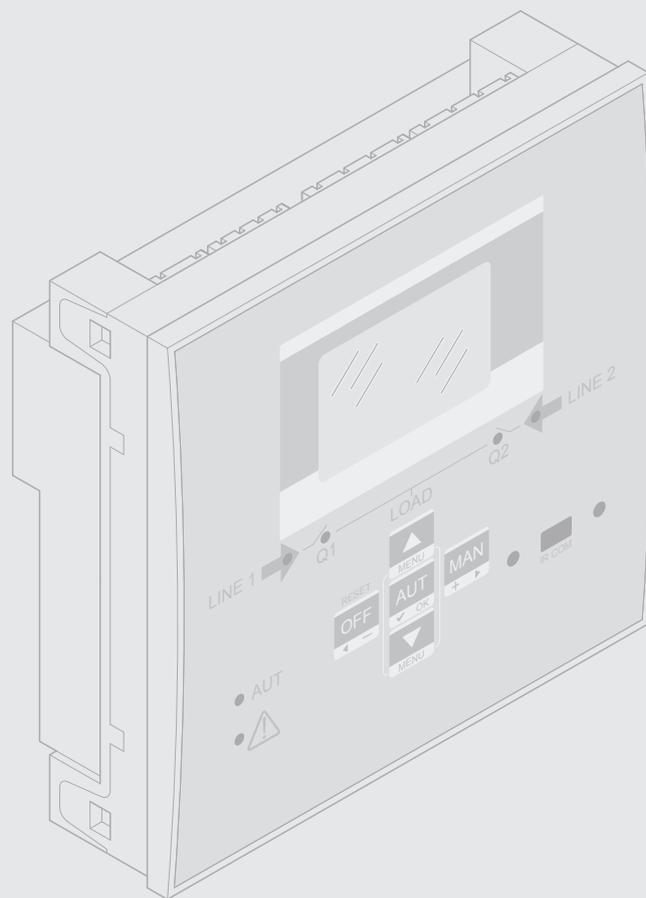


## Boîtier d'automatisme 4 226 82

FR

FRANÇAIS

3



## Table des matières

|   |           |
|---|-----------|
| <b>1. Protocole Modbus</b>                            | <b>4</b>  |
| <b>2. Réglage des paramètres</b>                      | <b>4</b>  |
| <b>3. Protocole Modbus RTU</b>                        | <b>5</b>  |
| <b>4. Fonctions Modbus</b>                            | <b>6</b>  |
| 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         |
| <b>5. Erreurs</b>                                     | <b>9</b>  |
| <b>6. Protocole Modbus ASCII</b>                      | <b>10</b> |
| <b>7. Tableaux</b>                                    | <b>11</b> |
| 7.1 Mesures données par le protocole de communication | 11        |
| 7.2 Bits d'état                                       | 13        |
| 7.3 Commandes   | 16        |
| 7.4 État général du dispositif                        | 18        |
| 7.5 Horloge à temps réel                              | 18        |
| <b>8. Lecture log événements</b>                      | <b>19</b> |
| <b>9. Réglage des paramètres</b>                      | <b>20</b> |
| A. Calcul CRC (Checksum pour RTU)                     | 22        |
| B. Calcul LRC (Checksum pour ASCII)                   | 23        |

## 1. Protocole Modbus

Le boîtier d'automatisme 4 226 82 est compatible avec les protocoles de communication Modbus (RTU et ASCII) sur l'interface optique et les modules d'extension :

- 4 226 87 Clé frontale IR-USB
- 4 226 88 Clé frontale IR-USB
- 4 226 89 RS485 Modules extensions

L'utilisation de cette fonction permet de lire l'état du dispositif et de commander l'unité par le logiciel de supervision ou autres dispositifs maître supportant le protocole Modbus (API par exemple).

## 2. Réglage des paramètres

Pour configurer le protocole Modbus, accéder au MENU RÉGLAGE et sélectionner le menu M08 du canal de communication (1 ou 2).

### Paramètres du menu

| M08- COMMUNICATION<br>(COMn, n=1...2) |                    | UdM | DÉFAUT       | PLAGE  |
|---------------------------------------|--------------------|-----|--------------|--|
| <b>P08.n.01</b>                       | Adresse série nœud |     | 05           | 01-247<br>(248 ... 255 usage interne)  |
| <b>P08.n.02</b>                       | Vitesse port série | bps | 19200        | 1200<br>2400<br>4800<br>9600<br>19200<br>38400<br>57600<br>115200                        |
| <b>P08.n.03</b>                       | Format donnée      |     | 8 bits, pair | 8 bits - pas de par.<br>8 bits, impair<br>8 bits, pair<br>7 bits, impair<br>7 bits, pair |
| <b>P08.n.04</b>                       | Bits stop          |     | 1            | 1-2  |
| <b>P08.n.05</b>                       | Protocole          |     | Modbus RTU   | Modbus RTU<br>Modbus ASCII   |

**Note:** ce menu est divisé en 2 sections pour les canaux de communication COM1...2. Le port de communication frontal IR pour la connexion avec **SW ACU** et **APP** via WiFi ou USB est fixe et ne nécessite aucun réglage.

- **P08.n.01** – Adresse série (nœud) du protocole de communication.
- **P08.n.02** – Vitesse de transmission du port de communication.
- **P08.n.03** – Format de la donnée. Uniquement réglage 7 bits pour protocole ASCII.
- **P08.n.04** – Nombre bit stop.
- **P08.n.05** – Sélection protocole de communication.
- **P08.n.06...P08.n.08** – Non disponible.
- **P08.n.09** – Non disponible.
- **P08.n.10** – Non disponible.
- **P08.n.11...P08.n.13** – Non disponible.

### 3. Protocole Modbus RTU

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

|        |                     |                      |                         |                  |        |
|--------|---------------------|----------------------|-------------------------|------------------|--------|
| T1T2T3 | Adresse<br>(8 bits) | Fonction<br>(8 bits) | Données<br>(N x 8 bits) | CRC<br>(16 bits) | 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 à l'esclave ou les données reçues de l'esclave en réponse à une demande (la longueur maximale pour le champ de données est de 80 registres de 16 bits, soit 160 octets).
- Le champ CRC permet aux dispositifs maître et esclave de contrôler l'intégrité du message.  
Si un message est corrompu 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 nécessaire à l'envoi de 3,5 caractères à la vitesse sélectionnée, le caractère qui suit est considéré comme le premier d'un nouveau message.

## 4. Fonctions Modbus

Les fonctions disponibles sont les suivantes :

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

Exemple :

Pour lire le nombre d'alarmes de commutation de l'interrupteur sur ligne 1 (Q1), présent sur la position 58 ( $3A_{hex}$ ), depuis le dispositif à adresse série 01, le message à envoyer est le suivant :

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 04 | 00 | 39 | 00 | 02 | A1 | C6 |
|----|----|----|----|----|----|----|----|

Où :

01 = adresse esclave

04 = fonction Modbus 'Read input register'

00 39 = Adresse du registre requis (nombre d'alarmes de commutation de l'interrupteur sur ligne 1) diminuée de 1

00 02 = Nombre de registres à lire à partir de l'adresse 22

A1 C6 = Checksum CRC

La réponse du boîtier d'automatisme sera la suivante :

|    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|
| 01 | 04 | 04 | 00 | 00 | 00 | 07 | BA | 46 |
|----|----|----|----|----|----|----|----|----|

Où :

01 = adresse boîtier d'automatisme (Esclave 01)

04 = fonction requise par le maître

04 = nombre de octets envoyés par le boîtier d'automatisme

00 00 00 07 = valeur Hex du nombre d'alarmes de commutation de l'interrupteur sur la ligne 1 = 7

BA 46 = 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).

| INTERROGATION MAÎTRE |                   |
|----------------------|-------------------|
| Adresse esclave      | 08 <sub>hex</sub> |
| Fonction             | 04 <sub>hex</sub> |
| Adresse MSB          | 00 <sub>hex</sub> |
| Adresse LSB          | 0F <sub>hex</sub> |
| Numéro registre MSB  | 00 <sub>hex</sub> |
| Numéro registre LSB  | 08 <sub>hex</sub> |
| LSB CRC              | C1 <sub>hex</sub> |
| MSB CRC              | 56 <sub>hex</sub> |

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

| RÉPONSE ESCLAVE     |                   |
|---------------------|-------------------|
| Adresse esclave     | 08 <sub>hex</sub> |
| Fonction            | 04 <sub>hex</sub> |
| Nombre de octets    | 10 <sub>hex</sub> |
| Registre MSB 10hex  | 00 <sub>hex</sub> |
| Registre LSB 10 hex | 00 <sub>hex</sub> |
| -----               | ----              |
| Registre MSB 17 hex | 00 <sub>hex</sub> |
| Registre LSB 17 hex | 00 <sub>hex</sub> |
| LSB CRC             | 8A <sub>hex</sub> |
| MSB CRC             | B1 <sub>hex</sub> |

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 le registre. Elle peut être utilisée avec les registres dont l'adresse est supérieure à 1000<sub>hex</sub>. 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 MAÎTRE      |                   |
|---------------------|-------------------|
| Adresse esclave     | 08 <sub>hex</sub> |
| Fonction            | 06 <sub>hex</sub> |
| Adresse MSB         | 2F <sub>hex</sub> |
| Adresse LSB         | 0F <sub>hex</sub> |
| Numéro registre MSB | 00 <sub>hex</sub> |
| Numéro registre LSB | 0A <sub>hex</sub> |
| LSB CRC             | 31 <sub>hex</sub> |
| MSB CRC             | 83 <sub>hex</sub> |

#### Réponse Esclave :

La réponse esclave est un écho de la requête, à savoir que l'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 MAÎTRE |                   |
|----------------------|-------------------|
| Adresse esclave      | 08 <sub>hex</sub> |
| Fonction             | 07 <sub>hex</sub> |
| LSB CRC              | 47 <sub>hex</sub> |
| MSB CRC              | B2 <sub>hex</sub> |

Le tableau ci-dessous indique la signification de l'octet d'état envoyé par le boîtier d'automatisme:

| BIT | SIGNIFICATION                      |
|-----|------------------------------------|
| 0   | Mode de fonctionnement OFF / Reset |
| 1   | Mode de fonctionnement MAN         |
| 2   | Mode de fonctionnement AUT         |
| 3   | Mode de fonctionnement TEST        |
| 4   | En erreur                          |
| 5   | Alimentation AC ok                 |
| 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 message unique ou de régler une valeur plus longue qu'un registre.

Exemple :

| MESSAGE MAÎTRE   |                   |
|--|-------------------|
| Adresse esclave  | 08 <sub>hex</sub> |
| Fonction   | 10 <sub>hex</sub> |
| Adresse registre MSB                                     | 20 <sub>hex</sub> |
| Adresse registre LSB                                     | 01 <sub>hex</sub> |
| Numéro registre MSB                                      | 00 <sub>hex</sub> |
| Numéro registre LSB                                      | 02 <sub>hex</sub> |
| Nombre de octets ( <i>le double de celui ci-dessus</i> ) | 04 <sub>hex</sub> |
| Données MSB  | 00 <sub>hex</sub> |
| Données LSB  | 00 <sub>hex</sub> |
| Données MSB  | 00 <sub>hex</sub> |
| Données LSB  | 00 <sub>hex</sub> |
| LSB CRC  | 85 <sub>hex</sub> |
| MSB CRC  | 3E <sub>hex</sub> |

| RÉPONSE ESCLAVE      |                   |
|----------------------|-------------------|
| Adresse esclave      | 08 <sub>hex</sub> |
| Fonction             | 10 <sub>hex</sub> |
| Adresse registre MSB | 20 <sub>hex</sub> |
| Adresse registre LSB | 01 <sub>hex</sub> |
| Nombre octet MSB     | 00 <sub>hex</sub> |
| Nombre octet LSB     | 02 <sub>hex</sub> |
| LSB CRC              | 1B <sub>hex</sub> |
| MSB CRC              | 51 <sub>hex</sub> |

#### 4.5 Fonction 17: report Slave ID

Cette fonction permet d'identifier le type de dispositif.

Exemple :

| INTERROGATION MAÎTRE |                   |
|----------------------|-------------------|
| Adresse esclave      | 08 <sub>hex</sub> |
| Fonction             | 11 <sub>hex</sub> |
| LSB CRC              | C6 <sub>hex</sub> |
| MSB CRC              | 7C <sub>hex</sub> |

| SLAVE RESPONSE                  |                   |
|---------------------------------|-------------------|
| Adresse esclave                 | 08 <sub>hex</sub> |
| Fonction                        | 11 <sub>hex</sub> |
| Compteur octets                 | 08 <sub>hex</sub> |
| Donnée 01 (Type) ❶              | 76 <sub>hex</sub> |
| Donnée 02 (version SW)          | 01 <sub>hex</sub> |
| Donnée 03 (version HW)          | 00 <sub>hex</sub> |
| Donnée 04 (révision paramètres) | 01 <sub>hex</sub> |
| Donnée 05 (type produit) ❷      | 04 <sub>hex</sub> |
| Donnée 06 (réservée)            | 00 <sub>hex</sub> |
| Donnée 07 (réservée)            | 00 <sub>hex</sub> |
| Donnée 08 (réservée)            | 00 <sub>hex</sub> |
| LSB CRC                         | B0 <sub>hex</sub> |
| MSB CRC                         | 2A <sub>hex</sub> |

❶  $118 - 76_{\text{hex}} = 4\ 226\ 82$

❷  $2 - 02_{\text{hex}} = \text{série Legrand}$

## 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<sub>hex</sub>, suivi d'un octet de code d'erreur.

Dans le tableau qui suit figurent les codes d'erreur envoyés au maître par l'esclave :

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

## 6. Protocole Modbus ASCII

Le protocole Modbus ASCII est principalement utilisé sur les applications qui nécessitent une communication par 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 saut à la ligne (Carriage Return CR) et un Line Feed LF à la place d'un pause de transmission.

Si le paramètre P7.05 est configuré comme protocole Modbus ASCII, le message de communication s'identifie de manière suivante :

|   |                           |                            |                           |                       |       |
|---|---------------------------|----------------------------|---------------------------|-----------------------|-------|
| : | Adresse<br>(2 caractères) | Fonction<br>(2 caractères) | Données<br>(N caractères) | LRC<br>(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 l'esclave doit exécuter.
- Le champ Données contient les données envoyées à l'esclave ou les données reçues de l'esclave en réponse à une demande. La longueur maximale admise est de 80 registres consécutifs.
- Le champ LRC permet aux dispositifs maître et esclave de contrôler l'intégrité du message. Si un message est corrompu 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 du courant de phase L3, présente sur la position 12 (0C<sub>hex</sub>), depuis l'esclave à adresse série 08, le message à envoyer est le suivant :

|   |    |    |    |    |    |    |    |      |
|---|----|----|----|----|----|----|----|------|
| : | 08 | 04 | 00 | 0B | 00 | 02 | E7 | CRLF |
|---|----|----|----|----|----|----|----|------|

Où :

: = ASCII 3A<sub>hex</sub> délimiteur début de message

08 = adresse esclave

04 = fonction Modbus 'Read input register'

00 0B = Adresse du registre requis (courant de phase L3) diminuée de 1

00 02 = Nombre de registres à lire à partir de l'adresse 04

E7 = Checksum LRC

CRLF = ASCII 0D<sub>hex</sub> 0A<sub>hex</sub> = Délimiteur de fin de message

La réponse est la suivante :

|   |    |    |    |    |    |    |    |    |      |
|---|----|----|----|----|----|----|----|----|------|
| : | 08 | 04 | 04 | 00 | 00 | A8 | AE | 9B | CRLF |
|---|----|----|----|----|----|----|----|----|------|

Où :

: = ASCII 3A<sub>hex</sub> délimiteur début de message

08 = adresse (Esclave 08)

04 = fonction requise par le maître

04 = nombre d'octets envoyés par le dispositif

00 00 A8 AE = valeur Hex du courant de phase de L3 (= 4.3182 A)

9B = checksum LRC

CRLF = ASCII 0D<sub>hex</sub> 0A<sub>hex</sub> = Délimiteur de fin de message

## 7. Tableaux

### 7.1 Mesures données par le protocole de communication

A utiliser avec fonctions 03 et 04

| ADRESSE           | WORDS | MESURE   | UNITÉ    | FORMAT        |
|-------------------|-------|--|----------|---------------|
| 02 <sub>hex</sub> | 2     | Tension Ligne 1 L1-N                           | V        | Unsigned long |
| 04 <sub>hex</sub> | 2     | Tension Ligne 1 L2-N                           | V        | Unsigned long |
| 06 <sub>hex</sub> | 2     | Tension Ligne 1 L3-N                           | V        | Unsigned long |
| 08 <sub>hex</sub> | 2     | Tension Ligne 1 L1-L2                          | V        | Unsigned long |
| 0A <sub>hex</sub> | 2     | Tension Ligne 1 L2-L3                          | V        | Unsigned long |
| 0C <sub>hex</sub> | 2     | Tension Ligne 1 L3-L1                          | V        | Unsigned long |
| 0E <sub>hex</sub> | 2     | Tension Ligne 2 L1-N                           | V        | Unsigned long |
| 10 <sub>hex</sub> | 2     | Tension Ligne 2 L2-N                           | V        | Unsigned long |
| 12 <sub>hex</sub> | 2     | Tension Ligne 2 L3-N                           | V        | Unsigned long |
| 14 <sub>hex</sub> | 2     | Tension Ligne 2 L1-L2                          | V        | Unsigned long |
| 16 <sub>hex</sub> | 2     | Tension Ligne 2 L2-L3                          | V        | Unsigned long |
| 18 <sub>hex</sub> | 2     | Tension Ligne 2 L3-L1                          | V        | Unsigned long |
| 1A <sub>hex</sub> | 2     | Fréquence Ligne 1 L3-L1                        | Hz/10    | Unsigned long |
| 1C <sub>hex</sub> | 2     | Fréquence Ligne 2 L3-L1                        | Hz/10    | Unsigned long |
| 1E <sub>hex</sub> | 2     | Tension de la batterie (alimentation DC)       | VDC / 10 | Unsigned long |
| 20 <sub>hex</sub> | 2     | Temps de fonctionnement total                  | s        | Unsigned long |
| 22 <sub>hex</sub> | 2     | Ligne 1 ok temps total                         | s        | Unsigned long |
| 24 <sub>hex</sub> | 2     | Ligne 2 ok temps total                         | s        | Unsigned long |
| 26 <sub>hex</sub> | 2     | Ligne 1 non ok temps total                     | s        | Unsigned long |
| 28 <sub>hex</sub> | 2     | Ligne 2 non ok temps total                     | s        | Unsigned long |
| 2A <sub>hex</sub> | 2     | Ligne 1 temps total interrupteur fermé         | s        | Unsigned long |
| 2C <sub>hex</sub> | 2     | Ligne 2 temps total interrupteur fermé         | s        | Unsigned long |
| 2E <sub>hex</sub> | 2     | Temps total interrupteur ouvert                | s        | Unsigned long |
| 30 <sub>hex</sub> | 2     | (non utilisé)                                  | --       | Unsigned long |
| 32 <sub>hex</sub> | 2     | Nombre de commutations interrupteur 1 en AUT   | nr       | Unsigned long |
| 34 <sub>hex</sub> | 2     | Nombre de commutations interrupteur 2 en AUT   | nr       | Unsigned long |
| 36 <sub>hex</sub> | 2     | Nombre de commutations interrupteur 1 en MAN   | nr       | Unsigned long |
| 38 <sub>hex</sub> | 2     | Nombre de commutations interrupteur 2 en MAN   | nr       | Unsigned long |
| 3A <sub>hex</sub> | 2     | Nombre d'alarmes de commutation interrupteur 1 | nr       | Unsigned long |
| 3C <sub>hex</sub> | 2     | Nombre d'alarmes de commutation interrupteur 2 | nr       | Unsigned long |
| 3E <sub>hex</sub> | 2     | (non utilisé)                                  | --       | Unsigned long |
| 40 <sub>hex</sub> | 2     | Alarmes <sup>(1)</sup>                         | bits     | Unsigned long |
| 50 <sub>hex</sub> | 2     | Tension batteries minimum                      | V        | Unsigned long |
| 52 <sub>hex</sub> | 2     | Tension batteries maximum                      | V        | Unsigned long |

(a continué)

## 7. Tableaux

| ADRESSE             | WORDS | MESURE                                    | UNITÉ | FORMAT        |
|---------------------|-------|---|-------|---------------|
| 54 <sub>hex</sub>   | 2     | Ligne 1 heures entretien                  | nr    | Unsigned long |
| 56 <sub>hex</sub>   | 2     | Ligne 2 heures entretien                  | nr    | Unsigned long |
| 58 <sub>hex</sub>   | 2     | Opérations d'entretien sur interrupteur 1 | nr    | Signed long   |
| 5A <sub>hex</sub>   | 2     | Opérations d'entretien sur interrupteur 2 | nr    | Signed long   |
| 21C0 <sub>hex</sub> | 1     | OR de toutes les limites                  | bits  | Unsigned int  |
| 21C1 <sub>hex</sub> | 1     | LIM 1                                     | bits  | Unsigned int  |
| 21C2 <sub>hex</sub> | 1     | LIM 2                                     | bits  | Unsigned int  |
| 21C3 <sub>hex</sub> | 1     | LIM 3                                     | bits  | Unsigned int  |
| 21C4 <sub>hex</sub> | 1     | LIM 4                                     | bits  | Unsigned int  |
| 1D00 <sub>hex</sub> | 2     | Compteur CNT 1                            | UM1   | long          |
| 1D02 <sub>hex</sub> | 2     | Compteur CNT 2                            | UM2   | long          |
| 1D04 <sub>hex</sub> | 2     | Compteur CNT 3                            | UM3   | long          |
| 1D06 <sub>hex</sub> | 2     | Compteur CNT 4                            | UM4   | long          |

<sup>(1)</sup> En lisant les words à partir de l'adresse 40<sub>hex</sub>, 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  | Panne de l'alimentation secourue externe      |
| 8   | A09  | Urgence                                       |
| 9   | A10  | Déclenchement protection interrupteur ligne 1 |
| 10  | A11  | Déclenchement protection interrupteur ligne 2 |
| 11  | A12  | Générateur non disponible ligne 1             |

| BIT     | CODE | ALARME                            |
|---------|------|-----------------------------------|
| 12      | A13  | Générateur non disponible ligne 2 |
| 13      | A14  | Heures entretien écoulées ligne 1 |
| 14      | A15  | Heures entretien écoulées ligne 2 |
| 15      | A16  | Opérations d'entretien ligne 1    |
| 16      | A17  | Opérations d'entretien ligne 2    |
| 17      | A18  | Panne tension auxiliaire          |
| 18      | UA1  | Alarme utilisateurs               |
| 19      | UA2  | Alarme utilisateurs               |
| 20      | UA3  | Alarme utilisateurs               |
| 21      | UA4  | Alarme utilisateurs               |
| 22 - 31 | ---  | Non utilisé                       |

### 7.2 Bits d'état

A utiliser avec les fonctions 03 et 04

| ADRESSE             | WORDS | FONCTION                              | FORMAT           |
|---------------------|-------|---------------------------------------|------------------|
| 2070 <sub>hex</sub> | 1     | État clavier panneau frontal ❶        | Unsigned integer |
| 2100 <sub>hex</sub> | 1     | État entrées numériques (par borne) ❷ | Unsigned integer |
| 2140 <sub>hex</sub> | 1     | État sorties numériques (par borne) ❸ | Unsigned integer |
| -                   | -     | -                                     | -                |
| 2074 <sub>hex</sub> | 1     | État tension ligne 1 ❹                | Unsigned integer |
| 2075 <sub>hex</sub> | 1     | État interrupteur ligne 1 ❺           | Unsigned integer |
| 2176 <sub>hex</sub> | 1     | État tension ligne 2 ❹                | Unsigned integer |
| 2177 <sub>hex</sub> | 1     | État interrupteur ligne 2 ❺           | Unsigned integer |
| 2078 <sub>hex</sub> | 2     | État fonctions entrées ❻              | Unsigned integer |
| 207A <sub>hex</sub> | 1     | État fonctions sorties ❼              | Unsigned integer |
| 207B <sub>hex</sub> | 1     | État messages écran ❽                 | Unsigned integer |
| 207C <sub>hex</sub> | 1     | État contrôleur général ❾             | Unsigned integer |
| 207E <sub>hex</sub> | 1     | État LED frontale                     | Unsigned integer |
| 207F <sub>hex</sub> | 1     | État LED frontale                     | Unsigned integer |

❶ Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 2070<sub>hex</sub> :

| BIT    | TOUCHE      |
|--------|-------------|
| 0      | HAUT        |
| 1      | OFF/RESET   |
| 2      | MAN         |
| 3      | BAS         |
| 4      | AUT/ENTER   |
| 5...15 | Non utilisé |

## 7. Tableaux

② Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 2100<sub>hex</sub> :

| BIT | ENTRÉE   |
|-----|----------|
| 0   | Entrée 1 |
| 1   | Entrée 2 |
| 2   | Entrée 3 |
| 3   | Entrée 4 |
| 4   | Entrée 5 |
| 5   | Entrée 6 |
| 6   | Entrée 7 |
| 7   | Entrée 8 |

| BIT   | ENTRÉE      |
|-------|-------------|
| 8     | Entrée 9    |
| 9     | Entrée 10   |
| 10    | Entrée 11   |
| 11    | Entrée 12   |
| 12    | Entrée 13   |
| 13    | Entrée 14   |
| 14-15 | Non utilisé |

③ Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 2072<sub>hex</sub> :

| BIT | SORTIE   |
|-----|----------|
| 0   | Sortie 1 |
| 1   | Sortie 2 |
| 2   | Sortie 3 |
| 3   | Sortie 4 |
| 4   | Sortie 5 |
| 5   | Sortie 6 |
| 6   | Sortie 7 |
| 7   | Sortie 8 |

| BIT | SORTIE      |
|-----|-------------|
| 8   | Sortie 9    |
| 9   | Sortie 10   |
| 10  | Sortie 11   |
| 11  | Sortie 12   |
| 12  | Sortie 13   |
| 13  | Sortie 14   |
| 14  | Sortie 15   |
| 15  | Non utilisé |

④ Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 2074<sub>hex</sub> (Ligne 1) et à l'adresse 2176<sub>hex</sub> (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.                 |

| BIT   | ÉTAT LIGNE                      |
|-------|---------------------------------|
| 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 de chaque bits du word se trouvant à l'adresse 2075<sub>hex</sub> (Ligne 1) et à l'adresse 2177<sub>hex</sub> (Ligne 2) :

| BIT           | CONDITION INTERRUPTEUR        |
|---------------|-------------------------------|
| <b>0</b>      | Interrupteur fermé            |
| <b>1</b>      | Alarme déclenchement          |
| <b>2</b>      | Non utilisé                   |
| <b>3</b>      | État commande (1 = fermeture) |
| <b>4</b>      | Sortie commande de fermeture  |
| <b>5</b>      | Sortie commande d'ouverture   |
| <b>6...15</b> | Non utilisé                   |

- ⑥ Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 2178<sub>hex</sub> :

| BIT      | ÉTAT FONCTIONS ENTRÉES                 |
|----------|--|
| <b>0</b> | Interrupteur ligne 1 fermé             |
| <b>1</b> | Déclenchement interrupteur ligne 1     |
| <b>2</b> | Non utilisé                            |
| <b>3</b> | Interrupteur ligne 2 fermé             |
| <b>4</b> | Déclenchement interrupteur ligne 2     |
| <b>5</b> | Non utilisé                            |
| <b>6</b> | Commutation sur ligne secondaire       |
| <b>7</b> | Inhibition retour sur ligne principale |

| BIT       | ÉTAT FONCTIONS ENTRÉES    |
|-----------|---------------------------|
| <b>8</b>  | Bouton d'urgence          |
| <b>9</b>  | Mise en marche générateur |
| <b>10</b> | Générateur 1 prêt         |
| <b>11</b> | Générateur 2 prêt         |
| <b>12</b> | Clavier verrouillé        |
| <b>13</b> | Blocage paramètres        |
| <b>14</b> | Non utilisé               |
| <b>15</b> | Alarmes bloqués           |

- ⑦ Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 207A<sub>hex</sub> :

| BIT      | ÉTAT FONCTIONS SORTIES      |
|----------|-----------------------------|
| <b>0</b> | Interrupteur ligne 1 ouvert |
| <b>1</b> | Interrupteur ligne 1 fermé  |
| <b>2</b> | Interrupteur ligne 2 ouvert |
| <b>3</b> | Interrupteur ligne 2 fermé  |
| <b>4</b> | Alarme générale             |
| <b>5</b> | Mise en marche générateur 1 |
| <b>6</b> | Mise en marche générateur 2 |
| <b>7</b> | ATS prêt                    |

| BIT       | ÉTAT FONCTIONS SORTIES |
|-----------|------------------------|
| <b>8</b>  | Délestage              |
| <b>9</b>  | Non utilisé            |
| <b>10</b> | Non utilisé            |
| <b>11</b> | Tout ouvrir            |
| <b>12</b> | Bobine sous tension 1  |
| <b>13</b> | Bobine sous tension 2  |
| <b>14</b> | Ligne 1 OK             |
| <b>15</b> | Ligne 2 OK             |

## 7. Tableaux

⑧ Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 207B<sub>hex</sub> :

| BIT | ÉTAT MESSAGES ÉCRAN          |
|-----|------------------------------|
| 0   | Mise en marche générateur 1  |
| 1   | Mise en marche générateur 2  |
| 2   | Refroidissement générateur 1 |
| 3   | Refroidissement générateur 2 |
| 4   | Commutation charge 2 → 1     |
| 5   | Commutation charge 1 → 2     |

⑨ Le tableau suivant indique la signification de chaque bits du word se trouvant à l'adresse 207C<sub>hex</sub> :

| BIT    | ÉTAT FONCTIONS SORTIES             |
|--------|------------------------------------|
| 0      | Mode de fonctionnement OFF / Reset |
| 1      | Mode de fonctionnement MAN         |
| 2      | Mode de fonctionnement AUT         |
| 3      | Mode de fonctionnement TEST        |
| 4      | En erreur                          |
| 5      | Alimentation AC présente           |
| 6      | Alimentation DC présente           |
| 7      | Alarme générale active (on)        |
| 8...15 | Non utilisé                        |

### 7.3 Commandes

A utiliser avec fonction 06

| ADRESSE             | WORDS | ÉTAT   |
|---------------------|-------|--|
| 4F00 <sub>hex</sub> | 1     | Réglage variable à distance REM1 ①                             |
| 4F01 <sub>hex</sub> | 1     | Réglage variable à distance REM2                               |
| .....               |       |  |
| 4F07 <sub>hex</sub> | 1     | Réglage variable à distance REM8                               |
| 2F00 <sub>hex</sub> |       | Changement de mode de fonctionnement ②                         |
| 2F0A <sub>hex</sub> | 1     | Simulation pression touches panneau frontal ③                  |
| 2F03 <sub>hex</sub> | 1     | Valeur 01 <sub>hex</sub> : sauvegarde mémoire                  |
|                     |       | Valeur 04 <sub>hex</sub> : reboot                              |
| 2F07 <sub>hex</sub> | 1     | Valeur 00 <sub>hex</sub> : Reset dispositif                    |
|                     |       | Valeur 01 <sub>hex</sub> : Reset dispositif et sauvegarde Fram |
| 2FF0 <sub>hex</sub> | 1     | Exécution menu de commande ④                                   |
| 28FA <sub>hex</sub> | 1     | Valeur 01 <sub>hex</sub> : Sauvegarde réglage heure/date       |

- ① En écrivant la valeur  $AA_{hex}$  à l'adresse indiquée, la variable à distance est réglée sur 1 en écrivant  $BB_{hex}$ , la variable à distance est réglée sur 0
- ② Le tableau suivant indique les valeurs à écrire à l'adresse  $2F00_{hex}$  pour obtenir la fonction correspondante :

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

- ③ Le tableau suivant indique la position de bit de la valeur à écrire à l'adresse  $2F0A_{hex}$  pour obtenir la fonction correspondante :

| BIT | SIGNIFICATION |
|-----|---------------|
| 0   | Touche Haut   |
| 1   | Mode MAN      |
| 2   | Touche Droite |
| 3   | START         |
| 4   | Mode TEST     |
| 5   | Mode OFF      |
| 6   | Mode AUT      |
| 7   | Mode STOP     |

- ④ En écrivant une valeur comprise entre 0 et 15 à l'adresse indiquée, la commande correspondante est exécutée :

| BIT | SIGNIFICATION                        |
|-----|--------------------------------------|
| 0   | Reset entretien 1                    |
| 1   | Reset entretien 2                    |
| 2   | Reset opérations entretien 1         |
| 3   | Reset opérations entretien 2         |
| 4   | Reset compteurs généraux CNTx        |
| 5   | Reset limites LIMx                   |
| 6   | Reset compteur heure ligne 1/ligne 2 |
| 7   | Reset compteur heure Q 1/ Q 2        |

| BIT | SIGNIFICATION                                |
|-----|--|
| 8   | Reset opération interrupteur                 |
| 9   | Reset liste événements                       |
| 10  | Reset paramètres par défaut                  |
| 11  | Sauvegarde paramètres dans mémoire de backup |
| 12  | Chargement paramètres de mémoire de backup   |
| 13  | I/O forçage                                  |
| 14  | Reset A03 – A04 alarmes                      |
| 15  | Simulation coupure de ligne                  |

## 7. Tableaux

### 7.4 État général du dispositif

A utiliser avec les fonctions 03 et 04.

| ADRESSE             | WORDS | ÉTAT                                   | FORMAT           |
|---------------------|-------|--|------------------|
| 2210 <sub>hex</sub> | 2     | État global dispositif (bit 0-bit31) ❶ | Unsigned integer |

❶ En lisant deux words à l'adresse 2210<sub>hex</sub>, 32 bits sont restitués (pour leur signification voir ci-dessous) :

| BIT    | SIGNIFICATION               |
|--------|-----------------------------|
| Bit 0  | Dispositif OFF              |
| Bit 1  | Dispositif en mode MAN      |
| Bit 2  | Dispositif en mode AUT      |
| Bit 3  | Dispositif en mode TEST     |
| Bit 4  | Tension Ligne 1 OK          |
| Bit 5  | Tension Ligne 2 OK          |
| Bit 6  | LED Ligne 1 on              |
| Bit 7  | LED Ligne 2 on              |
| Bit 8  | LED Ligne 1 sur la charge   |
| Bit 9  | LED Ligne 1 sur la charge   |
| Bit 10 | Contacteur secteur fermé    |
| Bit 11 | Contacteur générateur fermé |
| Bit 12 | Alarme générale             |
| Bit 13 | Alimentation AC             |
| Bit 14 | Start Générateur 1          |
| Bit 15 | Start Générateur 2          |

| BIT    | SIGNIFICATION            |
|--------|--------------------------|
| Bit 16 | Ligne 1 max Tens         |
| Bit 17 | Ligne 1 min Tens         |
| Bit 18 | Ligne 1 max Hz           |
| Bit 19 | Ligne 1 min Hz           |
| Bit 20 | Ligne 1 absence de phase |
| Bit 21 | Ligne 1 asym             |
| Bit 22 | Ligne 2 max Tens         |
| Bit 23 | Ligne 2 min Tens         |
| Bit 24 | Ligne 2 Hz max           |
| Bit 25 | Ligne 2 min Hz           |
| Bit 26 | Ligne 2 absence de phase |
| Bit 27 | Ligne 2 asym             |
| Bit 28 | (non utilisé)            |
| ...    | .....                    |
| Bit 31 | (non utilisé)            |

### 7.5 Horloge à temps réel

A utiliser avec fonctions 04 et 06.

Pour rendre effectives les modifications, il faut les mémoriser en utilisant la commande dédiée décrite dans le tableau 7.3.

| ADRESSE             | WORDS | FONCTION | PLAGE      |
|---------------------|-------|----------|------------|
| 28F0 <sub>hex</sub> | 1     | Année    | 2000..2099 |
| 28F1 <sub>hex</sub> | 1     | Mois     | 1-12       |
| 28F2 <sub>hex</sub> | 1     | Jour     | 1-31       |
| 28F3 <sub>hex</sub> | 1     | Heures   | 0-23       |
| 28F4 <sub>hex</sub> | 1     | Minutes  | 0-59       |
| 28F5 <sub>hex</sub> | 1     | Secondes | 0-59       |

## 8. Lecture log événements

Pour lire les événements, il est nécessaire de procéder comme suit :

1. Effectuer la lecture de 1 registre en utilisant la **fonction 04** à l'adresse 5030<sub>hex</sub>, l'octet le plus significatif (MSB) indique le nombre d'événements mémorisés (valeur comprise entre 0 et 100), l'octet le moins significatif est augmenté chaque fois qu'un événement est sauvegardé (valeur comprise entre 0 et 100). Une fois que 100 événements sont mémorisés, le MSB reste à 100 tandis que le LSB est remis à zéro puis continue à augmenter.
2. Régler l'indice de l'événement à lire (inférieur au nombre maximum d'événements mémorisés) à cet effet, exécuter la **fonction 06** à l'adresse 5030<sub>hex</sub> en indiquant l'événement à lire.
3. Effectuer une lecture de 43 registres (avec une unique **fonction 04**) à l'adresse 5032<sub>hex</sub>.
4. La valeur renvoyée est une chaîne de 86 caractères ASCII qui a la même description de l'événement que celle affichée sur l'écran du boîtier d'automatisme. L'indice de l'événement à lire est augmenté automatiquement après la lecture du registre 5032<sub>hex</sub>, à fin d'accélérer le téléchargement des événements.
5. Pour lire l'événement suivant répéter l'étape 4, pour lire tout autre événement effectuer la phase 3.

Exemple :

**Phase 1:** Lecture des événements mémorisés.

MASTER      Fonction = 4            (04<sub>hex</sub>)  
                  Adresse = 5030<sub>hex</sub>    (5030<sub>hex</sub> - 0001<sub>hex</sub> = 502F<sub>hex</sub>)  
                  Nb registres = 1        (01<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 04 | 50 | 2F | 00 | 01 | 11 | 03 |
|----|----|----|----|----|----|----|----|

Boîtier  
d'automatisme      Fonction = 4            (04<sub>hex</sub>)  
                  Nb octets = 1        (01<sub>hex</sub>)  
                  MSB = 100            (64<sub>hex</sub>)  
                  LSB = 2                (02<sub>hex</sub>)

|    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|
| 01 | 04 | 02 | 64 | 42 | 13 | C1 |
|----|----|----|----|----|----|----|

**Phase 2:** Régler l'indice de l'événement à lire.

MASTER      Fonction = 6            (06<sub>hex</sub>)  
                  Adresse = 5030<sub>hex</sub>    (5030<sub>hex</sub> - 0001<sub>hex</sub> = 502F<sub>hex</sub>)  
                  Valeur = 1            (01<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 50 | 2F | 00 | 01 | 68 | C3 |
|----|----|----|----|----|----|----|----|

Boîtier  
d'automatisme      Fonction = 6            (06<sub>hex</sub>)  
                  Adresse = 5030<sub>hex</sub>    (5030<sub>hex</sub> - 0001<sub>hex</sub> = 502F<sub>hex</sub>)  
                  Valeur = 1            (01<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 50 | 2F | 00 | 01 | 68 | C3 |
|----|----|----|----|----|----|----|----|

**Phase 3:** Lecture de l'événement.

MASTER      Fonction = 4            (04<sub>hex</sub>)  
                  Adresse = 5032<sub>hex</sub>    (5032<sub>hex</sub> - 0001<sub>hex</sub> = 5031<sub>hex</sub>)  
                  Nb registres = 43      (2B<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 04 | 50 | 31 | 00 | 2B | F0 | DA |
|----|----|----|----|----|----|----|----|

Boîtier  
d'automatisme      Fonction = 4            (04<sub>hex</sub>)  
                  Adresse = 5030<sub>hex</sub>    (5030<sub>hex</sub> - 0001<sub>hex</sub> = 502F<sub>hex</sub>)  
                  Nb octets = 86        (56<sub>hex</sub>)

Chaîne = 2012/07/18;09:34:52;E1100,CHANGE MODE TO: MODE OFF

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 01 | 04 | 56 | 32 | 30 | 31 | 30 | 2F | 30 | 31 | 2F | 30 | 31 | 3B | 30 | 30 | 3A | 31 | 34 | 3A |
| 30 | 31 | 3B | 45 | 30 |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

## 9. Réglage des paramètres

En utilisant le protocole Modbus, il est possible d'accéder aux paramètres des menus.

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.

### PROCÉDURE POUR LIRE LES PARAMÈTRES

1. Écrire la valeur du menu à lire en utilisant la **fonction 06** à l'adresse  $5000_{hex}$  ①.
2. Écrire la valeur du sous-menu voulu (si présent) en utilisant la **fonction 06** à l'adresse  $5001_{hex}$  ①.
3. Écrire la valeur du paramètre voulu en utilisant la **fonction 06** à l'adresse  $5002_{hex}$  ①.
4. Exécuter la **fonction 04** à l'adresse  $5004_{hex}$  avec un nombre de registres adapté à la longueur du paramètre (voir tableau).
5. Pour lire le paramètre suivant (dans le même menu/sous-menu), répéter l'étape 4, dans le cas contraire répéter l'étape 1.

### PROCÉDURE POUR ÉCRIRE LES PARAMÈTRES

1. Écrire la valeur du menu à modifier en utilisant la **fonction 06** à l'adresse  $5000_{hex}$  ①.
2. Écrire la valeur du sous-menu à modifier (si présent) en utilisant la **fonction 06** à l'adresse  $5001_{hex}$  ①.
3. Écrire la valeur du paramètre à modifier en utilisant la **fonction 06** à l'adresse  $5001_{hex}$  ①.
4. Exécuter la **fonction 16** à l'adresse  $5004_{hex}$  avec un nombre de registres adapté à la longueur du paramètre.
5. Pour écrire le paramètre suivant dans le même menu/sous-menu, répéter l'étape 4, dans le cas contraire passer à l'étape 6.
6. 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 (écrire la valeur 04 en utilisant la **fonction 06** à l'adresse  $2F03_{hex}$ ).

| TYPE DE PARAMÈTRE                                   | NOMBRE DE REGISTRE       |
|---|--------------------------|
| Longueur texte 6 caractères (ex. : M14.0x.06)       | 3 registres (6 octets)   |
| Longueur texte 16 caractères (ex. : M14.0x.05)      | 8 registres (16 octets)  |
| Longueur texte 20 caractères (ex. : (ex. M15.0x.03) | 10 registres (20 octets) |
| Abs (valeur numérique) < 32768 (par exemple M01.05) | 1 registre (2 octets)    |
| Abs (valeur numérique) > 32768 (par exemple M12.01) | 2 registres (4 octets)   |
| Adresse IP (ex. : M08.0x.06 M08.0x.07)              | 2 registres (4 octets)   |

- ① Il est possible de lire les menus, les sous-menus et les paramètres mémorisés aux adresses  $5000_{hex}$ ,  $5001_{hex}$  et  $5002_{hex}$  en utilisant la **fonction 04**.

Exemple :

Régler la valeur du paramètre M08.01.01 sur 8

**Phase 1** : Régler le menu 08.

MAITRE      Fonction = 6      ( $06_{hex}$ )  
 Adresse =  $5000_{hex}$       ( $5000_{hex} - 0001_{hex} = 4FFF_{hex}$ )  
 Valeur = 8      ( $08_{hex}$ )

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 4F | FF | 00 | 08 | AE | E8 |
|----|----|----|----|----|----|----|----|

Boîtier  
d'automatisme      Fonction = 6      ( $06_{hex}$ )  
 Adresse =  $5000_{hex}$       ( $5000_{hex} - 0001_{hex} = 4FFF_{hex}$ )  
 Valeur = 8      ( $08_{hex}$ )

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 4F | FF | 00 | 08 | AE | E8 |
|----|----|----|----|----|----|----|----|

**Phase 2 : Régler le sous-menu 01.**

MAITRE            Fonction = 6            (06<sub>hex</sub>)  
                     Adresse = 5001<sub>hex</sub>    (5001<sub>hex</sub> - 0001<sub>hex</sub> = 5000<sub>hex</sub>)  
                     Valeur = 1            (01<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 50 | 00 | 00 | 01 | 59 | 0A |
|----|----|----|----|----|----|----|----|

Boîtier  
d'automatisme    Fonction = 6            (06<sub>hex</sub>)  
                     Adresse = 5001<sub>hex</sub>    (5001<sub>hex</sub> - 0001<sub>hex</sub> = 5000<sub>hex</sub>)  
                     Valeur = 1            (01<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 50 | 00 | 00 | 01 | 59 | 0A |
|----|----|----|----|----|----|----|----|

**Phase 3 : Régler le paramètre 01.**

MAITRE            Fonction = 6            (06<sub>hex</sub>)  
                     Adresse = 5002<sub>hex</sub>    (5002<sub>hex</sub> - 0001<sub>hex</sub> = 5001<sub>hex</sub>)  
                     Valeur = 1            (01<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 50 | 01 | 00 | 01 | 08 | CA |
|----|----|----|----|----|----|----|----|

Boîtier  
d'automatisme    Fonction = 6            (06<sub>hex</sub>)  
                     Adresse = 5002<sub>hex</sub>    (5002<sub>hex</sub> - 0001<sub>hex</sub> = 5001<sub>hex</sub>)  
                     Valeur = 1            (01<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 06 | 50 | 01 | 00 | 01 | 08 | CA |
|----|----|----|----|----|----|----|----|

**Phase 3 : Régler valeur 8.**

MAITRE            Fonction = 16          (10<sub>hex</sub>)  
                     Adresse = 5004<sub>hex</sub>    (5004<sub>hex</sub> - 0001<sub>hex</sub> = 5003<sub>hex</sub>)  
                     Nb Registre = 1      (01<sub>hex</sub>)  
                     Nb octets = 2        (02<sub>hex</sub>)  
                     Valeur = 8            (0008<sub>hex</sub>)

|    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 01 | 10 | 50 | 03 | 00 | 02 | 04 | 00 | 00 | 00 | 08 | 4E | 7F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|

Boîtier  
d'automatisme    Fonction = 16          (10<sub>hex</sub>)  
                     Adresse = 5004<sub>hex</sub>    (5004<sub>hex</sub> - 0001<sub>hex</sub> = 5003<sub>hex</sub>)  
                     Valeur = 2            (02<sub>hex</sub>)

|    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|
| 01 | 10 | 50 | 03 | 00 | 02 | A0 | C8 |
|----|----|----|----|----|----|----|----|

**Phase 6 : Sauvegarde et reboot**

MAITRE            Fonction = 6            (06<sub>hex</sub>)  
                     Adresse = 2F03<sub>hex</sub>    (2F03<sub>hex</sub> - 0001<sub>hex</sub> = 2F02<sub>hex</sub>)  
                     Valeur = 4            (04<sub>hex</sub>)

|    |   |    |    |    |    |    |    |
|----|---|----|----|----|----|----|----|
| 01 | 6 | 2F | 02 | 00 | 04 | 21 | 1D |
|----|---|----|----|----|----|----|----|

Boîtier  
d'automatisme    Aucune réponse.

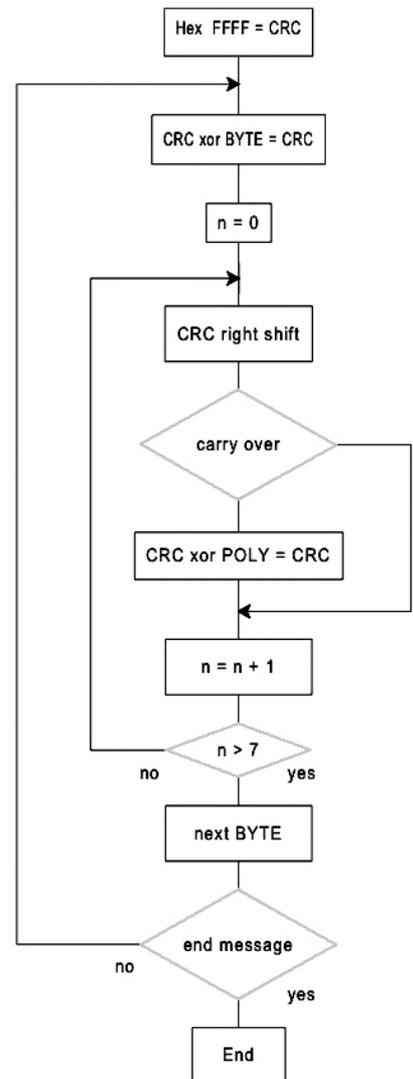
## 9. Réglage des paramètres

### A. CALCUL CRC (CHECKSUM pour RTU)

Algorithme de calcul CRC

Exemple : Frame = 0207<sub>hex</sub>

|   |             |             |                         |                         |
|---|-------------|-------------|-------------------------|-------------------------|
| <b>Initialisation CRC</b>                     | <b>1111</b> | <b>1111</b> | <b>1111</b>             | <b>1111</b>             |
| Charge premier octet                          |             |             | 0000                    | 0010                    |
| Exécution xor avec le premier octet du frame  | 1111        | 1111        | 1111                    | 1101                    |
| Exécution premier shift à droite              | 0111        | 1111        | 1111                    | 1110 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1101        | 1111        | 1111                    | 1111                    |
| Exécution 2 <sup>nd</sup> shift à droite      | 0110        | 1111        | 1111                    | 1111 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1100        | 1111        | 1111                    | 1110                    |
| Exécution 3 <sup>ème</sup> shift à droite     | 0110        | 0111        | 1111                    | 1111 <b>0</b>           |
| Exécution 4 <sup>ème</sup> shift à droite     | 0011        | 0011        | 1111                    | 1111 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1001        | 0011        | 1111                    | 1110                    |
| Exécution 5 <sup>ème</sup> shift à droite     | 0100        | 1001        | 1111                    | 1111 <b>0</b>           |
| Exécution 6 <sup>ème</sup> shift à droite     | 0010        | 0100        | 1111                    | 1111 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1000        | 0100        | 1111                    | 1110                    |
| Exécution 7 <sup>ème</sup> shift à droite     | 0100        | 0010        | 0111                    | 1111 <b>0</b>           |
| Exécution 8 <sup>ème</sup> shift à droite     | 0010        | 0001        | 0011                    | 1111 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Charge deuxième octet du frame                |             |             | 0000                    | 0111                    |
| Exécution xor avec le deuxième octet du frame | 1000        | 0001        | 0011                    | 1001                    |
| Exécution premier shift à droite              | 0100        | 0000        | 1001                    | 1100 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1110        | 0000        | 1001                    | 1101                    |
| Exécution 2 <sup>nd</sup> shift à droite      | 0111        | 0000        | 0100                    | 1110 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1101        | 0000        | 0100                    | 1111                    |
| Exécution 3 <sup>ème</sup> shift à droite     | 0110        | 1000        | 0010                    | 0111 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1100        | 1000        | 0010                    | 0110                    |
| Exécution 4 <sup>ème</sup> shift à droite     | 0110        | 0100        | 0001                    | 0011 <b>0</b>           |
| Exécution 5 <sup>ème</sup> shift à droite     | 0010        | 0100        | 0000                    | 1001 <b>1</b>           |
| Carry=1, charge polynôme                      | 1010        | 0000        | 0000                    | 0001                    |
| Exécution xor avec polynôme                   | 1001        | 0010        | 0000                    | 1000                    |
| Exécution 6 <sup>ème</sup> shift à droite     | 0100        | 1001        | 0000                    | 0100 <b>0</b>           |
| Exécution 7 <sup>ème</sup> shift à droite     | 0010        | 0100        | 1000                    | 0010 <b>0</b>           |
| Exécution 8 <sup>ème</sup> shift à droite     | 0001        | 0010        | 0100                    | 0001 <b>0</b>           |
| <b>Résultat CRC</b>                           | <b>0001</b> | <b>0010</b> | <b>0100</b>             | <b>0001</b>             |
|   |             |             | <b>12<sub>hex</sub></b> | <b>41<sub>hex</sub></b> |



**Note:** L'octet 41<sub>hex</sub> est envoyé en premier (y compris s'il s'agit du LSB), puis 12<sub>hex</sub> 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                |
| <b>Résultat LRC</b>            |       | <b>F5<sub>hex</sub></b> |

LEGRAND  
Pro and Consumer Service  
BP 30076 - 87002  
LIMOGES CEDEX FRANCE  
[www.legrand.com](http://www.legrand.com)

Cachet installateur