B' à l'état de repos
- RX	B'	

L'état de repos, dans les liaisons asynchrones, correspond à la transmission permanente de bits de stop sur les signaux de données (état MARK), et à la transmission de signaux de commande inactifs (circuits OUVERTS).

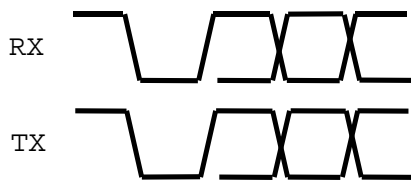
REMARQUE :

Les signaux différentiels +TX et +RX sont à un potentiel inférieur aux signaux -TX et -RX en état de repos. Cette remarque est particulièrement importante lors de la connexion à d'autres interfaces RS422A; en effet, beaucoup de confusion règne quant à la dénomination des signaux +TX, +RX, -TX et -RX (A,A',B et B' selon le CCITT) dans de nombreuses interfaces de divers constructeurs.

CHRONOGRAMME DES SIGNAUX RS422-RS485

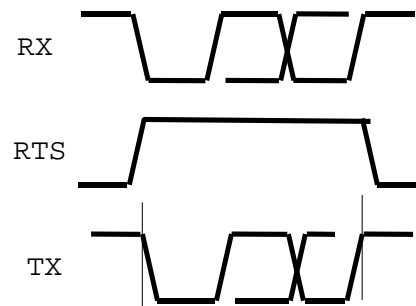
FULL-DUPLEX

Validation permanente de la transmission

**RS422A**

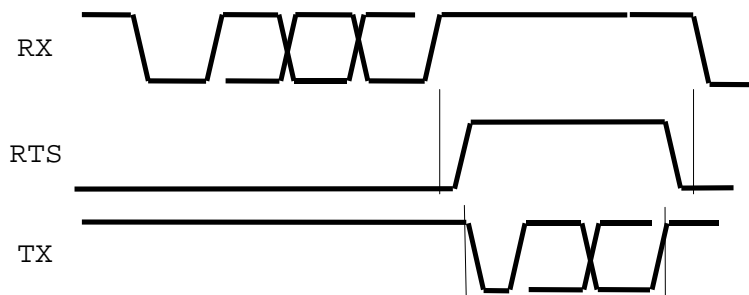
FULL-DUPLEX

Validation contrôlée de la transmission

**RS422A MULTIPOINT****RS485 2 paires**

HALF-DUPLEX

Validation contrôlée de la transmission

**RS485**

IX.2 PARTICULARITES DES LIGNES DE TRANSMISSION RS422A ET RS485

ATTENTION

Sur les produits ACKSYS, les résistances de terminaison et de polarisation sont intégrées au produit et sélectionnables par « switches ».

IX.2.1 Résistance de terminaison :

La résistance de terminaison de ligne pour le récepteur en mode RS422A (100 Ω) et pour le transmetteur/récepteur en mode RS485 (120 Ω) permet de réduire les réflexions perturbant la réception, générées dans une longue ligne à haut débit. La résistance de terminaison n'est pas nécessaire dans les cas suivants :

- Milieu exempt de perturbations.
- Distances et débits compris dans les limites 1000 m à 9600 bps et 100 m à 112 Kbps.

IX.2.2 Polarisation :

La polarisation de ligne est nécessaire afin d'avoir un état stable dans les deux cas suivants :

- En mode RS485, lors de la transition de la transmission à la réception.
- En mode RS422A, si plusieurs transmetteurs sont en bus et que, par conséquent, l'utilisation de l'état haute impédance est requise.

Une seule polarisation par ligne est nécessaire.

IX.2.3 Type de câble de raccordement :

- Paire(s) torsadée(s) avec ou sans blindage, jauge 22-24 AWG, impédance 100-120 Ω .
- Capacité entre conducteurs : 50 pF.
- Capacité entre blindage et conducteurs : 70 pF max.
- Ne jamais laisser certains fils du câble en l'air (non reliés) : relier les fils et paires non utilisés à la masse ou à la terre, à une extrémité au moins.

IX.2.4 Raccordement des masses électriques (GND) :

- soit par un fil commun (ou une paire disponible) reliant chaque équipement,
- soit en se reposant sur la prise de terre de chaque équipement ; il faut alors vérifier que la tension maximale en mode commun sur la ligne ne dépasse pas le niveau admissible par chaque équipement (c'est à dire qu'ils aient tous la même référence de terre),
- en cas d'utilisation homogène d'interfaces isolées galvaniquement, la connexion du GND n'est pas indispensable. La tension de mode commun sera limitée à la tension correspondant à l'isolement galvanique.

IX.2.5 Blindage :

Le blindage peut être raccordé directement sur le châssis des équipements ou sur la broche PGND des connecteurs. En cas de problèmes, toujours vérifier la qualité de la prise de terre car sur certains appareils, le châssis ou le PGND ne sont pas reliés directement à la terre.

Le blindage du câble est nécessaire dans un milieu industriel très perturbé. Il faut prendre en compte quatre paramètres contradictoires.

- La fréquence des perturbations. Les perturbations de basse fréquence (alimentation secteur, moteur, poste de soudure à l'arc, éclairage, appareils inductifs, etc.) peuvent être réduites si le blindage est raccordé en un seul point de la liaison. Par contre les

perturbations de haute fréquence (foudre, décharge électrostatique, TRS, etc.) sont réduites si le blindage est raccordé à la terre de place en place.

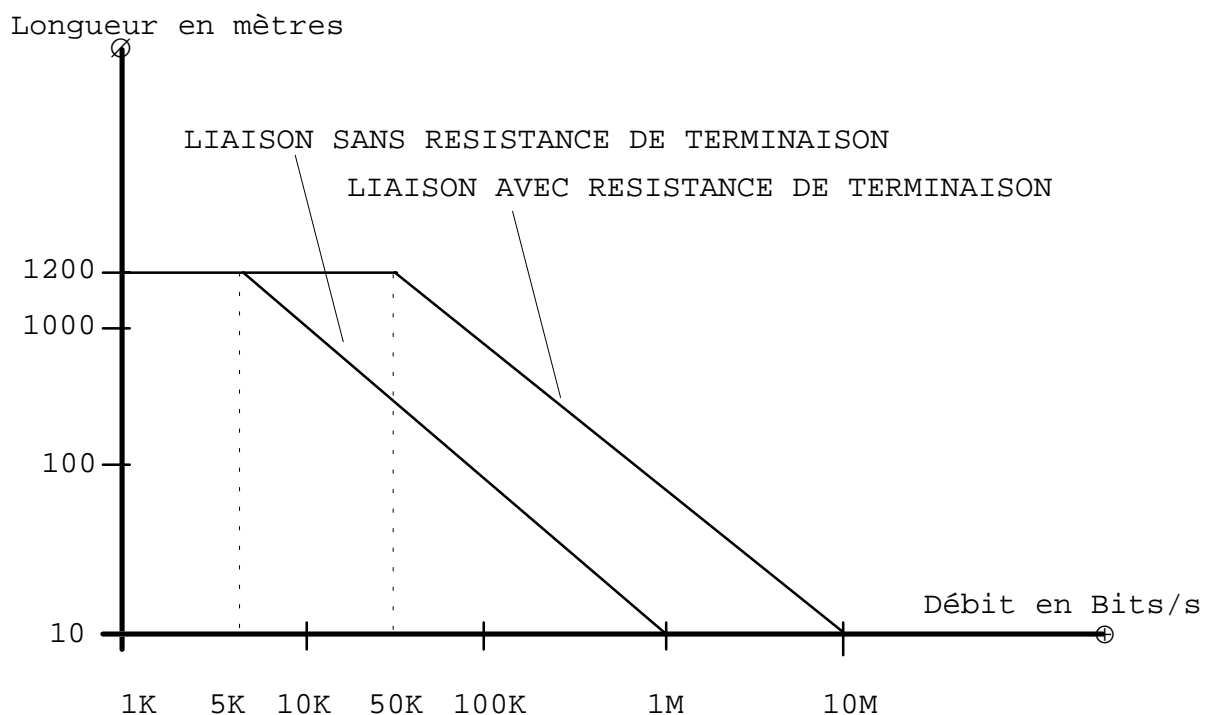
- La différence de potentiel entre les terres des différents équipements. Si toutes les terres ne sont pas prises au même point, il peut apparaître des courants de fuite qui passent par le blindage lorsqu'il est raccordé à plusieurs équipements à la fois. Si ces courants sont trop importants, ils peuvent rendre les manipulations dangereuses, et déclencher les protections d'alimentation (disjoncteurs).
- Les normes CE. Leur respect impose de placer suffisamment de raccords de terre pour éliminer les perturbations en moyenne et haute fréquence.
- La capacité entre blindage et conducteurs affaiblit les signaux en fonction de la vitesse de transmission et de la distance entre les équipements. Un blindage très capacitif pourra donc imposer une réduction de la distance ou de la vitesse (voir abaque ci-après).

IX.2.6 Masse et blindage :

Attention, certains équipement relient directement, de façon interne, la masse électrique (GND) et la prise de terre, mais ce n'est pas une règle générale. Ceci peut provoquer des courants de fuite également dans le fil de masse électrique.

IX.2.7 Longueur de câble pour une liaison RS422A OU RS485

Valeurs minimales pour un câble à paires torsadées de jauge 24 AWG



Les formules suivantes donnent approximativement la relation distance/débit :

$$\log d = 6,4 - (0,9 \times \log v) \quad (\text{sans résistance de terminaison})$$

$$\log d = 7,3 - (0,9 \times \log v) \quad (\text{avec résistance de terminaison})$$

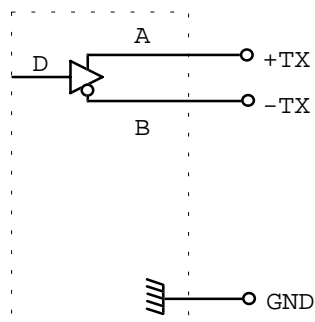
avec :

d : distance en mètres, limitée à 1200 m

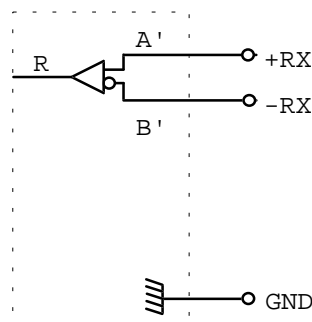
v : débit physique en bits par seconde

RACCORDEMENT RS422A (4 FILS)**SIMPLEX**

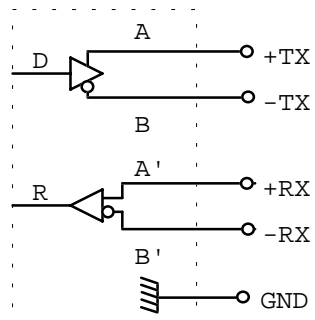
INTERFACE RS422A POUR LIAISON
POINT A POINT OU MULTIPPOINT
TYPE MAITRE

**SIMPLEX**

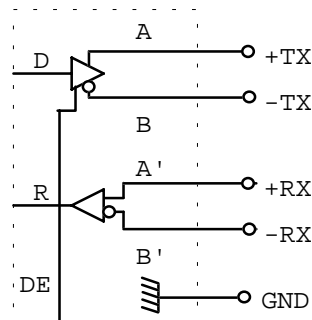
INTERFACE RS422A POUR LIAISON
POINT A POINT OU MULTIPPOINT
TYPE ESCLAVE

**FULL-DUPLEX**

INTERFACE RS422A POUR LIAISON
POINT A POINT OU MULTIPPOINT
TYPE MAITRE

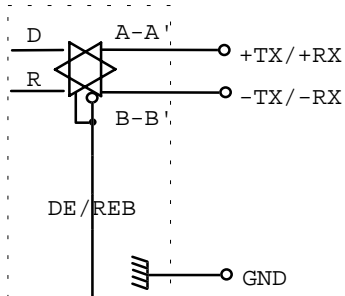
**FULL-DUPLEX**

INTERFACE RS422A POUR LIAISON
POINT A POINT OU MULTIPPOINT
TYPE MAITRE/ESCLAVE (POLLING-SELECTING)

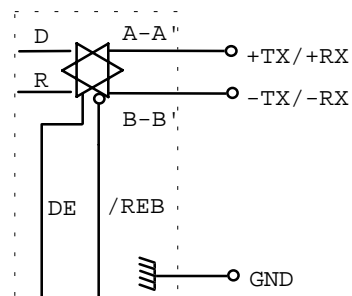


RACCORDEMENT RS485 (2 FILS)**HALF-DUPLEX**

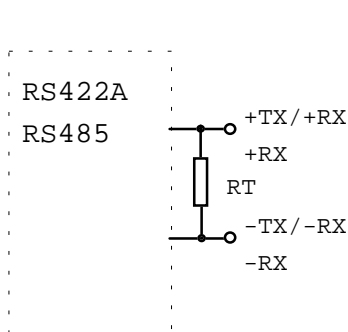
INTERFACE RS485 POUR LIAISON
POINT A POINT OU MULTIPPOINT
TYPE MAITRE/ESCLAVE (POLLING-SELECTING)

**HALF-DUPLEX**

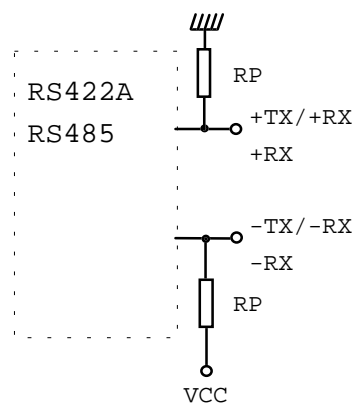
INTERFACE RS485 POUR LIAISON
POINT A POINT OU MULTIPPOINT
TYPE MULTIMAITRE AVEC ECHO

**RESISTANCE DE TERMINAISON RT**

$RT = 100-120 \text{ Ohms}$

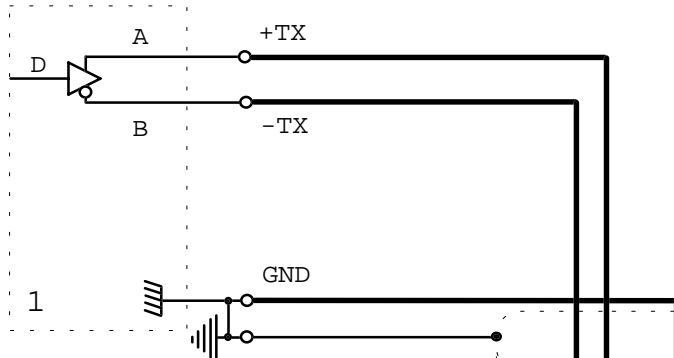
**RESISTANCE DE POLARISATION RP**

$RP = 1 \text{ KOhms max en RS422A}$
 $RP = 560 \text{ Ohms max en RS485}$

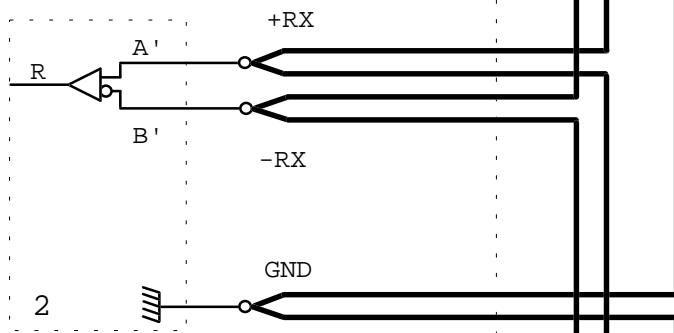


CABLAGE RS422A SIMPLEX MULTIPOINT

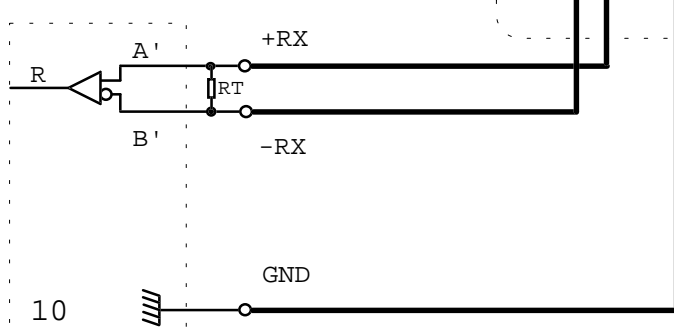
INTERFACE RS422A POUR
LIAISON MULTIPOINT
TYPE MAITRE



INTERFACE RS422A POUR
LIAISON MULTIPOINT
TYPE ESCLAVE INTERMEDIAIRE



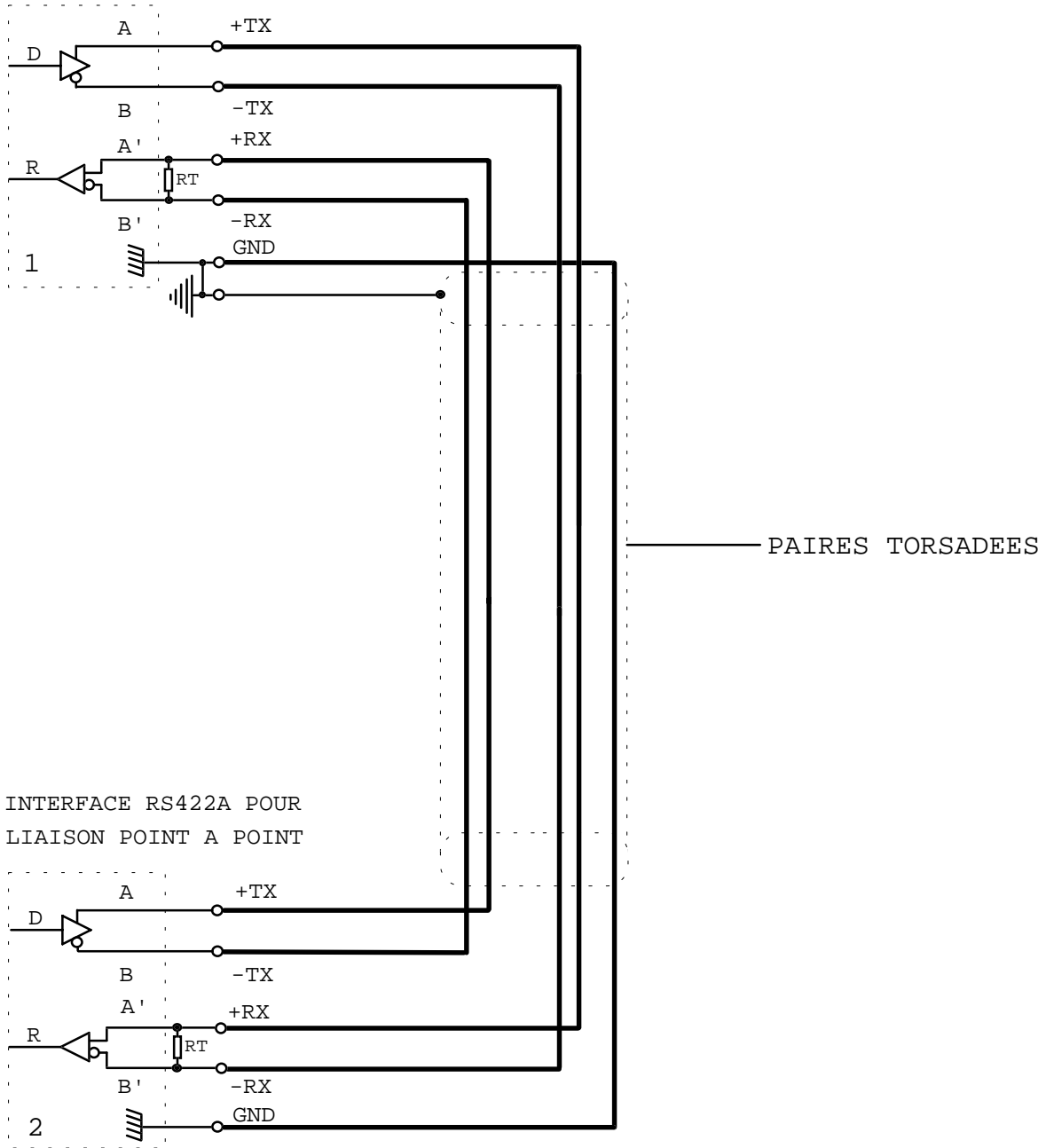
INTERFACE RS422A POUR
LIAISON MULTIPOINT
TYPE ESCLAVE TERMINAL



— PAIRES TORSADEES

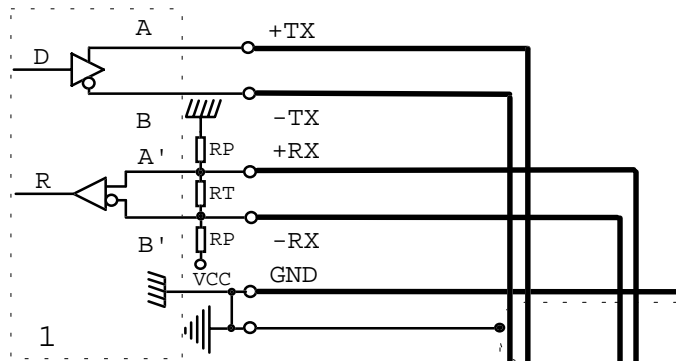
CABLAGE RS422A FULL-DUPLEX POINT A POINT

INTERFACE RS422A POUR
LIAISON POINT A POINT

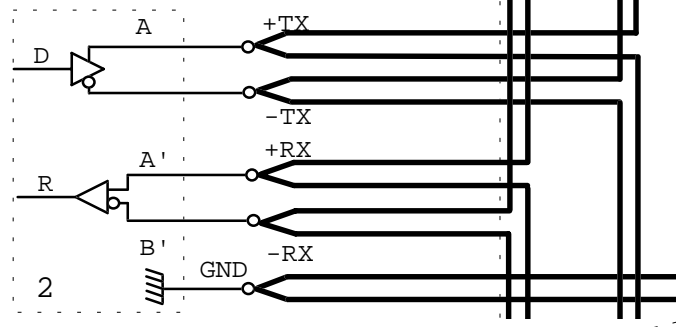


CABLAGE RS422A FULL-DUPLEX MULTIPOINT

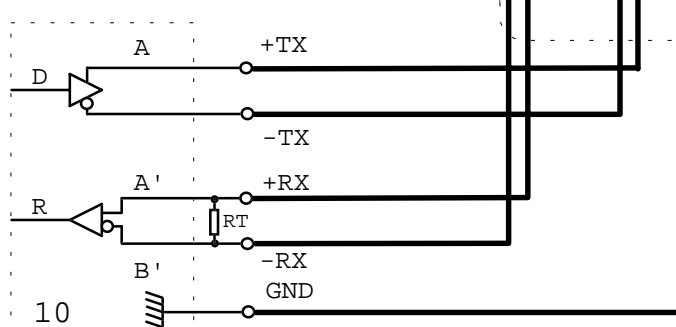
INTERFACE RS422A POUR
LIAISON MULTIPOINT TYPE MAITRE
(POLLING SELECTING)



INTERFACE RS422A POUR
LIAISON MULTIPOINT
TYPE ESCLAVE INTERMEDIAIRE



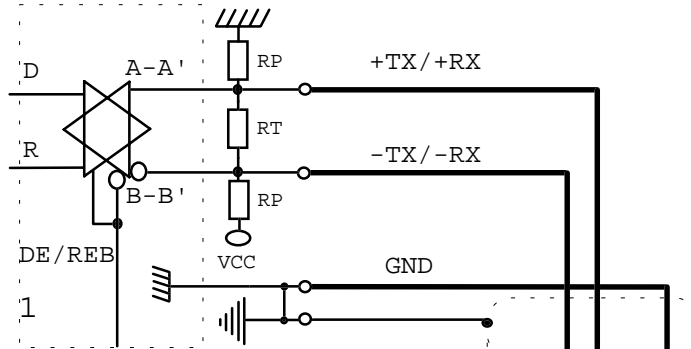
INTERFACE RS422A POUR
LIAISON MULTIPOINT
TYPE ESCLAVE TERMINAL



—— PAIRES TORSADEES

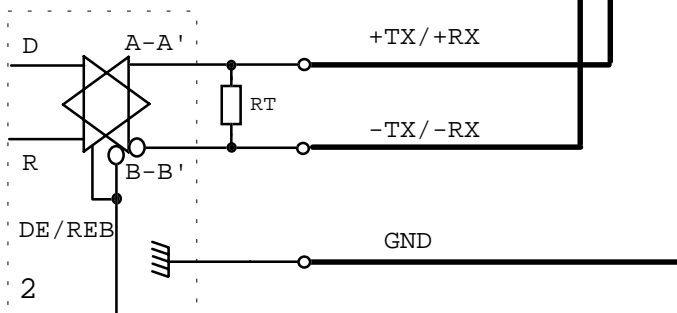
CABLAGE RS485 HALF-DUPLEX POINT A POINT

INTERFACE RS485 POUR
LIAISON POINT A POINT



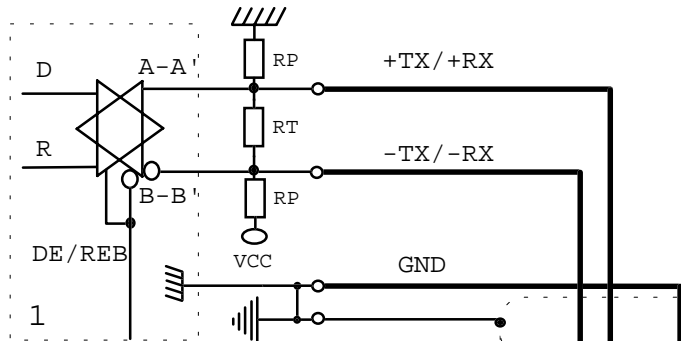
— PAIRES TORSADEES

INTERFACE RS485 POUR
LIAISON POINT A POINT

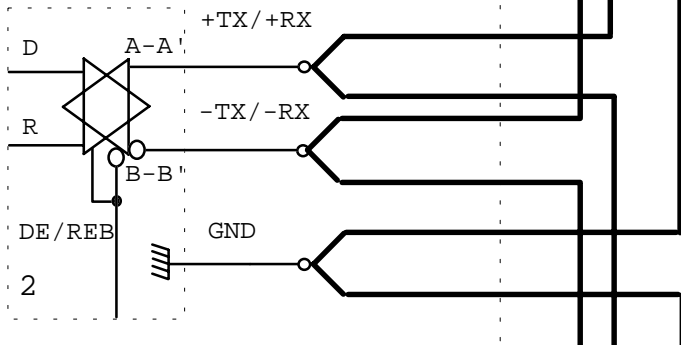


CABLAGE RS485 HALF-DUPLEX MULTIPOINT

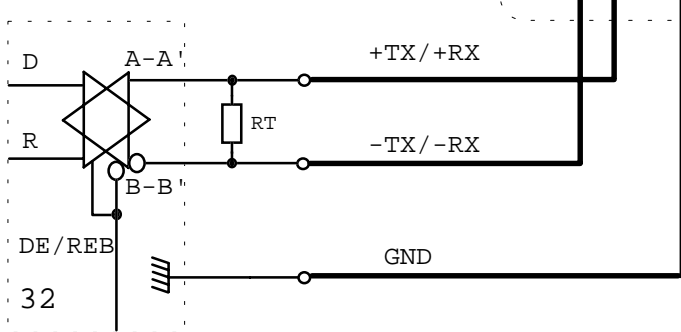
INTERFACE RS485 POUR LIAISON
MULTIPOINT TYPE MAITRE
(POLLING-SELECTING)



INTERFACE RS485 POUR LIAISON
MULTIPOINT TYPE ESCLAVE
INTERMEDIAIRE
(POLLING-SELECTING)



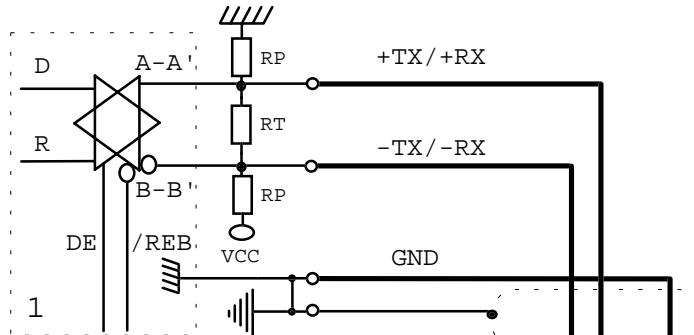
INTERFACE RS485 POUR LIAISON
MULTIPOINT TYPE ESCLAVE
TERMINAL
(POLLING-SELECTING)



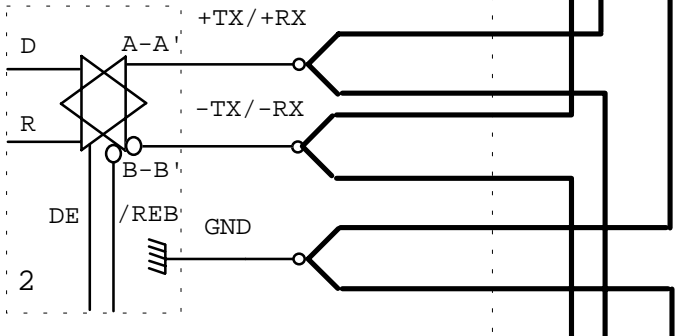
—— PAIRES TORSADEES

CABLAGE RS485 HALF-DUPLEX MULTIPOINT

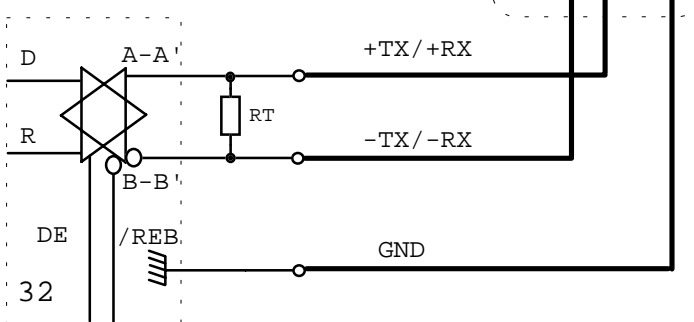
INTERFACE RS485 POUR LIAISON
MULTIPOINT TYPE MULTIMAITRE TERMINAL
AVEC ECHO (DETECTION DE COLLISION)



INTERFACE RS485 POUR LIAISON
MULTIPOINT TYPE MULTIMAITRE
TERMINAL AVEC ECHO
(DETECTION DE COLLISION)



INTERFACE RS485 POUR LIAISON
MULTIPOINT TYPE MULTIMAITRE
TERMINAL AVEC ECHO
(DETECTION DE COLLISION)



— PAIRES TORSADEES

X. EXTRAITS DE LA NORME EIA

Le tableau ci-dessous constitue un rappel des différentes caractéristiques de la norme EIA concernant les avis EIA/TIA-562 (RS232, EIA/TIA-574, EIA/TIA-561, etc.), RS422A et RS485.

Il est utile de s'y rapporter en ce qui concerne la longueur de câble maximale autorisée par les différentes normes.

SPECIFICATIONS		EIA/TIA-562 (RS232...)	RS-422A	RS-485
Type de communication		Unipolaire	Différentiel	Différentiel
Nombre de transmetteurs et récepteurs alloués par la ligne		1 transmetteur 1 récepteur	1 transmetteur 10 récepteurs	32 transmetteurs 32 récepteurs
Longueur max. de câble		16,5m	1320 m	1320 m
Débit maximum		64 Kbits/s	10 Mbits/s (sur 13 m)	10 Mbits/s (sur 13 m)
Tension en mode commun sur transmetteur		±25V	±7V	-7V à +12V
Niveau de Tension du transmetteur	En charge	±5V	±2V	±1,5V
	A vide	±15V	±5V	±5V
Résistance de charge sur le transmetteur		3 à 7 KΩ	100 Ω	54Ω
Courant de fuite du transmetteur	Sous tension	±100μA
	Hors tension	$V_{\max}/300\Omega$	±100μA	±100μA
Temps de montée		30V/μS max
Gamme de tension sur l'entrée du récepteur		±15V	-7V à +7V	-7V à +12V
Sensibilité du récepteur		±3V	±200mV	±200mV
Résistance d'entrée du récepteur		3 à 7 KΩ	4 KΩ min.	12 KΩ min.

XI. PERTURBATIONS RADIOELECTRIQUES (FCC)

Cet équipement génère et utilise des fréquences radioélectriques et, s'il n'est pas installé et utilisé correctement, en strict accord avec les spécifications du constructeur, peut engendrer des interférences sur la réception de la radio et de la télévision.

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la Classe A en accord avec les spécifications J.15 prescrites dans le règlement sur le brouillage radioélectrique édicté par les "FCC Rules and Regulations".

Si cet équipement provoque néanmoins des interférences sur la réception de la radio ou de la télévision, nous vous proposons d'essayer de corriger ces interférences en employant une ou plusieurs des mesures citées ci-dessous :

- Orientez l'antenne de réception différemment
- Eloignez votre ordinateur de l'antenne
- Branchez votre ordinateur sur une prise de courant différente de votre récepteur.

Si nécessaire, consultez ACKSYS ou un technicien RADIO-TELEVISION expérimenté pour d'autres suggestions.

XII. CARACTERISTIQUES DU 16C550

Les unités de communication intégrées sur les cartes passives ACKSYS (16C550, 16C552, 16C554) sont entièrement compatibles avec l'unité de communication de base 16C550.

XII.1 GENERALITES

- ◆ Communications asynchrones série.
- ◆ Technologie CMOS.
- ◆ Conçu pour un interfaçage simple avec les microprocesseurs les plus répandus (80286, 80386, 80486, PENTIUM, etc.).
- ◆ Emulation des modes 16C450 et 8250.
- ◆ Contrôle indépendant des unités de transmission, réception, état, interruptions.
- ◆ Générateur de bauds programmable.
- ◆ Horloge de réception indépendante.
- ◆ 16 octets de FIFO en réception comme en transmission.
- ◆ Format des données entièrement programmable :
 - ◆ 5, 6, 7 ou 8 bits par caractère,
 - ◆ Parité paire, impaire, absente ou forcée,
 - ◆ 1, 1+1/2, 2 bits d'arrêt.
- ◆ Détecteur de mauvais bit de départ.
- ◆ Lecture de l'état détaillé de la ligne de communication.
- ◆ Signaux TTL trois états et bus de données bi-directionnel.
- ◆ Détection et génération de "Break".
- ◆ Possibilité de rebouclage interne pour la détection de pannes.
- ◆ Détection d'erreur : "Break", parité, dépassement, trame.
- ◆ Système d'interruptions arbitré par priorité.
- ◆ Alimentation +5V unique.

Ce composant est utilisé pour sérialiser les données du côté transmetteur et pour les désérialiser côté récepteur.

Le format des données série transmission/réception est le suivant :

- 1 bit de départ,
- + De 5 à 8 bits de données (transmis du bit de poids faible au bit de poids fort),
- + 1 bit de parité si programmé,
- + 1 bit + 1/2 bit (si 5 bits de donnée) ou 2 bits d'arrêt.

Avec une horloge de base à 1.8432 MHz, la vitesse de transmission-réception maximale recommandée est de 115200 bits/s.

Des registres internes permettent au programmeur d'utiliser plusieurs types d'interruptions, différents formats de caractères, et une gestion simplifiée des signaux "Modem". Il est possible de lire les bits d'état du 16C550 à tout moment et de les modifier dynamiquement si nécessaire.

Le 16C550 intègre un générateur de bauds programmable qui est capable de diviser une fréquence extérieure, fournie par un quartz ou un signal TTL, par un facteur allant de 1 à $2^{16}-1$.

L'unité de communication asynchrone peut fonctionner aussi bien en mode "Polling" qu'en mode interruptions.

XII.2 ADRESSAGE DES REGISTRES

Le tableau ci-dessous indique les différents registres sélectionnés selon les configurations d'adresses :

DLAB	A2	A1	A0	Registre
0	0	0	0	Registre de réception (lecture) Registre de transmission (écriture)
0	0	0	1	Autorisation des interruptions
X	0	1	0	Identification des interruptions
X	0	1	1	Contrôle ligne
X	1	0	0	Contrôle Modem
X	1	0	1	Etat ligne
X	1	1	0	Etat Modem
X	1	1	1	Inutilisé*
1	0	0	0	Facteur de division (LSB)
1	0	0	1	Facteur de division (MSB)

** voir cependant le paragraphe XII.3 pour certaines cartes*

Note:

Le bit DLAB est le bit 7 du registre de contrôle ligne et il doit être positionné à 1 par le logiciel pour accéder aux registres du générateur de bauds.

Adresses des registres												
Bit	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	2	3	4	5	6	7	0 DLAB=1	1 DLAB=1
	Registre de réception	Registre d'émission	Registre de validation des interruptions	Registre d'identification des interruptions	Registre de contrôle FIFO	Registre de contrôle ligne	Registre de contrôle modem	Registre d'état ligne	Registre d'état modem	'Scratch Register'	Diviseur BRG Poids faibles	Diviseur BRG Poids forts
	RHR	THR	IER	ISR	FCR	LCR	MCR	LSR	MSR	SCR	DLL	DLH
0	Donnée bit 0 (Note 1)	Donnée bit 0	Autorise les interruptions sur réception (ERBFI)	'0' si présence d'une condition d'interruption	Validation des FIFO	Nombre de bits par caractères Bit 0 (WLS0)	Terminal de données prêt (DTR)	Donnée prête (DR)	Variation du signal 'Prêt à émettre' (DCTS)	Bit 0	Bit 0	Bit 8
1	Donnée bit 1	Donnée bit 1	Autorise les interruptions sur registre d'émission vide (ETBEI)	Identification de l'interruption Bit 0	Reset FIFO de réception	Nombre de bits par caractères Bit 1 (WLS1)	Demande pour émettre (RTS)	Ecrasement de caractère (OE)	Variation du signal 'Poste de donnée prêt' (DDSR)	Bit 1	Bit 1	Bit 9
2	Donnée bit 2	Donnée bit 2	Autorise les interruptions sur état ligne (ELSI)	Identification de l'interruption Bit 1	Reset FIFO d'émission	Nombre de bits de stop (STB)	(non utilisé)	Erreur de parité (PE)	Fin d'impulsion de sonnerie (TERI)	Bit 2	Bit 2	Bit 10
3	Donnée bit 3	Donnée bit 3	Autorise les interruptions sur état modem (EDSSI)	Identification de l'interruption Bit 2	Sélection Mode DMA	Autorise le contrôle de parité (PEN)	Autorisation des interruptions	Erreur de trame (FE)	Variation du signal de présence porteuse (DDCD)	Bit 3	Bit 3	Bit 11
4	Donnée bit 4	Donnée bit 4	0	0	Réservé	Sélection parité paire (EPS)	Boucle	Interruption par 'Break' (BI)	Prêt à émettre (CTS)	Bit 4	Bit 4	Bit 12
5	Donnée bit 5	Donnée bit 5	0	0	Réservé	Bit de parité forcé	0	Registre d'émission libre (THRE)	Poste de donnée prêt (DSR)	Bit 5	Bit 5	Bit 13
6	Donnée bit 6	Donnée bit 6	0	1 en mode FIFO	Seuil de réception (LSB)	Envoie un 'Break'	0	Transmetteur vide (TEMT)	Indicateur d'appel (RI)	Bit 6	Bit 6	Bit 14
7	Donnée bit 7	Donnée bit 7	0	1 en mode FIFO	Seuil de réception (MSB)	Accès aux registres du diviseur (DLAB)	0	0	Présence porteuse (DCD)	Bit 7	Bit 7	Bit 15

Note 1: Le bit 0 est le bit de poids faible. C'est le premier bit émis ou reçu en série

XII.2.1 Registre de contrôle de ligne

- **Bit 0 et 1 :**

Ces deux bits spécifient le nombre de bit dans chaque caractère lu ou transmis selon le tableau ci-dessous :

Bit 1	Bit 0	Longueur
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

- **Bit 2 :**

Ce bit spécifie le nombre de bits de stop à la fin de chaque caractère reçu ou émis par le 16C550.

- Bit 2 = 0 \Rightarrow 1 bit de stop

- Bit 2 = 1 \Rightarrow 1 + ½ bit de stop pour 5 bits de données
 \Rightarrow 2 bits de stop pour 6, 7 ou 8 bits de données.

- **Bit 3 :**

Ce bit autorise ou interdit le bit de parité.

- Bit 3 = 1 \Rightarrow 1 bit de parité généré

- Bit 3 = 0 \Rightarrow Pas de parité

- **Bits 4 et 5 :**

Bit 5	Bit 4	Parité
0	0	Impaire
0	1	Paire
1	0	Bit de parité forcé et contrôlé à 1
1	1	Bit de parité forcé et contrôlé à 0

- **Bit 6 :**

Quand le bit 6 est au 1 logique, le 16C550 génère sur la ligne un "Break" jusqu'à ce que le bit 6 soit remis à 0.

- **Bit 7 :**

Bit DLAB : Ce bit permet, lorsqu'il est à 1, d'accéder aux registres du générateur de bauds. Il doit être remis à 0 pour accéder aux registres d'interruption, de transmission et de réception.

XII.2.2 Le générateur de Bauds

Le 16C550 contient un générateur de bauds programmable capable de diviser une fréquence d'entrée issue d'un quartz ou bien un signal TTL par un facteur compris entre 1 et $2^{16}-1$.

La fréquence de sortie de générateur de bauds est 16 fois supérieure à la vitesse de transmission-réception. Deux registres 8 bits permettent d'écrire les poids forts et les poids faibles du facteur de division.

Vitesse	1,8432 MHz		6 MHz		8 MHz	
	Diviseur	% erreur	Diviseur	% erreur	Diviseur	% erreur
50	2304	0	7500	0	10000	0
75	1536	0	5000	0	6667	0,005
110	1047	0,026	3409	0,003	4545	0,010
150	768	0	2500	0	3333	0,010
300	384	0	1250	0	1667	0,020
600	192	0	625	0	833	0,040
1200	96	0	312	0,160	417	0,080
1800	64	0	208	0,160	278	0,080
2000	58	0,69	187	0,267	250	0
2400	48	0	156	0,160	208	0,160
3600	32	0	104	0,160	139	0,080
4800	24	0	78	0,160	104	0,160
7200	16	0	52	0,160	69	0,640
9600	12	0	39	0,160	52	0,160
19200	6	0	20	2,344	26	0,160
38400	3	0	10	2,344	13	0,160
56000	2	2,86	7	4,337	9	0,800
57600	2	0	7	6,994	9	3,680
115200	1	0	3	8,507	4	7,840
125000			3	0	4	0
187500			2	0		
250000					2	0
375000			1	0		
500000					1	0

XII.2.3 Le registre d'état ligne

Le registre d'état ligne donne au CPU toutes les informations relatives au transfert des données. La signification de chaque bit est définie ci-dessous :

- **Bit 0 :**
Ce bit signifie qu'une donnée est prête dans le registre du récepteur ou bien qu'elle a été transférée dans le FIFO de réception. Il est automatiquement remis à 0 par la lecture du caractère ou du FIFO.
- **Bit 1 :**
Il indique l'écrasement du caractère contenu dans le registre de réception par un autre caractère avant que le premier ait été lu par l'unité centrale. En mode FIFO, il signifie le débordement du FIFO. Ce bit est remis à 0 par la lecture du registre d'état ligne.
- **Bit 2 :**
Ce bit indique une erreur de parité. Le caractère reçu n'a pas la bonne parité. Ce bit est automatiquement remis à 0 par la lecture du registre d'état ligne. En mode FIFO, ce bit signifie que le caractère qui se trouve actuellement au sommet du FIFO est sujet à une erreur de parité.
- **Bit 3 :**
Il indique une erreur de trame, autrement dit, le bit d'arrêt qui suit le dernier bit de données ou de parité est détecté avec la valeur 0. Ce bit est remis à 0 par la lecture du registre d'état ligne. En mode FIFO, ce bit signifie que le caractère qui se trouve actuellement au sommet du FIFO est sujet à une erreur de trame.
- **Bit 4 :**
Ce bit instruit l'unité centrale qu'un "Break" à été détecté sur la ligne (tous les bits à 0 pendant une durée supérieure à un caractère). En mode FIFO, ce bit signifie que le caractère présent au sommet du FIFO a été engendré par le signal "Break".
Note : Les événements associés aux bits 0,1,2,3 peuvent produire, si on le désire, une interruption dite de réception lorsqu'ils sont détectés.
- **Bit 5 :**
Ce bit indique que l'organe de transmission est prêt à recevoir un caractère de l'unité centrale. De plus, ce bit peut aussi générer une interruption de transmission si le registre des interruptions est correctement programmé. En mode FIFO, il indique que le FIFO de transmission est vide ; ce bit est remis à zéro dès qu'au moins un caractère est présent dans le FIFO.
- **Bit 6 :**
Lorsqu'il est à 0, il indique que le transmetteur est actuellement en cours d'opération. Il est automatiquement remis à 1 quand le registre d'émission et le registre de décalage en émission sont simultanément vides.
En mode FIFO, il indique, lorsqu'il est positionné à 1, que le FIFO de transmission et le registre de décalage de transmission sont complètement vidés.
- **Bit 7 :**
En mode 16C450, ce bit vaut toujours zéro ; en mode FIFO, il indique qu'il y a au moins un caractère erroné dans le FIFO de réception. Ce bit est automatiquement remis à zéro par la lecture du registre d'état ligne.

XII.2.4 Le registre de contrôle du FIFO

Ce registre n'est accessible qu'en écriture. Il permet de contrôler le fonctionnement du FIFO :

- ◆ Validation du mode FIFO.
- ◆ Effacement du FIFO.
- ◆ Sélection du mode de fonctionnement du FIFO en réception.

Ce contrôle s'effectue de la manière suivante :

- **Bit 0 :**
Ce bit (FCR0), positionné à l'état logique 1, valide le fonctionnement du FIFO aussi bien en transmission qu'en réception. Si ce bit est positionné à zéro, le mode FIFO est alors interdit et tous les caractères contenus dans le FIFO sont écrasés.
Lorsque l'on passe du mode 16C450 au mode FIFO du 16C550 ou vice versa, toutes les données du FIFO sont effacées.
Ce bit doit être positionné à 1 lorsque d'autres bits de ce registre doivent être programmés.
- **Bit 1 :**
Ce bit (FCR1), positionné à 1, efface toutes les données contenues dans le FIFO de réception, le compteur de caractères est, lui aussi, remis à zéro. Néanmoins, le registre de décalage n'est pas détruit. Le bit FCR1 se remet automatiquement à zéro à la suite de son utilisation.
- **Bit 2 :**
Ce bit (FCR2), positionné à 1, efface toutes les données contenues dans le FIFO de transmission, le compteur de caractères est, lui aussi, remis à zéro. Néanmoins, le registre de décalage n'est pas détruit. Le bit FCR2 se remet automatiquement à zéro à la suite de son utilisation.
- **Bit 3 :**
Ce bit (FCR3) concerne les broches TXRDY et RXRDY qui ne sont pas utilisées dans cette carte ; ces deux signaux peuvent être utilisés pour déclencher des cycles de DMA.
- **Bit 4, 5 :**
Les bits 4 et 5 (FCR4 et FCR5) sont réservés pour de futures utilisations.
- **Bit 6, 7 :**
Ces deux bits (FCR6 et FCR7) sont utilisés pour indiquer au FIFO de réception le seuil à partir duquel il doit générer une interruption FIFO. Les valeurs admises sont données ci-dessous :

Bit 7	Bit 6	Nombre de caractères contenus dans le FIFO
0	0	1
0	1	4
1	0	8
1	1	14

XII.2.5 Le registre d'identification des interruptions

Ce registre se situe à la même adresse que le registre de contrôle FIFO mais en lecture uniquement. La signification des bits de ce registre est la suivante :

- **Bit 0 :**
Ce bit peut être utilisé en mode "Polling" et indique, lorsqu'il est à 1, qu'une condition d'interruption est active; le contenu du registre d'identification peut alors être utilisé en tant que pointeur sur la routine d'interruption appropriée.
- **Bits 1 et 2 :**
Ces deux bits du registre d'identification peuvent être utilisés pour connaître la source de l'interruption, comme indiqué dans la table ci-dessous.
- **Bit 3 :**
Dans le mode FIFO, ce bit est positionné à 1 pour indiquer qu'une interruption "Time out" est en attente.
- **Bits 4 et 5 :**
Ces deux bits sont toujours au niveau logique zéro.
- **Bits 6 et 7 :**
Ces bits sont positionnés à 1 lorsque le bit FCR0 du registre de contrôle du FIFO vaut 1.

Le tableau ci-après résume les différentes causes d'interruptions et indique comment les remettre à zéro.

Tableau d'identification des interruptions :

Bit 3	Bit 2	Bit 1	Bit 0	Priorité	Source	Remise à 0
0	0	0	1		Aucune	
0	1	1	0	1 ^{ère} (Haute)	Ecrasement Parité Trame Break	Lecture du registre d'état ligne
0	1	0	0	2 ^{ème}	Caractère prêt FIFO prêt	Lecture du registre de réception ou FIFO
1	1	0	0	2 ^{ème}	Time out FIFO	Lecture du registre de réception ou FIFO
0	0	1	0	3 ^{ème}	Transmetteur vide	Ecriture caractère ou lecture du registre d'IT
0	0	0	0	4 ^{ème} (Basse)	CTS DSR RING RLSD	Lecture registre d'état modem

XII.2.6 Registre de validation des interruptions

Le registre d'interruptions permet de valider les cinq sources possibles d'interruption du 16C550. Chaque interruption peut activer individuellement le signal d'interruption de l'UART. Il est possible de désactiver totalement les interruptions en mettant à 0 les bits 0, 1, 2, 3 de ce registre. Plusieurs sources d'interruption peuvent être activées simultanément :

Bit 0 : Ce bit à 1 autorise une interruption lorsqu'un caractère est prêt à être lu par l'unité centrale. En mode FIFO, il active également les interruptions 'TIME OUT'

Bit 1 : Ce bit à 1 autorise une interruption lorsque le transmetteur est prêt à recevoir un caractère du CPU.

Bit 2 : Ce bit à 1 autorise les interruptions fournies par le registre d'état ligne (LSR).

Bit 3 : Ce bit à 1 autorise les interruptions fournies par le registre d'état modem (MSR).

Bit 4-7: Ces bits sont toujours au 0 logique.

Gestion des FIFOS sous interruptions

Lorsque les interruptions FIFO de réception et récepteur sont validées, une interruption de réception est envoyée au CPU dès que le nombre de caractères contenus dans le FIFO atteint le nombre programmé dans le registre de contrôle FIFO (bits 6 et 7). L'interruption est remise à zéro, ainsi que le bit correspondant dans le registre d'identification des interruptions, dès que le nombre de caractères contenus dans le FIFO descend au-dessous de la valeur programmée.

Il faut noter que l'interruption d'état a une priorité plus forte que celle de réception caractère. Le bit caractère prêt est positionné à 1 aussitôt qu'un caractère est transféré depuis le registre à décalage vers le FIFO ; ce bit est remis à zéro dès que le FIFO est vidé.

Lorsque les interruptions FIFO de réception et récepteur sont validées, une interruption de "Time out" est générée dès que les conditions ci-dessous sont détectées simultanément :

- ◆ Au moins un caractère dans le FIFO.
- ◆ Le plus récent des caractères reçus est arrivé dans un temps supérieur à la durée de réception des quatre caractères précédents.

Le plus récent caractère à été lu par le CPU dans un temps supérieur à la durée de réception des quatre caractères précédents. Ceci donne un temps de 160 ms à 300 bits/s (avec des caractères de 12 bits) pour générer l'interruption "Time out". Lorsqu'une interruption "Time out" est générée, la lecture d'un caractère du FIFO remet à zéro le timer.

En fonctionnement normal, le compteur du "time out" est automatiquement remis à zéro à chaque fois qu'un caractère est reçu ou qu'un caractère est pris dans le FIFO.

Lorsque l'interruption FIFO de transmission est validée, l'interruption FIFO de transmission vide survient lorsque le FIFO est complètement vidé ; cette interruption est automatiquement remise à zéro lorsqu'un nouveau caractère est écrit dans le FIFO. Jusqu'à seize caractères peuvent être envoyés dans le FIFO au moment de la réception de l'interruption.

Les interruptions "time out" et réception FIFO ont la même priorité que l'interruption réception de caractère. L'interruption FIFO de transmission vidé à la même priorité que l'interruption de transmission caractère.

Gestion des interruptions en mode « polling »

Lorsque les interruptions ne sont pas validées, il est possible de contrôler le FIFO en mode "polling" ; comme le transmetteur et le récepteur sont contrôlés séparément, l'un d'entre eux seulement peut être utilisé en mode "polling".

En mode "polling", le programmeur doit venir tester les organes de transmission et de réception à l'aide du registre d'état ligne.

Le bit 0 du registre d'état ligne est positionné à 1 aussi longtemps qu'il reste au moins un caractère dans le FIFO de réception.

Les bits 1 à 4 du registre d'état ligne indiquent, le cas échéant, quelles erreurs sont survenues et la gestion des erreurs doit être effectuée comme sous interruptions.

Le bit 5 du registre d'état ligne indique que le FIFO de transmission est vide tandis que le bit 6 indique que le FIFO de transmission, ainsi que le registre de décalage de transmission, sont vides.

Enfin, le bit 7 indique qu'au moins un caractère erroné est contenu dans le FIFO de réception.

XII.2.7 Le registre de contrôle modem

Ce registre 8 bits contrôle l'interfaçage Modem. La signification de chaque bit est définie ci-dessous :

- **Bit 0 :**
Ce bit contrôle le signal "Data terminal ready DTR".
 - Bit 0 = 0 \Rightarrow /DTR niveau logique 1, signal OUVERT (inactif) sur la jonction.
 - Bit 0 = 1 \Rightarrow /DTR niveau logique 0, signal FERME (actif) sur la jonction.
- **Bit 1 :**
Il contrôle le signal "Request to send RTS".
 - Bit 0 = 0 \Rightarrow /RTS niveau logique 1, signal OUVERT (inactif) sur la jonction.
 - Bit 0 = 1 \Rightarrow /RTS niveau logique 0, signal FERME (actif) sur la jonction.
- **Bit 2 :**
Ce bit contrôle le signal "/OUT1", il n'est pas utilisé par les cartes ACKSYS.
- **Bit 3 :**
Il contrôle le signal "/OUT2". Ce signal est utilisé sur les cartes FAST400, 2RS232, 4RS232 et 8RS232 pour valider l'amplificateur de la ligne d'interruption. Ce bit doit impérativement être positionné à 1 pour utiliser ces cartes sous interruptions. Il n'est pas utilisé sur les cartes COMFAST et UNX232.
- **Bit 4 :**
Ce bit permet de positionner le 16C550 en mode de diagnostic interne. Les informations transitent directement de l'émetteur vers le récepteur, les signaux /DTR, /RTS, /OUT1, /OUT2 sont, de façon interne, connectés aux entrées Modem, ce qui permet d'effectuer le test des fonctions et des registres internes du circuit. Dans le mode "Diagnostic", les interruptions récepteur-transmetteur sont totalement opérationnelles.
- **Bit 5-7 :**
Ces bits sont toujours forcés au zéro logique.

XII.2.8 Le registre d'état modem

Ce registre 8 bits instruit le CPU de l'état des lignes de contrôle du Modem ou du périphérique raccordé sur la ligne de communication. Les bits de ce registre sont positionnés à 1 lorsqu'ils changent d'état et sont remis à 0 par la lecture de ce même registre.

- **Bit 0 :**
Ce bit est le "Delta clear to send DCTS". Il indique que l'entrée /CTS a changé d'état depuis la dernière lecture du CPU.
- **Bit 1 :**
Ce bit est le "Delta data set ready DDSR". Il indique que l'entrée /DSR a changé d'état.
- **Bit 2 :**
Ce bit indique que l'entrée "Ring indicator RI" est passée de l'état logique 1 à l'état logique 0, ce qui correspond, au niveau de la jonction, au passage du niveau FERME (actif) au niveau OUVERT (inactif).
- **Bit 3 :**
Ce bit est le "Delta received line signal detector DRLSD". Il indique que l'entrée /RLSD a changé d'état.

Note : Dès qu'un des bits 0, 1, 2, 3 passe à l'état 1, une interruption de type "Etat Modem" est générée.

- **Bit 4 :**
Ce bit est l'image inverse de l'entrée /CTS.
- **Bit 5 :**
Ce bit est l'image inverse de l'entrée /DSR.
- **Bit 6 :**
Ce bit est l'image inverse de l'entrée /RI.
- **Bit 7 :**
Ce bit est l'image inverse de l'entrée /RLSD.

Ces quatre derniers bits sont donc à 1 quand l'entrée correspondante est OUVERTE (active) sur la jonction, et à 0 quand l'entrée est FERMEE (inactive) sur la jonction.

XII.3 PARTICULARITES DES CARTES 8RS232 ET UNX232

Les cartes 8RS232 et UNX232 disposent d'un registre de polling permettant d'identifier rapidement les unités de communication dont la ligne d'interruption est active. Sur ces cartes, le signal envoyé sur le niveau d'IRQ sélectionné est un OU logique des lignes d'interruption issues de chacune des unités de communication.

Pour déterminer quels sont les ports qui ont déclenché une interruption, les registres d'identification des interruptions (IIR) du 16C550 ainsi que le registre de polling peuvent être consultés :

Le registre de polling est un registre dont chacun des 8 bits représente la ligne d'interruption d'un 16C550. Un bit à 1 indique que la ligne d'interruption de l'unité de communication correspondante est activée, et par conséquent qu'une condition d'interruption est présente sur ce port.

L'accès au registre de polling s'effectue à l'adresse Base+7, c'est à dire à l'adresse du "scratch register" de la voie 1. Le registre de polling fonctionne uniquement en lecture.

La carte UNX232 dispose de plus d'un registre appelé registre de masque. Il s'agit d'un registre 8 bits qui permet à l'utilisateur d'inhiber individuellement ou en combinaisons les interruptions de chaque port, et ce sans changer le registre de validation des interruptions des 16C550: lorsque la ligne d'interruption d'un port est masquée, elle n'est plus prise en compte pour la génération du signal d'interruption envoyé sur le bus du PC. Le masquage de l'interruption d'un port s'effectue par la mise à 0 du bit correspondant à la ligne voulue. De la même manière le démasquage s'effectue par la mise à 1 de ce bit.

ATTENTION: Le registre d'identification des interruptions (IIR) d'une unité de communication dont la ligne d'interruption est masquée indiquera toujours la présence des conditions d'interruption programmées, bien que le signal d'interruption ne soit pas envoyé vers le PC.

XIII. INSTALLATION DES CARTES SOUS WINDOWS NT

XIII.1 INSTALLATION

Nous vous conseillons de respecter l'ordre suivant : installation des paramètres de Windows NT, puis installation de la carte dans l'ordinateur, puis contrôle du SETUP du BIOS, enfin vérification de l'installation.

- 1) Choisir une adresse de base et une (ou des) interruption(s) qui ne soient pas déjà utilisées dans le P.C. Par exemple : BASE=280 (hexadécimal), IRQ=10 (décimal). Consultez dans ce manuel, le chapitre qui concerne la carte à installer, pour plus de précisions sur les adresses et interruptions possibles.
- 2) Insérez la disquette « Driver Cartes Multivoies Passives », puis cliquez sur **Démarrer, Exécuter**, et tapez **A:\WINNT\SRLESETUP**
- 3) Cliquez sur le bouton **Ajouter...**, choisissez le type de carte à installer et cliquez sur **Suivant**. Entrez les paramètres de la carte. Pour certains modèles de cartes, vous devrez donner l'interruption pour chacune des voies. Cliquez sur **Ajouter** pour terminer.
- 4) Cliquez sur **Quitter**. Un contrôle est alors effectué sur les conflits possibles entre les cartes série passives installées (cartes ACKSYS, COM1, COM2 et autres). Si un conflit est signalé, corrigez le problème. Notez que l'utilisation de la même interruption sur deux cartes différentes est possible mais à éviter, car toutes les cartes et tous les PC ne supportent pas cela.
- 5) Un message vous rappelle d'activer les interruptions dans le cas d'un BIOS Plug-And-Play, et vous demande d'arrêter l'ordinateur. Arrêtez-le en cliquant sur **Démarrer, Arrêter**, etc.).
- 6) Contrôlez les interrupteurs et les cavaliers de la carte par rapport à vos choix. Sur la carte UNX232, l'interrupteur 1 de SW1 doit être OFF (pas de registre de masque). Installez la carte dans le P.C.
- 7) Démarrez le P.C. Si il est équipé d'un BIOS Plug-And-Play (cas de la plupart des P.C. récents) :
 - Lancez le SETUP (par exemple, avec le BIOS AWARD®, tapez la touche F1 pendant l'auto-test).
 - Recherchez les paramètres de configuration PNP ou Plug-And-Play
 - Vous devez voir une liste d'IRQ. Assurez-vous que les IRQ utilisées par la carte série sont affectées au BUS ISA.
 - Sur certains P.C. (par exemple COMPAQ), il faut également réserver les plages d'adresse I/O utilisées.
 - N'oubliez pas de sauvegarder vos modifications.
- 8) Redémarrez le P.C. et lancez Windows NT. Lancez l'Observateur d'événements et cherchez les événements récents dont la source est « Serial », et lisez le détail. Les messages qui parlent du FIFO sont normaux. Les autres messages peuvent signaler des erreurs d'installation ou des conflits. Corrigez les problèmes décrits.

XIII.2 TEST D'UNE CARTE

Pour tester une carte, vous devez avoir terminé son installation et redémarré Windows, configuré correctement votre BIOS SETUP, et bien entendu la carte doit être correctement configurée et installée.

- 1) Installez la disquette « Driver Cartes Multivoies Passives », puis cliquez sur **Démarrer, Exécuter**, et tapez **A:\WINNT\SRSLSETUP**
- 2) Cliquez sur la carte à tester, puis sur le bouton **Tester**. Attendez la fin du test (moins de cinq secondes). Le résultat s'affiche pour chaque port installé.

XIII.3 RETRAIT D'UNE CARTE

Nous vous conseillons de respecter l'ordre suivant : suppression des paramètres de Windows NT, puis retrait de la carte de l'ordinateur.

- 3) Installez la disquette « Driver Cartes Multivoies Passives », puis cliquez sur **Démarrer, Exécuter**, et tapez **A:\WINNT\SRSLSETUP**
- 4) Cliquez sur la carte à supprimer, puis sur le bouton **Supprimer**. Attention, l'effet est immédiat.
- 5) Cliquez sur **Quitter**. Un contrôle est alors effectué sur les conflits possibles entre les cartes série passives restantes (cartes ACKSYS, COM1, COM2 et autres).

XIV. INSTALLATION DES CARTES SOUS WINDOWS 95

XIV.1 CARTES 4 ET 8 VOIES

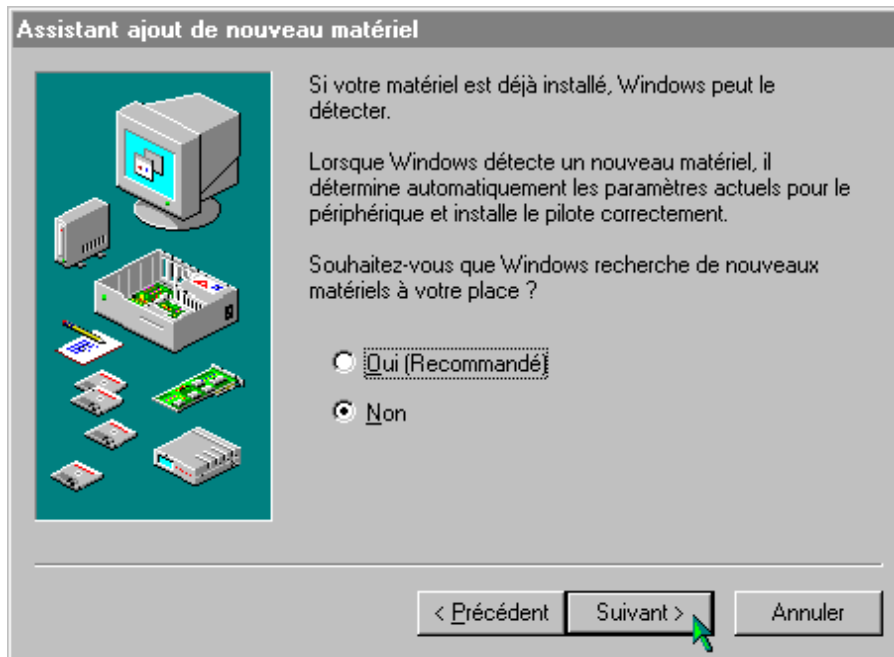
Cette section décrit la procédure d'installation des cartes 4RS232, 8RS232, et UNX232 sous Windows 95. Sélectionnez les adresses et l'interruption de la carte conformément à la documentation puis installez la carte dans le PC. Ouvrez le panneau de configuration de Windows 95 et lancez 'Ajout de périphérique' :



Cliquez sur <Suivant> pour commencer l'installation



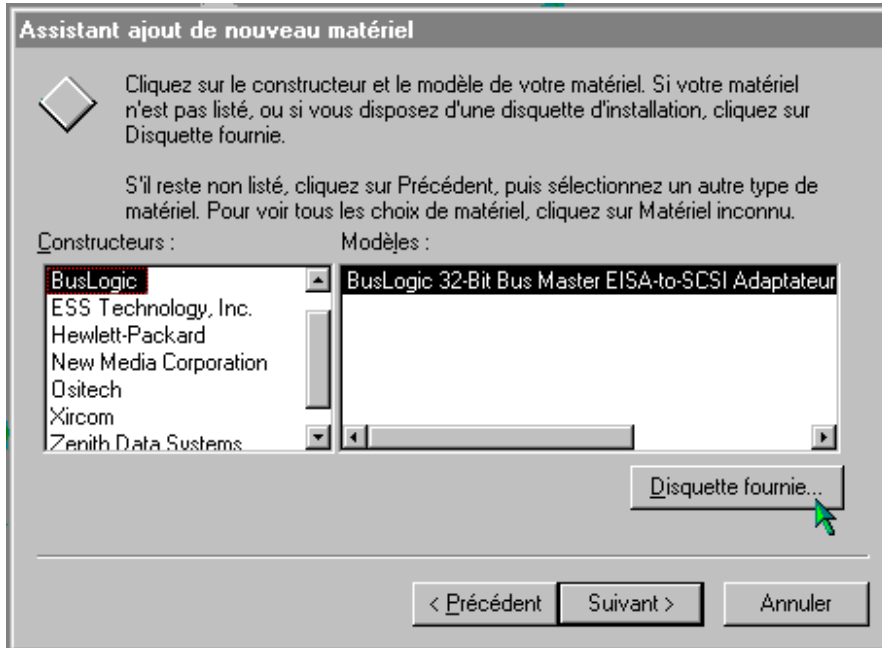
Choisissez 'Non' pour interdire à Windows d'effectuer la détection automatique de matériel, puis <Suivant> :



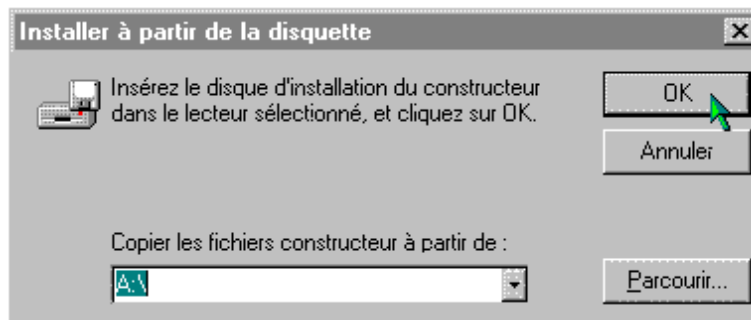
Sélectionnez 'Cartes adaptatrices multifonctions' puis <Suivant> :



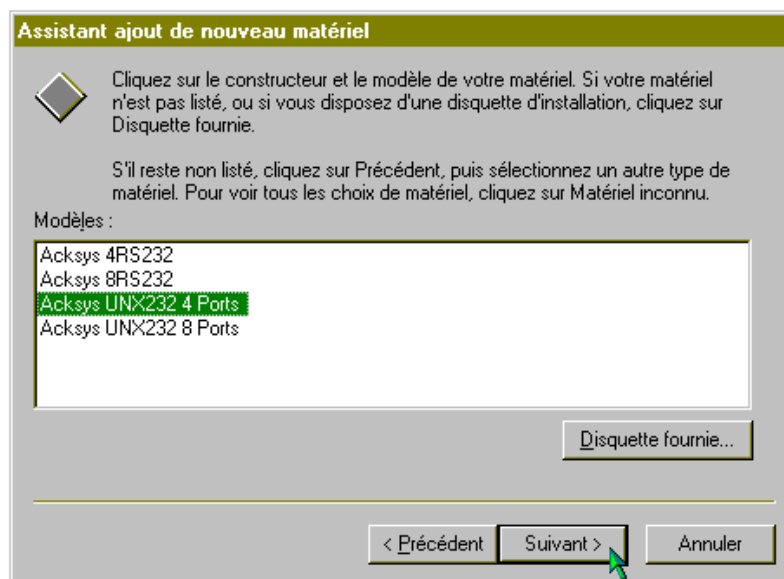
Cliquez sur 'Disquette fournie' :



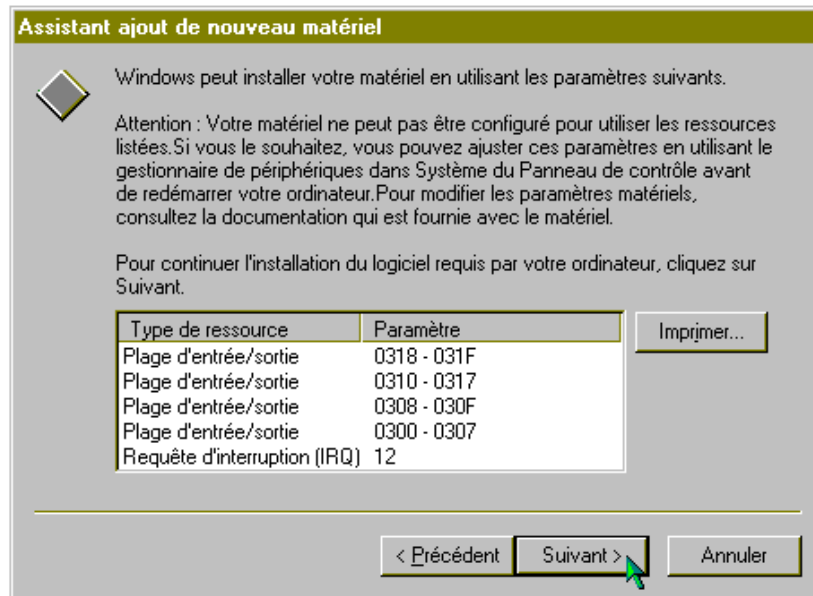
Insérez la disquette d'installation ACKSYS dans le lecteur, tapez A :\WIN95 puis cliquez sur 'OK' :



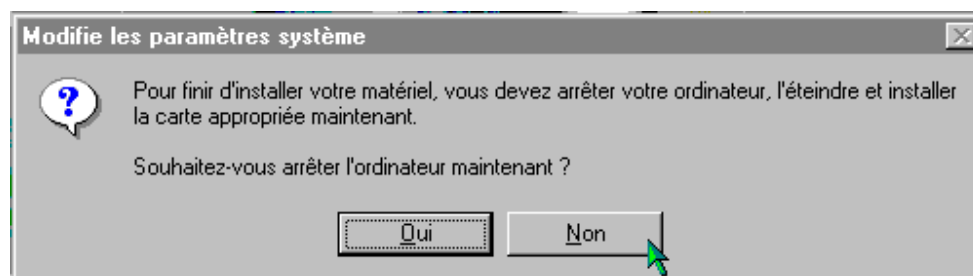
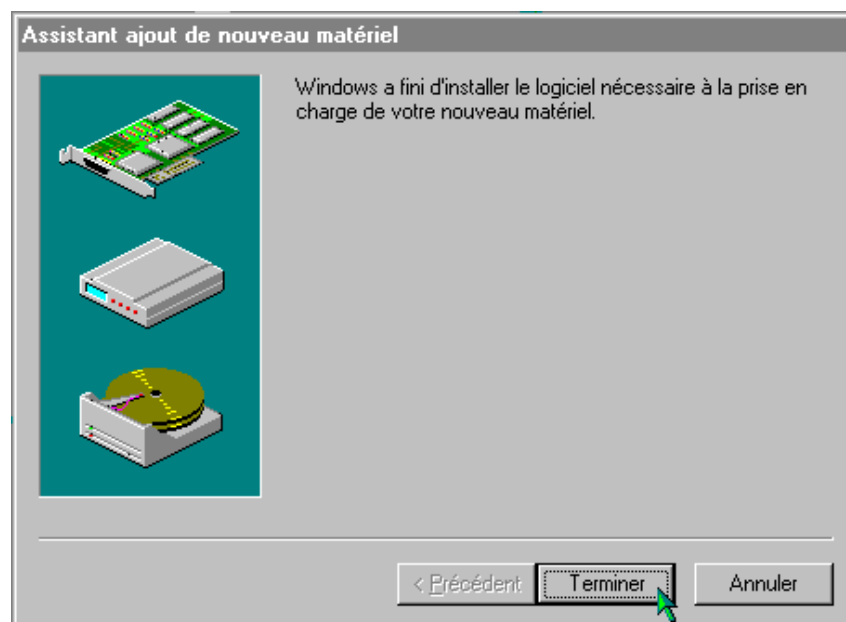
Sélectionnez votre carte dans la liste puis <Suivant> :



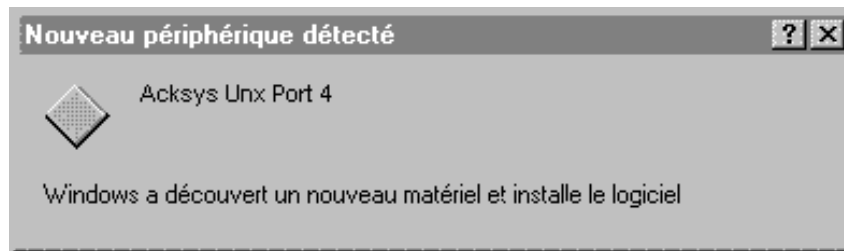
Les paramètres affichés dans l'écran suivant ne correspondent pas nécessairement à la configuration de la carte. Cliquez sur <Suivant>



Cliquez sur 'Terminer' puis sur 'Non' pour ne pas relancer le système :



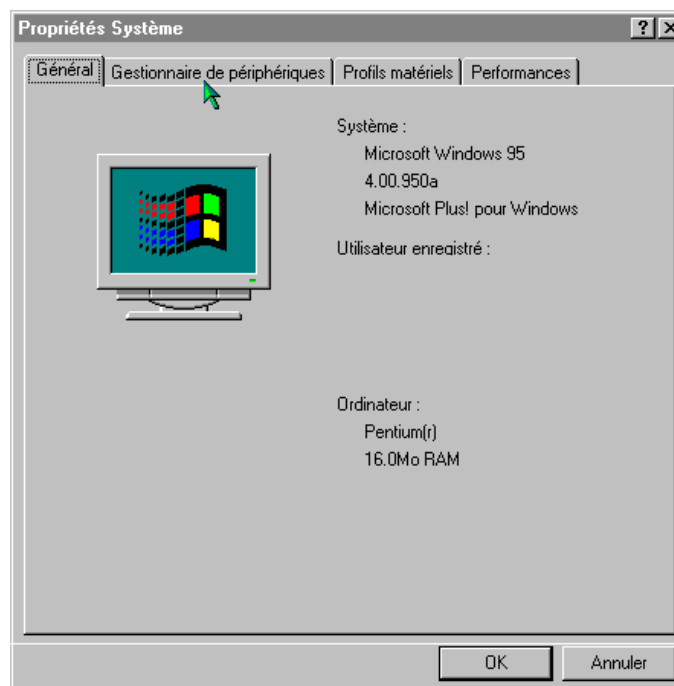
Si l'écran suivant n'apparaît pas après quelques secondes, l'énumération des ports de la carte aura lieu au prochain lancement du système :



Dans le panneau de configuration, lancez 'Système' :



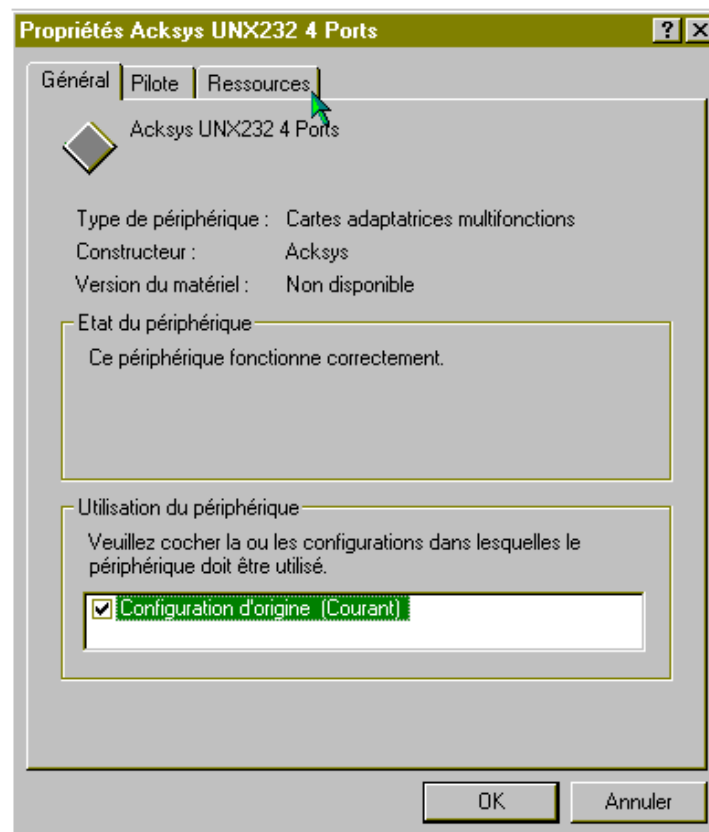
Sélectionnez 'Gestionnaire de périphériques' :

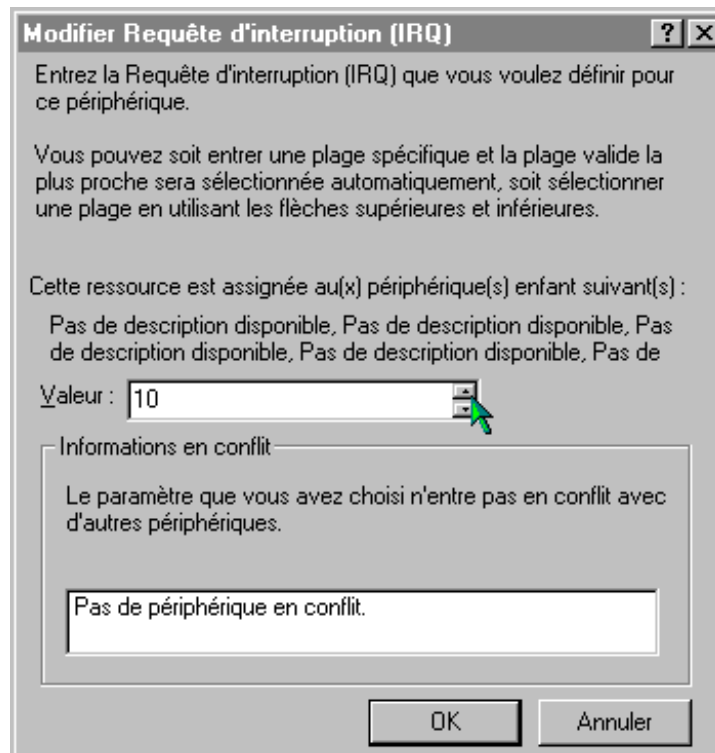


Sélectionnez votre carte dans 'Cartes adaptatrices multifonctions', puis cliquez sur 'Propriétés' :

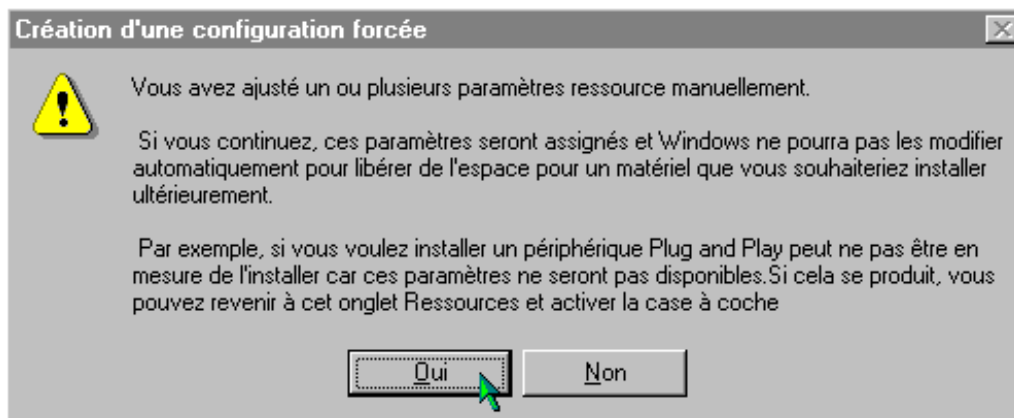


Selectionnez l'onglet 'Ressources' :

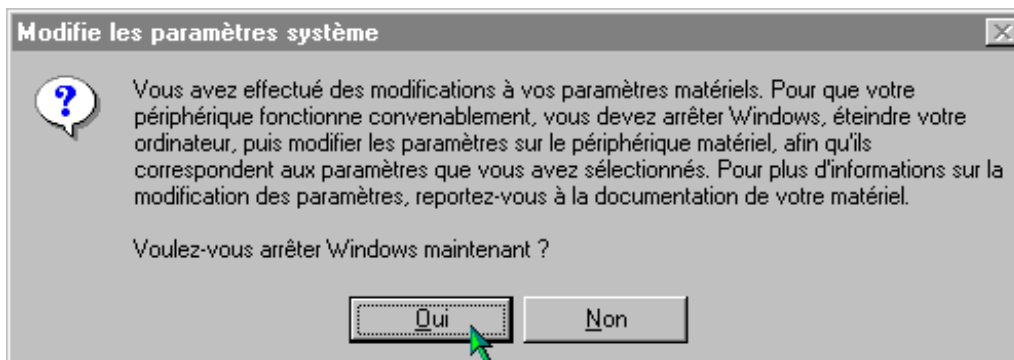




Lorsque tous les paramètres sont correctement définis, cliquez sur 'OK' puis sur 'Oui' pour accepter la configuration forcée:



Cliquez sur 'Oui' pour relancer le système afin de prendre en compte les modifications :



XIV.2 CARTES 2 VOIES

Pour chaque voie de votre carte, suivez la procédure standard d'installation de port série :

Dans le panneau de configuration, lancez 'Ajout de périphériques'. Dans l'assistant d'ajout de nouveau matériel, cliquez sur 'non' pour ne pas effectuer de détection automatique, puis sélectionnez 'Ports (COM & LPT)'. Sélectionnez '(Types de ports standard)' puis 'Port de communication' et terminez l'installation sans redémarrer le PC.

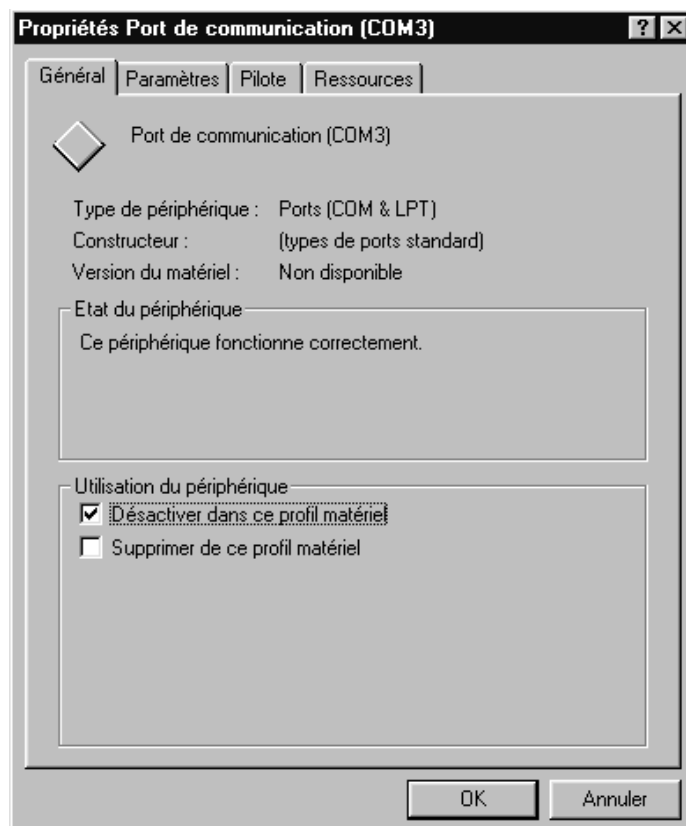
Lorsque les deux voies ont été installées, lancez 'Système' depuis le panneau de configuration, sélectionnez l'onglet 'Gestionnaire de périphériques' puis 'Ports (COM & LPT)'. Editez les propriétés des deux nouveaux ports et sélectionnez l'onglet 'Ressources'. Pour modifier la plage d'entrée/sortie et l'IRQ, vous devez sélectionner la configuration de base 8. Pour terminer l'installation, redémarrez le PC.

XIV.3 PROBLEMES D'INSTALLATION

Certaines versions de Windows 95 (OSR2) peuvent poser un problème lors de l'installation d'un port série configuré à une adresse conventionnelle (3F8h, 2F8h, 3E8h ou 2E8h) avec un niveau d'IRQ non standard : au lancement de Windows, l'énumérateur détecte automatiquement le nouveau port et l'installe systématiquement dans une configuration n'autorisant pas la changement d'IRQ (la configuration de base 8 n'existe pas). Pour pallier à ce problème, lancez 'Système' depuis le panneau de configuration, sélectionnez l'onglet 'Gestionnaire de périphériques' puis 'Ports (COM & LPT)'.

Editez les propriétés du port concerné et, dans l'onglet 'Général', cochez la case 'Désactiver dans ce profil matériel'. Le port apparaîtra alors barré d'une croix rouge dans l'onglet 'Gestionnaire de périphériques'.

Redémarrez le PC et suivez la procédure d'installation normale décrite ci-dessus.



XIV.4 SERVICES FICHIERS APPLICABLES AUX PORTS SERIE

Ces services sont des fonctions standard de Windows 95, ils sont documentés en détail dans les manuels et l'aide en ligne des outils de développement de ce système (dans le chapitre « Win32 reference »)

CreateFile
ReadFile
WriteFile
CloseHandle
FlushFileBuffers

XIV.5 SERVICES DE COMMUNICATION

Ces services sont des fonctions standard de Windows 95, ils sont documentés en détail dans les manuels et l'aide en ligne des outils de développement de ce système (dans le chapitre « Win32 reference »)

BuildCommDCB
BuildCommDCBAndTimeouts
ClearCommBreak
ClearCommError
CommConfigDialog
DeviceloControl
EscapeCommFunction
GetCommConfig
GetCommMask
GetCommModemStatus
GetCommProperties
GetCommState
GetCommTimeouts
GetDefaultCommConfig
PurgeComm
SetCommBreak
SetCommConfig
SetCommMask
SetCommState
SetCommTimeouts
SetDefaultCommConfig
SetupComm
TransmitCommChar
WaitCommEvent

XV. PILOTE POUR OS/2

XV.1 CARACTERISTIQUES GENERALES

Systemes supportés :

OS/2 2.0, OS/2 2.1, WARP

Cartes ACKSYS supportées :

- UNX232 4 ou 8 voies (adresse et interruption paramétrables)
- 8RS232 (adresse et interruption paramétrables)
- 4RS232 en 280.

Extensions futures (nous consulter) :

UNX/II (bus MCA), FAST400 et COMFAST (Supportées en standard par OS/2 en mode COM1/COM2, mais pas en mode COM3/COM4).

Particularités :

- Le FIFO (tampon matériel) n'est pas supporté dans cette version
- Support de l'API 16 et 32 bits pour ports série d'OS/2 : voir page XV-2
- Sous-système DOS : mêmes fonctions que l'API OS/2 lorsqu'elles sont accessibles par l'INT 21. Pas d'évolution prévue
- Gestion d'un buffer interne au driver (1ko en réception, 128o en émission, pour chaque voie)

Performances :

Les tests suivants ont été réalisés sur un 386 DX40 sous OS/2 2.0, carte 4 voies, émission en 8 bits, sans parité, 1 bit d'arrêt, pas de réception, le système étant inactif par ailleurs. Dans ces conditions le driver supporte :

- 4 voies à 19200 bauds ou 2 voies à 38400 bauds.
- Une voie à 57600 ou 115200 bauds avec une forte charge du système et un débit de 95% ou 88% (respectivement) du débit nominal.

XV.2 INSTALLATION

- 1) recopier le fichier **SRLDRV.SYS** fourni sur la disquette, sur le disque dur (par exemple dans **C:\OS2**)
- 2) ajouter dans le fichier **C:\CONFIG.SYS** une ligne selon le modèle suivant:

```
DEVICE={PATH}\SRLDRV.SYS -b{BASE} -i{INTERRUPT} -n{NOM} -s{INDICE}
```

où :

{PATH}	Emplacement où a été copié le driver.
{BASE}	Adresse de base de la carte (syntaxe du langage C), par défaut : 0x280
{INTERRUPT}	Interruption engendrée par la carte, (syntaxe C), par défaut : 10
{NOM}	Nom du pseudo-fichier associé à chaque voie, par défaut : COM
{INDICE}	Numéro qui, ajouté au nom, donne le nom de la première voie, par défaut : 3

Attention, il faut contrôler la concordance entre les indications fournies dans **CONFIG.SYS** et les interrupteurs de la carte (configuration usine de l'UNIX232 : adresse 0x280, IRQ 3).

Exemple de ligne dans **CONFIG.SYS** :

```
DEVICE=C:\OS2\SRLDRV.SYS -b0x280 -i10 -nCOM -s3
```

Si vous utilisez le nom **COM**, il est possible d'utiliser l'utilitaire standard **MODE.EXE** sur tous les noms compris entre **COM1** et **COM9**. Dans l'exemple ci-dessus, les noms des voies sont **COM3** pour la voie 1 de la carte **ACKSYS**, **COM4** pour la voie 2... **COM10** pour la voie 8 (cette voie n'est donc pas accessible pour **MODE.EXE**).

Contenu de la disquette, répertoire **OS2** :

- 1) répertoire **\driver** :

srldrv.sys	pilote de périphérique (driver).
config.sys	exemple de ligne à ajouter dans config.sys .

- 2) répertoire **\utils** :

srlmode.exe	programme de mise en oeuvre de DosDevIOctl pour un port série.
srlmode.doc	liste des mots-clé reconnus par srlmode.exe .
modetest.cmd	exemple d'utilisation de srlmode.exe
srltest.exe	programme de test des ports série.

- 3) répertoire **\exemples** :

srlmode.c	mise en oeuvre de DosDevIOctl , syntaxes 16 bits (MSC 6.0) et 32 bits.
srltest.c	sources du programme de test.

XV.3 SUPPORT DE L'API OS/2

Strategy Commands	Support
INIT	Oui
READ	Oui
NONDESTRUCTIVE READ	Oui ^(*)
INPUT STATUS	Oui ^(*)
INPUT FLUSH	Oui
WRITE	Oui ^(**)
WRITE VERIFY	Oui ^(**)

Strategy Commands	support
OUTPUT STATUS	Oui ^(*)
OUTPUT FLUSH	Oui
OPEN	Oui
CLOSE	Oui
DEINSTALL	Non
SHUTDOWN	Non

^(*)Ces fonctions, non testées, sont inaccessibles par les applications.

^(**)Les écritures multiples simultanées par plusieurs threads sur la même voie ne sont pas supportées.

Fonctions de DosDevIOctl :

code	GENIOCTL catégorie 1	support/version
41	Set rate	Oui
42	Set Line	Oui
43	Set Extended Rate	Oui ^(*)
44	Transmit Immediate	Oui
45	Break Off	Oui
46	Modem Control	Oui
47	Stop Transmit	Non (V3)
48	Start Transmit	Non (V3)
4B	Break On	Oui
53	Set DCB parameters	Voir plus bas
54	Set Enhanced	Non (V2,3)
61	Query rate	Oui

code	GENIOCTL catégorie 1	support/version
62	Query line	Oui
63	Query Extended rate	Oui
64	COM status	Oui
65	Query Tx Status	Oui
66	Query mdm output	Oui
67	Query mdm inputs	Oui
68	Query Rx char count	Oui
69	Query Tx char count	Oui
6D	Query error word	Oui
72	Query event info	Oui
73	Query DCB parameters	Oui
74	Query enhanced	Oui

^(*)Les fractions ne sont pas prises en compte dans la fonction 43 - Set Extended Baud Rate.

éléments du DCB	version driver
Write Timeout	Oui
Read Timeout	Oui
DTR modes	Oui
RTS modes	Oui
CTS handshake	Oui
DSR handshake	Oui
DCD handshake	Oui
DSR sensitivity	Oui
XON control for Tx buffer	Oui
XON control for Rx buffer	Oui

éléments du DCB	version driver
Error Replacement Char	Non(V2)
Nul Stripping	Non(V2)
Break Replacement Char	Non(V2)
Normal/Full duplex	Oui
Infinite Write Timeout	Oui
Read Modes	Oui
Enhanced modes	Non (V2,3)

Remarque : Les fonctions non supportées le seront dans la version indiquée entre parenthèses.

XVI. PILOTE POUR 386/IX ET UNIX SCO

XVI.1 CARTES SUPPORTEES ET CONFIGURATION

Ce pilote supporte les cartes UNX232, 8RS232 et 4RS232. Les cartes Comfast, FAST400 et 2RS232 peuvent être utilisées directement sous Unix SCO avec le driver 'sio' du système.

Les ports de communication de la carte doivent être à des adresses d'entrée/sortie consécutives, à partir de l'adresse 280h. Tous les ports sont sur le même niveau d'interruption (IRQ).

Avertissement

L'utilisation de certaines des extensions décrites peut dans certains cas, entrer en conflit avec le fonctionnement normal du driver et doit donc être réservée aux personnes expérimentées dans le domaine des drivers UNIX. Sous Unix SCO, la carte UNX232 peut être utilisée directement avec le driver 'sio' fourni par SCO, sous réserve d'adapter le fichier "/etc/conf/pack.d/sio/space.c". Dans ce cas, la présente documentation est sans objet.

XVI.2 INSTALLATION

Sous 386/IX sysVrel3.2 v2, utiliser la commande standard

```
# sysadm installpkg
```

Sous Unix SCO V.3 v2.0 et v4.2, utiliser les commandes suivantes (avec une disquette 3½ pouces) :

```
# mount /dev/fd0135ds18 /install -r
# /install/install/INSTALL
```

et après installation :

```
# umount /install
```

Le seul paramètre modifiable est le vecteur d'interruption. La valeur standard 3 entre en conflit avec le port COM2. Si vous avez un port COM2, utilisez une autre valeur (10 et 15 sont en général disponibles, sauf sur certains PC, en particulier ceux de la marque Compaq®). Attention, sur la carte UNX232, l'interrupteur **sw1-1** doit être **OFF** (pas de registre de masque).

XVI.3 UTILISATION

```
#include <unx232.h>
int ioctl (fd, opération, pvaleur)
int fd;          /* n° du fichier ouvert par la fonction open(2) */
int opération;   /* voir ci-dessous */
void *pvaleur;   /* voir ci-dessous */
```

La structure *UnxIesParameters_t* est définie de la façon suivante :

```
typedef struct {
    unsigned short mc_msec;    /* délai max micro-coupure          */
    unsigned short mf_msec;    /* délai max micro-fermeture        */
    unsigned char  char_on;    /* car. indiquant arrivée sig.      */
    unsigned char  char_off;   /* car. indiquant perte signal      */
    unsigned char  flags;      /* UNX_IES_CHAR_ON ou UNX_IES_CHAR_OFF */
    unsigned char  reserved;   /* réservé                          */
}UnxSigDesc_t;

typedef struct {
    unsigned char  version;     /* toujours zéro                    */
    unsigned char  état;       /* état interne au driver            */
    UnxSigDesc_t  sig[UNX_IES_NB_SIGS]; /* options de traitement          */
}UnxIesParameters_t;
```

Dans cette structure, les options de traitement des signaux sont également accessibles grâce aux macros UNX_SIG_CTS, UNX_SIG_DSR, UNX_SIG_DCD, UNX_SIG_RI

Par exemple :

```
UnxIesParameters_t ies_param;

ies_param.UNX_SIG_DSR.mc_msec = 100;
ies_param.UNX_SIG_DSR.flags = 0;

/* puis initialisation des autres signaux... */

ioctl(fd, TIOC_UNX_IES, &ies_param);
```

opération	pvaleur	description
TIOC_UNX_ECL	unsigned char *	Les bits 0 à 6 du caractère pointé par <i>pvaleur</i> sont écrits dans le registre Contrôle Ligne . Le bit 7 de ce registre est mis à zéro.
TIOC_UNX_ECM	unsigned char *	Le caractère pointé par <i>pvaleur</i> est écrit dans le registre de Contrôle MODEM .
TIOC_UNX_LEM	unsigned char *	Le registre d' Etat MODEM est lu et recopié à l'emplacement pointé par <i>pvaleur</i> .
TIOC_UNX_IES	UnxlesParameters_t *	Les paramètres de gestion des micro coupures et de détection d'événements sont installés dans le driver. L'état lisible par TIOC_UNX_LES est initialisé en fonction du contenu du registre d' Etat MODEM . Ces paramètres sont pris en compte jusqu'à la prochaine ouverture du port par la fonction open(2), lors de laquelle ils sont remis à zéro. Les caractères de détection d'événements sont décrits au paragraphe ' Détection par génération de caractères '.
TIOC_UNX_LES	unsigned char *	L'état "retardé" des signaux MODEM est copié à l'emplacement pointé par <i>pvaleur</i> . Un bit à 1 indique un signal ACTIF (circuit FERME). *pvaleur & UNIX_BIT_CTS indique l'état retardé de CTS (106) *pvaleur & UNIX_BIT_DSR indique l'état retardé de DSR (107) *pvaleur & UNIX_BIT_DCD indique l'état retardé de DCD (109) *pvaleur & UNIX_BIT_RI indique l'état retardé de RI (125) Définition de l'état retardé : - L'état retardé est initialisé par l'ioctl TIOC_UNX_IES. - L'état retardé est la simple transcription du registre d'état MODEM si aucune gestion de retards n'a été demandée par l'ioctl TIOC_UNX_IES. - Sinon, l'état retardé passe à 1, <i>mf_msec</i> millisecondes après que que le circuit correspondant soit passé à l'état fermé, à condition qu'il ne soit pas repassé à l'état ouvert entre-temps. L'état retardé passe à 0, <i>mc_msec</i> millisecondes après que que le circuit correspondant soit passé à l'état ouvert, à condition qu'il ne soit pas repassé à l'état fermé entre-temps. - La précision sur <i>mf_msec</i> et <i>mc_msec</i> ne peut dépasser celle de l'horloge système et dépend de la non-préemptivité du noyau UNIX.
TIOC_UNX_TXE	NULL	Si le tampon de transmission est vide, l'ioctl se termine immédiatement. Sinon, L'ioctl reste bloquant, sans consommer de temps d'UC, jusqu'à ce qu'un signal soit reçu ou que le tampon devienne vide. Un dernier caractère peut être alors en cours de transmission (dans le registre de sérialisation du composant série).
TIOC_UNX_ABS	int	Si la condition exprimée par les bits de <i>pvaleur</i> est réalisée, l'ioctl se termine immédiatement. Sinon, L'ioctl reste bloquant, sans consommer de temps d'UC, jusqu'à l'un des événements suivants : - La condition se réalise. - Un signal est reçu par le process en attente. La condition est un OU booléen d'un <i>mode</i> et d'un ou plusieurs <i>signaux modem</i> . Le mode peut être : UNIX_ABS_TOUS_ON : attendre que tous les circuits indiqués soient fermés. UNIX_ABS_TOUS_OFF : attendre que tous les circuits indiqués soient ouverts. UNIX_ABS_UN_ON : attendre qu'au moins l'un des circuits indiqués soit fermé. UNIX_ABS_UN_ON : attendre qu'au moins l'un des circuits indiqués soit ouvert. Les signaux modem sont les codes indiqués pour TIOC_UNX_LES. Exemple : la commande <pre>ioctl(fd, TIOC_UNX_ABS, UNIX_ABS_TOUS_ON UNIX_BIT_DSR UNIX_BIT_DCD);</pre> attend que les signaux retardés DSR et DCD soient simultanément FERMES.

XVI.3.1 Détection par génération de caractères

La structure `UnxIesParameters_t` passée à l'ioctl `TIOC_UNX_IES`, contient une sous-structure de description par signal d'entrée du port série :

`UNX_SIG_CTS`, `UNX_SIG_DSR`, `UNX_SIG_DCD`, `UNX_SIG_RI`.

Si le champ **flags** contient le bit `UNX_IES_CHAR_ON`, alors le caractère contenu dans **char_on** sera placé dans le tampon de réception à chaque fois que le circuit correspondant passe à l'état FERME.

Si le champ **flags** contient le bit `UNX_IES_CHAR_OFF`, alors le caractère contenu dans **char_off** sera placé dans le tampon de réception à chaque fois que le circuit correspondant passe à l'état OUVERT.

XVI.3.2 Influence de l'environnement d'utilisation ; Précautions d'usage

a. Précautions d'usage.

Le registre **Contrôle MODEM** est modifié par le driver dans certaines conditions, en particulier lors de l'appel des fonctions **open(...)**, **ioctl(..., TCSET..., ...)** modifiant **t_cflag**, **ioctl(..., TCSBRK, ...)**, **close(...)**.

Bien que le bit **OUT2** du registre de Contrôle MODEM ne soit pas utilisé sur la carte UNX232, il est conseillé de le laisser à 1 pour des raisons de compatibilité avec d'autres cartes (en particulier la MCX/LITE/U qui utilise ce bit pour autoriser les interruptions).

b. Mode -CLOCAL.

Le mode `-CLOCAL` (voir `termio(7)`) force le driver à annuler le contenu de ses tampons lorsque le circuit 109 (DCD) passe à l'état FERME. De plus, si un process est attaché au port (voir `setpgrp(2)`), il recevra le signal `SIGHUP` si le circuit 109 passe à l'état OUVERT. Attention, le mode `-CLOCAL` est automatique si le bit 5 du 'minor device number' du port vaut 1. Ceci peut être obtenu par l'utilitaire **setline** fourni.

```
# setline -m adr01 #           ajoute le bit 5 (gestion -CLOCAL) au port /dev/adr01
# setline -d adr01 #           retranche le bit 5 (gestion CLOCAL par défaut)
```

Dans le cas du mode `-CLOCAL` automatique, un **open(...)** sur le port reste bloqué tant que le circuit 109 n'est pas fermé, sauf si le flag `O_NDELAY` est passé à la fonction `open`.

c. Signal RI.

Les fonctions décrites ci-dessus s'appliquent imparfaitement au circuit 125 (RING INDICATOR) car aucune interruption n'est engendrée par l'ouverture de ce circuit.

d. Modes XMT1EN/RCV1EN

Ces modes, nommés également `RTSFLOW/CTSFLOW` sous Unix SCO, ne sont pas supportés dans ce driver.

XVI.3.3 Exemple d'utilisation

Exemple complet et testé de programme utilisant les fonctions spéciales :

```

#include <stdio.h>
#include <termio.h>
#include "unx232.h"

main()
{
    int fd;                                /* file descriptor du device a tester */
    UnxIesParameters_t iesParms;          /* parametres d'initialisation */
    int sig;                               /* indice d'init de iesParms */
    unsigned short e;                     /* evenement recu du driver */
    unsigned short olde = 0377;           /* valeur a comparer */
    char *file = "/dev/adr00";            /* nom du device ("adr00" par ex) */
    struct termio tio;                     /* pour parametrage UNIX de la ligne */
    int cm;                                /* registre de controle modem */

    /* ----- initialisation ----- */
    if( (fd=open(file,2)) == -1 ){         /* 'file' devient controlling device */
        perror(file); exit(1);
    }
    if( ioctl(fd,TCGETA,&tio) == -1 ){
        perror("TCGETA"); exit(1);
    }
    tio.c_lflag &= ~ICANON;
    tio.c_cc[VMIN] = 1;
    tio.c_cc[VTIME] = 0;
    tio.c_lflag &= ~ISIG;                 /* optimise vitesse driver */
    tio.c_cflag &= ~CBAUD;
    tio.c_cflag &= B300;
    if( ioctl(fd,TCSETA,&tio) == -1 ){
        perror("TCSETA"); exit(1);
    }

    /****** init standard *****/
    iesParms.version = 0;
    for( sig = 0; sig < UNX_IES_NB_SIGS; sig++ ){
        iesParms.sig[sig].flags = 0;      /* char_off inutilise */
        iesParms.sig[sig].mc_msec = 0;    /* inutilise */
        iesParms.sig[sig].mf_msec = 0;    /* inutilise */
    }

    /***** init non standard *****/
    iesParms.UNX_SIG_DCD.char_on = '\373';
    iesParms.UNX_SIG_DCD.char_off = '\374';
    iesParms.UNX_SIG_DCD.flags = UNX_IES_CHAR_ON|UNX_IES_CHAR_OFF;
    iesParms.UNX_SIG_DCD.mc_msec = 0;
    iesParms.UNX_SIG_DCD.mf_msec = 0;

    iesParms.UNX_SIG_DSR.char_off = '\375';
    iesParms.UNX_SIG_DSR.flags = UNX_IES_CHAR_OFF;
    iesParms.UNX_SIG_DSR.mc_msec = 100;
    iesParms.UNX_SIG_DSR.mf_msec = 100;
}

```

```
if( ioctl(fd,TIOC_UNX_IES,&iesParms) == -1 ){
    perror("TIOC_UNX_IES"); exit(1);
}

/* ----- tests ----- */

cm = UART8250_DTR|UART8250_RTS;
for(;;){
    if( ioctl(fd,TIOC_UNX_LES,&e) == -1 ){
        perror("TIOC_UNX_LES"); exit(1);
    }
    if( e != olde ){
        olde = e;
        while( (sig=ioctl(fd,FIORDCHK)) != 0 ){
            read(fd,&sig,1);
            printf("car. lu = %02x\n",sig);
        }
    }
    write(fd,"test en cours a 300 bauds...\n",29);
    if( ioctl(fd,TIOC_UNX_TXE,0) == -1 ){
        perror("TIOC_UNX_TXE"); exit(1);
    }
    cm ^= UART8250_DTR;          /* inversion du DTR a chaque emission */
    if( ioctl(fd,TIOC_UNX_ECM,&cm) == -1 ){
        perror("TIOC_UNX_ECM"); exit(1);
    }
}
}
```

XVII. PILOTE POUR MS-DOS

XVII.1 INTRODUCTION

Les programmes de commande de périphérique, ou "device drivers", permettent d'intégrer au système d'exploitation, MS-DOS, les routines de contrôle des périphériques. Ils ont pour objet d'isoler les plus hauts niveaux du système d'exploitation des caractéristiques spécifiques des différents périphériques installés dans la machine.

Dans ce contexte, le pilote de périphérique SRLDRV.SYS est destiné au pilotage de toutes les cartes et extensions ACKSYS basées sur le composant de transmission série 16C550 ou l'un de ses équivalents. Ceci concerne les cartes passives pour PC et certaines extensions pour MCX-*Lite*/0, lorsqu'elles sont utilisées sous MCXDOS.

SRLDRV.SYS permet d'ouvrir en lecture et en écriture chacun des canaux de communication, comme s'il s'agissait de fichiers. De nombreux exemples de programmation en BASIC et en assembleur accompagnent la description des commandes du driver.

Chacun des canaux de communication est considéré par MS-DOS comme une entité unique et indépendante des autres canaux, portant un nom, unique également, permettant de l'identifier quel que soit le niveau de programmation. Ainsi, l'ouverture du fichier correspondant au premier canal s'écrira en BASIC:

```
OPEN "SRL1" FOR RANDOM AS 1
```

ou en C :

```
dh = open ("SRL1", O_RDWR | O_BINARY) ;
```

XVII.2 CONFIGURATION DES CARTES

Le programme de commande SRLDRV.SYS est capable de gérer jusqu'à quatre ports série situés à des adresses consécutives et sur le même niveau d'interruption. Les configurations d'adresse et d'IRQ possibles pour les différentes cartes sont données ci-dessous. Vous trouverez le positionnement correspondant des interrupteurs et cavaliers dans la section de la présente documentation traitant de votre carte.

XVII.2.1 La Carte 2RS232

SRLDRV.SYS supporte les deux configurations d'adresse de la carte 2RS232 :

- Adresses 3F8h, 2F8h (ST1 en position 1-2)
- Adresses 3E8h, 2E8h (ST1 en position 2-3)

- Toutes les IRQ sont supportées, mais les quatre ports doivent être placés sur le même niveau d'interruption (IRQ 3, 4, 5, 7, 9, 10, 11, 15).

XVII.2.2 La Carte 4RS232

SRLDRV.SYS supporte les deux configurations d'adresse de la carte 4RS232 :

- Adresses 3F8h, 2F8h, 3E8h, 2E8h (ST1 en position 1-2)
- Adresses 280h, 288h, 290h, 298h (ST1 en position 2-3)

- Toutes les IRQ sont supportées, mais les quatre ports doivent être placés sur le même niveau d'interruption (IRQ 3, 4, 5, 7, 9, 10, 11, 15).

XVII.2.3 La Carte 8RS232

Toutes les adresses et IRQ de la carte 8RS232 sont supportées mais les huit ports doivent être placés sur le même niveau d'interruption.

- Adresse base fixée par inter DIL SW1.
- IRQ 3, 4, 5, 7, 10, 11, 12, 15

XVII.2.4 La Carte UNX232

Comme pour la 8RS232, toutes les adresses et IRQ de la carte UNX232 sont supportées.

- Adresse base fixée par inter DIL SW2.
- IRQ 3, 4, 5, 7, 9, 10, 11, 12, 15

XVII.2.5 La Carte FAST400

Les deux ports de la carte FAST400 peuvent être configurés en COM1 & COM2 ou COM3 & COM4 mais les deux ports doivent être sur le même niveau d'interruption:

- Adresses 3F8h et 2F8h (com1 & com2 : sw1-1 sur OFF)
- Adresses 3E8h et 2E8h (com3 & com4 : sw1-1 sur ON)
- IRQ9, IRQ10, IRQ11 ou IRQ12 (respectivement sw1-5, 6, 7 ou 8)

XVII.2.6 La Carte COMFAST

SRLDRV.SYS supporte la carte COMFAST lorsque les deux ports sont sur IRQ5. Les deux configurations d'adresse sont supportées :

- Adresses 3F8h et 2F8h (com1 & com2 : sw1-1 sur OFF)
- Adresses 3E8h et 2E8h (com3 & com4 : sw1-1 sur ON)
- IRQ 5 sur les deux ports (sw1-2, sw1-3 sur OFF, sw1-4 sur ON)

XVII.2.7 L'extension *Lite-485* pour *MCX-Lite*

L'extension *Lite* 485 de la carte MCX peut être gérée par SRLDRV.SYS dans le cadre d'une utilisation sous MCXDOS® ou AUTOMCX® (le pilote doit alors être chargé sur la carte). SRLDRV.SYS supporte les deux configurations d'adresse de la carte *Lite* 485 :

- Adresses 3F8h et 2F8h (sw1-1 sur OFF)
- Adresses 280h, 288h (sw1-1 sur ON)
- Les deux ports sur IRQ5 (sw1-4 sur ON)

XVII.2.8 L'extension *Lite-UNIX* pour *MCX-Lite*

L'extension *Lite* UNIX de la carte MCX peut être gérée par SRLDRV.SYS dans le cadre d'une utilisation sous MCXDOS® ou AUTOMCX® (le pilote doit alors être chargé sur la carte). SRLDRV.SYS supporte toutes les IRQ de la carte *Lite* UNIX :

- Adresse 280h
- IRQ 3, 4, 5 ou 7 (commune à tous les ports, inter DIL SW1)
- Registre de masque activé (ST1 en position 1-2, option /p du programme)
- Registre de masque désactivé (ST1 en position 2-3)

XVII.2.9 L'extension *Lite-104* pour *MCX-Lite*

L'extension *Lite* 104 de la carte MCX peut être gérée par SRLDRV.SYS dans le cadre d'une utilisation sous MCXDOS® ou AUTOMCX® (le pilote doit alors être chargé sur la carte). SRLDRV.SYS supporte les deux configurations d'adresse de la carte *Lite* 104 :

- Adresses 3F8h, IRQ 4 (com1 : ST1 en position 1-2)
- Adresses 2F8h, IRQ 3 (com2 : ST1 en position 2-3)

XVII.3 INSTALLATION

Le programme de commande, ou 'device driver', des cartes ACKSYS multivoies, SRLDRV.SYS, est chargé automatiquement par le DOS au moyen d'une commande incluse dans le fichier descriptif du logiciel, CONFIG.SYS. Si ce fichier n'existe pas dans le répertoire principal de la disquette ou du disque à partir duquel le dos est lancé, il doit être créé (par exemple par une commande COPY CON CONFIG.SYS).

SRLDRV.SYS peut gérer jusqu'à six ports série situés à des adresses consécutives et sur le même niveau d'interruption. Il supporte également les adresses des ports COM standard du PC (3F8h, 2F8h, 3E8h, 2E8h).

REMARQUE : La gestion par SRLDRV.SYS d'un port situé à une adresse standard (COM1, COM2, COM3 ou COM4) peut perturber le fonctionnement d'un programme qui tenterait d'y accéder par d'autres moyens que les fonctions de l'interruption 14h. C'est le cas notamment des pilotes de souris .

Il est possible de gérer des ports configurés sur des niveaux d'IRQ différents en chargeant plusieurs fois le driver.

XVII.3.1 Paramètres de lancement

La syntaxe de la ligne de commande de lancement de SRLDRV.SYS est la suivante :

```
DEVICE = [path]\SRLDRV.SYS adr1, [adr2, adr3, adr4, adr5] irq, [rxsize, txsize] [/#dinop /cn /snn /rxxx]
```

path représente le chemin d'accès au programme (ex : C:\DRIVERS\)

adr1-adr5 Adresses des unités de communication (voir §XVII.2). Dans le cas des adresses non consécutives (3F8h, 2F8h, 3E8h, 2E8h), spécifiez l'adresse de chacun des ports. La première adresse différente de 3F8h, 2F8h, 3E8h ou 2E8h est considérée comme adresse de base: les ports suivants sont aux adresses consécutives par incréments de 8. Quelles que soit les adresses de la carte, tous les ports doivent impérativement être situés sur le même niveau d'IRQ. Les adresses peuvent être entrées sous forme décimale ou sous forme hexadécimale (dans ce cas suivie de 'h' . Ex : 280h).

irq Ligne d'IRQ utilisée (voir §XVII.2). Les valeurs admises sont : 3, 4, 5, 6, 7, 9, 10, 11, 12 et 15. Tous les ports de la carte doivent être sur le même niveau d'interruption.

rxsize Ce paramètre est optionnel. S'il est spécifié, il représente la taille des tampons de réception. La valeur peut être entrée sous forme décimale ou sous forme hexadécimale (suivie de 'h'). Elle doit être comprise entre 32 (20h) et 32768 (8000h). La taille par défaut des tampons de réception est de 256 octets (configuration standard).

txsize Représente la taille des tampons d'émission. Ce paramètre est optionnel également. La valeur peut être entrée sous forme décimale ou sous forme hexadécimale (suivie de 'h'). Elle doit être comprise entre 32 (20h) et 32768 (8000h). La taille par défaut des tampons d'émission est de 256 octets (configuration standard). L'espace mémoire occupé par le programme de commande de la carte ACKSYS est directement dépendant de la taille des tampons d'émission et de réception et est égal à :

$(\text{rxsize} + \text{txsize}) \times N + 8000$ Avec N = nombre de voies de la carte.

/# Avec # = 1 à 9. Indique le nombre de ports de la carte.

/d Lorsque cette option est spécifiée, les valeurs retournées par les commandes IOCTL en lecture sont sous forme décimale. Si /d n'est pas spécifié, les valeurs sont retournées sous forme hexadécimale.

/n ou /b Lorsque cette option est spécifiée, les valeurs retournées par les commandes IOCTL en lecture sont numériques. Si /n n'est pas spécifié, les valeurs sont retournées codées ASCII .

/i Cette option autorise l'installation du handler de l'interruption 60H.

/o Cette option désactive la détection automatique des 16C550.

/cn Cette option autorise l'installation du handler de l'interruption 14H. Le paramètre n indique le numéro de port attribué à la voie 1 de la carte ACKSYS. Par exemple, si le système dispose de deux ports standards com1 et com2, les numéro de port sont attribués comme suit :

com1= 0, com2= 1	(interruption 14 du BIOS)
SRL1= 2, SRL2= 3, SRL3= 4, SRL4= 5	(interruption 14 de SRLDRV)

Dans ce cas n vaut 2

/snn Cette option permet de changer le numéro initial de 'device', notamment dans le cas où plusieurs cartes sont installées. Par exemple, avec /s5 le nom des ports sera SRL5, SRL6, SRL7, SRL8.

Avec /s20, le nom des ports sera : SRL20, SRL21, SRL22, SRL23.

/rxxx L'option /r permet de renommer les 'device'. Le nom par défaut est SRL#. Cette option permet d'assurer la compatibilité avec le pilote UNXDRV. Attention, si vous utilisez un nom de device système, tel que COM, ce dernier ne sera plus accessible.

/p Support du registre de polling pour les cartes UNX232 et 8RS232.

Exemple 1 :

Carte quatre ports installée à l'adresse 280h, utilisant IRQ5, tampons de réception de 512 octets, tampons d'émission de 1024 octets, retour des valeurs IOCTL en décimal ASCII, installation du handler de l'int 14H :

```
DEVICE = C:\DRIVERS\SRLDRV.SYS 280h, 5, 512, 1024 /nc2
```

ou encore :

```
DEVICE = C:\DRIVERS\SRLDRV.SYS 280h,5,200h,400h /n/c2
```

Exemple 2 :

Une carte 4RS232 configurée en 3F8h, 2F8h, 3E8h, 2E8h, IRQ 10 et une carte 4RS232 à l'adresse 280h, IRQ12, tailles des tampons par défaut :

```
DEVICE = SRLDRV.SYS 3F8h,2F8h,3E8h,2E8h, 10 /4      (1ère carte)
```

```
DEVICE = SRLDRV.SYS 280h, 12 /4s9                    (2ème carte)
```

REMARQUES:

- *Sous DOS 5, il est possible de charger le driver en mémoire haute. Dans ce cas, remplacez 'DEVICE =' par 'DEVICEHIGH ='. Attention: si la mémoire disponible dans les UMB est insuffisante, le driver affiche un message d'erreur et ne se charge pas. Dans ce cas, diminuez la taille des tampons à l'aide de SRLSET.EXE ou chargez le driver en mémoire conventionnelle.*
- *Après insertion de la ligne de commande dans le fichier CONFIG.SYS, une séquence de reset de la machine (Ctrl-alt-del) est nécessaire pour installer le programme de commande.*

XVII.3.2 Compatibilité avec UNXDRV.SYS

Toutes les options du pilote UNXDRV.SYS sont reconnues par SRLDRV.SYS. Pour remplacer le premier par le second, il suffit donc d'ajouter les paramètres /rUNIX /p.

Exemple :

```
DEVICE=UNXDRV.SYS 300h,11 /s9
```

Peut être remplacé par :

```
DEVICE=SRLDRV.SYS 300h,11 /s9 /rUNIX/p
```

Pour obtenir un fonctionnement identique.

XVII.3.3 Initialisation du programme de commande

L'initialisation de SRLDRV.SYS est exécutée lors du chargement du DOS, immédiatement avant le chargement de l'interpréteur de commandes (COMMAND.COM). Les informations suivantes sont alors affichées:

Le numéro de révision du programme de commande
Le nom de la configuration (16 caractères)
Le nom de chacun des canaux installés (SRL1 à SRL4)

Si une erreur se produit pendant l'installation, un message d'erreur indique son origine et éventuellement le numéro du canal concerné. S'il s'agit d'une erreur concernant le niveau d'IRQ ou l'adresse de la carte, le driver ne reste pas résident en mémoire. Dans ce cas, aucune des voies de la carte n'est utilisable.

Le nom de la configuration permet d'identifier les configurations particulières effectuées à l'aide du programme SRLSET.EXE. La configuration STANDARD initiale est la suivante :

- 9600 bits/seconde
- 8 bits/caractères
- 2 bits de stop
- pas de parité
- pas de handshake
- Tampons de réception = 256 octets par voie
- Tampons d'émission = 256 octets par voie

Après installation du programme de commande, la carte ACKSYS est opérationnelle et peut être immédiatement utilisée, par exemple depuis le processeur de commandes avec des commandes telles que :

DIR > SRL1

ou COPY FICHER.TXT SRL2

XVII.4 UTILISATION DU PROGRAMME DE COMMANDE

XVII.4.1 Depuis le processeur de commandes

Le processeur de commandes ('SHELL') permet d'effectuer les opérations élémentaires d'entrée et de sortie de caractères. Il n'est pas possible de modifier les paramètres de transmission directement à ce niveau. Le transfert d'informations entre la carte et le système est réalisé caractère par caractère. Les commandes supportées sont COPY, CTTY, PRINT ainsi que les redirections d'entrées/sorties (< et >).

Exemples :

Envoi du fichier ACKSYS.TXT sur SRL1 :

```
COPY ACKSYS.TXT SRL1
```

Equivalent à :

```
TYPE ACKSYS.TXT > SRL1
```

Ou encore :

```
PRINT /d:SRL1 /b:128 ACKSYS.TXT
```

Il est possible d'envoyer des commandes IOCTL au driver depuis le processeur de commandes à l'aide du programme COMCTL.COM. Par exemple, pour modifier les paramètres de transmission :

```
COMCTL 1, vinit: 19200, 8, 1, np
```

Le programme COMCTL.COM est décrit au chapitre XVII.6.

XVII.4.2 Utilisation depuis l'assembleur

SRLDRV.SYS permet de piloter les ports série des cartes ACKSYS depuis l'assembleur par des appels au système d'exploitation (system calls) du type INT 21h. Avant de pouvoir effectuer des transferts d'informations entre l'application et un canal donné, celui-ci doit tout d'abord être ouvert par une fonction OPEN. Un code de gestion ('handle') de 16 bits est alors associé à ce canal et devra être utilisé pour toute opération ultérieure de lecture ou d'écriture. Les appels de fonction suivants sont supportés par le programme de commande des cartes ACKSYS multivoies:

Fonction 3Dh - Ouverture de canal

Ouvre le canal spécifié par la chaîne ASCII¹ donnée. Retourne le code de gestion (« handle ») utilisé par le programme pour les opérations ultérieures sur le canal.

Appel avec:

AH	=	3Dh
AL	=	Mode d'accès : 00 = accès en lecture uniquement 01 = accès en écriture uniquement 02 = accès en lecture et écriture
DS:DX	=	Segment:offset du nom de canal ASCIIZ

Retourne

	Si ouverture réussie
Carry	= 0
AX	= Code de gestion (handle)
	Si non
Carry	= 1
AX	= Code d'erreur

Fonction 3Eh - Fermeture de canal

Ferme le canal spécifié par le code de gestion obtenu par la fonction d'ouverture (3Dh).

Appel avec:

AH	=	3Eh
BX	=	Code de gestion

Retourne

	Si fermeture réussie
Carry	= 0
	Si non
Carry	= 1
AX	= Code d'erreur

¹ ASCIIZ : Chaîne de caractères ASCII terminée par un zéro (0h).

Fonction 3Fh - Lecture du canal

Transfère les données depuis le tampon de réception du canal associé au code de gestion donné, vers le tampon spécifié. Si le nombre de caractères demandé n'est pas reçu dans les 10 secondes (voir commande ioctl TIMEO) suivant l'appel de la fonction, le nombre de caractères réellement transférés est retourné dans AX, mais le flag Carry n'est pas positionné (en d'autres termes, la sortie sur 'Time Out' de la fonction de lecture n'est pas considéré comme une erreur, même si aucun caractère n'a été reçu. Il revient au programmeur de vérifier le nombre de caractères effectivement lus).

Appel avec:

AH	=	3Fh
BX	=	Code de gestion
CX	=	Nombre de caractères à lire
DS:DX	=	Segment:offset du tampon destination

Retourne

Si lecture réussie	
Carry	= 0
AX	= Nombre de caractères lus
Si non	
Carry	= 1
AX	= Code d'erreur

Fonction 40h - Ecriture sur un canal

Transfère les caractères spécifiés vers le tampon de transmission du canal donné, puis lance l'émission. Si le tampon de transmission ne peut recevoir le nombre de caractère spécifié dans les 10 secondes (voir commande ioctl TIMEO) suivant l'envoi de la commande, le nombre de caractères effectivement transférés est retourné dans AX.

Appel avec:

AH	=	40h
BX	=	Code de gestion
CX	=	Nombre de caractères à écrire
DS:DX	=	Segment:offset du tampon à transférer

Retourne

Si écriture réussie	
Carry	= 0
AX	= Nombre de caractères transférés
Si non	
Carry	= 1
AX	= Code d'erreur

Sous-fonction 03h - Envoi d'une commande IOCTL au driver

Permet d'envoyer une des commandes IOCTL dont la description est donnée au chapitre XVII.5.2. Les commandes sont des chaînes de caractères ASCII.

Appel avec:

AH	=	44h
AL	=	03h
BX	=	Code de gestion obtenu par la fonction 3Dh
CX	=	Longueur de la commande (nombre de caractères)
DS:DX	=	Segment:offset de la chaîne de commande

Retourne

Si fonction réussie	
Carry	= 0
AX	= Nombre d'octets transférés
Si non	
Carry	= 1
AX	= Code d'erreur

Sous-fonction 02h - Lecture de la réponse à une commande IOCTL

Permet d'obtenir le code de retour et éventuellement des informations à la suite de l'envoi d'une commande IOCTL (sous-fonction 03). Le nombre de caractères à lire doit être fixé à 20 dans CX. DS:DX doit pointer sur un tampon de 20 caractères destiné à recevoir la chaîne de caractères.

Appel avec:

AH	=	44h
AL	=	02h
BX	=	Code de gestion obtenu par la fonction 3Dh
CX	=	Nombre de caractères à lire
DS:DX	=	Segment:offset du tampon

Retourne

Si fonction réussie	
Carry	= 0
AX	= Nombre d'octets transférés dans le tampon
Si non	
Carry	= 1
AX	= Code d'erreur

Exemple :

Initialisation du canal 2 à 1200 Bauds, transmission d'un bloc de données puis réception d'un bloc après vidage du tampon. Le passage en mode 'RAW' supprime le filtrage des caractères par le DOS et autorise les transmissions par blocs de caractères.

```

      .
      .
      ; OUVERTURE DU CANAL SRL2
      mov  ax,3D02h      ; Ouverture en lecture/écriture
      mov  dx,offset CHANNEL ; Pointe sur le nom du canal
      mov  ds,DSEG      ; DS sur segment de données
      int  21h          ; Appel de fonction (system call)
      jc   open_fail    ; Saut si erreur à l'ouverture
      mov  bx,ax        ; BX = Code de gestion ('handle')

      ; PASSAGE EN MODE 'RAW'
      mov  ax,4400h     ; Fonction 44h, subfonction 0
      int  21h          ; 'get device informations'
      mov  dh,0         ; Raz dh
      or   dl,20h       ; Passe en mode raw
      mov  ax,4401h     ; Fonction 44h, subfonction 1
      int  21h          ; 'Set device informations'

      ; INIT PARAMETRES DE TRANSMISSION
      mov  si,offset STRINIT ; Pointe sur chaîne d'initialisation
      call ioctl_write     ; Sous-programme I/O CONTROL
      jc   ioctl_fail     ; Saut si erreur

      ; TRANSMISSION D'UN BLOC DE 128 CAR
      mov  dx,offset TXBLOC ; Pointe sur bloc à transmettre
      mov  cx,80h          ; Longueur = 128 octets
      mov  ah,40h         ; Fonction écriture sur canal
      int  21h            ; Appel de fonction DOS
      jc   write_fail     ; Saut si erreur à l'écriture

      ; VIDAGE DU BUFFER DE RECEPTION
      mov  si,offset STRFLUSH ; Pointe sur chaîne de commande
      call ioctl_write     ; Appel du sous-programme
      jc   ioctl_fail     ; Saut si erreur

      ; RECEPTION D'UN BLOC DE 128 CAR
      mov  dx,offset RXBLOC ; Pointe sur destination bloc Reçu
      mov  cx,80h          ; Lecture 128 caractères
      mov  ah,3Fh         ; Fonction lecture du canal
      int  21h            ; Appel de fonction DOS
      jc   read_fail     ; Saut si erreur à la lecture

```

```

; SOUS-PROGRAMME I/O CONTROL
; Transmet la commande pointée par DS:SI et teste le code de retour.
; Positionne CF si erreur.
;
;
ioctl_write proc near
    mov     cx,[si]           ; Nombre de caractères dans CX
    mov     dx,si            ;
    add     dx,2             ; dx pointe sur début de la chaîne
    mov     ax,4403h         ; Code Ecriture IOCTL
    int     21h             ; Appel de fonction DOS
    jc     ioexit           ; Saut si erreur
    mov     ax,4402h         ; Code Lecture IOCTL
    mov     dx,offset RETCODE ; Destination du code de retour
    mov     cx,16           ; Nombre de caractères (maximum)
    int     21h             ; Appel dos
    jc     ioctlexit        ; Saut si erreur
    cmp     [RETCODE], '00'  ; Test code de retour = 0
    jz     ioctlexit        ; Sortie si oui
    stc                     ; Si non, positionne CF (erreur)
ioexit: ret                 ; Retour au programme principal
ioctl_write endp

    .
    .

; ZONE DE DONNEES
;
DSEG segment para public 'DATA'
;
CHANNEL    db     'SRL2',0   ; Nom du canal format ASCIIZ
STRINIT    dw     16         ; Longueur de la chaîne
           db     'vin: 1200,7,1,odd' ; VINIT: 1200 Bauds,7 bits,1 stop, impaire
STRFLUSH   dw     03         ; Longueur de la chaîne
           db     'clr'      ; CLRRX, 2 caractères
RETCODE    dw     8 dup ?    ; Code de retour et réponse
TXBLOC     db     16 dup ' ACKSYS ' ; Bloc à transmettre
RXBLOC     dw     64 dup ?   ; Reserve 128 octets pour RX

DSEG ends

```

XVII.4.3 Utilisation depuis le basic

En basic, les canaux de communication des cartes ACKSYS sont ouverts comme des fichiers. La plupart des instructions valides pour les fichiers sur disque le sont également pour les canaux des cartes ACKSYS.

L'ouverture d'un canal est réalisée par l'instruction OPEN. Le mode d'entrée et de sortie peut être indifféremment séquentiel ou direct.

Les entrées et sorties de caractères en mode séquentiel sont réalisées par les instructions suivantes :

```
PRINT #      PRINT # USING      WRITE #  
INPUT #      LINE INPUT #      INPUT$
```

En mode direct, utilisez les instructions PUT et GET.

Le transfert de chaînes de données de contrôle, permettant le traitement des commandes détaillées au chapitre XVII.5.2, est réalisé par l'instruction IOCTL en écriture (du programme vers le driver) et par la fonction IOCTL\$ en lecture (du driver vers le programme).

La commande d'envoi d'une chaîne de contrôle s'écrit sous la forme :

```
IOCTL # N,"STRING"
```

N représente le numéro spécifié dans la commande OPEN et STRING la chaîne de caractères de la commande.

La lecture de la réponse à une commande IOCTL est réalisée en BASIC par la fonction IOCTL\$. Le nombre de caractères renvoyés par le driver varie en fonction de la commande et du format de retour (options /d et /n de la ligne d'invocation du driver). Sauf indication particulière dans la description des commandes IOCTL, le driver renvoie un octet indiquant si la commande a été comprise (soit deux caractères en hexadécimal ASCII). La longueur maximale de la chaîne retournée est de 16 caractères.

Exemple 1

Dans l'exemple suivant, la commande CNTRX permet d'obtenir le nombre de caractères présents dans le tampon de réception du canal SRL5. Le nombre de caractères reçus par SRL5 est affecté à la variable NBCAR.

```
' Ouverture du canal  
  OPEN "SRL5" FOR RANDOM AS 1  
' Envoi de la commande  
  IOCTL #1,"CNTRX"  
' Lecture et analyse de la réponse  
  REP$= IOCTL$(1)  
  IF LEN(REP$) <> 4 THEN GOTO IOERROR  
  NBCAR= VAL("&h" +REP$)  
  IF NBCAR < 0 THEN NBCAR = 65536 + NBCAR
```

Exemple 2

Initialisation du canal 1 à 4800 Bauds, 7 bits/caractère, 2 stops, pas de parité, mode de handshake XON/XOFF. Initialisation du canal 2 à 300 Bauds, 5 bits/caractères, 1.5 stops, parité paire, handshake STX/ETX. Transmission de tous les caractères reçus sur SRL1 vers SRL2.

```
'Ouverture des canaux de communication
  OPEN "SRL1" FOR INPUT AS #1
  OPEN "SRL2" FOR OUTPUT AS #2

'Initialisations
  IOCTL #1,"vinit: 4800,7b,2s,np"
  IOCTL #1,"vmode:xon/xoff"
  IOCTL #1,"timeo:-"

  IOCTL #2,"vinit:300, 5b, 1.5s, ep"
  IOCTL #2,"vmode:stx/etx"

'Réception sur SRL1, Emission sur SRL2
  DO
    A$= INPUT$(1,1)
    PRINT #2,A$;
  LOOP WHILE INKEY$ = ""
```

Exemple 3

Transmission du fichier "DEMO.TXT" sur le canal SRL3 par blocs de 128 octets. SRL3 est initialisé à 4800 Bd, 8 bits/caractère, 2 stops, pas de parité.

```
'Ouverture du canal et du fichier'
  OPEN "SRL3" FOR RANDOM AS 1 LEN= 128
  FIELD #1, 128 as A$

OPEN "DEMO.TXT" FOR RANDOM AS 2 LEN= 128
  FIELD #2, 128 as B$

'Initialisation du canal
  IOCTL #1,"VINIT: 4800"

'Lecture du fichier et transmission
  WHILE NOT EOF(2)
    GET #2
    RSET A$= B$
    PUT #1
  WEND
.
.
.
```

XVII.4.4 Utilisation sous quick basic 4.0 microsoft

A partir de la version 4.0, QUICK BASIC offre la possibilité d'ouvrir les périphériques (ainsi que les fichiers) en mode binaire. Cette particularité présente l'avantage d'autoriser des transferts par blocs, comme en mode d'accès aléatoire, mais avec des blocs de taille variable.

Dans ce mode les instructions utilisées pour la lecture sont :

```
GET [#]dh, rec, variable  
INPUT$(n, [#]dh)
```

Avec dh = Numéro de fichier (instruction OPEN)
rec = Numéro d'enregistrement (non significatif: rec=1)
n = Nombre de caractères à lire

Dans le cas de l'instruction GET, le nombre de caractères lus dépend de la longueur de la variable : s'il s'agit d'un entier, deux octets seront lus; s'il s'agit d'une chaîne de caractères, GET lit un nombre de caractères égal à la longueur de la chaîne.

L'instruction PUT est utilisée pour l'émission des caractères.

```
PUT [#]dh, rec, variable
```

EXEMPLE :

```
DEFINT A-Z  
N = 0  
  
' OUVERTURE CANAL SRL1'  
OPEN "SRL1" FOR BINARY ACCESS READ WRITE AS 1  
  
' ENVOI D'UN BLOC DE 8 CARACTERES'  
A$ = "ACKSYS" + CHR$(13) + CHR$(10)  
PUT #1, 1, A$  
  
' RECEPTION DE 10 CARACTERES'  
  
WHILE N < 10  
IOCTL 1, "cnt": N = VAL("&h" + RIGHT$(IOCTL$(1), 4))  
WEND  
  
B$ = INPUT$(10, 1)  
  
END
```

XVII.5 LES COMMANDES IOCTL

XVII.5.1 Généralités

Les fonctions IOCTL permettent au programme d'envoyer des commandes au driver, par exemple pour modifier les paramètres de transmission, et d'obtenir des informations telles que le nombre de caractères reçus ou l'état d'un port de communication. La syntaxe de chacune de ces commandes est décrite ci-après. Le format général est le suivant :

COMMANDE: PARAM1, PARAM2, PARAM3,

Il s'agit d'une chaîne de caractères ASCII composée d'une commande et suivie éventuellement de paramètres. La commande est constituée de cinq caractères ASCII suivis obligatoirement de deux points lorsqu'il y a des paramètres. Ces derniers sont séparés par des virgules ou des espaces et peuvent être omis pour certaines commandes. Par exemple, la commande VINIT peut s'écrire:

VINIT: , , , odd

Les trois premiers paramètres prendront dans ce cas les valeurs par défaut, soit respectivement :

- 9600 bits par seconde
- 8 bits par caractère
- 2 bits de stop

Les paramètres numériques ne peuvent excéder 65535 ; ils peuvent être exprimés sous forme hexadécimale (sauf indication contraire) et sont dans ce cas suivis de la lettre h (accolée : 2000h). Ainsi 512 est équivalent à 200h.

Les trois premiers caractères de la commande sont suffisants pour son identification: par exemple, vinit peut s'écrire vin. Les majuscules et minuscules ne sont pas différenciées.

La réponse à une commande est obtenue par appel d'une fonction 'IOCTL' en lecture (IOCTL\$ en BASIC). Le nombre de caractères renvoyés varie en fonction de la commande et du format de retour (options /d et /n de la ligne d'invocation du driver), mais ne peut excéder 16 caractères. Sauf indication particulière dans la description des commandes IOCTL, le driver renvoie un octet, que nous appellerons 'code de retour', indiquant si la commande a été comprise ou non, soit deux caractères en hexadécimal ASCII.

Les commandes renvoyant une valeur, telles que CNTRX, TFREE ou BSTAT, ne renvoient pas de code de retour mais directement une valeur ou une chaîne de caractères (commandes RELRP, CFGRP...). Le code de retour peut prendre les valeurs suivantes :

- 00 : *La commande a été correctement exécutée.*
- C0 : *Erreur sur un des paramètres de la commande (non déterminé)*
- C1 : *Premier paramètre erroné ou manquant*
- C2 : *Second paramètre erroné ou manquant*
- Cn : *n^{ième} paramètre erroné ou manquant*
- D0 : *Commande non reconnue*

Si l'option /d est spécifiée dans la ligne d'invocation du programme de commande, les valeurs sont retournées converties en décimal. Le nombre de caractères est par conséquent différent : les valeurs sur un octet, telles que les codes de retour, sont retournées sur trois caractères ('000' à '255'), les valeurs sur deux octets sont retournées sur cinq caractères ('00000' à '65535').

Si l'option /n est spécifiée, les valeurs et codes de retour ne sont pas convertis en ASCII. Si l'option /d est spécifiée, les valeurs sur un octet sont exprimées par deux caractères (00 00 à 02 55) et les autres valeurs sur trois caractères (00 00 00 à 06 55 35).

Exemple : Retour de la fonction CNTRX, valeur retournée 1024 :

Sans options :	30 34 30 30	soit 0400 en ASCII
Option /d :	30 31 30 32 34	soit 01024 ASCII
Option /u :	00 04	soit 0400h
Option /d/u :	24 10 00	soit 001024

REMARQUE : Lorsque l'option /u est spécifiée, les valeurs sont retournées avec l'octet de poids le plus faible à l'adresse la plus basse (poids forts à droite).

XVII.5.2 Description des commandes IOCTL

Commande VINIT

Définition des paramètres de transmission. Cette commande vide les tampons d'émission et de réception. Elle ne doit pas être envoyée pendant une émission ou une réception de caractères (risque de désynchronisation).

Format: VINIT: Bd, len, stp, pty

- Bd : Vitesse de transmission en Bauds. Les valeurs possibles sont 50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600, 14400, 19200, 28800, 38400, 57600 et 115200 Bauds. Cette valeur ne peut pas être exprimée en hexadécimale. La valeur par défaut est 9600 Bauds.
- len : Nombre de bits par caractère. Les valeurs admises sont 5, 6, 7 et 8 éventuellement suivies de la lettre b. La valeur par défaut est 8.
- stp : Nombre de bits de stop. Les valeurs admises sont 1, 1.5 et 2, éventuellement suivies de la lettre s. La valeur par défaut est 2.
- pty : Parité : parité paire = paire ou even (abrégé pa ou ev)
 parité impaire = impaire ou odd (abrégé im ou od)
 pas de parité = sansparité ou npty (abrégé sa ou np).

Exemple 1

Initialisation en basic de la voie 3 : rx et tx 7 bits/car, 1 stop, parité impaire, 1200 Bauds.

```
OPEN "R",#1,"SRL3"  
IOCTL #1,"VINIT: 1200, 7b, 1s, odd "
```

Exemple 2 :

Initialisation en assembleur de la voie 2: rx, tx = 8 bits/car, 1 stop, pas de parité, 300 Bauds.

```
ASSUME CS:CSEG,DS:CSEG
```

```

mov  ax,3D02h           ; Ouverture en lecture/écriture
mov  dx,offset CHANNEL ; Pointe sur le nom du canal
int  21h               ; Appel system DOS
jc   open_fail        ; Saut si erreur
mov  bx,ax             ; 'handle' dans bx
mov  si,offset STRING  ; Pointe sur la chaîne de caractères
call ioctl_write       ; Sous-programme envoie de commande
.
.
.

```

```
; SOUS-PROGRAMME I/O CONTROL
```

```
;
; Transmet la commande pointée par DS:DX et teste le code de retour.
; Positionne CF si erreur.
;
```

```
ioctl_write proc near
```

```

mov  cx,[si]           ; Nombre de caractères dans CX
mov  dx,si             ;
add  dx,2              ; Pointe sur début de la chaîne
mov  ax,4403h         ; Code Ecriture IOCTL
int  21h              ; Appel de fonction DOS
jc   ioctlexit        ; Saut si erreur
mov  ax,4402h         ; Code Lecture IOCTL
mov  dx,offset RETCODE ; Destination du code de retour
mov  cx,18            ; Nombre de caractères (maximum)
int  21h              ; Appel dos
jc   ioctlexit        ; Saut si erreur
cmp  [RETCODE],'00'   ; Test code de retour = 0
jz   ioctlexit        ; Sortie si oui
stc                          ; Positionne CF (erreur)

```

```
ioctlexit:
```

```
ret ; Retour au programme principal
```

```
ioctl_write endp
```

```
.
```

```

CHANNEL byte 'SRL2',0 ; Non du canal format ASCIIZ
STRING  word 12       ; Longueur de la chaîne
        byte 'vin: 300,8,1' ; 1 stop, 300 Bd, par défaut pas de parité
RETCODE dword ?      ; Espace réservé pour le code de retour
        byte 16 dup ? ;

```

Commande DIMBF

Redéfinition de la taille des tampons d'émission et de réception de tous les canaux.

Format: DIMBF: rrrr, tttt
 DIMBF: @ssss, aaaa

Dans le premier cas, rrrr représente la taille des tampons de réception pour tous les canaux, tttt représente la taille des tampons d'émission pour tous les canaux. Dans le second cas, ssss représente le segment et aaaa l'adresse d'une table contenant les tailles des tampons de réception et d'émission des ports 1 à 4 (soit 16 octets). Les tailles des tampons doivent être comprises entre 32 et 32767 octets. La taille totale ne peut excéder la taille initiale définie au chargement (celle-ci peut être connue par envoi de la commande BSIZE). Le code de retour renvoyé par la commande DIMBF peut prendre les valeurs suivantes :

- 00 = Commande correctement exécutée.
- C1 = Erreur sur le premier paramètre (rrrr)
- C2 = Erreur sur le second paramètre (tttt)
- C0 = Taille totale trop grande.

Exemple :

```

DIM ts%(8)                                ' Table
RESTORE TSize
FOR N = 0 TO 7 : READ ts%(N) : NEXT N      ' Remplissage de la table

OPEN "SRL1" FOR BINARY AS 1                ' Ouverture canal 1

IOCTL 1, "dim:@" + HEX$(VARSEG(ts%(0))) + "h," + HEX$(VARPTR(ts%(0))) + "h"

...

```

TSize :

```

DATA 1024, 512      ' Port 1 : reception = 1024 octets, émission = 512 o
DATA 2048, 1024    ' Port 2 : rx = 2048, tx = 1024
DATA 384, 768     ' Port 3 : rx = 384, tx = 768
DATA 256, 512     ' Port 4 : rx = 256, tx = 512

```

Commande VMODE

Initialisation du mode de contrôle de flux ('handshake').

Format: VMODE: con/coff

Cinq modes de 'handshake' sont possibles, déterminés par le paramètre 'con/coff' :

- 1 - XON/XOFF : Transmission autorisée = caractère 11h (xon)
Arrêt de la transmission = caractère 13h (xoff)
- 2 - XANY/XOFF : Transmission autorisée = caractère quelconque
Arrêt de la transmission = caractère 13h (xoff)
- 3 - ACK/NAK : Transmission autorisée = caractère 06 (ack)
Arrêt de la transmission = caractère 15h (nak)
- 4 - STX/ETX : Transmission autorisée = caractère 02h (stx)
Arrêt de la transmission = caractère 03h (etx)
- 5 - RTS/CTS : 'handshake hardware', contrôlé par les lignes RTS et CTS
- 6 - SC ee/dd : Transmission autorisée = caractère ee
Arrêt de la transmission = caractère dd

Le mode RTS/CTS n'est pas autorisé lorsque les modes 'contrôle modem' ou 'open/close' sont sélectionnés (voir commande STMOD) .

Le dernier mode (SC ee/dd) permet à l'utilisateur de définir ses propres caractères de contrôle; ceux-ci sont désignés par leur code ASCII.

Lorsque con/coff est omis, il n'y a pas de 'handshake'.

Exemples

Initialisation de la voie 1 en mode STX/ETX, en basic

```
OPEN "SRL1" FOR RANDOM AS 1
IOCTL #1,"vmode: stx/etx "
```

Handshake avec caractères Ctrl V et Ctrl X :

```
IOCTL #1,"vmode:sc 16h/18h"
```

Suppression du contrôle de flux :

```
IOCTL #1,"vmode"
```

Commande STMOD

Sélection du mode contrôle des signaux modem et 'open/close'

Format : STMOD: modem, opcl
 STMOD: nomodem, nopcl

modem : passage en mode contrôle modem
opcl : passage en mode 'open/close'
nomodem : Quitte le mode contrôle modem
nopcl : Quitte le mode 'open/close'

Mode MODEM : Dans ce mode, l'émission des caractères n'est autorisée que si les signaux /CTS, /DSR et /DCD sont activés. Lorsque le tampon de réception est plein, le signal /RTS est désactivé. Le contrôle de flux est autorisé.

Mode 'open/close' : Lorsque ce mode est sélectionné, la réception sur un port n'est autorisée qu'après ouverture du canal par OPEN. Les tampons d'émission et de réception sont vidés et les signaux /DTR et /RTS sont activés. A la fermeture du canal (CLOSE), les signaux /DTR et /RTS sont désactivés, la réception est interdite et le tampon de réception est vidé. Si un mode de contrôle de flux est sélectionné (xon/xoff par exemple) le caractère d'arrêt (xoff) est envoyé sur la ligne; dans ce cas, le caractère de reprise (xon) sera envoyé à l'ouverture suivante. Le tampon d'émission n'est pas vidé lors de la fermeture de la ligne.

Commande TIMEO

Permet de fixer le temps d'attente des caractères en entrée (INPUT) et le temps d'attente de disponibilité des tampons en sortie (OUTPUT). Les valeurs sont exprimées en secondes, et comprises entre 1 et 3600 secondes. Si la valeur du time-out en sortie est omise, elle sera égale la valeur du time out en entrée. La valeur par défaut, à l'initialisation, est de 10 secondes. Un temps d'attente infini peut être obtenu avec le caractère '-' en paramètre.

Format : TIMEO : nnnn, mmmm
 TIMEO : - (Temps d'attente infini)

avec nnnn = Time out en entrée (0 à 3600)
 mmmm = Time out en sortie (0 à 3600)

Exemples :

Time out en entrée = 30 secondes, pas de time out en sortie (0 secondes) :

```
OPEN "SRL1" FOR BINARY AS 1
IOCTL #1, "TIMEO: 30, 0"
```

Time out en entrée = time out en sortie = 60 secondes :

```
IOCTL #1, "TIMEO: 60"
```

Time out en entrée infini, time out en sortie infini :

```
IOCTL #1, "TIMEO: -, -"
```

Commande RXENB

Activation ou désactivation d'un canal de réception

```
Format: RXENB: ON    (Activation)
        RXENB: OFF   (Désactivation)
```

Lorsque la ligne est désactivée, les caractères reçus ne sont pas mémorisés dans le tampon de réception. Les caractères de contrôle de flux sont cependant détectés.

Commande CLRRX

Effacement du tampon de réception. Tous les pointeurs concernant ce tampon sont remis à zéro. Pas de paramètres.

Format : CLRRX

Commande FLUSH

Effacement du tampon d'émission. Tous les pointeurs sont remis à zéro.

Format : FLUSH

Sur réception de cette commande, la carte stoppe toute transmission sur ce canal et vide le tampon d'émission. Pas de paramètres.

Commande TFREE

Lecture de la place disponible dans le tampon de transmission.

Format : TFREE

Le format de la valeur retournée est fonction des options de la ligne d'invocation du programme de commande (CONFIG.SYS), ou du format spécifié par la commande EXFMT :

- *Pas d'options (exf: asc hex) : Sous forme hexadécimale, quatre caractères ASCII.*
- *Option /d (exf: asc dec) : Sous forme décimale, sur cinq caractères ASCII*
- *Option /n (exf: num hex) : Sous forme hexadécimale, sur deux caractères*
- *Option /nd (exf: num bcd) : Sous forme BCD, sur trois caractères.*

Commande CNTRX

Lecture du nombre de caractères présents dans le tampon de réception.

Format : CNTRX

Format de la réponse : Voir commande TFREE ci-dessus.

Commande ERLOC

Lorsque SRLDRV détecte un erreur de réception, la position dans le tampon du caractère erroné est mémorisée. Lors de la lecture du tampon par le système, si le mauvais caractère est compris dans le bloc de caractères à lire, la lecture s'arrête sur ce caractère et un code d'erreur est retourné au DOS, provoquant l'affichage d'un message du type :

Read fault reading device SRL1

La commande ERLOC permet de connaître la position du prochain caractère défectueux dans le tampon de la carte et éventuellement de considérer ce caractère comme bon. Dans ce cas, la commande ERLOC doit être suivie du signe '!'.
Format : ERLOC: [-]

Format de la réponse : Voir commande TFREE ci-dessus. Si la valeur 0 est retournée, il n'y a pas eu d'erreur de réception.

Commande LSTAT et ISTAT

Lecture de l'état des unités de communication

Format : LSTAT [: b][-]
 ISTAT [: b]

Les informations renvoyées sont codées sur un mot de 16 bits, de la façon suivante :

- bit 15 (8000h) Etat du signal /DCD (Data Carrier Detect). Signal entrant.
- bit 14 (4000h) Etat du signal /RI (Ring Indicator). Signal entrant.
- bit 13 (2000h) Etat du signal /DSR (Data Set Ready). Signal entrant.
- bit 12 (1000h) Etat du signal /CTS (Clear To Send). Signal entrant.
- bit 11 (0800h) A 1, indique qu'une variation du signal /DCD a eu lieu.
- bit 10 (0400h) A 1, indique qu'une variation du signal /RI a eu lieu.
- bit 9 (0200h) A 1, indique qu'une variation du signal /DSR a eu lieu.
- bit 8 (0100h) A 1, indique qu'une variation du signal /CTS a eu lieu.
- bit 7 (0080h) Etat du signal /DTR (Data Terminal Ready). /DTR est un signal sortant.
- bit 6 (0020h) A 1, indique que le registre de transmission de l'UART (THRE) est libre.
- bit 4 (0010h) A 1, indique qu'un break a été détecté sur la ligne (BI).
- bit 3 (0008h) Erreur trame (FE) : indique qu'un caractère de format incorrect a été reçu
- bit 2 (0004h) Erreur de parité (PE) : Indique qu'un caractère avec mauvaise parité a été reçu.
- bit 1 (0002h) Indique un écrasement de caractères au niveau du contrôleur de communication (OR). Ce phénomène se produit lorsque l'unité centrale n'a pas pu venir lire assez rapidement les caractères reçus, et que d'autres caractères sont arrivés, écrasant les précédents.
- bit 0 (0001h) Indique qu'un caractère est disponible dans le registre de réception de l'unité de communication (DR).

Différence entre LSTAT et ISTAT : LSTAT renvoie des informations correspondant aux valeurs des registres de l'unité de communication lors de la dernière interruption les concernant : Interruption sur réception de caractères (DR, OR, PE, FE, BI, THRE) ou interruption sur changement d'état des signaux de contrôle. Les informations renvoyées par ISTAT, par contre, correspondent à une lecture de l'unité de communication effectuée instantanément. L'option '-' de LSTAT permet la remise à zéro des bits indiquant une transition des signaux de contrôle

Le format de la valeur retournée est fonction des options de la ligne d'invocation du driver ou du format spécifié par la commande EXFMT :

- *Pas d'options (exf: asc hex) : Sous forme hexadécimale, quatre caractères ASCII*
- *Option /d (exf: asc dec) : Sous forme décimale, sur cinq caractères ASCII*
- *Option /n (exf: num hex) : Sous forme hexadécimale, sur deux caractères*
- *Option /nd (exf: num bcd) : Sous forme BCD, sur trois caractères.*

Si le format décimal est sélectionné (option /d), le décryptage de la valeur peut être simplifié en spécifiant le paramètre 'b'. Dans ce cas, la valeur est retournée sous forme binaire, représentée par une chaîne de seize caractères ASCII '0' ou '1'. Le premier caractère (caractère de gauche de la chaîne) représente le bit 15, le caractère suivant le bit 14, etc...

Commande BSTAT

La commande BSTAT retourne des informations concernant la gestion des tampons d'émission et de réception.

Format : BSTAT [: b]

- bit 7 (80h) Ce bit indique, lorsqu'il est positionné à 1, que le tampon de réception est plein. Si un mode de contrôle de flux est sélectionné (xon/xoff par exemple), cela signifie que le caractère d'arrêt (xoff) n'a pas été pris en compte par l'équipement distant. Dans ce cas, les prochains caractères reçus ne seront pas mémorisés. S'il n'y a pas de contrôle de flux, les caractères reçus sont mémorisés en fin de tampon et les caractères en début de tampon sont perdus. Ce bit est remis à zéro par une lecture du canal ou une commande de vidage du tampon (CLRRX).
- bit 6 (40h) Indique lorsqu'il est à 1 qu'il y a eut écrasement ou perte de caractère en réception : un caractère a été reçu alors que le tampon de réception était plein (bit 7 positionné). Ce bit est remis à zéro par une lecture du canal ou une commande de vidage du tampon (CLRRX).
- bit 5 (20h) Indique qu'un caractère a été mal reçu : erreur de parité ou de format. Ce bit est remis à zéro après lecture du caractère ou effacement du tampon de réception.
- bit 4 (10h) Ce bit indique que la réception des caractères est interdite sur ce canal. Il est positionné par la commande RXENB ou par le driver en mode 'open/close'.
- bit 3 (08h) Indique qu'une transmission de caractères est en cours sur ce canal. remis à zéro par un vidage du tampon d'émission (FLUSH) ou après émission de tous les caractères du tampon.
- bit 2 (04h) Réserve
- bit 1 (02h) Emission de caractère interdite : un caractère d'arrêt de la transmission a été reçu (contrôle de flux). Ce bit est remis à zéro après réception du caractère de reprise de transmission ou à la suite d'un vidage du tampon (commande FLUSH).
- bit 0 (01h) Réserve

Le format de la valeur retournée est fonction des options de la ligne d'invocation du driver ou du format spécifié par la commande EXFMT :

- *Pas d'options (exf: asc hex) : Sous forme hexadécimale, sur deux caractères ASCII.*
- *Option /d (exf: asc dec) : Sous forme décimale, sur trois caractères ASCII*
- *Option /n (exf: num hex) : Sous forme hexadécimale, sur un caractère*
- *Option /nd (exf: num bcd) : Sous forme BCD, sur deux caractères.*

Si le format décimal est sélectionné (option /d), le décryptage de la valeur peut être simplifié en spécifiant le paramètre 'b'. Dans ce cas, la valeur est retournée sous forme binaire, représentée par une chaîne de huit caractères ASCII '0' ou '1'. Le premier caractère (caractère de gauche de la chaîne) représente le bit 7, le caractère suivant le bit 6, etc...

Commande BSIZE

Cette commande permet de connaître les dimensions des tampons de réception et d'émission.

Format : BSIZE : type

- type = RX : Retourne la taille du tampon de réception.
- type = TX : Retourne la taille du tampon d'émission
- type non spécifié : Retourne la taille totale (tampon rx et tx de toutes les voies)
en paragraphes (taille/16)

Le format de la valeur retournée est fonction des options de la ligne d'invocation du driver ou du format spécifié par la commande EXFMT :

- *Pas d'options (exf: asc hex) : Sous forme hexadécimale, sur quatre caractères.*
- *Option /d (exf: asc dec) : Sous forme décimale, sur six caractères ASCII*
- *Option /n (exf: num hex) : Sous forme hexadécimale, sur deux caractères*
- *Option /nd (exf: num bcd) : Sous forme BCD, sur trois caractères.*

Commande STSIG

Contrôle des signaux RTS et DTR.

Format : STSIG: RTS, /DTR

- RTS : Active le signal RTS
- DTR : Active le signal DTR
- /RTS : Désactive le signal RTS
- /DTR : Désactive le signal DTR

Lorsque le nom d'un signal est omis, celui-ci est désactivé; ainsi l'expression :

'stsig: /rts, dtr'

est équivalente à : 'stsig: dtr'

Commande BREAK

Envoi d'un BREAK sur la ligne (Mise à zéro de la ligne d'émission)

Format : BREAK: L
BREAK: S

L signifie 'Long break' soit 4 secondes
S signifie 'Short break' soit 1 seconde

Commande RELRP

RELRP renvoie le numéro de révision du programme de commande et le nombre de voies installées.

Format : RELRP

Les informations sont retournées sous forme d'une chaîne de 3 caractères ASCII. Les deux premiers caractères représentent le numéro de révision du programme de commande : par exemple 14 = révision 1.4 . Le troisième caractère est le nombre de voies installées.

Commande CFGRP

Cette commande retourne la chaîne d'identification de la configuration du programme de commande.

Format : CFGRPT

Revoie la chaîne de seize caractères ASCII entrée lors de la configuration du programme de commande par SRLSET.EXE. Cette chaîne permet d'identifier les configurations particulières du driver. La chaîne ' STANDARD ' correspond à la configuration initiale du programme de commande à la livraison.

Commande EXFMT

Définition du format des données retournées par les fonctions IOCTL en lecture. Cette commande permet de modifier dynamiquement le format spécifié par la ligne d'invocation du programme de commande dans le fichier CONFIG.SYS.

Format : EXFMT: hexa, décimal, bcd, ascii, num

hexa : Les données sont retournées sous forme hexadécimale (base 16). Format par défaut au chargement du driver.

décimal: Les données sont retournées sous forme décimale (base 10). Equivalent à /d dans la ligne d'invocation du driver.

bcd : décimal.

ascii : Les données sont retournées sous forme d'une chaîne de caractères ascii. Format par défaut au chargement du driver.

num : Les valeurs retournées sont numériques (non ascii). Equivalent à /n.

Le format par défaut est hexa ascii.

Commande RDREG

La commande RDREG permet de lire un des registres de l'unité de communication.

Format : RDREG: registre

'registre' est le nom du registre à lire :

- **rbr** = Registre de réception de caractère.
- **ier** = Registre de validation des interruptions.
- **iir** = Registre d'identification des interruptions.
- **lcr** = Registre de contrôle ligne
- **mcr** = Registre de contrôle modem
- **lsr** = Registre d'état ligne
- **msr** = Registre d'état modem
- **scr** = 'Scratch register'
- **pol** = Registre de polling (voie 1 uniquement)
- **dll** = Partie basse du diviseur de fréquence.
- **dlh** = Partie haute du diviseur de fréquence.

Le format de la valeur retournée est fonction des options de la ligne d'invocation du driver ou du format spécifié par la commande EXFMT :

- *Pas d'options (exf: asc hex) : Sous forme hexadécimale, sur deux caractères ASCII.*
- *Option /d (exf: asc dec) : Sous forme décimale, sur trois caractères ASCII*
- *Option /n (exf: num hex) : Sous forme hexadécimale, sur un caractère*
- *Option /nd (exf: num bcd) : Sous forme BCD, sur deux caractères.*

Exemples :

Lecture du registre d'état modem du canal 4 . Option /d :

```
OPEN "SRL4" FOR BINARY AS 1
IOCTL #1, " rdreg : msr "
MSR = VAL ( IOCTL$(1) )
```

Lecture du diviseur de fréquence (permet de connaître la fréquence de transmission) de la voie 2. Format de retour hexadécimal ASCII (pas d'options) :

```
OPEN " SRL2" FOR BINARY AS 1
IOCTL #1, " rdr : dlh "
DLH$ = IOCTL$(1)
IOCTL 1, " rdr : dll "
DIV = VAL ("&h" + DLH$ + IOCTL$(1))
```

Commande WRREG

Ecriture dans un registre de l'unité de communication.

Format : WRREG : registre, valeur

'registre' est le nom du registre :

- thr = Registre de transmission de caractères
- ier = Registre de validation des modes d'interruption.
- lcr = Registre de contrôle ligne
- mcr = Registre de contrôle modem
- scr = 'Scratch register'
- msk = Registre de masque (voie 1 uniquement)
- dll = Partie basse du diviseur de fréquence.
- dlh = Partie haute du diviseur de fréquence.

la valeur doit être comprise entre 0 et 255.

Exemple :

Désactivation du signal RTS de la voie 3 :

```
OPEN "SRL3" FOR BINARY AS 1
IOCTL #1, " rdreg : mcr "
MCR = VAL("&h" + IOCTL$(1) )
IOCTL 1, " wrreg : mcr, "+str$(MCR AND &hFD)
```

**ATTENTION : CETTE COMMANDE PEUT
PERTURBER LE FONCTIONNEMENT DU
DRIVER**

XVII.5.3 Exemple de programmation des fonctions ioctl du driver

```

#include <dos.h>
#include <stdio.h>
#include <fcntl.h>
#include <io.h>
#include <stdlib.h>
#include <string.h>

int ioctl(int dh, char _far *commande);
void setraw(int dh);

union REGS rin, rout;
struct SREGS segregs;

char _far *exfmt = "exfmt:hexa,num";
char _far *vinit = "vinit: 1200 7b 1s ep";
char _far *tfree = "tfree";
char _far *cntrx = "cntrx";
char *txstr = "ACKSYS ";
int ioctlin; /* utilisée pour la lecture IOCTL */
int far *iorep; /* Pointeur sur ioctlin */
char rdbuf[512];

void main()
{
    int dh1, dh2, cnt;
    unsigned nbchar;

    dh1 = open("SRL1",O_RDWR | O_BINARY); /* ouverture SRL1 */
    dh2 = open("SRL2",O_RDWR | O_BINARY); /* ouverture SRL2 */

    iorep = &ioctlin; /* Init iorep pointeur sur ioctlin */
    setraw(dh1); /* Passage mode raw SRL1 */
    setraw(dh2); /* Passage mode raw SRL2 */
    ioctl(dh1, exfmt); /* Fixe le format de retour */
    ioctl(dh1, vinit); /* Commande vinit sur SRL1 */
    ioctl(dh2, vinit); /* Commande vinit sur SRL2 */

    if ((nbchar = write(dh1, txstr, strlen(txstr))) != strlen(txstr))
        printf ("Erreur d'écriture sur SRL1 nb car = %d\n\r",nbchar);

    printf ("Ecriture sur SRL1 nb car = %d\n\r",nbchar);

    while ((nbchar= ioctl(dh2, cntrx)) < strlen(txstr) && cnt++ < 20000)
        printf ( "%d \ r",nbchar);

    read(dh2, rdbuf, ioctl(dh2, cntrx));
    rdbuf[strlen(txstr)] = 0x00;
    printf("Caracteres reçus : '%s'\n\r",rdbuf);
}

```

ioctl passe la chaîne de commande au driver par l'intermédiaire de l'interruption 21H, fonction 44H

```

ioctl(int dh, char far *commande)
{
    ioctlin = 0;                /* Init ioctlin à zéro          */
    rin.h.ah = 0x44;           /* Numéro de la fonction       */
    rin.h.al = 0x03;           /* sous fonction écriture      */
    rin.x.bx = dh;             /* Code de gestion             */
    rin.x.cx = strlen(commande); /* Nombre de caractères        */
    rin.x.dx = FPOFF(commande); /* Adresse de la chaîne        */
    segregs.ds = FPSEG(commande); /* Segment de la chaîne        */
    intdosx(&rin, &rout, &segregs); /* Envoi interruption          */
    rin.h.ah = 0x44;           /* Numéro de la fonction       */
    rin.h.al = 0x02;           /* sous-fonction lecture       */
    rin.x.bx = dh;             /* Code de gestion             */
    rin.x.cx = 16;             /* Nombre de caractères        */
    rin.x.dx = FPOFF(iorep);   /* Adresse réponse             */
    segregs.ds = FPSEG(iorep); /* Segment                      */
    intdosx(&rin, &rout, &segregs); /* Appel système                */
    return(ioctlin);          /* Retourne la valeur          */
}

```

setraw passe le périphérique en mode 'RAW'. Supprime le filtrage des caractères et autorise les transferts par blocs

```

void setraw(int dh)
{
    rin.x.ax = 0x4400;         /* 'Get device informations'    */
    rin.x.bx = dh;             /* Code de gestion              */
    intdosx(&rin, &rout, &segregs); /* Envoie interruption 21h     */
    rin.x.ax = 0x4401;         /* 'Set device informations'    */
    rin.h.dh = 0x00;           /* Force dh = 0                 */
    rin.h.dl = rout.h.dl | 0x20; /* Bit mode binaire             */
    intdosx(&rin, &rout, &segregs); /* Appel système                */
}

```

XVII.6 LE PROGRAMME COMCTL.COM

COMCTL.COM permet d'envoyer une commande IOCTL à la carte ACKSYS directement à partir du processeur de commandes ('shell') ou d'un fichier de commande (procédure). Le format est le suivant :

```
COMCTL n,<commande>
```

n représente le numéro de canal, <commande> la commande IOCTL. Par exemple, l'initialisation des paramètres de transmission du canal 2 (SRL2) s'écrirait :

```
COMCTL 2, vinit: 1200, 7, 2, o
```

Si une erreur est détectée, le code de retour hexadécimal est affiché. Les informations renvoyées par certaines commandes sont également affichées.

COMCTL.COM retourne un code de sortie pouvant être testé par la commande IF ERRORLEVEL. La signification du code de sortie est la suivante :

- 1 = Aucune erreur détectée, code de retour 00
- 2 = Aucune erreur détectée, informations retournées
- 3 = Erreur de syntaxe dans la ligne de commande
- 4 = Erreur à l'ouverture du canal
- 5 = Erreur d'écriture IOCTL
- 6 = Erreur de lecture IOCTL
- 7 = Erreur Time out
- 8 = Autre erreur (Code de retour différent de zéro)

Exemple :

Procédure INIT.BAT

```
@echo off

:debut

comctl %1,vinit: 600, 7, 2, o
if errorlevel 3 goto erreur

comctl %1, vmode: xon/xoff
if errorlevel 3 goto erreur

comctl %1, timeo: 15, 20
if errorlevel 3 goto erreur

comctl %1,clrrx
if errorlevel 3 goto erreur
```

```
echo SRL%1 initialisée
shift
if not %1 == 0 goto debut
goto fin

:erreur
echo Erreur à l'initialisation de SRL%1

:fin
echo Initialisation terminée
```

Cette procédure appelée sous la forme :

```
INIT 1 2 3 4 0
```

provoquera l'initialisation des canaux 1 à 4 si aucune erreur n'est rencontrée.

XVII.7 INTERRUPTIONS 60H ET 14H

L'option /i de la ligne d'invocation du programme de commande permet d'installer le 'handler' de l'interruption 60H pour les cartes ACKSYS. Les fonctions supportées par ce handler sont similaires aux fonctions de l'interruption 14H du BIOS, ce qui permet d'adapter très rapidement des programmes existants aux cartes ACKSYS. Il est possible également d'utiliser directement l'interruption 14H, à l'aide de l'option /cn de la ligne d'invocation du programme de commande. Dans ce cas, n indique le numéro de port attribué à la voie 1 de la carte: par exemple, si vous disposez déjà des ports COM1 et COM2, leur numéro de port pour l'interruption 14H sera respectivement 0 et 1; le premier numéro de port attribué à la carte ACKSYS doit donc être: 2.

XVII.7.1 Description des fonctions

Fonction 00H - Initialisation des paramètres de communication.

Permet de définir la vitesse de communication, la parité, le nombre de bits par caractère et le nombre de bits de stop.

Appel avec: AH = 00H
AL = Paramètres de communication

bits 7 6 5	vitesse bits/sec	bits 4 3	parité	bit 2	stops	bits 1 0	bits/ car
0 0 0	110	x 0	sans	0	1 bit	0 0	5
0 0 1	150	0 1	impaire	1	2 bits	0 1	6
0 1 0	300	1 1	paire			1 0	7
0 1 1	600					1 1	8
1 0 0	1200						
1 0 1	2400						
1 1 0	4800						
1 1 1	9600						

DX = Numéro du port de la carte (0 - 7 ou n - n+7)

Retourne: AH = Etat de la ligne
d0 = Donnée disponible (dans le tampon de réception)
d1 = Ecrasement de caractères détecté
d2 = Erreur de parité sur le premier caractère du tampon
d3 = Erreur trame sur le premier caractère du tampon
d4 = Break détecté
d5 = Tampon de transmission disponible (au moins 1 car)
d6 = Tampon de transmission vide
d7 = Erreur time-out

AL = Etat Modem
d0 = Changement d'état du signal /CTS
d1 = Changement d'état du signal /DSR
d2 = Détection de sonnerie signal /RI
d3 = Changement d'état du signal /DCD
d4 = Etat du signal /CTS
d5 = Etat du signal /DSR
d6 = Etat du signal /RI
d7 = Etat du signal /DCD

Fonction 01H - Envoi d'un caractère

La fonction 01H permet d'envoyer un caractère sur le canal spécifié. Retourne l'état de la ligne. A la différence de la fonction 01h de l'interruption 14h du BIOS, si le transmetteur du contrôleur de communication n'est pas libre, le caractère est mémorisé dans le tampon de transmission puis envoyé ultérieurement.

Appel avec: AH = 01H
AL = Caractère
DX = Numéro du port de la carte (0 - 7 ou n - n+7)

Retourne: AL = Caractère envoyé
AH = Etat de la ligne

d0 = Donnée disponible (dans le tampon de réception)
d1 = Ecrasement de caractères détecté
d2 = Erreur de parité sur le premier caractère du tampon rx
d3 = Erreur trame sur le premier caractère du tampon rx
d4 = Break détecté
d5 = Tampon de transmission disponible (au moins 1 car)
d6 = Tampon de transmission vide
d7 = Erreur time-out. Si ce bit est à 1, l'écriture du caractère a échoué.

Fonction 02H - Lecture d'un caractère

La fonction 02H permet de lire un caractère sur le canal spécifié. Retourne l'état de la ligne. A la différence de la fonction 02h de l'interruption 14H du BIOS, le caractère est lu depuis le tampon de réception et non depuis le contrôleur de communication.

Appel avec: AH = 02H
DX = Numéro du port de la carte (0 - 7 ou n - n+7)

Retourne: AH = Etat de la ligne (voir fonction 00H). Si le bit 7 est à 1, aucun caractère n'a pu être lu (time out).
AL = Caractère envoyé

Fonction 03H - Lecture de l'état du port de communication

Retourne l'état du port de communication spécifié.

Appel avec: AH = 03H
DX = Numéro du port de la carte (0 - 7 ou n - n+7)

Retourne: AH = Etat de la ligne (voir fonction 00H).
AL = Etat modem (voir fonction 00H)

Fonction 04H - Initialisation étendue

Définition de la vitesse de communication, de la parité, du nombre de bits par caractère et du nombre de bits de stop.

Appel avec: AH = 04H

- AL = 'break flag'
00H = pas de break
01H = Envoi d'un break sur la ligne
- BH = parité
00H = pas de parité
01H = parité impaire (odd)
02H = parité paire (even)
03H = parité impaire inversée (stick parity odd)
04H = parité paire inversée (stick parity even)
- BL = Bits de stop
00H = 1 bit de stop
01H = 2 bits de stop (6 à 8 bits par caractères)
01H = 1,5 bits de stop (si 5 bits par caractères)
- CH = Nombre de bit par caractère
00H = 5 bits par caractère
01H = 6 bits par caractère
02H = 7 bits par caractère
03H = 8 bits par caractère
- CL = Vitesse de transmission
00H = 110 bits par seconde
01H = 150 bits par seconde
02H = 300 bits par seconde
03H = 600 bits par seconde
04H = 1200 bits par seconde
05H = 2400 bits par seconde
06H = 4800 bits par seconde
07H = 9600 bits par seconde
08H = 19200 bits par seconde
- DX = Numéro du port de la carte (0 - 7 ou n - n+7)

Retourne: AH = Etat de la ligne (voir fonction 00H).
AL = Etat Modem (voir fonction 00H)

Fonction 05H - Contrôle du port de communication

Lecture ou écriture du registre de contrôle des signaux modem du port de communication spécifié.

Appel avec: AH = 05H
AL = Sous fonction
 00H *Lecture du registre de contrôle modem*
 01H *Écriture du registre de contrôle modem*
BL = Contenu du registre de contrôle modem (si AL = 01H)
 d0 = *Etat du signal DTR (Data Terminal Ready)*
 d1 = *Etat du signal RTS (Request To Send)*
 d2 .. d7 = 0
DX = Numéro du port de la carte (0 - 7 ou n - n+7)

Retourne: Si appel avec AL = 00H

BL = Contenu du registre de contrôle modem (voir ci-dessus)

Si appel avec AL = 01H

AH = Etat de la ligne (voir fonction 00H).

AL = Etat Modem (voir fonction 00H)

XVII.8 CONFIGURATION DU PROGRAMME DE COMMANDE

Certains paramètres du programme de commande SRLDRV.SYS peuvent être modifiés de façon à adapter la configuration initiale aux applications. Pour modifier ou simplement visualiser ces paramètres, lancez le programme COMSET.EXE. Il s'agit de modifications directes ('patch') des valeurs par défaut du driver; par conséquent, le programme de configuration a besoin de savoir sur quel disque et dans quel répertoire se trouve SRLDRV.SYS. Dès que vous avez entré le chemin d'accès complet au driver, la page de saisie des paramètres est affichée :

Tampon TX : Taille du tampon de transmission. Elle doit être comprise entre 32 et 32768 octets. Vous pouvez taper directement une nouvelle valeur ou faire défiler par incréments de 256 octets à l'aide des touches [+] et [-]. Le nombre entré doit être un multiple de 16. Si ce n'est pas le cas, le programme arrondit à la valeur inférieure. Utilisez la touche [TAB] pour passer au port suivant.

Tampon RX : Taille du tampon de réception. Elle doit être comprise entre 32 et 32768 octets. Vous pouvez taper directement une nouvelle valeur ou faire défiler par incréments de 256 octets à l'aide des touches [+] et [-]. Le nombre entré doit être un multiple de 16. Si ce n'est pas le cas, le programme arrondit à la valeur inférieure.

Vitesse : Vitesse de transmission exprimée en bits par seconde. Les vitesses admises sont les suivantes : 50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200 et 38400 bits/sec. Sélectionnez la vitesse désirée à l'aide des touches [+] et [-].

Parité : Sélection de la parité à l'aide des touches [+] et [-]. Les modes possibles sont les suivants :

- *sans* : pas de contrôle de parité
- *paire* : parité paire ('even')
- *imp.* : parité impaire ('odd')
- */paire* : bit de parité forcé à zéro ('stick parity bit')
- */imp.* : bit de parité forcé à un ('stick parity bit')

Bits/car : Sélection du nombre de bits par caractère à l'aide des touches [+] et [-]. Les valeurs autorisées sont 5, 6, 7 ou 8 bits par caractère.

Bits Stop : Sélection du nombre de bits de stop à l'aide des touches [+] et [-]. Avec 5 bits par caractère, le nombre de bit de stop peut être 1 ou 1,5. Avec 6, 7 ou 8 bits par caractère, le nombre de bits de stop peut être 1 ou 2.

XIX. FICHE ERREUR

Nous avons besoin de vos commentaires et suggestions pour améliorer la qualité et la facilité d'utilisation de nos documentations.

Nous vous serions très reconnaissants de remplir cette fiche d'appréciation, et de nous la retourner par courrier ou télécopie. Nous vous en remercions par avance.

**Destinataire : ACKSYS,
Département R&D
N° de télécopie : (33) 01 39 11 47 96**

Votre société	Tél
Personne à contacter	Fonction

Indiquez clairement la version de la carte, la version des logiciels et de la documentation dont vous parlez :

Nom de la carte	
Système d'exploitation	
Documentation révision	B-02 du 12.11.1998

Vos remarques :
