



DATA I/O

## PROGRAMMATEUR D'EPROM

MODELE 201

### GUIDE D'UTILISATION

## I. MISE EN OEUVRE

### 1.1 Généralités.

Pour programmer une mémoire dans ce mode d'utilisation, l'opérateur se sert du clavier et de l'écran d'affichage en suivant une procédure simple:

- a) Une phase d'initialisation qui comprend la séquence d'auto-test du programmeur 201.
- b) Une phase de chargement dans la mémoire interne du 201 des données destinées au composant à programmer. Données provenant soit d'une mémoire mère (duplication), soit d'un ordinateur ou d'un terminal à distance.
- c) Une phase de recopie dans la mémoire vierge suivie d'une vérification de la conformité des données transférées.

Lorsque les données proviennent d'un ordinateur ou d'un terminal, elles sont transmises au 201 via le port série RS-232-C. La phase b commencera alors par l'établissement des paramètres de transmission, au lieu de la sélection de l'algorithme correspondant à la mémoire mère.

Lorsque les données sont dans la mémoire du 201, il est possible d'y apporter des modifications en utilisant les fonctions d'édition (EDIT).

Les données présentes dans la mémoire du 201 peuvent être transmises à un ordinateur via la liaison série RS232.

La plupart des opérations en cours peuvent être stoppées et abandonnées à tout moment en appuyant sur une des touches alphanumériques.

Si, au cours de la procédure, le programmeur rencontre un défaut, il stoppe et affiche un message d'erreur (voir à la fin de ce guide).

## 1.2 Initialisation.

A la mise en marche, le programmeur effectue automatiquement un auto-test. A condition toutefois que le support du composant soit vide, sinon l'écran indique:

```
DEVICE IN SOCKET  
REMOVE DEVICE
```

Il faut, dans ce cas, enlever le composant puis appuyer sur ENTER pour lancer l'auto-test. Durant l'auto-test, l'écran affiche:

```
SELF TESTING
```

Quand c'est terminé, il indique pendant quelques instants:

```
SELF TEST OK  
DATA I/O 201 N
```

Où N est le numéro de la version du logiciel interne. Ensuite, apparaît le message:

```
LOAD FROM MASTER
```

Ce qui signifie: Chargement depuis une Mémoire Mère.

### NOTA:

Cette fonction est la première des six fonctions du menu principal. A savoir:

LOAD FROM MASTER	= chargement depuis une mémoire mère
PROGRAM	= recopie dans un composant vierge
VERIFY	= vérification des contenus
BLANK CHECK	= vérification de la virginité
RS232	= utilisation du port série
EDIT	= édition (pour faire des modifications dans les données)

## 1.3 Chargement depuis une mémoire mère.

- 1) Mettre l'appareil sous tension (marche).
- 2) Attendre la fin de l'auto-test.
- 3) L'écran affiche: LOAD FROM MASTER.
- 4) Appuyer sur ENTER. L'écran affiche: F/P CODE ...  
Ce qui signifie type et brochage du composant.  
Il y a trois manières de répondre:

a. Le composant mère contient un identificateur électronique. Taper FFFF et appuyer sur ENTER. L'algorithme correspondant à ce composant sera automatiquement sélectionné.

- b. Le composant n'a pas d'identificateur incorporé. Appuyer sur une touche de défilement jusqu'à l'apparition du nom du fabricant (ex. INTEL). Appuyer sur ENTER. Appuyer à nouveau sur une touche de défilement pour faire apparaître le type du composant (ex. INTEL 2732). Appuyer sur ENTER.
- c. Taper au clavier le code de ce composant (voir au préalable la Device List). Puis appuyer sur ENTER.

- 5) Insérer le composant mère sur le support et abaisser le levier de blocage.
- 6) Appuyer sur ENTER, le contenu du composant mère se charge dans la RAM interne du 201.
- 7) Après le bip sonore, retirer le composant mère.

Si le chargement s'est correctement effectué, le voyant vert s'allume et une valeur s'inscrit sur l'écran (ex. SUMCHECK = BA25). Prendre note de ce total, il servira de référence pour vérifier une programmation ultérieure.

S'il y a problème, le voyant rouge s'allume accompagné d'un message d'erreur.

#### 1.4 Programmation d'un composant.

- 1) Appuyer sur une touche de défilement jusqu'à l'apparition de la fonction PROGRAM.
- 2) Appuyer sur ENTER. L'écran affiche: F/P CODE...
- 3) Reprendre les instructions du point 4 de la procédure de chargement depuis une mémoire mère.
- 4) Insérer un composant mémoire vierge(\*) sur le support et abaisser le levier de blocage.
- 5) Appuyer sur ENTER, les données chargées dans la RAM sont copiées dans le composant vierge.
- 6) Après le bip sonore, la programmation est terminée. Un total de vérification s'inscrit sur l'écran. S'assurer qu'il est identique à celui noté à la fin du chargement dans la RAM.

(\*) Pour vérifier la virginité d'un composant, procéder comme suit:

- a) Faire défiler les fonctions jusqu'à BLANK CHECK. Appuyer sur ENTER. Si le F/P CODE qui apparaît sur l'écran est incorrect, appliquer les instructions du point 4 de la procédure du paragraphe 1.3.
- b) Insérer le composant sur le support puis appuyer sur ENTER. A la fin de la vérification, l'écran indique BLANK CHECK OK si le composant est vierge. Sinon c'est le message NONBLANK accompagné du voyant rouge qui survient.

#### 1.5 Chargement par le port série.

- 1) Relier le programmeur 201 à l'ordinateur.
- 2) Etablir les paramètres de la communication comme suit:  
(A titre d'exemple, nous supposons que les paramètres requis sont: vitesse=9600 baud; bit de parité=aucun; nombre de bits de donnée=8; nombre de bits d'arrêt=1)

- a. Faire défiler les fonctions jusqu'à RS232 PORT
  - b. Appuyer sur ENTER puis faire défiler jusqu'à PORT SETTING.
  - c. Appuyer sur ENTER puis faire défiler jusqu'à BAUD RATE = 9600.
  - d. Appuyer sur ENTER puis faire défiler jusqu'à PARITY CHK.= NONE.
  - e. Appuyer sur ENTER puis faire défiler jusqu'à DATA BITS = 8.
  - f. Appuyer sur ENTER puis faire défiler jusqu'à STOP BIT = 1.
  - g. Appuyer sur ENTER.
- 3) Revenir à la fonction RS232 et appuyer sur ENTER.
  - 4) Faire défiler jusqu'à DOWNLOAD puis appuyer sur ENTER.
  - 5) Faire défiler les formats jusqu'à celui qui convient. Les formats de transfert de données acceptés par le programmeur sont: Intel Intelec 8/MDS, Intel MCS-86 Hexadecimal Object, Motorola Exorciser, Motorola Exomax et Tektronix Hexadecimal.
  - 6) Appuyer sur ENTER.
  - 7) Taper l'adresse de début dans la RAM. (Si l'on tape par exemple 1000H, la première donnée transférée se placera à cette adresse et non en 0000H). Par défaut, l'écran affiche: BEG RAM ADD=0000. Appuyer ensuite sur ENTER.
  - 8) Taper la taille du bloc de données à charger: BLCK SIZE = ..., puis appuyer sur ENTER.
  - 9) Taper l'adresse de début des données dans le fichier transmis par le calculateur: OFFSET = ...., et appuyer sur ENTER. Le terminal affiche: LOADING PORT, le chargement peut commencer.

## 1.6 Vérification des contenus.

Cette opération consiste à comparer les données programmées dans le composant à celles contenues dans la RAM du programmeur.

- 1) Appuyer sur une touche de défilement jusqu'à l'apparition de la fonction VERIFY.
- 2) Appuyer sur ENTER, l'écran indique un F/P CODE. S'il n'est pas correct, suivre les instructions du point 4 du chargement depuis une mémoire mère au paragraphe 1.3.
- 3) Appuyer sur ENTER puis insérer le composant sur le support et appuyer à nouveau sur ENTER pour démarrer la vérification.

Si celle-ci se termine avec succès, le voyant vert s'allume et l'écran affiche le total de vérification (SUMCHECK = ...). Sinon il y a un message d'erreur (VERIFY ERROR NN).

## 1.7 Emission par le port série.

Pour envoyer les données de la RAM à un calculateur, suivre une procédure voisine de celle du chargement de la RAM via le port série. Les paramètres de communication, le format, l'adresse de début, la taille du bloc et l'adresse dans la mémoire du calculateur sont à programmer de la même façon (points 1 à 9 du paragraphe 1.5, excepté qu'au point 4, la fonction se nomme ici UPLOAD au lieu de DOWNLOAD).

Lorsque l'émission est terminée, le programmeur affiche:  
SENDING COMPLETE SUMCHECK = HHHH (où HHHH désigne le total de  
vérification en hexadécimal).

### 1.8 Edition des données.

Les fonctions d'édition servent à effectuer des modifications dans les données chargées dans la RAM du programmeur. Ces fonctions sont:

- SPLIT - Division d'un bloc de 16 bits en 2 blocs adjacents de 8 bits.
- SHUFFLE - Opération inverse de la précédente, regroupement de deux blocs de 8 bits en un seul de 16 bits.
- COMPLEMENT - Inversion de données par rapport à FF.
- INSERT - Insertion de données entre des adresses consécutives.
- BLOCK INSERT - Insertion d'un bloc de données.
- DELETE - Effacement de données octet par octet.
- BLOCK DELETE - Effacement d'un bloc de données.
- BLOCK FILL - Chargement d'une valeur unique dans un groupe d'adresses.
- DATA SEARCH - Recherche de données (jusqu'à 8 octets).
- BLOCK MOVE - Recopie d'un bloc de données d'une adresse à une autre dans la RAM.
- DATA EDIT - Edition du contenu de la RAM adresse par adresse.

#### 1.81 Division (SPLIT):

- 1) Appuyer sur une touche de défilement jusqu'à l'apparition de la fonction EDIT. Appuyer ensuite sur ENTER.
- 2) Si nécessaire, faire défiler les fonctions d'édition jusqu'à SPLIT et appuyer sur ENTER.
- 3) Taper au clavier l'adresse médiane autour de laquelle se divisera en deux le bloc de 16 bits (ex.: MIDDLE ADR.=1000). Par défaut, ce point est le milieu de la RAM (8000H).
- 4) Appuyer sur ENTER pour exécuter la commande. Lorsque la répartition est terminée, l'écran affiche: SPLIT COMPLETED.

#### 1.82 Groupement (SHUFFLE):

Pour souder deux octets en un bloc de 16 bits, suivre la même procédure que pour la répartition (SPLIT). L'adresse médiane sera celle du milieu du nouveau bloc, et le mot SHUFFLE remplacera SPLIT sur l'écran.

#### 1.83 Complément:

L'inversion logique d'un groupe de données se fait sur la base de mot de 8 bits. Le 0 devient F et F devient 0, etc.

- 1) Faire défiler jusqu'à la fonction EDIT. Appuyer sur ENTER.
- 2) Faire défiler à nouveau pour obtenir la fonction COMPLEMENT. Appuyer sur ENTER.
- 3) Taper au clavier l'adresse de la première donnée à inverser: START ADR.= ..., puis appuyer sur ENTER.
- 4) Taper au clavier l'adresse de la dernière donnée à inverser: END ADR.= ..., puis appuyer sur ENTER. L'opération commence. Lorsque c'est terminé, l'écran affiche: COMPLEMENT COMPLETED.

### 1.84 Insertion:

Pour insérer une donnée à une adresse ou dans un groupe d'adresses adjacentes, suivre la procédure ci-dessous. Il est à noter que la donnée, ou les données, précédentes sont décalées et que celle(s) situées à l'extrémité supérieure de la RAM sont perdues.

- 1) Faire défiler jusqu'à la fonction EDIT, puis appuyer sur ENTER.
- 2) Chercher à l'aide d'une touche de défilement la fonction INSERT ou BLOCK INSERT selon le cas. Appuyer sur ENTER.
- 3) Dans le cas de INSERT:  
Taper au clavier l'adresse où l'on désire insérer la donnée:  
INSERT ADDRESS = ...  
Dans le cas de BLOCK INSERT:  
Taper au clavier l'adresse du début de l'insertion (BLOCK INSERT START ADR. = ...) puis celle de fin (END ADR. = ...), en appuyant sur ENTER après chaque entrée.
- 4) Taper la valeur que l'on désire insérer (DATA = ...).
- 5) Appuyer sur ENTER pour exécuter l'instruction.

### 1.85 Effacement:

Lorsque l'on efface dans la RAM une donnée ou un groupe de données, les suivantes se décalent pour remplir les adresses vidées et la valeur FF est mise dans la ou les dernières adresses de la RAM, adresses libérées à la suite de l'effacement.

- 1) Faire défiler jusqu'à la fonction EDIT, puis appuyer sur ENTER.
- 2) Chercher ensuite la fonction DELETE ou BLOCK DELETE selon le cas. Appuyer sur ENTER.
- 3) Dans le cas de DELETE:  
Taper l'adresse de la donnée à effacer.  
Dans le cas de BLOCK DELETE:  
Taper les adresses de début puis de fin du bloc à effacer.
- 4) Appuyer sur ENTER pour exécuter l'instruction.

### 1.86 Chargement d'une valeur dans un groupe d'adresses:

Cette opération consiste à remplacer les données contenues dans un groupe d'adresses par une donnée unique.

- 1) Choisir la fonction EDIT, puis appuyer sur ENTER.
- 2) Chercher la fonction BLOCK FILL et appuyer sur ENTER.
- 3) Taper l'adresse de début du bloc, puis celle de fin en appuyant sur ENTER après chaque entrée.
- 4) Taper la valeur de la donnée de remplacement et appuyer sur ENTER.

### 1.87 Recherche de données:

- 1) Chercher la fonction EDIT, puis appuyer sur ENTER.
- 2) Sélectionner la fonction DATA SEARCH. Appuyer sur ENTER.
- 3) Taper l'adresse de l'emplacement de début de recherche et appuyer sur ENTER.
- 4) Taper l'adresse de la fin de recherche et appuyer sur ENTER.
- 5) Taper la valeur de la donnée, ou des données à rechercher.

Nota: Jusqu'à 8 octets peuvent être recherchés; la virgule est placée par le programmeur.

Exemples: DATA = 01,

DATA = 01,02,03,04,05,06,07,08

6) Appuyer sur ENTER pour déclencher la recherche.

Quand la donnée est trouvée, le programmeur affiche l'adresse de la première rencontre. Appuyer sur ENTER pour reprendre la recherche.

Si la donnée n'est pas rencontrée, l'écran indique: DATA SEARCH FAIL.

#### 1.88 Recopie de données d'une adresse à une autre:

Dans cette opération, les données copiées remplacent celles d'origine dans les adresses spécifiées. Il est recommandé de s'assurer à l'avance que la taille du bloc de données à copier à partir d'une première adresse de destination n'excède pas la capacité restante de la RAM.

- 1) Sélectionner la fonction EDIT. Appuyer sur ENTER.
- 2) Sélectionner la fonction BLOCK MOVE. Appuyer sur ENTER.
- 3) Taper l'adresse du début du bloc. Appuyer sur ENTER.
- 4) Taper l'adresse de la dernière donnée du bloc. Appuyer sur ENTER.
- 5) Taper la première adresse de destination. (Le reste du bloc sera copié dans les adresses suivantes)
- 6) Appuyer sur ENTER pour exécuter la copie.

#### 1.89 Edition individuelle de donnée:

Ce mode permet d'éditer et de visualiser individuellement chaque donnée contenue dans la RAM pour la modifier.

- 1) Sélectionner la fonction EDIT. Appuyer sur ENTER.
- 2) Sélectionner la fonction DATA EDIT. Appuyer sur ENTER.
- 3) Taper l'adresse de la première donnée à éditer, puis appuyer sur ENTER. La valeur actuelle de la donnée s'affiche sur la deuxième ligne de l'écran (ex.: DATA 05-->\_ \_).
- 4) Pour changer la donnée, taper la nouvelle valeur puis appuyer sur ENTER. (ex.: DATA 05--> 9C).

Si aucune nouvelle valeur n'est entrée, la donnée reste inchangée après avoir appuyé sur ENTER.

Si la touche ENTER n'est pas actionnée après avoir tapé la nouvelle valeur, celle-ci sera ignorée.



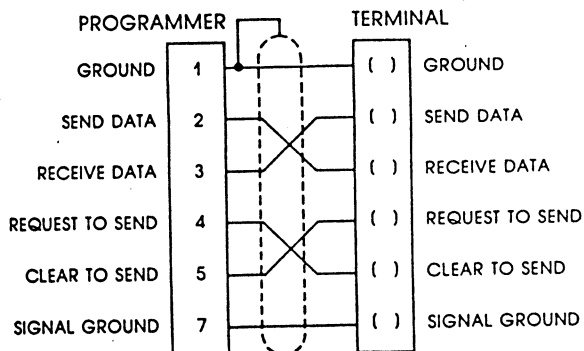
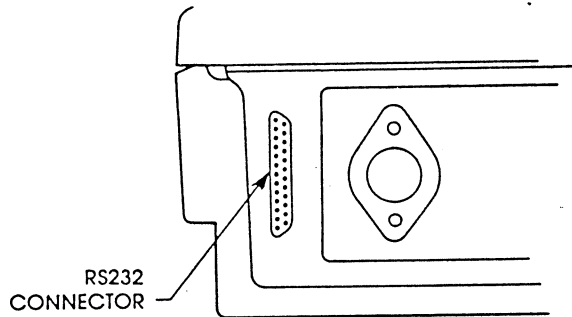
## II. COMMANDE A DISTANCE PAR TERMINAL

### 2.1 Etablissement de la liaison série.

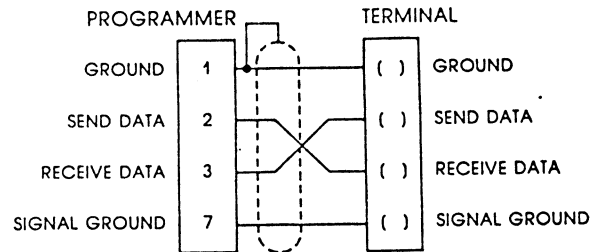
Il est recommandé d'utiliser un câble blindé pour diminuer les effets des interférences électromagnétiques éventuelles. La communication entre le programmeur et le terminal (ou l'ordinateur) pouvant comporter ou non un protocole d'échange (modes "handshake" ou "non-handshake"), le brochage des connecteurs se fera en conséquence. Voir les schémas ci-dessous.

#### NOTE

*For continued compliance with class B limits of VDE 0871, use a properly shielded interconnect cable and peripheral equipment complying with VDE 0871 class B.*



Half/Full Duplex, With Handshake



Half/Full Duplex, Without Handshake

#### NOTE

*All signals are named with respect to the originating unit.*

Après avoir réalisé la liaison matérielle, il faut établir les paramètres de la communication comme indiqué au paragraphe 1.6. Par défaut, le programmeur prend les valeurs suivantes: vitesse 9600 bauds, pas de parité, 8 bits de donnée et 1 bit d'arrêt.

## 2.2 Généralités.

### a) Règles d'entrée des commandes.

- Chaque commande doit être entrée à la suite du caractère > sur l'écran, et être suivie d'un retour chariot (Return).
- Toutes les commandes doivent être tapées en majuscule.
- Le paramètre qui suit une lettre, ou un groupe de lettres, de commande peut être ou non précédé par un espace. Ainsi l'on peut taper LS4 ou LS 4 indifféremment.
- Les paramètres qui suivent le premier doivent être précédés d'un espace. Par exemple, la commande d'affichage de la mémoire doit être tapée M O F ou MO F..
- Il n'est pas nécessaire de taper les zéros commençant un paramètre numérique telle qu'une adresse. Par exemple 0010 a le même effet que 10.
- Tous les paramètres sont à entrer en hexadécimal, sauf indication contraire.

### b) Touches d'exécution.

- BACKSPACE: Cette touche sert à effacer le ou les caractères que l'on vient d'entrer. Si le clavier ne comporte pas cette touche, utiliser Ctrl-H.
- ESCAPE : Sert à annuler une commande en cours d'exécution ou une ligne en cours de frappe. Dans les deux cas, le signe > est affiché par l'écran qui attend une nouvelle entrée.
- CTRL-S : Permet de suspendre temporairement l'affichage des données sur l'écran.
- CTRL-Q : Permet de reprendre l'affichage des données suspendu par CTRL-S.

### e) Indications d'erreur.

L'expression SYNTAX ERROR apparaît sur l'écran quand le format donné à une commande est incorrect ou que des caractères illégaux sont inclus dans une commande. Le caractère ^ s'affiche à côté du premier caractère inconnu.

Le voyant lumineux placé sous le support indique le résultat des opérations. Normalement, il s'allume en vert et est accompagné de deux beeps sonores. Mais si l'opération est défectueuse, le voyant est rouge et trois beeps se font entendre.

#### f) Valeurs par défaut.

Lorsque les paramètres accompagnant une commande ne sont pas spécifiés, ils prennent par défaut les valeurs suivantes:

<start address>	(adresse de départ)	0000H
<end address>	(adresse de fin)	1FFFH
<destination>		0000H
<number of device>	(nombre de mémoires mères)	
	pour un jeu	1
	pour un mot de 16 bits	2
	pour un mot de 32 bits	4

### 2.3 Procédures de commande à distance.

La suite de ce chapitre traite de la commande à distance par terminal des opérations suivantes:

- Entrée et sortie du mode commande à distance par terminal.
- Menu d'assistance.
- Sélection du type de composant.
- Opérations de programmation.
- Vérification des contenus.
- Edition du contenu de la mémoire du programmeur.
- Calcul du total hexadécimal des données (Sumcheck) ou somme des OU exclusifs (Checksum ou EXOR).

### 2.4 Entrée et sortie du mode commande à distance par terminal.

Pour entrer dans ce mode:

- 1) Relier le programmeur au terminal comme indiqué en 2.1.
- 2) Mettre les appareils sous tension (ON = marche).
- 3) Lorsque l'auto-test (SELF TESTING) est terminé, faire défiler les fonctions à l'aide d'une des touches de défilement du programmeur jusqu'à l'apparition de RS232 PORT. Appuyer sur ENTER.
- 4) Faire défiler à nouveau jusqu'à TERMINAL CONTROL. Appuyer sur ENTER.
- 5) Sur le clavier du terminal, taper la commande M <CR>. (<CR> est le symbole de la touche de retour chariot "Return").

Pour sortir du mode:

Taper R<CR> sur le clavier du terminal ou appuyer sur une des touches du programmeur. Celui-ci affichera RS232 PORT.

## 2.5 Menu d'aide.

En réponse à la commande ?<CR> tapée au clavier du terminal, le tableau ci-dessous s'affiche sur l'écran. Il dresse la liste des commandes et indique leur format.

### \*\*\* 201 PROGRAMMER COMMAND HELP \*\*\*

SELECT DEVICE TYPE A?		\	SUMCHECK (TOTAL)	CT	A1	A2	{	A3	}
BLANK CHECK	B	\	CHECKSUM (EXOR)	CX	A1	A2	{	A3	}
LOAD SINGLE	L	[	MEMORY DISPLAY	M	A1	A2			
LOAD SET	LS	[	MEMORY MODIFY	M	A1				
LOAD WORD	LW	[	MEMORY FILL	M	A1	A2	D1	[..D8]	
LOAD LONG WORD	LL	[	INSERT	I	A1	D1	[..D8]		
PROGRAM SINGLE	P	[	DELETE	D	A1	[	A2		
PROGRAM SET	PS	[	TRANSFER	T	A1	A2	A3		
PROGRAM WORD	PW	[	MEMORY SEARCH	MS	A1	A2	D1	[..D8]	
PROGRAM LONG WORD	PL	[	RETURN TO LOCAL	R					
VERIFY SINGLE	V	[	HELP	H					
VERIFY SET	VS	[							
VERIFY WORD	VW	[							
VERIFY LONG WORD	VL	[							

DEVICE TYPE: INTEL 2764      F/P: 79/33      MASTER #: 01

A1 = adresse de départ      D1..D8 = Mots de données  
A2 = adresse de fin      # = Nombre de mémoires mères  
A3 = adresse de destination      H/L = Choix de Vcc (H ou L)

## 2.6 Sélection du type de composant.

- 1) Taper la commande A? <CR>.
- 2) La liste des fabricants apparaît sur l'écran. Taper le chiffre ou le nombre correspondant à celui désiré. Exemple: 7 <CR>.
- 3) La liste des composants de ce fabricant s'inscrit sur l'écran. Taper le chiffre ou le nombre désiré. Exemple: 1 <CR>.

Le programmeur détermine alors automatiquement l'algorithme et le brochage corrects.

Par défaut, le programmeur prend ceux du composant mémoire INTEL 2764.

## 2.7 Opérations de programmation.

Sous ce nom générique sont regroupés les trois types d'opération suivants: vérification de la virginité du ou des composants mémoires à programmer, chargement des données dans la RAM du programmeur et recopie de ces données dans la ou les mémoires vierges.

### a) Vérification de la virginité.

- 1) Après en avoir sélectionné le type comme expliqué en 2.6, insérer le composant dans le support et abaisser le levier de blocage.
- 2) Taper la commande B <CR>.
- 3) Si la mémoire est vierge, le terminal affichera

```
FUNCTION COMPLETED  
BLANK CHECK OK
```

Sinon, un message d'erreur remplacera la seconde ligne et le voyant du support sera rouge.

- 4) Relever le levier et enlever la mémoire.

### b) Chargement dans la RAM.

Les données à charger dans la mémoire du programmeur proviennent soit d'une seule mémoire mère, soit de jeux de mémoires mères. Dans le cas d'une seule mémoire, le chargement se fait par mots de 8 bits. Dans l'autre cas, les mots peuvent être de 8, 16 ou 32 bits. Les mémoires composant un jeu doivent être du même type et la sélection de ce type (voir en 2.6) être effectuée avant toute opération de chargement (et de recopie).

Pour charger depuis une seule mémoire mère:

- 1) Après avoir sélectionné son type, insérer la mémoire mère dans le support et abaisser le levier.
- 2) Taper la commande L.
- 3) Taper l'adresse du début du chargement dans la RAM si celle-ci n'est pas l'adresse 0, auquel cas passer au point 6.
- 4) Taper un espace puis l'adresse de fin.
- 5) Taper un espace puis l'adresse de la mémoire mère où se trouve la première donnée à charger.
- 6) Appuyer sur <CR> pour exécuter la commande.  
Quand le chargement est complet, le terminal affiche

```
FUNCTION COMPLETED  
THE SUMCHECK (TOTAL) IS: HHHH
```

HHHH est le total hexadécimal des données chargées dans la RAM. Noter cette valeur, elle servira de référence pour vérifier une programmation ultérieure.

- 7) Relever le levier et enlever la mémoire mère.

• Pour charger depuis un jeu par mots de 8 bits:

- 1) Sélectionner le type des mémoires.
- 2) Taper la commande LS suivie du nombre de mémoires composant le jeu. Le tableau ci-après indique les adresses dans la RAM assignées à chaque mémoire.
- 3) Appuyer sur <CR>. Le terminal affiche

INSERT DEVICE NO: 01

- 4) Insérer le composant mère numéro 1 puis appuyer sur <CR>. Quand le contenu sera chargé, le terminal affichera

FUNCTION COMPLETED  
INSERT DEVICE NO: NN

NN est le numéro de la mémoire suivante.

- 5) Lorsque toutes ont été chargées, noter le total hexadécimal SUMCHECK pour pouvoir vérifier une programmation ultérieure.

Numéro du composant	Capacité en K bits.					
	16K	32K	64K	128K	256K	512K
1	0000 07FF	0000 0FFF	0000 1FFF	0000 3FFF	0000 7FFF	0000 FFFF
2	0800 0FFF	1000 1FFF	2000 3FFF	4000 7FFF	8000 FFFF	
3	1000 17FF	2000 2FFF	4000 5FFF	8000 BFFF		
4	1800 1FFF	3000 3FFF	6000 7FFF	C000 FFFF		
	.	.	.			
	.	.	.			
8	3800 3FFF	7000 7FFF	E000 FFFF			
	.	.	.			
	.	.	.			
16	7800 7FFF	F000 FFFF				
	.	.				
	.	.				
32	F800 FFFF					

### Pour charger des mots de 16 bits:

Chaque mot est scindé en deux octets selon le poids des bits. Les mémoires sont groupées par deux, l'une d'elle, dite la première, contient les octets de plus faible poids. Ils seront chargés, à partir de l'adresse 0, dans les adresses paires de la RAM. (Voir page 21).

- 1) Sélectionner le type des composants.
- 2) Taper la commande LW suivie du nombre de mémoires mères (nombre obligatoirement pair). Exemple: LW 4<CR>. Le terminal affichera

INSERT DEVICE NO: 01

- 3) Insérer la mémoire contenant les octets de faible poids puis appuyer sur <CR>. Quand le contenu sera chargé, le terminal affichera

FUNCTION COMPLETED  
INSERT DEVICE NO: NN

NN est le numéro de la mémoire suivante.

- 4) Insérer la seconde mémoire de la paire et appuyer sur <CR>.
- 5) Répéter les points 3 et 4 pour toutes les paires du jeu.
- 6) A la fin du chargement, noter le total hexadécimal pour pouvoir vérifier une programmation ultérieure.

NOTA: La somme totale des capacités des mémoires mères ne peut dépasser la capacité de la RAM du programmeur. Prendre garde à respecter cette limitation.

### Pour charger des mots de 32 bits:

Les mots sont scindés en quatre octets selon le poids des bits. Les octets de même poids sont dans une même mémoire. Les mémoires sont groupées par quatre pour composer les mots de 32 bits. Chaque mémoire d'un groupe est numérotée de un à quatre selon le poids de ses octets. Le chargement se fera dans le même ordre, c'est à dire à partir de l'octet de plus faible poids. (Voir page 21).

- 1) Sélectionner le type des composants.
- 2) Taper la commande LL suivie du nombre de mémoires mères (nombre obligatoirement multiple de quatre). Exemple: LL 8<CR>. Le terminal affichera

INSERT DEVICE NO: 01

- 3) Insérer la mémoire contenant les octets de plus faible poids puis appuyer sur <CR>. Quand le contenu sera chargé, le terminal affichera

FUNCTION COMPLETED  
INSERT DEVICE NO: NN

NN est le numéro de la mémoire suivante.

- 4) Insérer la seconde mémoire du groupe et appuyer sur <CR>.
- 5) Répéter les points 3 et 4 pour les autres mémoires du groupe, puis pour les groupes suivants du jeu.
- 6) A la fin du chargement, noter le total hexadécimal pour pouvoir vérifier une programmation ultérieure.

NOTA: La somme totale des capacités des mémoires mères ne peut dépasser la capacité de la RAM du programmeur. Prendre garde à respecter cette limitation.

c) Programmation de composant(s) vierge(s).

Après avoir sélectionné le type des mémoires, vérifié leur virginité et chargé les données dans la RAM du programmeur, la programmation proprement dite peut commencer. La procédure à suivre dépend de la façon dont les données ont été chargées: depuis une seule mémoire mère, depuis un jeu, par mot de 8, 16 ou 32 bits, etc. A la fin de chaque programmation, un total hexadécimal (SUMCHECK) sera affiché afin de comparer à celui indiqué après le chargement.

- 1) Lorsque le chargement a été fait depuis une seule mémoire mère:

- Insérer la mémoire vierge dans le support.
- Taper la lettre de commande P.
- Taper l'adresse de la RAM où commencent les données à recopier.
- Taper un espace puis l'adresse de la dernière donnée à copier.
- Taper un espace puis l'adresse de destination, c'est à dire celle où doit commencer la copie dans le ou les composants vierges.

NOTA: Ces trois adresses ne sont pas nécessaires si l'adresse de départ et celle(s) de destination sont les adresses 0.

- Appuyer sur la touche (CR).
- A la fin de la programmation, le total hexadécimal SUMCHECK doit être égal à celui noté après le chargement.

- 2) Lorsque les données proviennent de plus d'une mémoire mère, les mémoires vierges devront être insérées successivement comme l'avaient été les mémoires mères lors du chargement. Leur contenu suivra la répartition des données indiquées au tableau de la page 10.

- Taper la commande PS puis le nombre de mémoires.
- Appuyer sur <CR> et insérer les mémoires au fur et à mesure que le terminal le demande.

- 3) Dans le cas d'un chargement de mots de 16 bits par des mémoires mères appariées, les mémoires vierges à programmer doivent aussi être groupées par deux et obéir aux mêmes règles d'insertion.

- Taper la commande PW.
- Taper le nombre de mémoires puis appuyer sur <CR>.
- Insérer la mémoire vierge destinée à recevoir les octets de faible poids et appuyer sur <CR>.
- Le terminal indique PROGRAMMING pendant l'opération.
- Insérer la seconde mémoire de la première paire lorsque le terminal le demande et appuyer sur <CR>.
- Insérer ensuite dans le même ordre les mémoires des paires suivantes au fur et à mesure de la demande du terminal.



**Cablage RS232C programmeur DATA I/O <-----> PC**

Coté Programmeur				Coté Pc	
Signaux	DB25 Male			DB25 Femelle	Signaux
<b>Masse</b>	<b>1</b>	<-----	----->	<b>1</b>	<b>Masse</b>
<b>TX</b>	<b>2</b>	<-----	----->	<b>3</b>	<b>RX</b>
<b>RX</b>	<b>3</b>	<-----	----->	<b>2</b>	<b>TX</b>
<b>RTS</b>	<b>4</b>	<-----	----->	<b>5</b>	<b>CTS</b>
<b>CTS</b>	<b>5</b>	<-----	----->	<b>4</b>	<b>RTS</b>
<b>GND</b>	<b>7</b>	<-----	----->	<b>7</b>	<b>GND</b>
				<b>20 ---- 6</b>	<b>DTR + DSR</b>

Coté Programmeur				Coté Pc	
Signaux	DB25 Male			DB9 Femelle	Signaux
<b>Masse</b>	<b>1</b>	<-----	----->	<b>MM</b>	<b>Masse</b>
<b>TX</b>	<b>2</b>	<-----	----->	<b>2</b>	<b>RX</b>
<b>RX</b>	<b>3</b>	<-----	----->	<b>3</b>	<b>TX</b>
<b>RTS</b>	<b>4</b>	<-----	----->	<b>8</b>	<b>CTS</b>
<b>CTS</b>	<b>5</b>	<-----	----->	<b>7</b>	<b>RTS</b>
<b>GND</b>	<b>7</b>	<-----	----->	<b>5</b>	<b>GND</b>
				<b>4 ---- 6</b>	<b>DTR + DSR</b>

# Procédure de transfert entre PC et Data I/O

## Paramètre de hyperterminal

Lancer Hyperterminal.

Aller dans fichier ---> Propriété ---> Se connecter en utilisant : COM1 ---> Configurer :  
9600 , 8 , aucun , 1 , Xon/Xoff ---> OK ---> OK.

## Pour envoyer un fichier vers le programmeur

Avant toute chose il faut que le programmeur soit configuré prêt à recevoir

« **DOWLOAD** ».

Aller dans transfert ---> envoyer un fichier texte ---> choisir le fichier à envoyer (Fichier de type Motorola renommé avec attribut .txt) le fichier doit avoir le format de ce type  
« **27C64-Matra\_Mot.txt** » ---> Ouvrir et le transfert vers la RAM du programmeur démarre si le programmeur et en configuration de Recevoir.

## Pour recevoir un fichier venant de la RAM du programmeur.

Aller dans transfert ---> Capturer un fichier Text ---> Donner un nom au fichier à recevoir du genre « **27C64-Matra\_Mot.txt** ---> Démarrer.

Configurer le programmeur sur « **Upload** ».

Une fois le transfert terminé aller dans transfert ---> Capturer le texte ---> Arrêter.

# 5. Error Messages

The following is a list of error codes and their corresponding messages. The circumstances which cause an error message to be displayed are described in the Description column and corrective action to take upon receiving the error message is explained in the right-hand column. The error codes are listed in numerical order.

<b>Code</b>	<b>Name</b>	<b>Description</b>	<b>Corrective Action</b>
17	BANK ERROR	The bank selector circuit is unable to switch banks.	Contact your local Data I/O Service Center.
20	NONBLANK DEVICE	Device failed blank test.	Erase the device.
21	ILLEGAL BIT	Unable to program device due to already programmed bit of incorrect polarity.	Erase the device if possible or discard it.
22	PROGRAM FAIL	The device failed to program properly.	Replace device if faulty.
23	VERIFY ERROR	The programmed PROM data failed to verify against the master data in RAM.	Reprogram the device, if possible, or try another device.
27	BLOCK LIMIT ERROR	PROM or RAM size is insufficient to perform the current operation using the begin RAM address, block size, or set size specified.	Make sure the begin RAM address, block size, and set size are set correctly. You may need to have your RAM serviced.

## Error Messages

Code	Name	Description	Corrective Action
31	OVERCURRENT	The device to be programmed is drawing excessive current.	Indicates that the device is faulty. Replace the device.
32	VCC ERROR	When 5 V Vcc is applied to pin 26/28, the voltage drops to an unacceptable level.	Replace device if faulty.
33	VPP ERROR	The programming power supplies cannot be set at the proper levels.	Replace device if faulty.
34	BAD DEVICE CODE	An incorrect family and pinout code was entered.	Consult the Device List for the correct family and pinout codes and enter the correct code.
35	NOT TRISTATE™	Device failed to TRISTATE™ all data pins.	Indicates that the device is faulty. Replace the device.
36	DEVICE NOT ENABLED	All the device's data pins are not driving when the chip is enabled.	Indicates that the device is faulty. Replace the device.
37	BAD INSERTION	The device was inserted in the socket incorrectly.	Align the bottom-most pin of the device with the bottom of the socket. Pin 1 should be at the top of the device.

# Error Messages

Code	Name	Description	Corrective Action
41	FRAME ERROR	The serial interface detected a start bit but the stop bit was in the wrong position.	Check the current baud rate and stop bit settings.
42	OVERRUN ERROR	The serial interface received characters when the programmer was unable to accept them.	Check the serial port connections. Make sure handshake lines are properly connected and try again.
46	I/O TIMEOUT	No characters, or only nulls and rubouts, were received upon serial input for 25 seconds after pressing the ENTER key; or, no characters could be transmitted for a period of 25 seconds due to the state of the handshake lines.	Check all connections and attempt transmission again.
48	BUFFER FULL	The serial port input buffer received more characters than the programmer was able to process.	Make sure the handshake lines are hooked up and operative.
52	I/O VERIFY ERROR	The data from the serial port does not match the data in RAM.	Reload data to RAM. If the problem persists, service the programmer or contact your local Data I/O Service Center.

## Error Messages

Code	Name	Description	Corrective Action
64	DRAM ERROR	The self-test shows a dynamic RAM read/write verification error.	Contact your local Data I/O Service Center.
67	I/O FORMAT ERROR	Data sent to the programmer are non-hexadecimal characters. Input data is in the incorrect format.	Specify the correct data translation format for the data being transferred, or correct the data in the file.
71	PIN 1 ERROR	The self-test indicates that the socket pin 1 driver is at an inaccurate level.	Contact your local Data I/O Service Center.
72	ADDRESS SHORT	The device contains a short on its address line.	Indicates that the device is faulty. Replace the device.
73	DATA SHORT	The device contains a short on its data line.	Indicates that the device is faulty. Replace the device.
74	DATA BUS FAIL	Indicates that the part data bus may be damaged or the contact to ZIF socket is faulty.	Replace the faulty device or contact your local Data I/O Service Center.
75	PIN 22 ERROR	The self-test indicates that the socket pin 22 driver is at an inaccurate level.	Contact your local Data I/O Service Center.

# Error Messages

Code	Name	Description	Corrective Action
76	PIN 23 ERROR	The self-test indicates that the socket pin 23 driver is at an inaccurate level.	Contact your local Data I/O Service Center.
77	PIN 26 ERROR	The self-test indicates that the socket pin 26 driver is at an inaccurate level.	Contact your local Data I/O Service Center.
78	PIN 24 ERROR	The self-test indicates that the socket pin 24 driver is at an inaccurate level.	Contact your local Data I/O Service Center.
79	PIN 28 ERROR	The self-test indicates that the socket pin 28 driver is at an inaccurate level.	Contact your local Data I/O Service Center.
81	PARITY ERROR	The incoming data has incorrect parity.	Check the parity setting and re-attempt transmission.
82	I/O FORMAT ERROR	The checksum field received by the programmer does not agree with its own calculated checksum.	Reload data to RAM. If the problem persists, service the programmer or contact your local Data I/O Service Center.
90	INVALID FORMAT	Non-existent or unsupported I/O format is selected in Computer Remote Control.	Check the list of supported I/O formats and enter a valid format code.

## Error Messages

Code	Name	Description	Corrective Action
91	I/O FORMAT ERROR	The programmer received an invalid address field.	Check all connections, check the data format and data source, and then try again.
92	5.6 V FAIL	The 5.6 V power supply circuit failed.	Contact your local Data I/O Service Center.
93	ID VOLT FAIL	The electronic ID power supply circuit failed.	Contact your local Data I/O Service Center.
94	VPP FAIL	The programming pulse power supply circuit failed.	Contact your local Data I/O Service Center.
95	DAC FAIL	The digital analog converter circuit failed.	Contact your local Data I/O Service Center.
98	VCC FAIL	The Vcc supply circuits are not working properly.	Contact your local Data I/O Service Center.
A1	NO ID FOUND	The installed device has no electronic ID.	Check the device type. Consult the Device List for the correct family and pinout codes.
A2	INVALID ID	The device's electronic ID is incompatible with the family and pinout codes selected.	Consult the Device List for the correct family and pinout codes and make sure the correct codes are entered. Programming with incorrect codes could damage a part.