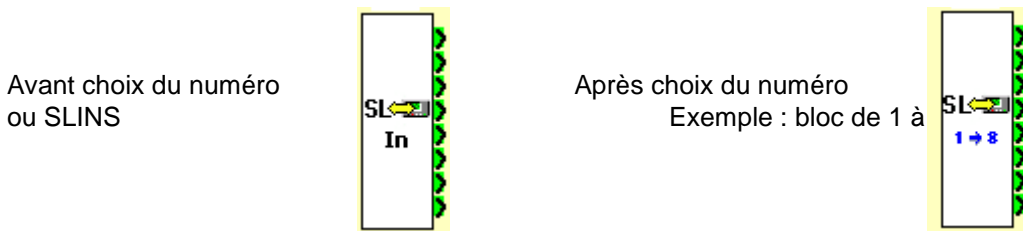


# 1 DESCRIPTION DES FB "FUNCTION BLOCK" SLIN SLOUT

SLIN SLOUT : c'est un protocole de communication, disponible sur un Millenium 3 par la prise de programmation située en façade (liaison série). 24 données sont accessibles en lecture et 24 autres en lecture / écriture. Les données sont au format 16 bits et sont lues et/ou écrites dans l'application avec les FB SLIN, SLINS / SLOUT

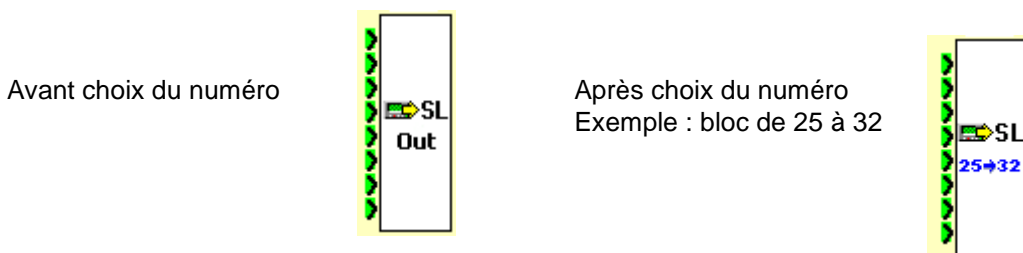
## 1.1 ENTREE LIAISON SERIE

3 blocs sont disponibles en entrée de 1 à 8, de 9 à 16 et de 17 à 24



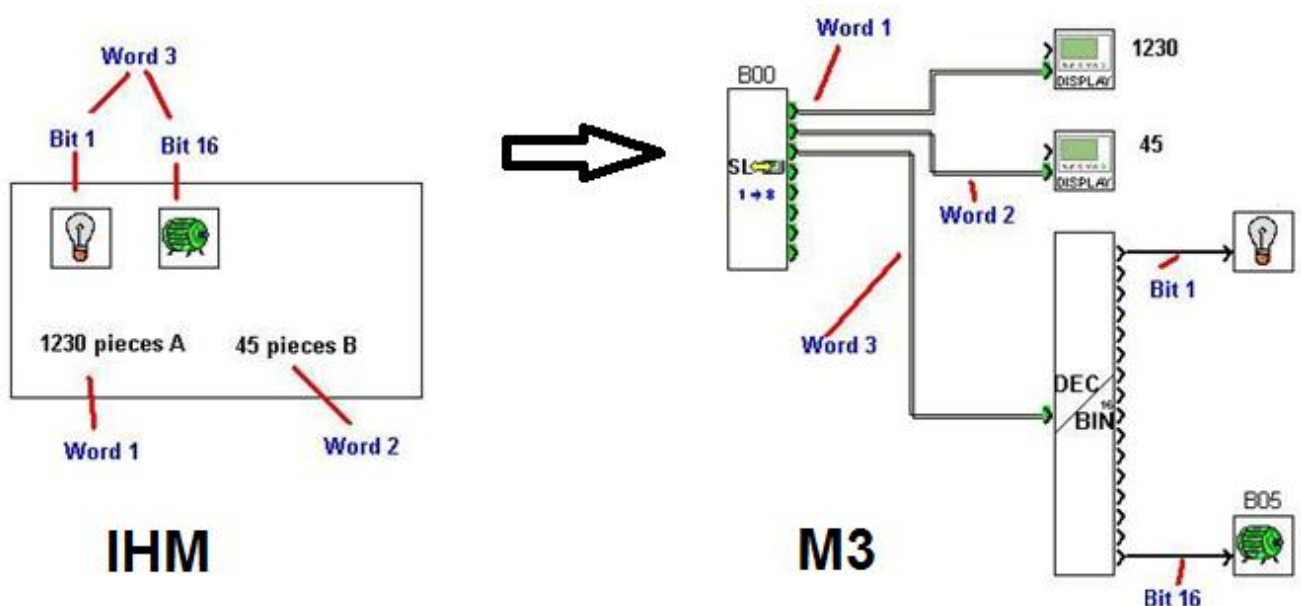
## 1.2 SORTIE LIAISON SERIE

3 blocs sont disponibles en sortie de 25 à 32, de 33 à 40 et de 41 à 48



On pourra donc écrire 24 valeurs dans le M3 et lire aussi 24 valeurs

## 1.3 EXEMPLES D'APPLICATION :



## 2 LIAISON SERIE

Le protocole utilisé entre le M3 et l'hôte est basé sur un protocole type MODBUS© ASCII.

Configuration de la liaison série :

Vitesse de dialogue : 115 200 bauds

Format : 7 bits, parité paire, 1 bit de stop, ASCII

Le Millenium 3 est Esclave, c'est le PC ou l'IHM qui est maître de la liaison.

Gestion Half-duplex : le poseur de question doit attendre une réponse avant d'envoyer une autre question

En mode ASCII, la trame commence par « 2 points » ( : ) (ASCII 3A hex) et termine ,par CR et LF (ASCII 0x0D and 0x0A hex) avec juste avant un LRC pour garantir l'intégrité de la trame.

:	Données	LRC	'CR'	'LF'
---	---------	-----	------	------

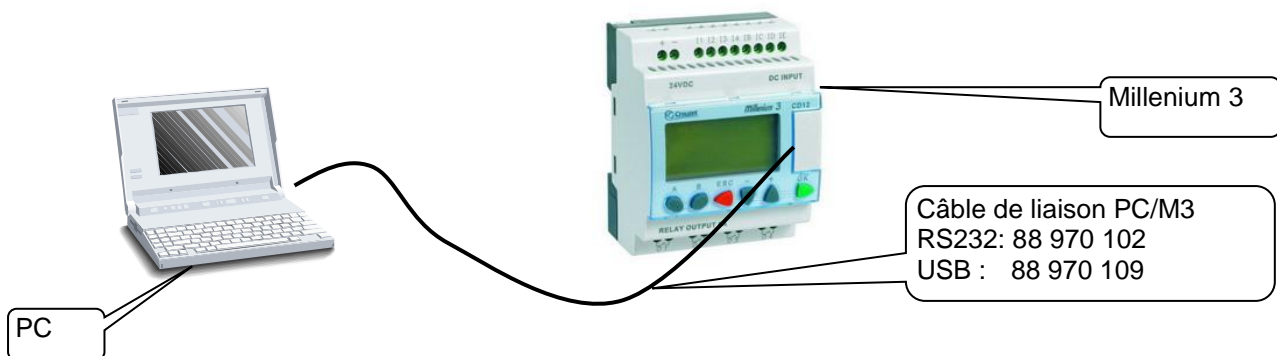
LRC : Longitudinal Redundancy Check

CR : Carriage return ( 0x0D )

LF : Line Feed ( 0x0A )

**Note : L'envoi est fait octet par octet**

### 2.1 EXEMPLE AVEC UN PC CONNECTE A UN MILLENIUM 3



### 3 CONSTRUCTION DES TRAMES SLIN SLOUT

#### 3.1 TRAME D'ECRITURE

La trame d'écriture à envoyer au contrôleur est la suivante :

Début trame	Adresse	Commande écriture	Adresse données	Nbre d'octets	Données	LRC	Fin trame
:	0x04	0x10	0x00 0x00 0xFF 0xaa	0xnn	0xd1H d1L d2H ... dnnL	0xcc	CR LF

Réponse du contrôleur :

Début trame	Adresse	Commande écriture	Adresse données	Nbre d'octets	LRC	Fin trame
:	0x04	0x10	0x00 0x00 0xFF 0xaa	0xnn	0xcc	CR LF

Exemple : écrire dans la donnée n° 3 du M3 la valeur 8569 (2179 en Hexa) (2 octets) :

Question	:	0x04	0x10	0x00 0x00 0xFF	0x02	0x02	0x21 0x79	0x4F	CR LF
Réponse	:	0x04	0x10	0x00 0x00 0xFF	0x02	0x02	0xE9		CR LF

**Calcul LRC :**

$04 + 10 + 00 + 00 + FF + 02 + 02 + 21 + 79 = 1B1$  on ne garde que B1, B1 complété = 4E,  $4E + 1 = 4F$

Excepté les délimiteurs chaque octet est envoyé sous la forme de deux caractères ASCII. Ce qui donne :

Question	3A 30 34 31 30 30 30 30 30 46 46 30 32 30 32 32 31 37 39 34 46 0D 0A
Réponse	3A 30 34 31 30 30 30 30 30 46 46 30 32 30 32 45 39 0D 0A

#### 3.2 TRAME DE LECTURE

La trame de lecture à envoyer au contrôleur est la suivante :

Note : les 48 registres peuvent être lus avec la même trame.

Début trame	Adresse	Commande lecture	Adresse données	Nbre d'octets	LRC	Fin trame
:	0x04	0x03	0x00 0x00 0xFF 0xaa	0xnn	0xcc	CR LF

Réponse du contrôleur

Début trame	Adresse	Commande lecture	Nbre d'octets	Données	LRC	Fin trame
:	0x04	0x03	0xnn	0xd1H d1L d2H ... dnnL	0xcc	CR LF

exemple : lire 1 donnée 16 bits à partir de la donnée n° 2 : (avec la donnée égale à 123 : 7B en hexa)

Question	:	0x04	0x03	0x00 0x00 0xFF	0x01	0x02	0xF7	CR LF
Réponse	:	0x04	0x03	0x02	0x00	0x7B	0xE0	CR LF

Question	3A 30 34 30 33 30 30 30 30 46 46 30 31 30 32 46 37 0D 0A
Réponse	3A 30 34 30 33 30 32 30 30 37 42 37 43 0D 0A

### 3.3 DETAILS

Adresse : 0x04 : trame SLIN/SLOUT  
0x01 : autres trames

0xaa : Adresse des données = adresse de l'entrée / sortie du bloc SLIN/SLOUT en hexa moins 1 (-1)	0xaa = 0x00 à 0x17 en écriture (SLIN) 0xaa = 0x00 à 0x2F en lecture (SLIN & SLOUT)
---	---

3 FB en entrée (SL IN) de 1 à 24 (24 - 1 = 23 = 17 hexa), 3 FB en sortie (SL OUT) de 25 à 48 (48 - 1 = 47 = 2F hexa)

Les blocs SLIN sont en écriture mais aussi en lecture.

Nombre d'octets	C'est le nombre de données à écrire. Chaque valeur est constituée de 2 octets. (Octet poids faible en tête)
-----------------	---

LRC	Somme complétée augmentée de 1, des octets entre <b>Adresse</b> et la dernière des données à écrire.
-----	--

Note : en cas d'erreur lors de la conception des trames, faire un ON/OFF du M3

## 4 AUTRES TRAMES

### 4.1 TABLE GENERALE D'ADRESSAGE

Adresse ( hexa )	Description
0000 6900	Valeur horloge
0000 6B00	Etat M3
0000 6C00	Commande

### 4.2 VALEUR HORLOGE

Adresse du mot	Contenu	Valeurs
6900	Seconde	00 → 59
6901	Minute	00 → 59
6902	Heure	00 → 23
6903	Jour semaine	00 → 06 0 = lundi
6904	Date jour	01 → 31
6905	Mois	01 → 12
6906	Année	01 → 99 1 = 2001
6980	Nombre Minute	60 (décalage d'une heure)
6981	Changement	0 ou 1
6982	Mois Eté	01 → 12 ex 3 = mars
6983	Date Eté	01 → 05 n° du dimanche (5 = dernier)
6984	Mois Hivers	01 → 12 ex 10 = octobre
6985	Date Hivers	01 → 05 n° du dimanche (5 = dernier)
6990	Dérive	0 → +/- 31 (en seconde)

**Lecture / Ecriture** : Autorisée quel que soit le mode de fonctionnement du M3

Ecriture de l'horloge :

: 01 10 00 00 69 00 03 seconde minute heure LRC.  
: adresse commande adresse mot nombre octet seconde minute heure LRC

Lecture horloge :

: 01 03 00 00 69 00 07 8C

### 4.3 ETAT M3

Adresse du mot	Contenu
6B00	IS163_Etat M3
6B01	IS163_Etat M3

**Lecture** : Autorisée quel que soit le mode de fonctionnement du M3

**Ecriture** : interdite

b0 RUN  
b10 ERREUR  
b11 DEF\_ALIM

### 4.4 COMMANDE

Adresse du mot	Contenu
6C00	Commande

**Valeur de Commande** : 01 : STOP  
02 : RUN  
03 : RUN avec INIT

**Lecture** : Non autorisée

**Ecriture** : Autorisée quel que soit le mode de fonctionnement du M3

RUN  
: 01 10 00 00 6C 00 01 02 80  
: adresse commande adresse mot nombre octet Run LRC

STOP  
: 01 10 00 00 6C 00 01 01 81

### 4.5 SIMULATION DES BOUTONS DE FACADE

Il faut envoyer le code "Touche x" puis le code "No\_Touche"

: 01 10 00 00 6F 4E 01 20 11  
: adresse commande adresse mot nombre octet Touche A LRC

Touche OK : (01) :011000006F4E010130

Touche + : (02) :011000006F4E01022F

Touche - : (04) :011000006F4E01042D

Touche Esc : (08) :011000006F4E010829

Touche B : (10) :011000006F4E011021

Touche A : (20) :011000006F4E012011

No\_Touche : (00) :011000006F4E010031