

Boîtier d'automatisme 4 226 80

FR

FRANÇAIS

3

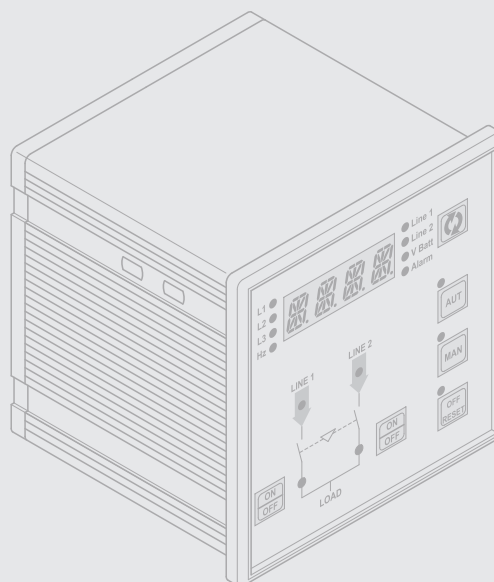


Table des matières

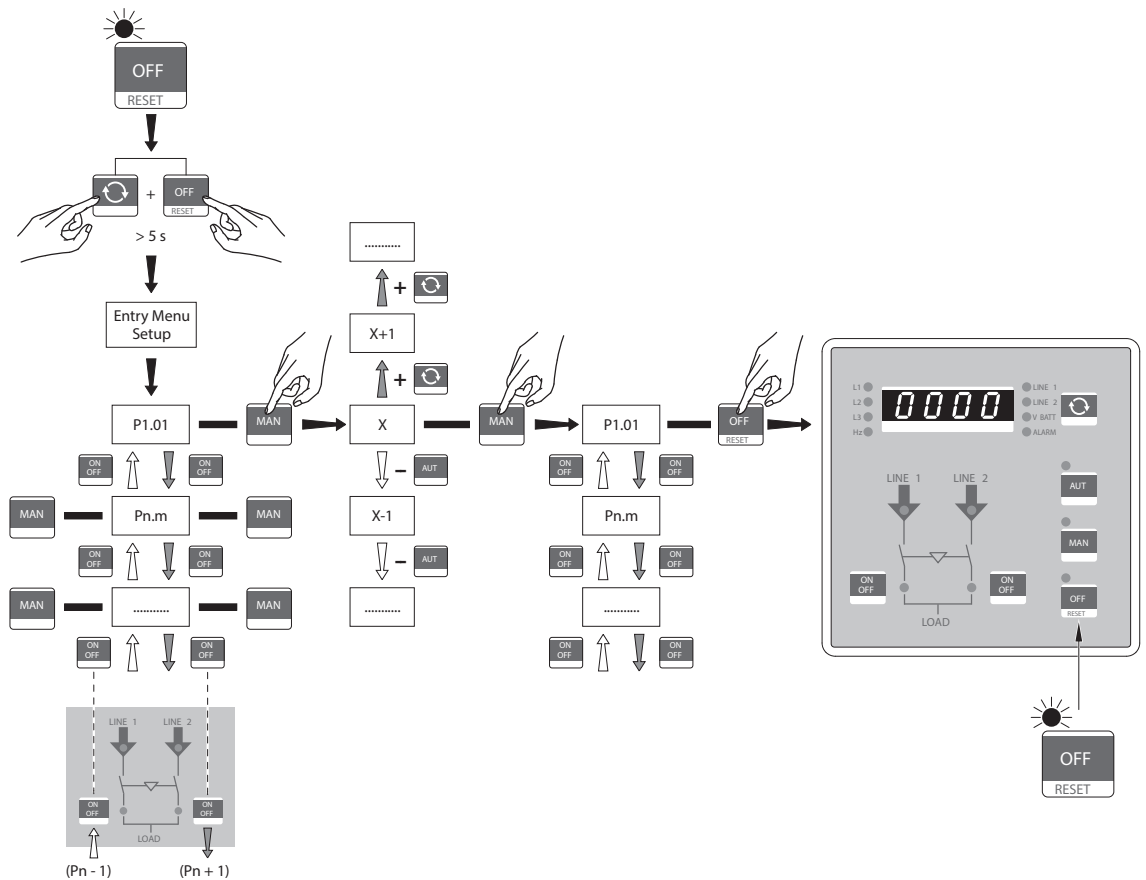
1. Protocole Modbus	4
2. Réglage des paramètres	4
3. Protocole Modbus RTU	2
4. Fonctions Modbus	4
4.1 Fonction 04: read input register	6
4.2 Fonction 06: preset single register	7
4.3 Fonction 07: read exception status	8
4.4 Fonction 16: preset multiple register	8
4.5 Fonction 17: report Slave ID	9
5. Erreurs	9
6. Protocole Modbus ASCII	10
7. Tableaux	11
7.1 Mesures fournies par le protocole de communication	11
7.2 Bits d'état	12
7.3 Commandes	16
7.4 Réglage des paramètres	17
7.4.1 Réglage des paramètres	17
A. Calcul CRC (Checksum pour RTU)	20
B. Calcul LRC (Checksum pour ASCII)	21

1. Protocole Modbus

Le boîtier d'automatisme 4 226 80 est compatible avec les protocoles de communication Modbus (RTU et ASCII) sur port RS232, à l'aide du câble 4 226 96 (contacter Legrand pour plus d'informations).
L'utilisation de cette fonction permet de lire l'état du dispositif et de commander l'unité à l'aide du logiciel de supervision ou autres dispositifs maitres compatibles avec le protocole Modbus (API par exemple).

2. Réglage des paramètres

Pour configurer le protocole Modbus, accéder au menu P7 – Communication. Pour accéder aux menus et passer d'un menu à un autre, merci de suivre la procédure ci-dessous :



Menu P7 – Communication

PAR	FONCTION	PLAGE	DEFAULT
P7.01	Adresse RS-232	1 ..245	1
P7.02	RS-232 Baud Rate	2400 4800 9600 19200 38400	9600 baud
P7.03	Protocole RS-232	RTU ASC – ASCII Mod – ASCII + modem	RTU
P7.04	Parité RS-232	Non – Aucun Impair EVe - Pair	NON

3. Protocole Modbus RTU

Pour le protocole Modbus RTU, le type du message de communication est la suivante :

T1T2T3	Adresse (8 bit)	Fonction (8 bit)	Données (N x 8 bit)	CRC (16 bit)	T1T2T3
--------	--------------------	---------------------	------------------------	-----------------	--------

- Le champ Adresse contient l'adresse série du dispositif esclave destinataire.
- Le champ Fonction contient le code de la fonction que l'esclave doit exécuter.
- Le champ Données contient les données envoyées aux esclaves ou les données reçues par l'esclave suite à une interrogation (la longueur maximale pour le champ données est de 60 registres de 16 bit, soit 120 octets).
- Le champ CRC permet aux dispositifs maître et esclave de contrôler l'intégrité du message.
Si un message est interrompu par des interférences ou du champ électrique, le champ CRC permet aux dispositifs de détecter l'erreur et d'ignorer le message.
- La séquence T1 T2 T3 correspond à une durée pendant laquelle aucune donnée ne doit être échangée sur le bus de communication pour permettre aux dispositifs connectés de reconnaître la fin d'un message ou le début d'un autre.
Cette durée doit être d'au moins 3,5 fois le temps nécessaire à l'envoi d'un caractère.

Le boîtier d'automatisme mesure le temps qui s'écoule entre la réception d'un caractère et le suivant. Si cette durée dépasse celle limitée à l'envoi de 3,5 caractères au baud rate sélectionné, le caractère qui suit est considéré comme le premier d'un nouveau message.

4. Fonctions Modbus

Les fonctions disponibles sont les suivantes :

03 = Read Multiple Holding Registers	Permet la lecture des registres internes du boîtier d'automatisme
04 = Read input register	Permet la lecture des mesures du boîtier d'automatisme
06 = Preset single register	Permet d'écrire des paramètres
07 = Read exception	Permet de lire l'état des dispositifs
10 = Preset multiple register	Permet d'écrire plusieurs paramètres
17 = Report Slave ID	Permet la lecture d'informations relatives au dispositif

Exemple :

Pour lire la valeur de la tension des batteries, présente sur la position 30 ($1E_{hex}$), depuis le boîtier d'automatisme à adresse série 01, le message à envoyer est le suivant :

01	04	00	1D	00	02	E1	CD
----	----	----	----	----	----	----	----

Où :

01 = adresse Esclave
 04 = fonction Modbus 'Read input register'
 00 1D = Adresse du registre requis (tension des batteries) diminuée de 1
 00 02 = Nombre de registres à lire à partir de l'adresse 30
 E1 CD = Checksum CRC

La réponse du boîtier d'automatisme sera la suivante :

01	04	04	00	00	00	7C	FA	65
----	----	----	----	----	----	----	----	----

Où :

01 = adresse boîtier d'automatisme (Esclave 01)
 04 = fonction requise par le master
 04 = nombre de octets envoyés par le boîtier d'automatisme
 00 00 00 7C = Valeur hex de la tension des batteries = 124 = 12.4V
 FA 65 = checksum CRC

4.1 Fonction 04: read input register

La fonction 04 Modbus permet de lire un ou plusieurs registres consécutifs dans la mémoire Esclave. L'adresse de chaque mesure est indiquée dans le tableau 7.1. Conformément au standard Modbus, l'adresse indiquée dans le message de requête doit être diminuée de 1 par rapport à celle effectivement indiquée dans le tableau.

Si l'adresse de mesure ne figure pas dans le tableau ou si le nombre de registres requis dépasse le nombre maximum admis, le boîtier d'automatisme renvoie un code d'erreur (voir tableau des erreurs dans le chapitre 5).

Exemple :

INTERROGATION MASTER (MAITRE)	
Adresse Esclave	08 _{hex}
Fonction	04 _{hex}
Adresse MSB	00 _{hex}
Adresse LSB	0F _{hex}
Numéro registre MSB	00 _{hex}
Numéro registre LSB	08 _{hex}
MSB CRC	21 _{hex}
LSB CRC	57 _{hex}

Sur l'exemple ci-dessus, à l'esclave 08, sont demandés 8 registres consécutifs commençant par l'adresse 10_{hex}. Aussi, les registres de 10_{hex} à 17_{hex} seront lus. Comme prévu, le message se termine par le checksum CRC.

RÉPONSE SLAVE (ESCLAVE)	
Adresse Esclave	08 _{hex}
Fonction	04 _{hex}
Nombre de octets	10 _{hex}
Registre MSB 10 _{hex}	00 _{hex}
Registre LSB 10 _{hex}	00 _{hex}
-----	----
Registre MSB 17 _{hex}	00 _{hex}
Registre LSB 17 _{hex}	00 _{hex}
MSB CRC	5E _{hex}
LSB CRC	83 _{hex}

La réponse est toujours constituée de l'adresse esclave, du code fonction requis par le maître et des données des registres requis. La réponse se termine par le CRC.

4.2 Fonction 06: preset single register

Cette fonction permet d'écrire dans les registres. Elle peut être utilisée avec les registres dont l'adresse est supérieure à 1000_{hex}. Il est ainsi possible de modifier le réglage des paramètres. Si la valeur n'est pas comprise dans le bon intervalle, le boîtier d'automatisme répond par un message d'erreur. De la même manière, si l'adresse du paramètre n'est pas reconnue, le boîtier d'automatisme envoie une réponse d'erreur.

L'adresse et l'intervalle correct de chaque paramètre sont indiqués dans le tableau 7.3.

Exemple :

MESSAGE MAITRE	
Adresse Esclave	08 _{hex}
Fonction	06 _{hex}
Adresse MSB	16 _{hex}
Adresse LSB	03 _{hex}
Numéro registre MSB	00 _{hex}
Numéro registre LSB	1E _{hex}
MSB CRC	FD _{hex}
LSB CRC	13 _{hex}

Réponse Esclave :

La réponse d'esclave est un écho de la requête, à savoir que le Esclave renvoie au maître l'adresse et la nouvelle valeur de la variable.

4. Fonctions Modbus

4.3 Fonction 07: read exception status

Cette fonction permet de lire l'état du boîtier d'automatisme.

Exemple :

INTERROGATION MAITRE	
Adresse Slave	08 _{hex}
Fonction	07 _{hex}
MSB CRC	47 _{hex}
LSB CRC	B2 _{hex}

Le tableau ci-dessous indique la signification du octet d'état envoyé comme réponse par le boîtier d'automatisme.

BIT	SIGNIFICATION
0	Mode de fonctionnement OFF / Reset
1	Mode de fonctionnement MAN
2	Mode de fonctionnement AUT
3	(non utilisé)
4	En erreur
5	(non utilisé)
6	Alimentation DC ok
7	Alarme générale active (on)

4.4 Fonction 16: preset multiple register

Cette fonction permet de modifier plusieurs paramètres par un unique message ou de régler une valeur plus longue qu'un registre.

Exemple :

MESSAGE MAITRE	
Adresse Esclave	08 _{hex}
Fonction	10 _{hex}
Adresse registre MSB	20 _{hex}
Adresse registre LSB	01 _{hex}
Numéro registre MSB	00 _{hex}
Numéro registre LSB	02 _{hex}
Données MSB	01 _{hex}
Données LSB	F4 _{hex}
Données MSB	06 _{hex}
Données LSB	83 _{hex}
MSB CRC	55 _{hex}
LSB CRC	3A _{hex}

RÉPONSE ESCLAVE	
Adresse Esclave	08 _{hex}
Fonction	10 _{hex}
Adresse registre MSB	20 _{hex}
Adresse registre LSB	01 _{hex}
Nombre octet MSB	00 _{hex}
Nombre octet LSB	04 _{hex}
MSB CRC	9B _{hex}
LSB CRC	53 _{hex}

4.5 Fonction 17: report Slave ID

Cette fonction permet d'identifier le type de dispositif.

Exemple :

INTERROGATION MAITRE	
Adresse Esclave	08 _{hex}
Fonction	11 _{hex}
MSB CRC	C6 _{hex}
LSB CRC	7C _{hex}

RÉPONSE ESCLAVE	
Adresse Esclave	08 _{hex}
Fonction	11 _{hex}
Compteur octets	04 _{hex}
Donnée 01 (Type) ❶	5F _{hex}
Donnée 02 (version SW)	04 _{hex}
Donnée 03 (version HW)	00 _{hex}
Donnée 04 (révision paramètres)	01 _{hex}
MSB CRC	... _{hex}
LSB CRC	... _{hex}

❶ 95 - 5F_{hex} = 4 226 80

5. Erreurs

Dans le cas où l'esclave recevrait un message incorrect, il répond par un message constitué de la fonction *OR-ée* requise en OR avec 80_{hex}, suivi d'un octet de code d'erreur.

Dans le tableau qui suit figurent les codes d'erreur envoyés au master par le Slave :

CODE	ERREUR
01	Fonction non valable
02	Adresse non valable
03	Paramètre hors intervalle
04	Exécution fonction impossible
06	Slave occupé, fonction momentanément non disponible

6. Protocole Modbus ASCII

Le protocole Modbus ASCII est en principe utilisé sur les applications qui nécessitent une communication avec deux modems.

Les fonctions et adresses disponibles sont les mêmes que celles de la version RTU, sauf que les caractères transmis sont au format ASCII et que la fin du message est délimité par un à-la-ligne (Carriage Return CR) et un Line Feed LF à la place d'un pause de transmission.

Si le paramètre P7.04 est configuré comme protocole Modbus ASCII, le message de communication sur le port de communication correspondant à la structure suivante :

:	Adresse (2 caractères)	Fonction (2 caractères)	Données (N caractères)	LRC (2 caractères)	CR LF
---	---------------------------	----------------------------	---------------------------	-----------------------	-------

- Le champ Adresse contient l'adresse série du dispositif esclave destinataire.
- Le champ Fonction contient le code de la fonction que le Slave doit exécuter.
- Le champ Données contient les données envoyées à l'esclave ou les données reçues par l'esclave suite à une interrogation. La longueur maximale admise est de 80 registres consécutifs.
- Le champ LRC permet aux dispositifs master et Slave de contrôler l'intégrité du message. Si un message est interrompu par des interférences ou du champ électrique, le champ LRC permet aux dispositifs de détecter l'erreur et d'ignorer le message.
- Le message se termine toujours par le caractère de contrôle CRLF (0D 0A).

Exemple :

Pour lire la valeur de la tension équilibrée, présente sur la position 04 (04_{hex}) depuis l'esclave à adresse série 08, le message à envoyer est le suivant :

:	08	04	00	03	00	02	EF	CRLF
---	----	----	----	----	----	----	----	------

Où :

: = ASCII 3A_{hex} délimiteur début de message
 08 = adresse esclave
 04 = fonction Modbus "Read input register"
 00 03 = Adresse du registre requis (L2 tension de la ligne 1) diminuée de 1
 00 02 = Nombre de registres à lire à partir de l'adresse 04
 EF = Checksum LRC
 CRLF = ASCII 0D_{hex} 0A_{hex} = Délimiteur de fin de message

La réponse est la suivante :

:	08	04	04	00	00	01	A0	4F	CRLF
---	----	----	----	----	----	----	----	----	------

Où :

: = ASCII 3A_{hex} délimiteur début de message
 08 = adresse (Esclave 08)
 04 = fonction requise par le master
 04 = nombre de octets envoyés par le dispositif
 00 00 01 A0 = valeur Hex de la tension enchainée (= 416V)
 4F = checksum LRC
 CRLF = ASCII 0D_{hex} 0A_{hex} = Délimiteur de fin de message

7. Tableaux

7.1 Mesures fournies par le protocole de communication

A utiliser avec fonctions 03 et 04

ADRESSE	WORDS	MESURE	UNITÉ	FORMAT
02 _{hex}	2	Tension Ligne 1 L1-N	V	Unsigned long
04 _{hex}	2	Tension Ligne 1 L2-N	V	Unsigned long
06 _{hex}	2	Tension Ligne 1 L3-N	V	Unsigned long
08 _{hex}	2	Tension Ligne 1 L1-L2	V	Unsigned long
0A _{hex}	2	Tension Ligne 1 L2-L3	V	Unsigned long
0C _{hex}	2	Tension Ligne 1 L3-L1	V	Unsigned long
0E _{hex}	2	Tension Ligne 2 L1-N	V	Unsigned long
10 _{hex}	2	Tension Ligne 2 L2-N	V	Unsigned long
12 _{hex}	2	Tension Ligne 2 L3-N	V	Unsigned long
14 _{hex}	2	Tension Ligne 2 L1-L2	V	Unsigned long
16 _{hex}	2	Tension Ligne 2 L2-L3	V	Unsigned long
18 _{hex}	2	Tension Ligne 2 L3-L1	V	Unsigned long
1A _{hex}	2	Fréquence Ligne 1 L3-L1	Hz/10	Unsigned long
1C _{hex}	2	Fréquence Ligne 2 L3-L1	Hz/10	Unsigned long
1E _{hex}	2	Tension de la batterie (alimentation DC)	VDC / 10	Unsigned long
20 _{hex}	2	(non utilisé)	-	-
22 _{hex}	2	(non utilisé)	-	-
24 _{hex}	2	(non utilisé)	-	-
26 _{hex}	2	(non utilisé)	-	-
28 _{hex}	2	(non utilisé)	-	-
2A _{hex}	2	(non utilisé)	-	-
2C _{hex}	2	(non utilisé)	-	-
2E _{hex}	2	(non utilisé)	-	-
30 _{hex}	2	(non utilisé)	-	-
32 _{hex}	2	Nombre de commutations interrupteur 1 en AUT	nr	Unsigned long
34 _{hex}	2	Nombre de commutations interrupteur 2 en AUT	nr	Unsigned long
36 _{hex}	2	Nombre de commutations interrupteur 1 en MAN	nr	Unsigned long
38 _{hex}	2	Nombre de commutations interrupteur 2 en MAN	nr	Unsigned long
3A _{hex}	2	Nombre d'alarmes de commutation interrupteur 1	nr	Unsigned long
3C _{hex}	2	Nombre d'alarmes de commutation interrupteur 2	nr	Unsigned long
3E _{hex}	2	(non utilisé)	-	-
40 _{hex}	2	Alarmes ⁽¹⁾	bits	Unsigned long

7. Tableaux

⁽¹⁾ En lisant les words à partir de l'adresse 40_{hex}, 32 bits sont restitués (pour leur signification voir ci-dessous) :

BIT	CODE	ALARME
0	A01	Tension de la batterie trop faible
1	A02	Tension de la batterie trop haute
2	A03	Timeout interrupteur ligne 1
3	A04	Timeout interrupteur ligne 2
4	A05	Erreur ordre de phases Ligne 1
5	A06	Erreur ordre de phases Ligne 2
6	A07	Timeout charge non alimentée
7	A08	Groupe électrogène non disponible
8	A09	Urgence
9	A10	Déclenchement interrupteur 1
10	A11	Déclenchement interrupteur 2
11..31	--	(non utilisé)

7.2 Bits d'état

A utiliser avec fonctions 03 et 04

ADRESSE	WORDS	FONCTION	FORMAT
2070 _{hex}	1	État clavier panneau frontal ❶	Unsigned integer
2071 _{hex}	1	État entrées numériques (par borne) ❷	Unsigned integer
2072 _{hex}	1	État sorties numériques (par borne) ❸	Unsigned integer
-	-	-	-
2074 _{hex}	1	État tension ligne 1 ❹	Unsigned integer
2075 _{hex}	1	État interrupteur ligne 1 ❺	Unsigned integer
2076 _{hex}	1	État tension ligne 2 ❻	Unsigned integer
2077 _{hex}	1	État interrupteur ligne 2 ❼	Unsigned integer
2078 _{hex}	2	État fonctions entrées ❽	Unsigned integer
207A _{hex}	1	État fonctions sorties ❿	Unsigned integer
207B _{hex}	1	État messages écran ⓫	Unsigned integer

❶ Le tableau suivant indique la signification des bits de words à l'adresse 2070_{hex} :

BIT	TOUCHE
0	Ligne 2 ON-OFF
1	Mode OFF
2	Mode MAN
3	Mode AUT
4	(non utilisé)
5	Ligne 1 ON-OFF
6	Sélection mesures ligne 2
7	Sélection mesures ligne 1
8...15	(non utilisé)

❷ Le tableau suivant indique la signification des bits de words à l'adresse 2071_{hex} :

BIT	ENTRÉE
0	État borne d'entrée 2.1
1	État borne d'entrée 2.2
2	État borne d'entrée 2.3
3	État borne d'entrée 2.4
4	État borne d'entrée 2.5
5	État borne d'entrée 2.6
6...15	(non utilisé)

❸ Le tableau suivant indique la signification des bits de words à l'adresse 2072_{hex} :

BIT	SORTIE
0	État borne de sortie 4.1
1	État borne de sortie 4.3
2	État borne de sortie 5.1
3	État borne de sortie 5.3
4	État borne de sortie 3.1
5	État borne de sortie 3.4
6...15	(non utilisé)

7. Tableaux

④ Le tableau suivant indique la signification des bits de mots à l'adresse 2074_{hex} (Ligne 1) et à l'adresse 2076_{hex} (Ligne 2) :

BIT	ÉTAT LIGNE
0	Valeurs ligne dans les limites
1	Valeurs ligne dans les limites avec retard
2	Tension dans les limites
3	Tension ok
4	Fréquence dans les limites
5	Fréquence ok
6	Tension inférieure au min.
7	Tension supérieure au max.
8	Asymétrie de la tension
9	Absence de phase tension
10	Fréquence inférieure au min.
11	Fréquence supérieure au max.
12	Erreur ordre de phases
13	Tous les paramètres de ligne ok
14-15	(non utilisé)

⑤ Le tableau suivant indique la signification des bits de mots à l'adresse 2075_{hex} (Ligne 1) et à l'adresse 2077_{hex} (Ligne 2) :

BIT	CONDITION INTERRUPTEUR
0	Interrupteur fermé
1	Alarme déclenchement
2	(non utilisé)
3	État commande (1 = fermeture)
4	Sortie commande de fermeture
5	Sortie commande d'ouverture
6...15	(non utilisé)

⑥ Le tableau suivant indique la signification des bits de words à l'adresse 2078_{hex} :

BIT	ÉTAT FONCTIONS ENTRÉES
0	Interrupteur ligne 1 fermé
1	Déclenchement interrupteur ligne 1
2	(non utilisé)
3	Interrupteur ligne 2 fermé
4	Déclenchement interrupteur ligne 2
5	(non utilisé)
6	Commutation sur ligne secondaire
7	Inhibition retour sur ligne principale
8	Bouton d'urgence
9	Mise en marche générateur
10	Générateur 1 prêt
11	(non utilisé)
12	Clavier verrouillé
13	Blocage paramètres
14	(non utilisé)
15	(non utilisé)

⑦ Le tableau suivant indique la signification des bits de words à l'adresse 207A_{hex} :

BIT	ÉTAT FONCTIONS SORTIES
0	Interrupteur ligne 1 ouvert
1	Interrupteur ligne 1 fermé
2	Interrupteur ligne 2 ouvert
3	Interrupteur ligne 2 fermé
4	Alarme générale
5	(non utilisé)
6	Mise en marche générateur 2
7	ATS prêt
8...15	(non utilisé)

⑧ Le tableau suivant indique la signification des bits de words à l'adresse 207B_{hex} :

BIT	ÉTAT MESSAGES ÉCRAN
0	(non utilisé)
1	Mise en marche générateur 2
2	(non utilisé)
3	Refroidissement générateur 2
4	Commutation charge 2 → 1
5	Commutation charge 1 → 2

7. Tableaux

7.3 Commandes

A utiliser avec fonction 06

ADRESSE	WORDS	ÉTAT	FORMAT
2F00h	1	Changement de mode de fonctionnement ❶	Unsigned integer
2F01h	1	Reset dispositif (warm boot) ❷	Unsigned integer
2F02h	1	Rétablissement paramètres par défaut ❷	Unsigned integer
2F03h	1	Sauvegarde paramètres en mémoire ❸	Unsigned integer
2F04h	1	Reset compteur heures ❹	Unsigned integer
2F05h	1	Reset compteurs fonctionnement ❹	Unsigned integer
2F06h	1	(non utilisé)	-
2F07h	1	(non utilisé)	-
2F08h	1	Verrouillage du clavier ON/OFF ❺	Unsigned integer
...
2F0Dh	1	Simulation pression touches panneau frontal ❻	Unsigned integer

❶ Le tableau suivant indique les valeurs à écrire à l'adresse 2F00_{hex} pour obtenir les fonctions correspondantes :

VALEUR	FONCTION
0	Commutation en mode OFF
1	Commutation en mode MAN
2	Commutation en mode AUT

❷ En écrivant la valeur 01_{hex} à l'adresse indiquée, la fonction correspondante est exécutée.

❸ En écrivant la valeur AA_{hex} à l'adresse indiquée, la fonction correspondante est exécutée.

❹ En écrivant la valeur FF_{hex} à l'adresse indiquée, la fonction correspondante est exécutée.

❺ Le tableau qui suit indique les valeurs à écrire à l'adresse 2F08_{hex} pour obtenir les fonctions correspondantes :

VALEUR	FONCTION
0	Clavier débloqué
1	Verrouillage du clavier

❻ Le tableau suivant indique les positions de bit de la valeur à écrire à l'adresse 2F0D_{hex} pour obtenir les fonctions correspondantes :

BIT	SIMULATION PRESSION TOUCHES
0	Commutation manuelle Ligne 2
1	Mode OFF
2	Mode MAN
3	Mode AUT
4	(non utilisé)
5	Commutation manuelle Ligne 1
6	Sélection mesures ligne 2
7	Sélection mesures ligne 1
8...15	(non utilisé)

7.4 Réglage des paramètres

En utilisant le protocole Modbus, il est possible d'accéder aux paramètres des menus. Dans les tableaux suivants, figurent les plages numériques de tous les paramètres. Pour interpréter correctement la correspondance entre valeur numérique et fonction sélectionnée et/ou unité de mesure, faire référence au manuel d'utilisation du boîtier d'automatisme.

Pour rendre effectives les modifications apportées au réglage des paramètres, il est nécessaire de mémoriser les valeurs en mémoire, en utilisant à cet effet la commande dédiée décrite dans le tableau 7.3.

7.4.1 Réglage des paramètres

A utiliser avec fonctions 04 et 06.

ADRESSE	WORDS	PARAMÈTRES	PLAGE	FORMAT
3000 _{hex}	1	P1.01 Tension nominale	100...690	Unsigned integer
3001 _{hex}	1	P1.02 Rapport VT	100...999	Unsigned integer
3002 _{hex}	1	P1.03 Configuration câblage	0...3 ❶	Unsigned integer
3003 _{hex}	1	P1.04 Mode contrôle tension	0...2 ❶	Unsigned integer
3004 _{hex}	1	P1.05 Fréquence nominale	0...1 ❶	Unsigned integer
3005 _{hex}	1	P1.06 Tension nominale des batteries	0...3 ❶	Unsigned integer
3006 _{hex}	1	P1.07 Langue	0...5 ❶	Unsigned integer
3100 _{hex}	1	P2.01 Type application	0...2 ❶	Unsigned integer
3101 _{hex}	1	P2.02 Contrôle ordre de phases	0...2 ❶	Unsigned integer
3102 _{hex}	1	P2.03 Sélection ligne principale	0...1 ❶	Unsigned integer
3103 _{hex}	1	P2.04 Temps d'inter-verrouillage Ligne1->Ligne2	1...900	Unsigned integer
3104 _{hex}	1	P2.05 Temps d'inter-verrouillage Ligne2->Ligne1	1...900	Unsigned integer
3105 _{hex}	1	P2.06 Stratégie de commutation	0...1 ❶	Unsigned integer
3106 _{hex}	1	P2.07 Mode de contrôle commutation	0...2 ❶	Unsigned integer
3107 _{hex}	1	P2.08 Timeout ouverture / fermeture commutation	1...900	Unsigned integer
3108 _{hex}	1	P2.09 Durée impulsion ouverture commutation	1...600	Unsigned integer
3109 _{hex}	1	P2.10 Durée impulsion fermeture commutation	1...600	Unsigned integer
310A _{hex}	1	P2.11 Timeout charge non alimentée	0...3600 ❷	Unsigned integer
310B _{hex}	1	P2.12 Blocage re-transfert automatique	0...1 ❶	Unsigned integer
310C _{hex}	1	P2.13 Temps pré-transfert	0...300 ❷	Unsigned integer
310D _{hex}	1	P2.14 Temps post-transfert	0...300 ❷	Unsigned integer
310E _{hex}	1	P2.15 Retard mise en marche générateur	0...900	Unsigned integer
310F _{hex}	1	P2.16 Temps de refroidissement générateur	1...3600	Unsigned integer
3110 _{hex}	1	P2.17 Temps rotation générateur	0...14 ❶	Unsigned integer
3111 _{hex}	1	P2.18 Heures rotation générateur	0...23	Unsigned integer
3112 _{hex}	1	P2.19 Minutes rotation générateur	0...59	Unsigned integer
3113 _{hex}	1	P2.20 Tension minimum des batteries	69...100 ❷	Unsigned integer

❶ L'association entre une valeur numérique et une fonction doit être effectué dans un ordre donné, en tenant compte de la fonction indiquée dans le manuel d'utilisation. La première fonction est obtenue en réglant la valeur sur 0, tandis que la dernière fonction est obtenue en réglant la valeur maximale admise par l'intervalle.

❷ Pour sélectionner OFF, régler la valeur numérique minimale admise par l'intervalle.

❸ Pour sélectionner OFF, régler la valeur numérique maximale admise par l'intervalle.

❹ L'association entre une valeur numérique et une fonction doit être effectué dans un ordre donné, en tenant compte de la fonction indiquée dans le manuel d'utilisation. La première fonction est obtenue en réglant la valeur maximale admise par l'intervalle, tandis que la dernière fonction est obtenue en réglant la valeur sur 0.

7. Tableaux

ADRESSE	WORDS	PARAMÈTRES	PLAGE	FORMAT
3200 _{hex}	1	P3.01 Chute de tension min.	70... 98	Unsigned integer
3201 _{hex}	1	P3.02 Prélèvement tension min.	75... 100	Unsigned integer
3202 _{hex}	1	P3.03 Retard tension min.	1 ... 9000	Unsigned integer
3203 _{hex}	1	P3.04 Chute de tension max.	102...121 ⑤	Unsigned integer
3204 _{hex}	1	P3.05 Prélèvement tension max.	100...115	Unsigned integer
3205 _{hex}	1	P3.06 Retard tension max.	1...9000	Unsigned integer
3206 _{hex}	1	P3.07 Seuil absence phase	59...85 ②	Unsigned integer
3207 _{hex}	1	P3.07 Retard absence phase	1...300	Unsigned integer
3208 _{hex}	1	P3.09 Seuil asymétrie	2...21 ③	Unsigned integer
3209 _{hex}	1	P3.10 Retard asymétrie	1...9000	Unsigned integer
320A _{hex}	1	P3.11 Fréquence min.	79...100 ②	Unsigned integer
320B _{hex}	1	P3.12 Retard fréquence min.	1 ...9000	Unsigned integer
320C _{hex}	1	P3.13 Fréquence max.	100...121 ⑤	Unsigned integer
320D _{hex}	1	P3.14 Retard fréquence max.	1...9000	Unsigned integer
320E _{hex}	1	P3.15 Retard ligne 1 OK (quand ligne 2 non OK)	1...3600	Unsigned integer
320F _{hex}	1	P3.16 Retard ligne 1 OK (quand ligne 2 OK)	1...3600	Unsigned integer
3300 _{hex}	1	P4.01 Chute de tension min.	70... 98	Unsigned integer
3301 _{hex}	1	P4.02 Prélèvement tension min.	75... 100	Unsigned integer
3302 _{hex}	1	P4.03 Retard tension min.	1 ... 9000	Unsigned integer
3303 _{hex}	1	P4.04 Chute de tension max.	102...121 ⑤	Unsigned integer
3304 _{hex}	1	P4.05 Prélèvement tension max.	100...115	Unsigned integer
3305 _{hex}	1	P4.06 Retard tension max.	1...9000	Unsigned integer
3306 _{hex}	1	P4.07 Seuil absence phase	59...85 ②	Unsigned integer
3307 _{hex}	1	P4.08 Retard absence phase	1...300	Unsigned integer
3308 _{hex}	1	P4.09 Seuil asymétrie	2...21 ③	Unsigned integer
3309 _{hex}	1	P4.10 Retard asymétrie	1...9000	Unsigned integer
330A _{hex}	1	P4.11 Fréquence min.	79...100 ②	Unsigned integer
330B _{hex}	1	P4.12 Retard fréquence min.	1 ...9000	Unsigned integer
330C _{hex}	1	P4.13 Fréquence max.	100...121 ⑤	Unsigned integer
330D _{hex}	1	P4.14 Retard fréquence max.	1...9000	Unsigned integer
330E _{hex}	1	P4.15 Retard ligne 1 OK (quand ligne 1 non OK)	1...3600	Unsigned integer
330F _{hex}	1	P4.16 Retard ligne 1 OK (quand ligne 1 OK)	1...3600	Unsigned integer
3400 _{hex}	1	P5.01.01 Entrée programmable 1	0...13 ①	Unsigned integer
3401 _{hex}	1	P5.01.02 Entrée programmable 1	0...1 ①	Unsigned integer
3402 _{hex}	1	P5.01.03 Entrée programmable 1	0...250 ①	Unsigned integer
3403 _{hex}	1	P5.01.04 Entrée programmable 1	0...250 ①	Unsigned integer
3404 _{hex}	1	P5.02.01 Entrée programmable 2	0...13 ①	Unsigned integer
3405 _{hex}	1	P5.02.02 Entrée programmable 2	0...1 ①	Unsigned integer
3406 _{hex}	1	P5.02.03 Entrée programmable 2	0...250 ①	Unsigned integer
3407 _{hex}	1	P5.02.04 Entrée programmable 2	0...250 ①	Unsigned integer
3408 _{hex}	1	P5.03.01 Entrée programmable 3	0...13 ①	Unsigned integer
3409 _{hex}	1	P5.03.02 Entrée programmable 3	0...1 ①	Unsigned integer

ADRESSE	WORDS	PARAMÈTRES	PLAGE	FORMAT
340A _{hex}	1	P5.03.03 Entrée programmable 3	0...250 ❶	Unsigned integer
340B _{hex}	1	P5.03.04 Entrée programmable 3	0...250 ❶	Unsigned integer
340C _{hex}	1	P5.04.01 Entrée programmable 4	0...13 ❶	Unsigned integer
340D _{hex}	1	P5.04.02 Entrée programmable 4	0...1 ❶	Unsigned integer
340E _{hex}	1	P5.04.03 Entrée programmable 4	0...250 ❶	Unsigned integer
340F _{hex}	1	P5.04.04 Entrée programmable 4	0...250 ❶	Unsigned integer
3410 _{hex}	1	P5.05.01 Entrée programmable 5	0...13 ❶	Unsigned integer
3411 _{hex}	1	P5.05.02 Entrée programmable 5	0...1 ❶	Unsigned integer
3412 _{hex}	1	P5.05.03 Entrée programmable 5	0...250 ❶	Unsigned integer
3413 _{hex}	1	P5.05.04 Entrée programmable 5	0...250 ❶	Unsigned integer
3414 _{hex}	1	P5.06.01 Entrée programmable 6	0...13 ❶	Unsigned integer
3415 _{hex}	1	P5.06.02 Entrée programmable 6	0...1 ❶	Unsigned integer
3416 _{hex}	1	P5.06.03 Entrée programmable 6	0...250 ❶	Unsigned integer
3417 _{hex}	1	P5.06.04 Entrée programmable 6	0...250 ❶	Unsigned integer
3500 _{hex}	1	P6.01.01 Sortie programmable 1	0...11 ❶	Unsigned integer
3501 _{hex}	1	P6.01.02 Sortie programmable 1	0...1 ❶	Unsigned integer
3502 _{hex}	1	P6.02.01 Sortie programmable 2	0...11 ❶	Unsigned integer
3503 _{hex}	1	P6.02.02 Sortie programmable 2	0...1 ❶	Unsigned integer
3504 _{hex}	1	P6.03.01 Sortie programmable 3	0...11 ❶	Unsigned integer
3505 _{hex}	1	P6.03.02 Sortie programmable 3	0...1 ❶	Unsigned integer
3506 _{hex}	1	P6.04.01 Sortie programmable 4	0...11 ❶	Unsigned integer
3507 _{hex}	1	P6.04.02 Sortie programmable 4	0...1 ❶	Unsigned integer
3508 _{hex}	1	P6.05.01 Sortie programmable 5	0...11 ❶	Unsigned integer
3509 _{hex}	1	P6.05.02 Sortie programmable 5	0...1 ❶	Unsigned integer
350A _{hex}	1	P6.06.01 Sortie programmable 6	0...11 ❶	Unsigned integer
350B _{hex}	1	P6.06.02 Sortie programmable 6	❶	Unsigned integer

- ❶ L'association entre une valeur numérique et une fonction doit être effectuée dans un ordre donné, en tenant compte de la fonction indiquée dans le manuel d'utilisation. La première fonction est obtenue en réglant la valeur sur 0, tandis que la dernière fonction est obtenue en réglant la valeur maximale admise par l'intervalle.
- ❷ Pour sélectionner OFF, régler la valeur numérique minimale admise par l'intervalle.
- ❸ Pour sélectionner OFF, régler la valeur numérique maximale admise par l'intervalle.

ADRESSE	WORDS	PARAMÈTRES	PLAGE	FORMAT
3600 _{hex}	1	P7.01 Adresse série RS232	1...245	Unsigned integer
3601 _{hex}	1	P7.02 Vitesse série RS232	0...4 ❶	Unsigned integer
3602 _{hex}	1	P7.03 Protocole RS232	0...4 ❶	Unsigned integer
3603 _{hex}	1	P7.04 Parité RS232	0...4 ❶	Unsigned integer

- ❶ L'association entre une valeur numérique et une fonction doit être effectuée dans un ordre donné, en tenant compte de la fonction indiquée dans le manuel d'utilisation. La première fonction est obtenue en réglant la valeur sur 0, tandis que la dernière fonction est obtenue en réglant la valeur maximale admise par l'intervalle.

7. Tableaux

A. Calcul CRC (Checksum pour RTU)

Algorithme de calcul CRC

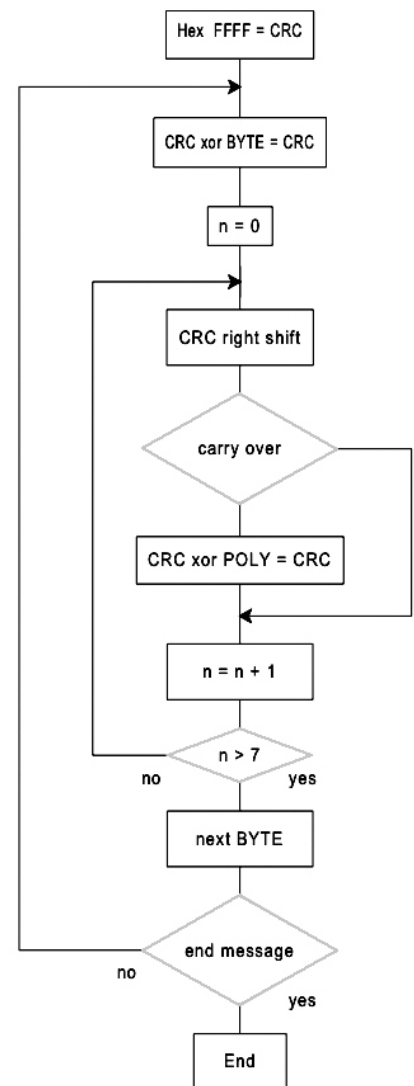
Exemple : Trame = 0207_{hex}

Initialisation CRC

Charge premier bit	1111	1111	1111	1111
Exécution xor avec le premier octet du trame	1111	1111	1111	1101
Exécution premier shift à droite	0111	1111	1111	1110 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1101	1111	1111	1111
Exécution 2 nd shift à droite	0110	1111	1111	1111 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1100	1111	1111	1110
Exécution 3 ^{ème} shift à droite	0110	0111	1111	1111 0
Exécution 4 ^{ème} shift à droite	0011	0011	1111	1111 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1001	0011	1111	1110
Exécution 5 ^{ème} shift à droite	0100	1001	1111	1111 0
Exécution 6 ^{ème} shift à droite	0010	0100	1111	1111 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1000	0100	1111	1110
Exécution 7 ^{ème} shift à droite	0100	0010	0111	1111 0
Exécution 8 ^{ème} shift à droite	0010	0001	0011	1111 1
Carry=1, charge polynôme	1010	0000	0000	0001
Charge deuxième octet du trame			0000	0111
Exécution xor avec le deuxième octet du trame	1000	0001	0011	1001
Exécution premier shift à droite	0100	0000	1001	1100 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1110	0000	1001	1101
Exécution 2 nd shift à droite	0111	0000	0100	1110 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1101	0000	0100	1111
Exécution 3 ^{ème} shift à droite	0110	1000	0010	0111 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1100	1000	0010	0110
Exécution 4 ^{ème} shift à droite	0110	0100	0001	0011 0
Exécution 5 ^{ème} shift à droite	0010	0100	0000	1001 1
Carry=1, charge polynôme	1010	0000	0000	0001
Exécution xor avec polynôme	1001	0010	0000	1000
Exécution 6 ^{ème} shift à droite	0100	1001	0000	0100 0
Exécution 7 ^{ème} shift à droite	0010	0100	1000	0010 0
Exécution 8 ^{ème} shift à droite	0001	0010	0100	0001 0

Résultat CRC

0001 0010 0100 0001
12_{hex} **41_{hex}**

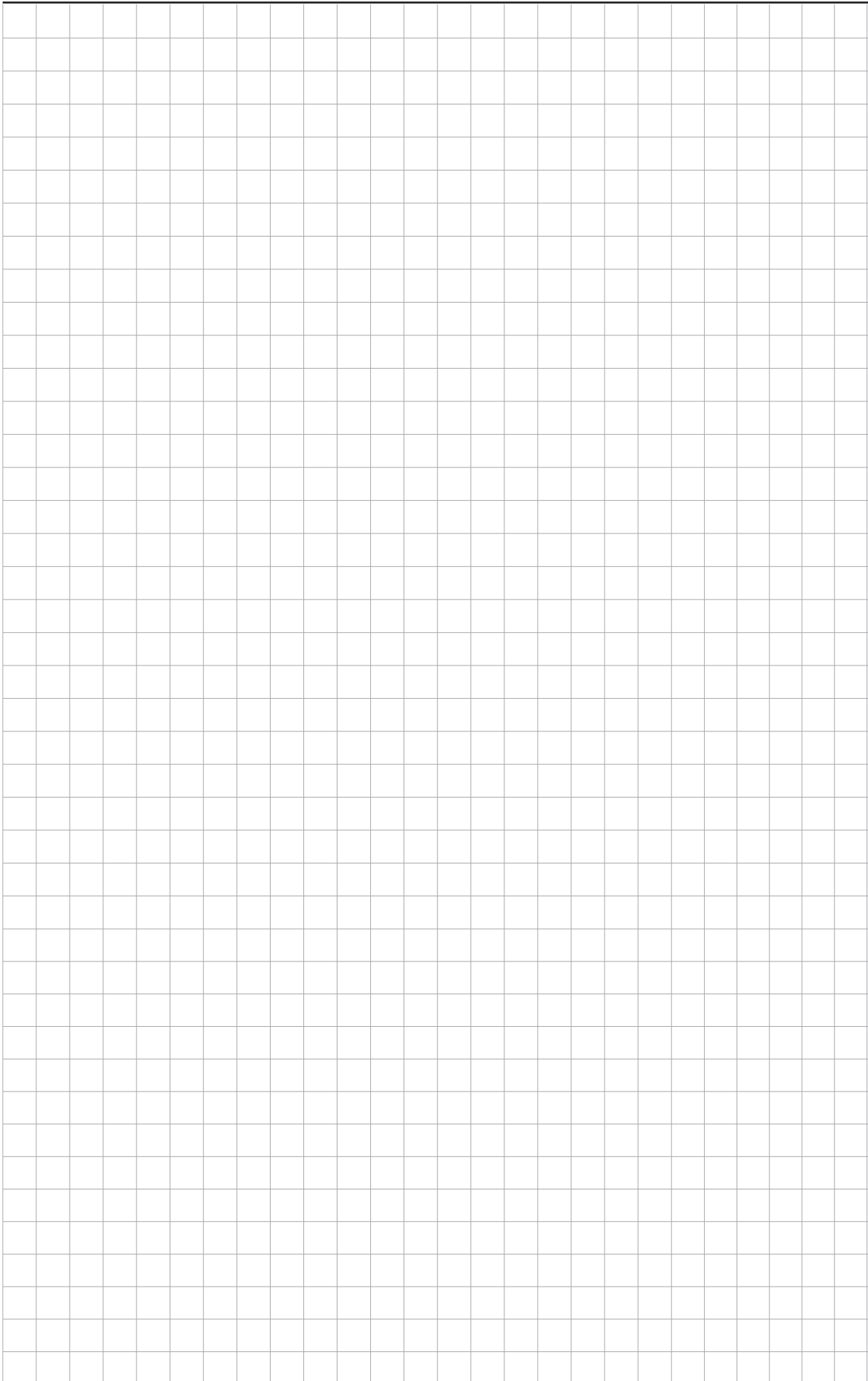


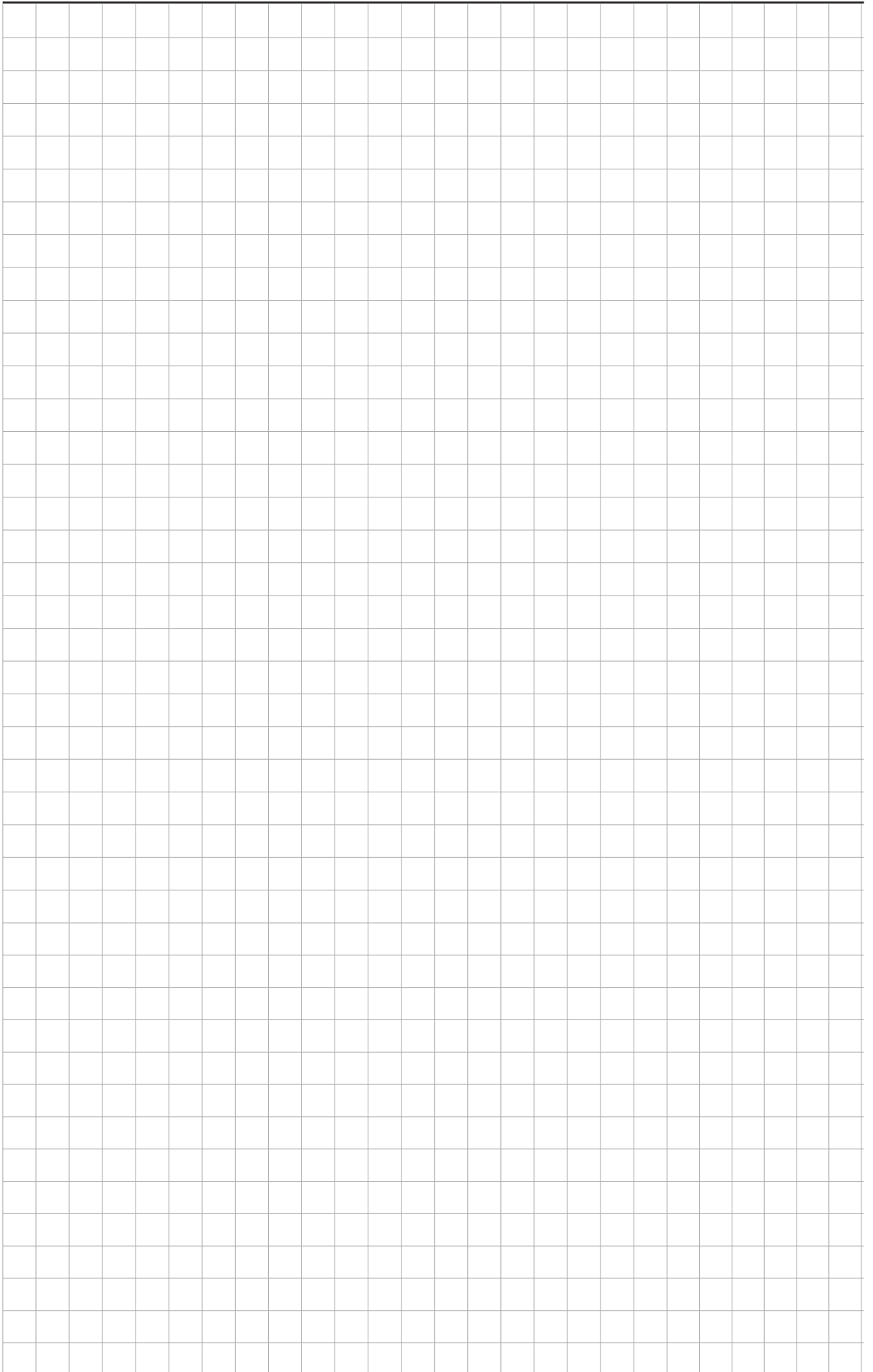
Note : Le bit 41_{hex} est envoyé en premier (y compris s'il s'agit du LSB), puis 12_{hex} est envoyé.

B. Calcul LRC (Checksum pour ASCII)

Exemple :

Adresse	01	00000001
Fonction	04	00000100
Adresse de mise en marche haut	00	00000000
Adresse de mise en marche bas	00	00000000
Nombre de registres	08	00001000
		Somme 00001101
	1. complément	11110010
		+ 1 00000001
	2. complément	11110101
Résultat LRC		F5_{hex}





LEGRAND
Pro and Consumer Service
BP 30076 - 87002
LIMOGES CEDEX FRANCE
www.legrand.com

Cachet installateur