

**NOTICE
D'UTILISATION**

Datalog H60

Centrale de communication

FR



<http://www.socomec.com/en/datalog-H60>



1. GLOSSAIRE	7
2. REMARQUES CONCERNANT CE MANUEL	8
2.1. Champ d'application	8
2.2. Versions de produits	8
2.3. Groupe cible	8
2.4. Principe général de fonctionnement	8
2.5. Spécifications techniques	9
3. CONSIGNES DE SÉCURITÉ	10
4. DÉBALLAGE	11
4.1. Contenu de la livraison	11
4.2. Identification du datalogger	11
4.2.1. Étiquette signalétique	11
4.2.2. Version du logiciel	11
5. MONTAGE	12
5.1. Choix de l'environnement	12
5.2. Montage du Datalog H60 sur pied	12
5.3. Montage du Datalog H60 sur rail DIN	13
5.4. Montage murale du Datalog H60	13
5.5. Montage de l'antenne GPRS/GSM standard coudée	14
5.5.1. Spécifications	14
5.6. Montage de l'antenne GSM/GPRS déportée (option)	15
6. RACCORDEMENT ÉLECTRIQUE	16
6.1. Aperçu des interfaces externes et des LEDs	16
6.1.1. Face avant du boîtier	16
6.1.2. Face arrière du boîtier	17
6.2. Raccordement du datalogger DATALOG H60 au réseau Internet pour la communication avec le serveur distant	17
6.2.1. Connexion à Internet via une ligne ADSL	17
6.2.2. Connexion à Internet via une liaison GPRS ou 3G	18
6.3. Raccordement de la sortie TOR du datalogger	19
6.4. Raccordement des entrées TOR du datalogger	19
6.5. Raccordement du datalogger aux équipements Modbus via la liaison RS485	19
6.6. Raccordement du datalogger aux équipements via la liaison RS232	20
7. RACCORDEMENT À UN PC LOCAL VIA ETHERNET POUR LA CONFIGURATION ..	21
7.1. Connexion directe du PC au datalogger	21
7.2. Connexion du PC au datalogger dans un réseau local	21
7.3. Configuration réseau du PC pour accéder au datalogger	22
8. RACCORDEMENT DE L'ALIMENTATION	23
9. MISE EN SERVICE DU DATALOGGER	24
9.1. Connexion à l'interface Web intégrée	24
9.2. Configuration de la connectivité du datalogger	25
9.2.1. Configuration de la connexion modem	26
9.2.2. Configuration de la connexion Ethernet	27
9.2.3. Configuration du serveur FTP distant	28
9.2.4. Configuration du fuseau horaire	28
9.2.5. Configuration des transferts de données	29
9.2.6. Bouton request	29
9.3. Configuration système	30
9.3.1. Configuration des ports	30
9.4. Configuration des alarmes	30
9.4.1. Alarmes Système	31
9.4.2. Alarmes Entrées TOR	32
9.4.3. Alarmes Sortie TOR	32

9.5. Configuration des schedules	33
9.6. Configuration du metering	36
9.6.1. Entrées impulsionsnelles	36
9.6.2. M-Bus	36
9.6.3. Wireless Mbus 868MHz	37
9.7. Configuration Modbus	38
9.8. Actions exécutables	40
9.8.1. Demande de connexion au serveur distant	40
9.8.2. Demande de scan M-Bus	41
9.8.3. Demande de reboot	41
9.8.4. Téléchargement de fichiers système	41
9.9. Demande de connexion au serveur distant	41
10. VÉRIFICATION DU BON FONCTIONNEMENT DU DATALOGGER	42
11. SERVEUR FTP LOCAL	43
12. ARRÊT DU DATALOG H60	44
13. COMMUNICATION AVEC LE SERVEUR DISTANT	45
13.1. Modes de connexion	45
13.1.1. Gestion du code PIN de la carte SIM	45
13.2. Serveur FTP distant	46
13.3. Serveur Web Services distant	46
13.4. Connexion au serveur distant	47
13.4.1. Process de téléchargement sur le serveur distant (Upload)	48
13.4.2. Inbox	50
13.5. Bouton Request	50
14. COMMANDES	51
14.1. Commandes spécifiques Modbus	51
14.2. Commande Status	52
14.3. Commande de mise à jour	52
14.4. Commande d'un contact sec (sortie numérique)	53
15. CONFIGURATION DU DATALOGGER	54
15.1. Paramètres	54
15.2. Précisions sur les numéros de téléphone	55
15.3. Configuration par SMS	56
15.4. Configuration locale	57
15.5. Configuration distante	57
15.6. Contrôle de l'accès local	58
15.7. Configuration des ports	58
16. IMPULSIONS	59
17. WIRED M-BUS	60
18. WIRELESS M-BUS	61
19. TAGS RFID ACTIFS	62
20. MODBUS	63
20.1. Configuration	63
20.2. Ensembles de données Modbus	63
20.2.1. Variables	63
20.2.2. Boundaries	65
20.3. Esclaves Modbus	65
20.4. Adresse des variables	65
21. SCHEDULER	66
21.1. Moteur d'alarme	68
22. FICHIERS DE LOG	70
23. SYNCHRONISATION DE L'HORLOGE INTERNE	71
24. MISE À JOUR DU FIRMWARE DE LA PASSERELLE	72
25. SUPPORT	73

1. GLOSSAIRE

Nom	Description
APN	Access Point Name Nom du point d'accès permettant au datalogger de se connecter au réseau Internet par liaison mobile.
FTP	File Transfer Protocol Protocole de communication destiné à l'échange informatique de fichiers sur un réseau TCP/IP.
GPRS	General Packet Radio Service Norme pour la téléphonie mobile dérivée du GSM permettant un débit de données plus élevé. Appelé aussi 2,5G. Download : max. 86 Kbps Upload : max. 43 Kbps
GSM	Global System for Mobile Communications Réseau commuté du téléphone mobile.
HTTP	HyperText Transfer Protocol Protocole de communication client-serveur développé pour le Web.
IP	Internet Protocol Protocole de messages responsable de l'adressage et de la transmission de paquets TCP sur le réseau.
PSSTT	Public Switched Telephone Network Voir RTC.
Rail DIN	Rail métallique standardisé de 35 mm utilisé en Europe dans les équipements industriels de contrôle en racks.
RTC	Réseau téléphonique commuté Réseau commuté du téléphone fixe.
TCP	Transmission Control Protocol Protocole orienté connexion sur Internet qui offre les services de segmentation des données en paquets que le protocole IP transmet sur le réseau. Ce protocole fournit un service fiable de transfert de données. Voir aussi IP.
TCP/IP	Transmission Control Protocol/Internet Protocol Ensemble de protocoles réseaux qui fournissent des services d'interconnexion entre des ordinateurs d'architectures matérielles et de systèmes d'exploitation différents. TCP/IP inclut des normes de communication entre ordinateurs et des conventions pour l'interconnexion des réseaux et le routage.
TIC	Télé-Information Client Sortie d'information numérique des compteurs ERDF qui diffuse en permanence les paramètres contractuels gérés, ainsi que les grandeurs de consommation mesurées par l'appareil.

2. REMARQUES CONCERNANT CE MANUEL

Ce guide décrit le montage, l'installation et la mise en service de l'appareil. Veuillez garder ces instructions toujours disponibles.

2.1. Champ d'application

La présente description technique est valable pour les dataloggers Datalog H60.

2.2. Versions de produits

Suivant le type de modem GSM et de carte radio, il peut exister plusieurs versions de produits.

Version produit SOCOMEC :

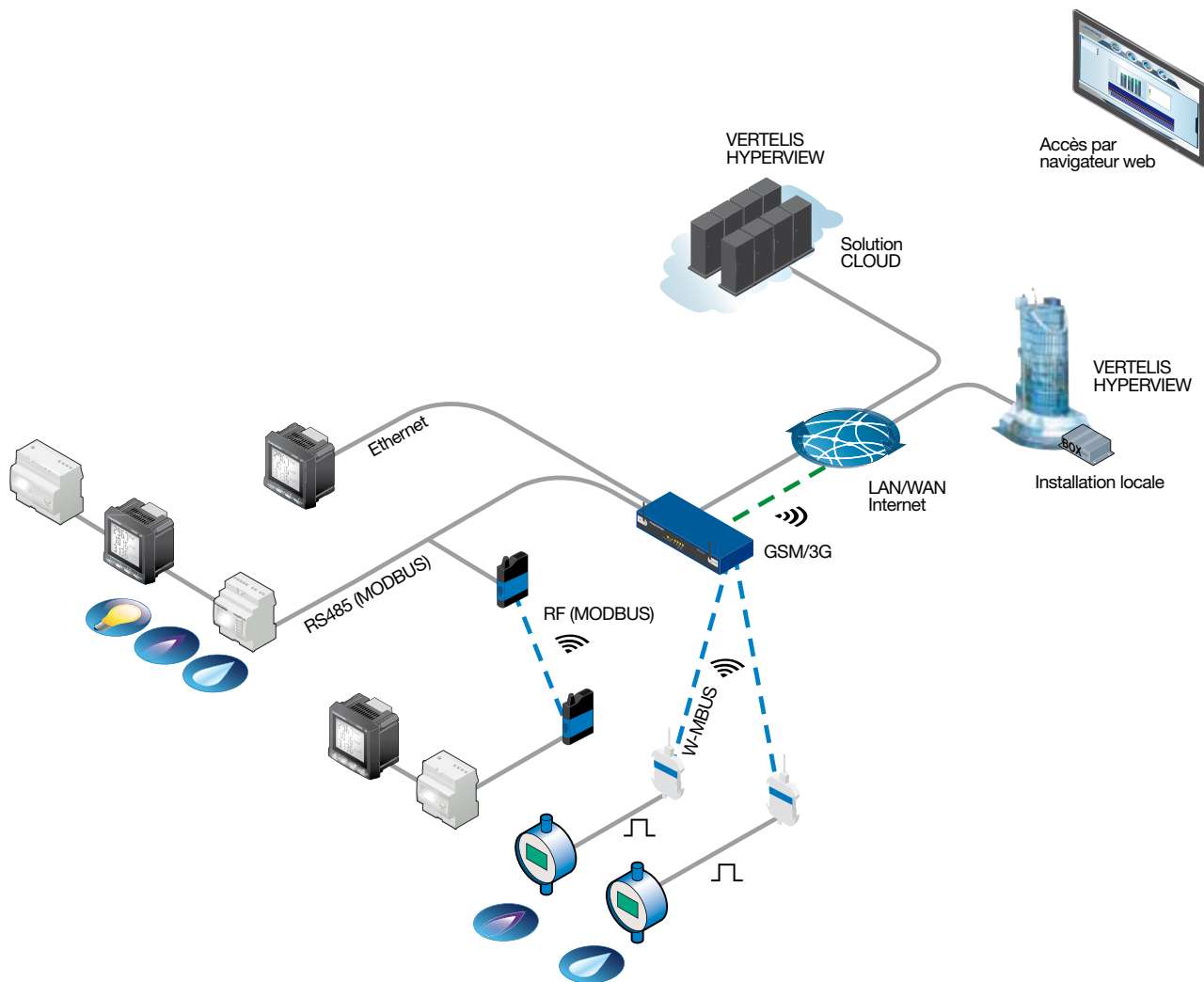
- - Datalog H60 -WirelessMbus-868MHz-3G
- - Carte radio RFID Modem 2G – RFID
- - Accessoires commun à tous les produits :
 - Bloc Alimentation externe 12v
 - Alimentation 24V Rail Din
 - Antenne GPRS bi bandes déportée avec 5m de câble
 - Antenne GPRS bi bandes déportée avec 10m de câble
 - Antenne GPRS bi bandes déportée avec 20m de câble

2.3. Groupe cible

Ce guide s'adresse aux installateurs.

2.4. Principe général de fonctionnement

Le datalogger H60 est un concentrateur dédié aux réseaux sans fil. Collecter des données de compteurs ou de capteurs, contrôler des E/S sont les fonctionnalités de base du datalogger DATALOG H60.




2.5. Spécifications techniques

	Caractéristiques	Protocole	Connectique
Interface radio Wireless M-Bus	Bandes de fréquence : 868,3 et 868,95 MHz 169MHz	Mode opératoire S1 et T1 Protocole EN 13757-4 Mode opératoire N Protocole EN 13757-4:2011	Antenne - SMA
Interface E/S Entrées Numériques Sortie Numérique	3 entrées 1 sortie	TOR - Impulsions Relais	Bornier à vis Bornier à vis
Interface série Port 1 Port 2	RS485 RS232	Modbus Modbus/MBus	Bornier à vis DB9
Interface réseau Ethernet GSM/GPRS 3G	10/100Mbps/s 850/900/1800/1900 MHz 900/2100 MHz	Services IP Services IP Services IP	RJ45 Antenne - SMA Antenne - SMA
Caractéristiques générales Alimentation Batterie interne Batterie externe Températures Dimensions Agréments RTTE	[+12/24v] DC fourni par une alimentation externe (Accessoire) Li-Ion 650mAH Gestion d'un UPS externe par trois entrées TOR dédiées Fonctionnement : -20°C à +70°C - Stockage : -20°C à +85°C 20cm (L) x 12cm (P) x 3,2cm (H) 1995/5/CEEE		

3. CONSIGNES DE SÉCURITÉ

Respectez impérativement toutes les consignes de sécurité figurant dans ce guide.

Tout non-respect de ces consignes peut causer un endommagement des appareils et représenter un danger pour les personnes.

	ATTENTION	<p>Le Datalogger Datalog H60 peut être endommagée par des décharges électrostatiques (ESD).</p> <ul style="list-style-type: none">• Tous les travaux d'installation du Datalog H60 ne doivent être effectués que par un électricien professionnel habilité.• Évitez tout contact avec les raccords des composants et les contacts des connecteurs.
---	------------------	---

	ATTENTION	<p>Le Datalogger Datalog H60 contient une batterie Lithium. Il y a risque d'explosion si la batterie est remplacée par une batterie de type incorrect.</p>
---	------------------	--

4. DÉBALLAGE

4.1. Contenu de la livraison

Vérifiez que la livraison soit complète et que l'appareil ne soit pas endommagé. Contactez Socomec si la livraison est incomplète ou endommagée.



DATALOG H60

2 antennes coudées (RF/Modem)



Alimentation externe 12V



Kit de fixation RailDin et mural

En option:



Antenne GPRS bi bandes déportée avec 5m ou 20m de câble

4.2. Identification du datalogger

4.2.1. Étiquette signalétique



Vous pouvez identifier le datalogger Datalog H60 grâce à son étiquette signalétique, qui se trouve sur le devant du boîtier.

- DATALOG H60: Nom du produit.
- WEN071300614B: Numéro de série
- 20/13: Semaine/Année de production.
- MAC: Adresse Mac du produit.
- Le code-barres correspond au numéro de série du produit.

4.2.2. Version du logiciel

Vous trouverez la version du logiciel sur l'interface Web du datalogger. La version du logiciel est indiquée dans la page d'accueil Overview.

5. MONTAGE

5.1. Choix de l'environnement

Respectez les conditions environnantes suivantes.

- Température d'utilisation : -10 °C à +45 °C.
Les performances de la batterie sont dégradées pour des températures inférieures à 0 °C et supérieures à 45 °C.
- Température de stockage : -20 °C à +45 °C (pour une durée de moins d'un mois). -20 °C - +35 °C (pour une durée de plus d'un mois)
- L'humidité ambiante doit être comprise entre 10 à 90 % RH.
- Protégez le Datalog H60 contre la poussière, l'humidité, les substances agressives et la buée.
- La distance entre le Datalog H60 et les équipements Modbus ne doit pas dépasser la distance maximale autorisée pour le type d'interface correspondant (RS485 ou RS232).
- En cas d'utilisation de la liaison cellulaire, veillez à ce que la réception soit optimale lors du montage. Les indications correspondantes figurent aux chapitres respectifs.

Le datalogger Datalog H60 doit envoyer les données de votre installation au serveur distant automatiquement et à intervalles réguliers, il vous faudra remplir les conditions suivantes :

- Pour envoyer les données de l'installation via Ethernet, le réseau doit comprendre un routeur permettant l'accès à Internet.
- Pour envoyer les données de l'installation via liaison GSM, la couverture réseau doit être satisfaisante.
- Dans le cas d'une utilisation de l'option antenne GSM déportée reportez-vous au chapitre montage de l'antenne déportée avant de monter le datalogger.



ATTENTION

Afin d'optimiser la sensibilité de réception GSM il est impératif de laisser un espace vide autour de l'antenne de 20 cm.

5.2. Montage du Datalog H60 sur pied

Le Datalog H60 peut se poser sur pied sur une surface plane.

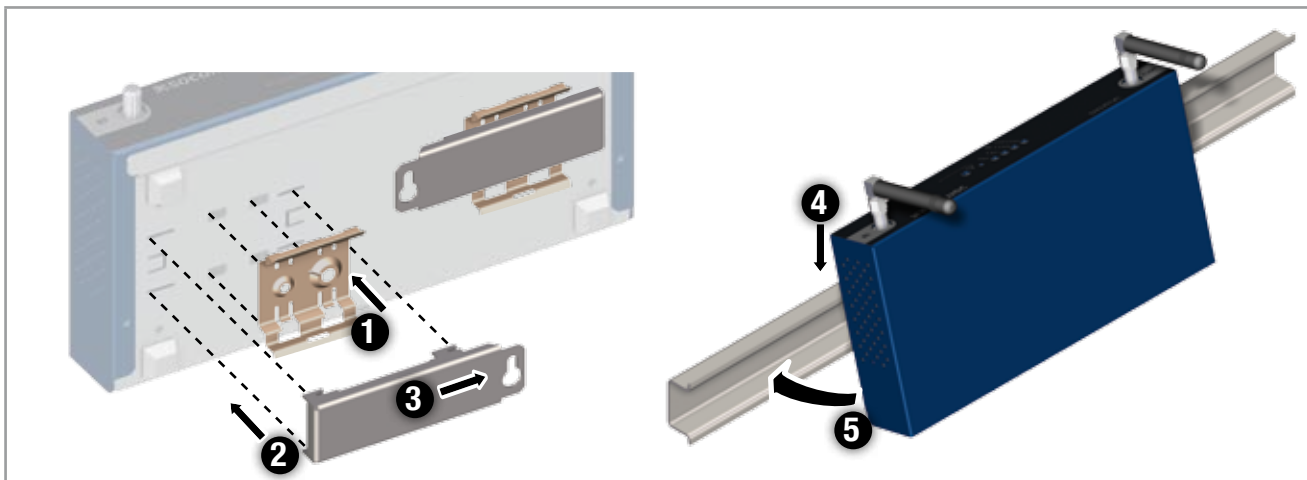


5.3. Montage du Datalog H60 sur rail DIN

Le datalogger Datalog H60 se monte sur un rail métallique standardisé (RAIL DIN) de 35 mm.

Ce montage sur rail nécessite l'accessoire de montage supplémentaire Réf : KFIX01.

1. Monter le kit de fixation sur le datalogger :
2. Accrochez le Datalog H60 au rail à l'aide des deux supports.

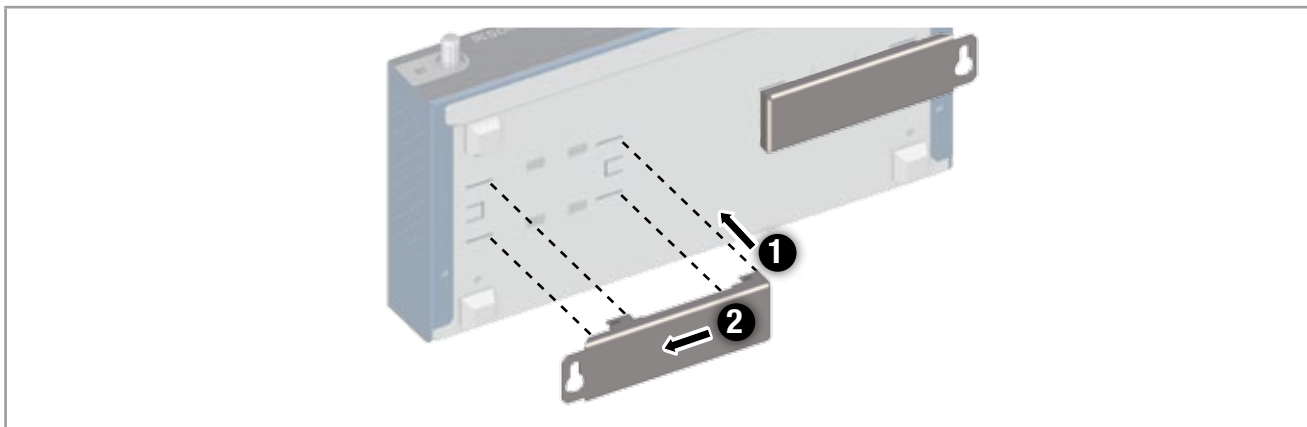


5.4. Montage murale du Datalog H60

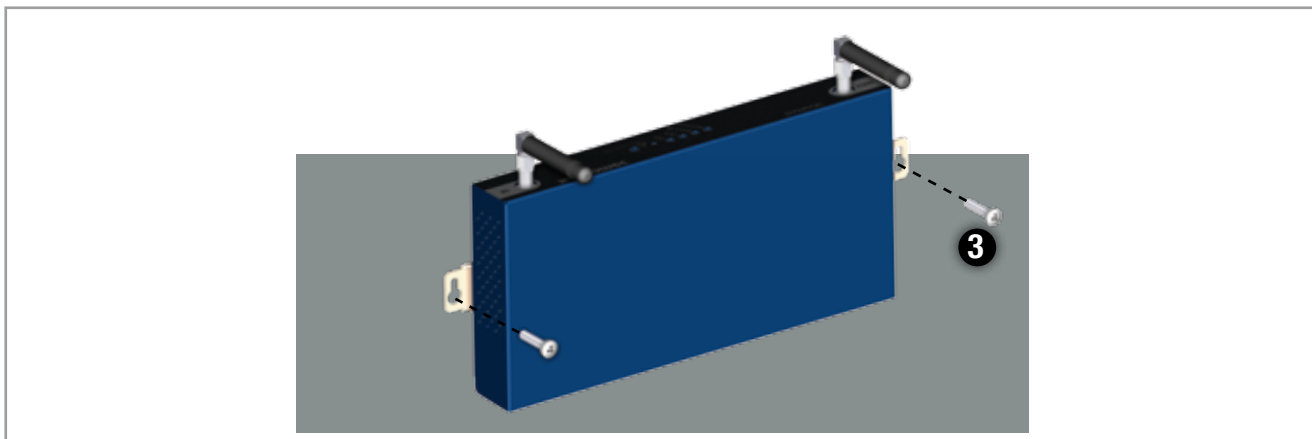
Le Datalog H60 peut être fixée sur un mur.

Ce montage sur rail nécessite l'accessoire de montage supplémentaire Réf : KFIX01.

1. Monter le kit de fixation sur la le datalogger :
2. Lever les 2 languettes pour bloquer les fixations :

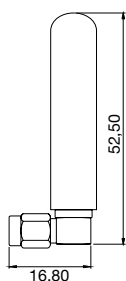


3. Déterminez la position du datalogger, et marquez les trous de perçage à l'aide d'un crayon.
4. Percez les trous (diamètre : 6 mm) aux emplacements marqués et insérez les chevilles.
5. Fixez le Datalog H60 au mur à l'aide de vis.



5.5. Montage de l'antenne GPRS/GSM standard coudée

Le datalogger est livré en standard avec deux antennes coudées.



5.5.1. Spécifications

Fréquence/MHZ	900/1800
Gain	0db
VSWR	< 1: 2.0
Impédance	50 Ohms
Patten	Omni direction
Polarisation	Vertical
Connecteur	SMA R/A Male

Vissez le câble de l'antenne GSM à la main à la prise d'antenne du datalogger repérée par le label GSM en face avant.

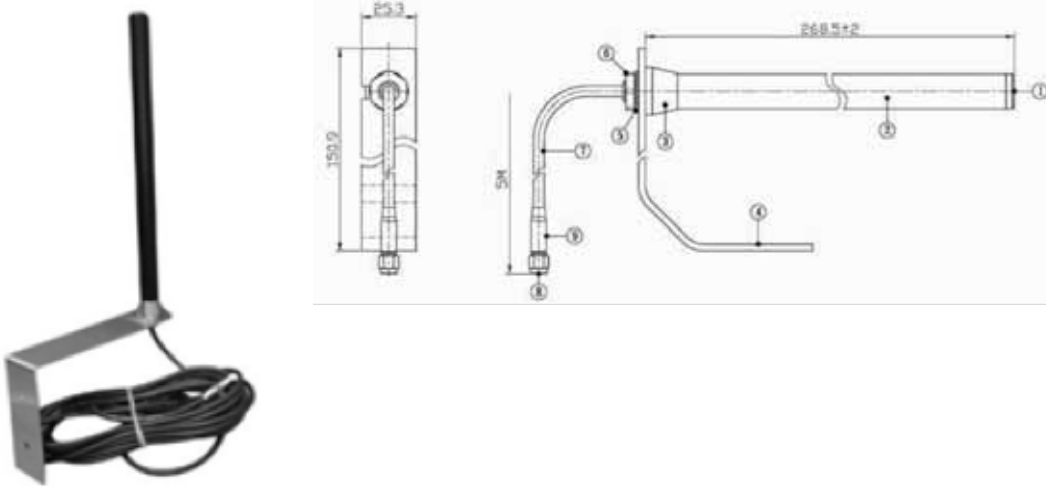


ATTENTION

Afin d'optimiser la sensibilité de réception GSM il est impératif de laisser un espace vide autour de l'antenne de 20 cm.

5.6. Montage de l'antenne GSM/GPRS déportée (option)

Uniquement en cas de commande en option : Kit antenne déportée.



Contenu du kit : 1 antenne et 1 équerre d'antenne avec câble d'antenne.

Déterminez le lieu de montage optimal de l'équerre d'antenne.

1. Déterminez la position de l'antenne en utilisant l'équerre sur le mur, et marquez les trous de perçage à l'aide d'un crayon.
2. Percez les trous (diamètre : 6 mm) aux emplacements marqués et insérez les chevilles.
3. Fixez l'équerre d'antenne au mur à l'aide de vis.
4. Vissez l'antenne sur le câble d'antenne.
5. Vissez le câble sur la prise antenne.

Câble : RG-58.

Fréquences : 824-960 MHz ; 1710-1990 MHz.

6. RACCORDEMENT ÉLECTRIQUE



ATTENTION

- Tous les travaux de câblage doivent impérativement être effectués par un électricien qualifié spécialisé.
- Avant l'installation, tous les appareils raccordés au bus de communication correspondant doivent être déconnectés des deux côtés (DC et AC).
- Veuillez respecter toutes les consignes de sécurité figurant dans la documentation de l'onduleur.

6.1. Aperçu des interfaces externes et des LEDs

6.1.1. Face avant du boîtier



Antennes

ANT RF	antenne pour liaison RF
ANT GSM	antenne pour liaison GPRS ou 3G

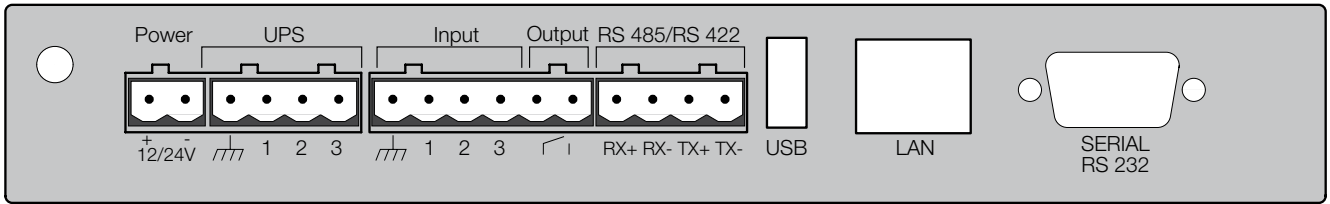
Boutons

REQUEST	Appui court => Demande de connexion Appui long (3s) => Affiche le niveau de réception du signal GSM (voir LED GSM) 3 appuis longs successifs => Retour usine des paramètres
RESET	Redémarrage matériel du datalogger

LEDs

POWER	S'allume lorsque le produit est alimenté
CPU	S'allume suivant l'activité CPU
RF	Eteinte par défaut et clignote sur trafic RF ou lorsqu'un ordinateur est connecté en mode transparent.
GSM	S'allume en cours de connexion GSM S'allume pendant 1 seconde sur la réception d'un SMS Sur un appui long du bouton Request elle indique le niveau du signal reçu RSSI par un nombre de clignotement (0 à 5 fois) 0 – puissance du signal ≤ -112 dBm 1 – puissance du signal entre -111 dBm et -96 dBm 2 – puissance du signal entre -96 dBm et -81 dBm 3 – puissance du signal entre -81 dBm et -66 dBm 4 – puissance du signal entre -66 dBm et -51 dBm 5 - puissance du signal > -51 dBm

6.1.2. Face arrière du boîtier



12/24V	Alimentation du datalogger.
UPS	Entrées de statuts onduleur.
Input	3 entrées TOR.
Output	1 sortie TOR.
RS485/RS422	1 port RS485/RS422.
USB	Prise USB.
LAN	Prise LAN.
Serial RS232	port RS232 type DB9.

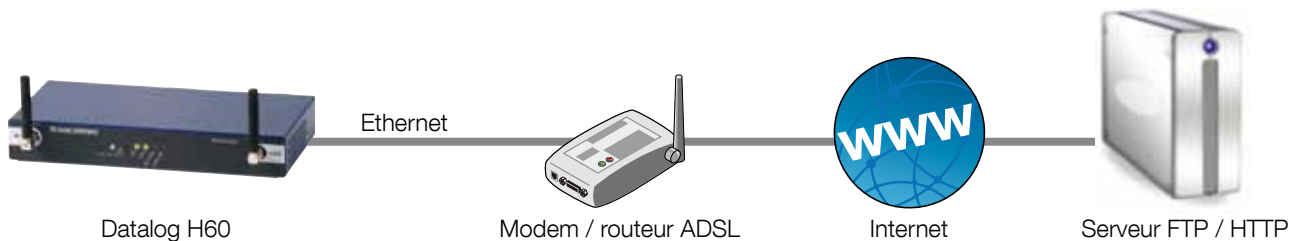
6.2. Raccordement du datalogger DATALOG H60 au réseau Internet pour la communication avec le serveur distant

Le présent chapitre décrit le raccordement du Datalog H60 au réseau Internet pour la communication avec le serveur distant. Cette communication peut se faire au choix selon trois media différents :

- Ethernet via une ligne ADSL.
- 3G (D'autres connections sont également possible contacter SOCOMEC)

6.2.1. Connexion à Internet via une ligne ADSL

Raccordez le datalogger Datalog H60 au réseau ADSL en utilisant la prise Ethernet repérée par le label LAN sur la face arrière.



La longueur et la qualité du câble peuvent avoir des répercussions négatives sur la qualité du signal.

Utilisez un câble Ethernet respectant les remarques suivantes :

- Un câble droit est conseillé.
- Utilisez un câble de qualité, au minimum un câble blindé torsadé par paire (« shielded twisted pair ») de catégorie 5 (STP Cat 5) ou supérieure.
- La longueur maximale autorisée du câble Ethernet est de 100 m par segment.



Les réglages du datalogger Datalog H60 à la livraison sont les suivants :

Adresse IP : 192.168.1.12
 Masque de sous réseau : 255.255.255.0
 DHCP : Désactivé

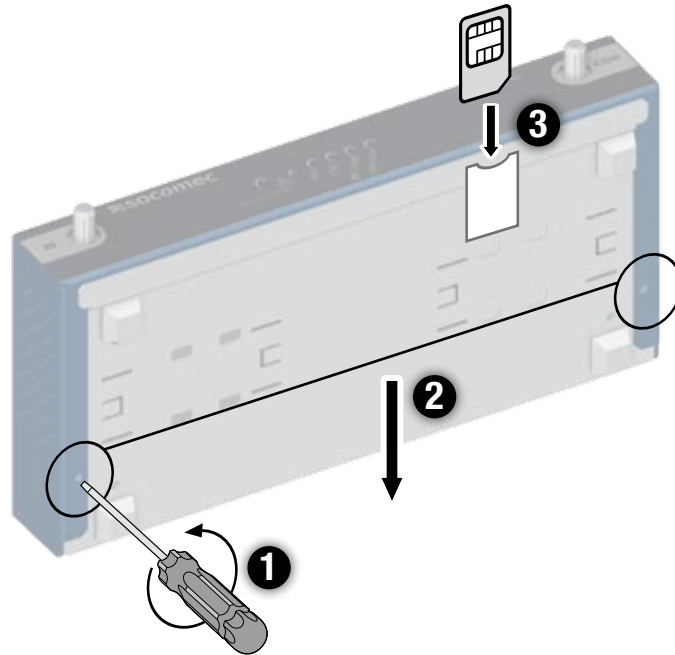


Administrateur réseau

Si votre réseau local est géré par un administrateur réseau, contactez-le avant d'intégrer le datalogger Datalog H60 dans votre réseau.

6.2.2. Connexion à Internet via une liaison GPRS ou 3G

Pour utiliser la liaison GPRS ou la liaison 3G et permettre au datalogger DATALOG H60 de communiquer avec le serveur distant, il faut insérer une carte SIM dans le tiroir en interne du datalogger.



Afin d'assurer le bon fonctionnement du datalogger Datalog H60 en mode 3G ou autres, vous devez insérer une carte SIM présentant les caractéristiques suivantes :

- Possibilité de recevoir et d'envoyer des SMS.
- Communication en adéquation avec l'abonnement adéquat.

Insérez la carte SIM dans le produit.

La configuration usine du datalogger, la gestion du code est PIN Mode = off (voir chapitre Configuration Webserver).

Trois cas sont possibles :

- Le code PIN est désactivé : la communication modem est active.
- Le code PIN est activé et égal à 0000 : la communication modem est active.
- Le code PIN est activé et différent de 0000 : la communication modem est en erreur.



AVERTISSEMENT

Si la carte SIM a un code PIN activé mais différent de 0000 au premier démarrage du datalogger, elle sera bloquée après 3 tentatives. Vous pouvez la débloquer en utilisant un téléphone portable avec le code PUK fourni par votre opérateur.

Pour communiquer avec le serveur distant via GSM, le datalogger doit être positionné dans une zone de couverture GSM optimale.

Pour connaître la qualité de signal reçu par le datalogger, utiliser le bouton REQUEST par un appui long (3s).

Le niveau de réception du signal GSM (RSSI) s'affiche sur la LED GSM par un nombre de clignotement (0 à 5 fois).

La correspondance entre nombre de clignotement et puissance du signal est reportée ci-dessous :

- 0 – puissance du signal ≤ -112 dBm
- 1 – puissance du signal entre -111 dBm et -96 dBm
- 2 – puissance du signal entre -96 dBm et -81 dBm
- 3 – puissance du signal entre -81 dBm et -66 dBm
- 4 – puissance du signal entre -66 dBm et -51 dBm
- 5 - puissance du signal > -51 dBm



AVERTISSEMENT

La puissance du signal reçu doit être au minimum égal à -96 dBm soit 2 clignotements.

6.3. Raccordement de la sortie TOR du datalogger

Le datalogger Datalog H60 présente 1 sortie TOR.
Cette sortie peut être utilisée en sortie numérique.

6.4. Raccordement des entrées TOR du datalogger

Le datalogger Datalog H60 présente 3 entrées TOR.
Ces entrées peuvent être utilisées en entrée numérique pour déclencher des alarmes ou en compteur d'impulsions (voir le manuel d'exploitation pour plus de détails).



AVERTISSEMENT

Pour éviter toute détérioration du datalogger, ne pas injecter de courant ou de tension sur les entrées TOR.

En mode compteur d'impulsion, la durée de l'impulsion doit être au minimum de 10 ms et maximum de 1000ms pour être comptabilisée.

6.5. Raccordement du datalogger aux équipements Modbus via la liaison RS485

Le présent chapitre décrit le raccordement du datalogger Datalog H60 au bus de communication RS485 pour la communication avec les équipements Modbus.

L'interface RS485 utilisée est repérée par le label RS485/RS 422.

Cette interface est 4 fils ou 2 fils.

Le choix entre 4 fils ou 2 fils se fait par automatiquement par le logiciel embarqué..

Le datalogger peut se trouver à l'extrémité du bus de communication RS485 ou en milieu de bus.

Afin d'assurer le fonctionnement du bus de données RS485, ce dernier doit être terminé aux deux extrémités par une résistance de fin de ligne de 120 Ohms.

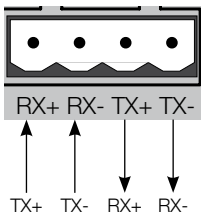
Suivant le positionnement du datalogger sur le bus, cette résistance de fin de ligne doit être placée à l'extérieur du boîtier.

Consultez la documentation des équipements Modbus concernant leur principe de raccordement et de câblage RS485.

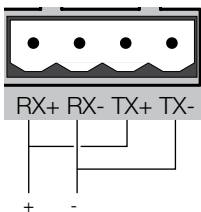
Une fois le câble RS485 disponible près du datalogger :

- 1. Dénudez la gaine du câble de communication RS485 sur env. 4 cm.
- 3. Raccourcissez le blindage jusqu'à la gaine de câble.
- 4. Dénudez les fils sur env. 6 mm.
- 5. Raccordez les conducteurs au connecteur repéré RS485/RS422 en respectant les affectations dans votre bus de communication RS485.

Montage en RS485 4 fils:



Montage en RS485 2 fils:



6.6. Raccordement du datalogger aux équipements via la liaison RS232

Le port RS232 peut être utilisé pour collecter des données d'un équipement M-Bus. Un transceiver M-Bus doit être connecté au port RS232 et ce port doit être configuré en mode M-Bus .

Les équipements M-Bus doivent être configurés avec une adresse unique sur le bus.

Un scan du bus doit être initié depuis l'interface web. Les équipements M-Bus découverts durant ce scan seront interrogés à chaque occurrence du scheduler associé. Si des équipements sont enlevés ou ajoutés sur le bus, un nouveau scan doit être initié afin que le datalogger prenne en compte la modification.

7. RACCORDEMENT À UN PC LOCAL VIA ETHERNET POUR LA CONFIGURATION

Lors de la première mise en service, le datalogger Datalog H60 doit être configurée via le réseau Ethernet avec un PC (portable ou fixe).

Le PC peut être raccordé au datalogger directement ou intégré au réseau local en particulier si la liaison ADSL est utilisée pour la communication avec le serveur distant.



Recommandations relatives au câblage Ethernet :

La longueur et la qualité du câble peuvent avoir des répercussions négatives sur la qualité du signal :

- Un câble droit est nécessaire.
- Utilisez un câble de qualité, au minimum un câble blindé torsadé par paire (« shielded twisted pair ») de catégorie 5 (FTP Cat 5) ou supérieure.
- La longueur maximale autorisée du câble Ethernet est de 100 m par segment.

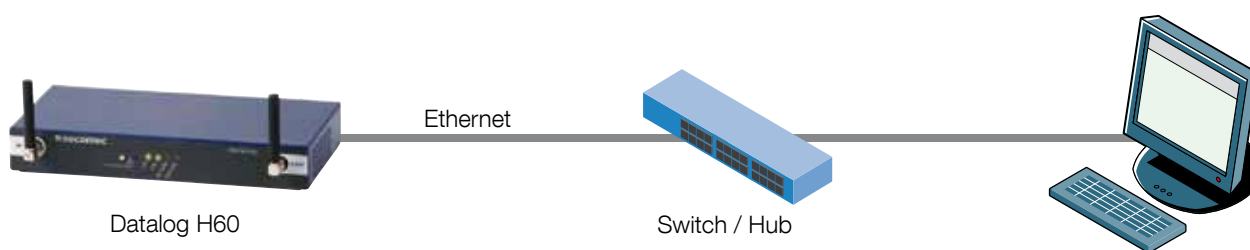
Ce chapitre décrit les deux types de raccordement d'un PC au datalogger.

7.1. Connexion directe du PC au datalogger



Raccordez le datalogger Datalog H60 directement sur le PC en utilisant l'interface Ethernet du boîtier et un câble Ethernet (câble croisé). Pour repérer la prise Ethernet du PC référez-vous à la documentation du PC.

7.2. Connexion du PC au datalogger dans un réseau local



Raccordez le datalogger Datalog H60 au réseau local en utilisant l'interface Ethernet du boîtier et un câble Ethernet (câble droit). Pour repérer la prise Ethernet du Switch/Hub référez-vous à sa documentation.

7.3. Configuration réseau du PC pour accéder au datalogger

En connexion directe du PC au datalogger ou en connexion du PC au datalogger dans un réseau local, il est nécessaire de configurer d'une adresse IP fixe sur le PC dans la même plage d'adresse IP et dans le même sous réseau que le datalogger Datalog H60.



Les réglages du datalogger Datalog H60 à la livraison sont les suivants :

Adresse IP : 192.168.1.12
Masque de sous réseau : 255.255.255.0
DHCP : Désactivé



Administrateur réseau

Si votre réseau local est géré par un administrateur réseau, contactez-le avant d'intégrer le datalogger Datalog H60 dans votre réseau.

L'étape suivante permet de configurer l'adresse réseau d'un PC pour accéder au datalogger Datalog H60 :

Configuration d'une deuxième adresse IP sur le PC :

- 1. Sous Windows (2000 ou XP) cliquez sur Démarrer/Paramètres/Connexions réseau et accès à distance. La fenêtre Connexions réseau et accès à distance s'affiche.
- 2. Cliquez avec le bouton droit sur Réseau local puis cliquez sur Propriétés.
- 3. Sélectionnez Protocole Internet (TCP/IPv4) puis cliquez sur Propriétés.
- 4. Cliquez ensuite sur Avancé.
- 5. Dans la zone Adresse IP cliquez sur Ajouter.
- 6. Entrez l'adresse IP 192.168.1.xxx (xxx entre 1 et 254 et différent de 12) et le masque de sous-réseau 255.255.255.0.
- 7. Cliquez sur Ajouter.
- 8. Pour valider les réglages, cliquez sur OK dans chacune des trois fenêtres.
- 9. Fermez la fenêtre Connexion réseau et accès à distance.

8. RACCORDEMENT DE L'ALIMENTATION

Raccordez l'alimentation seulement lorsque tous les autres appareils sont raccordés avec le datalogger Datalog H60.

Vérifier si les raccordements suivants (si utilisés) sont branchés correctement avant d'alimenter le datalogger :

- Sortie TOR.
- Entrées TOR.
- Entrée RS485 utilisée pour le raccordement d'équipements.
- Entrée RS232 utilisée pour le raccordement d'équipements.

1-Brancher le câble sur l'entrée 12/24V du connecteur à vis débrochable du datalogger en respectant la polarité.

2-Branchez l'alimentation utilisée au secteur.



AVERTISSEMENT

Phase de démarrage du datalogger Datalog H60 :

Après le branchement du bloc d'alimentation, la LED POWER s'allume. Le datalogger met environ 20 secondes à démarrer. Son initialisation est terminée lorsque la LED CPU clignote avec une période de 0,5s.



AVERTISSEMENT

Informations sur les caractéristiques de l'alimentation à utiliser :

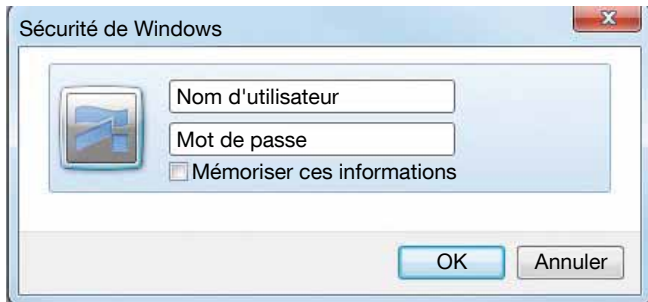
L'alimentation doit délivrer entre 12 et 24V et 15W.

9. MISE EN SERVICE DU DATALOGGER

La configuration du datalogger Datalog H60 est réalisée via l'interface Web intégrée.

9.1. Connexion à l'interface Web intégrée

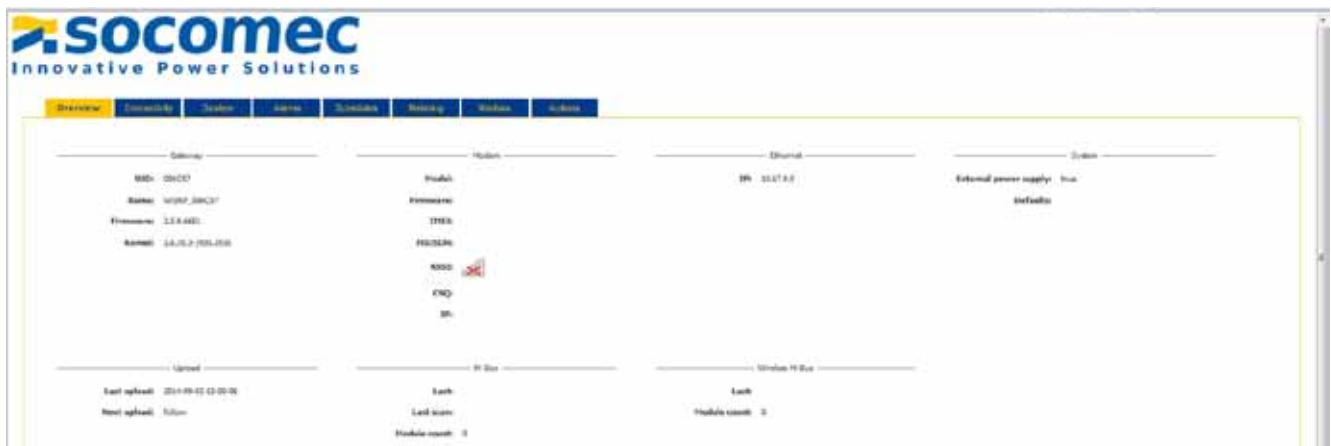
- 1- Lancez le navigateur web. L'interface web est compatible avec les dernières versions des navigateurs : Firefox, Chrome et Internet Explorer. Les versions plus anciennes peuvent fonctionner mais ne sont plus supportées (par exemple IE 7).
- 2- Appelez la page d'accueil du datalogger Datalog H60 via la ligne d'adresse du navigateur <http://192.168.1.12>.
- 3- La fenêtre suivante s'affiche :



ATTENTION

Les réglages du datalogger Datalog H60 à la livraison sont les suivants :
Utilisateur : admin
Mot de passe : high

4. La page d'accueil suivante s'affiche :



Si le datalogger n'est pas encore opérationnel le message suivant s'affiche :



ATTENTION

Si l'accès aux pages web est effectué pendant la phase d'initialisation du datalogger, un message d'alerte s'affiche en haut de page du type : « le datalogger est en cours d'initialisation. ». Attendre que le datalogger soit complètement initialisé pour accéder aux pages web.

9.2. Configuration de la connectivité du datalogger

La page Connectivity permet de configurer le datalogger afin qu'il communique avec le serveur distant.

The screenshot shows the Socomec web interface for configuring connectivity. The interface includes a navigation menu and several configuration panels:

- Modem:** Includes fields for Modem Mode (OFF), Modem Type (GSM), Call Number (997777), PIN, SIM PIN, Password, Power (OFF), Disconnect Delay (30), Call ID (V-Main), and Call ID.
- Ethernet:** Includes fields for IP (192.168.1.1), Netmask (255.255.255.0), Gateway (192.168.1.254), and DNS server (192.168.1.1).
- FTP:** Includes fields for Address (192.168.1.1), User (SOCMECHS), Password (SOCMECHS), Host mode (Web Cam), and Port (21).
- Time:** Includes fields for Daylight (GMT+1), Alarm threshold (0), and NTP server (172.16.0.1).
- Request button:** Includes fields for Connection request (checked), Upload delay (checked), and FTP Data request.

9.2.1. Configuration de la connexion modem

Modem

PIN Mode:

PIN Code:

Call Number:

APN:

Login:

Password:

Mode:

Disconnect delay:

Caller ID Whitelist

Caller ID:

Paramètres	Description
PIN Mode	Off : Le code PIN de la carte SIM doit être désactivé
	Manual : Le code PIN de la carte SIM doit être renseigné dans la case PIN Code
	Automatic : non-implémenté
PIN Code	Code PIN de la carte SIM à renseigner si Manual est sélectionné dans PIN Mode
Call Number	Numéro d'appel GPRS. Le numéro par défaut *99***1# est valable dans la majorité des cas.
APN	Nom de l'APN de votre opérateur mobile
Login	Nom d'utilisateur APN de votre opérateur mobile
Password	Mot de passe APN de votre opérateur mobile
Mode	OnDemand : Le datalogger établit la connexion uniquement lorsqu'il doit communiquer avec le serveur distant. Il la coupe lorsque le transfert de données est terminé après un délai configurable dans Disconnect delay.
	AlwaysOn : non-implémenté
	AlwaysOff : Ce mode est à utiliser en cas de connexion avec le serveur distant via Ethernet. Cette connexion ne se fait jamais via le modem mais si une carte SIM valide est présente, le datalogger peut recevoir des appels ou des SMS entrants et émettre des SMS.
	Off : Le modem est toujours éteint. Ce mode est à utiliser en cas de connexion avec le serveur distant via Ethernet sans possibilité de recevoir des appels ou des SMS et émettre des SMS.
Disconnect delay	Valeur en seconde du délai d'attente en mode OnDemand entre la fin des échanges de données et la fin de connexion.
Caller ID	Liste des numéros de téléphone autorisés pour les appels et les SMS entrants.



AVERTISSEMENT

Consultez votre opérateur mobile pour obtenir les informations (APN, login, mot de passe) relatives à votre carte SIM.

9.2.2. Configuration de la connexion Ethernet

Ethernet

IP: • • •

Netmask: • • •

Gateway: • • •

Use DHCP

DNS

DNS servers: • • •

Paramètres	Description
IP	Entrez l'adresse IP à laquelle le datalogger Datalog H60 est accessible.
Netmask	Entrez le masque de sous-réseau de votre réseau. Ce masque limite le réseau Ethernet à des adresses IP définies et sépare les plages réseau les unes des autres.
Gateway	Entrez l'adresse de la passerelle de votre réseau. L'adresse de la passerelle est l'adresse IP de l'appareil qui établit la connexion à Internet. En général, l'adresse entrée ici est celle de votre routeur ADSL.
Use DHCP	Vous avez la possibilité d'obtenir les paramètres Ethernet automatiquement si l'infrastructure du réseau le permet. Dans ce cas sélectionnez le mode dynamique et reportez-vous à la configuration de votre serveur DHCP pour connaître l'adresse IP de votre passerelle.
DNS servers	Le serveur DNS (Domain Name System) traduit les adresses Internet explicites (par ex. www.SOCOMECCOM) en adresses IP correspondantes. Entrez ici les adresses des serveurs DNS que vous avez reçue de votre fournisseur d'accès à l'Internet (FAI). Vous pouvez également entrer l'adresse IP de votre routeur.

9.2.3. Configuration du serveur FTP distant

Ce champ permet de renseigner la configuration du serveur FTP distant permettant l'échange de données.

Paramètres	Description
Address	Adresse IP ou nom du serveur FTP distant
Login	Nom d'utilisateur utilisé par le datalogger pour la connexion au serveur FTP distant
Password	Mot de passe utilisé par le datalogger pour la connexion au serveur FTP distant
Mode	Actif ou passif
Use FTPS	A cocher si le serveur distant est de type FTPS
Root	Répertoire de racine sur le serveur FTP distant
WS notification	None : Aucune notification Web Services
	Put : Notification via Web Services lors du dépôt de données par le datalogger sur le serveur FTP
	Get: Notification via Web Services lors du téléchargement de données par le datalogger à partir du serveur FTP
	Both : Put+Get

9.2.4. Configuration du fuseau horaire

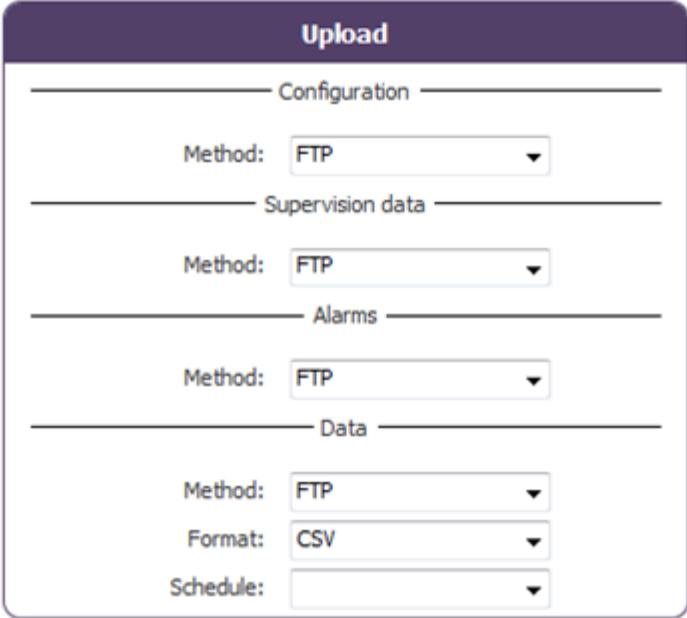
Paramètres	Description
Timezone	Champ texte pour indiquer le fuseau horaire. Le format doit respecter le format TZ, voir le lien :
Alarm threshold (s)	Différence en seconde entre l'heure du datalogger et l'heure de synchronisation NTP au-delà de laquelle une alarme est émise
NTP servers	Adresses IP des serveurs NTP utilisés pour la synchronisation de l'horloge du datalogger

9.2.5. Configuration des transferts de données

Le datalogger peut déposer sur le serveur distant 4 types de données :

- Données de configuration du datalogger.
- Données de supervision du datalogger.
- Alarmes
- Données RF, Modbus, Index.

Pour chaque type de donnée, le datalogger peut utiliser la méthode FTP.



Upload

Configuration

Method: FTP

Supervision data

Method: FTP

Alarms

Method: FTP

Data

Method: FTP

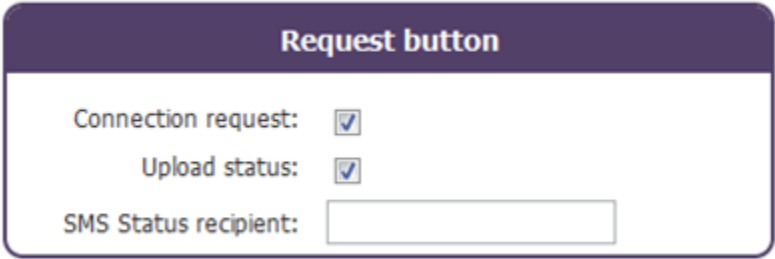
Format: CSV

Schedule:

Le format des données est XML.

Le dépôt des données doit être associé à un Schedule en renseignant son identifiant unique configuré.

9.2.6. Bouton request



Request button

Connection request:

Upload status:

SMS Status recipient:

Par défaut, un appui sur le bouton "Request" en face avant du produit déclenche une connexion au serveur distant et un téléchargement des données enregistrées et le dépôt d'un fichier de statut. Les deux peuvent sélectivement être désactivés.

Un SMS d'état peut également être envoyé à un destinataire spécifié.

9.3. Configuration système

9.3.1. Configuration des ports

The screenshot shows a configuration window titled 'Ports'. It is divided into three sections:

- RS232:** A dropdown menu set to 'Off'.
- RS485:** A section with several dropdown menus: 'Mode' set to 'Modbus', 'Baudrate' set to '9600', 'Data bits' set to '8', 'Parity' set to 'None', and 'Stop bits' set to '1'.
- Digital inputs:** A section with three dropdown menus labeled 'Input #1:', 'Input #2:', and 'Input #3:', all set to 'Digital input'.

Les ports du datalogger peuvent être configurés avec les paramètres suivants :

Port	Valeurs	Description
rs232	Off Mbus	RS232 désactivé RS232 activé en mode MBus
rs485	Off Modbus	RS485 désactivé RS232 activé en mode Modbus
Input #1/mode	Digital input Pulse	Mode entrée numérique Mode entrée impulsion
Input #2/mode	Digital input Pulse	Mode entrée numérique Mode entrée impulsion
Input #3/mode	Digital input Pulse	Mode entrée numérique Mode entrée impulsion

Les paramètres du port RS485 sont :

Nom	Valeurs
Mode	off , modbus
Baudrate	4800, 9600, 19200 , 38400, 57600, 115200
Data bits	8
Parity	odd, even , none
stop_bit	1 , 2

9.4. Configuration des alarmes

Le datalogger peut générer des alarmes :

- Alarmes Système.
- Alarmes Entrée TOR.
- Etc

9.4.1. Alarmes Système

The screenshot shows a configuration window titled "System alarms". It contains four dropdown menus: "Power" set to "Off", "Modem IP" set to "Off", "MSISDN" set to "Off", and "SW Version" set to "On". Below these is a horizontal line with the word "Defaults" centered. Underneath the line are two text input fields: "Ignored:" and "Delayed:".

Les alarmes Système sont de 5 types :

- Power : alarme générée sur perte et retour d'alimentation.
- Modem IP : alarme générée si l'adresse IP du datalogger change.
- MSISDN : alarme générée si la carte SIM insérée dans le datalogger est changée.
- SW Version : alarme générée si la version du firmware du datalogger change (lors d'une mise à jour)
- Défaut.

Chaque source d'alarme peut être activée individuellement et être transférée immédiatement sur serveur distant (On) ou à la connexion suivante (Delayed).

Le datalogger génère également des alarmes de type défaut, dont les codes sont reportés ci-dessous :

Code	Description
D_MODEM	Défaut du modem
D_ETHERNET	Défaut de l'interface Ethernet
D_INTERNAL_BAT	Défaut batterie interne

Chaque défaut est transféré immédiatement sur serveur distant par le datalogger.

Dans la case Ignored peuvent être listées les codes défauts ignorés par le datalogger. Dans le cas où plusieurs codes défaut sont saisis, ils doivent être séparés par le caractère ',' (virgule).

Dans la case Delayed peuvent être listées les codes défauts transférés à la connexion suivante par le datalogger. Dans le cas où plusieurs codes défaut sont saisis, ils doivent être séparés par le caractère ',' (virgule).

9.4.2. Alarmes Entrées TOR

Des alarmes multiples peuvent être configurées pour les entrées numériques.

Une alarme pour une entrée numérique peut être configurée comme suit :

Nom	Description
Index	Index de l'entrée numérique
Label	Nom de l'alarme (uniquement informatif)
Mode	On : Envoi immédiat Off : Envoi désactivé Delayed : Envoi à la prochaine connexion
Type	None : Détection désactivée Raising : Détection sur front montant Falling : Détection sur front descendant Both : Raising+Falling

9.4.3. Alarmes Sortie TOR

Une alarme peut être configurée pour la sortie numérique.

Nom	Description
Label	Nom de l'alarme (uniquement informatif)
Mode	On : Envoi immédiat Off : Envoi désactivé Delayed : Envoi à la prochaine connexion
Type	None : Détection désactivée Raising : Détection sur front montant Falling : Détection sur front descendant Both : Raising+Falling

9.5. Configuration des schedules

Le scheduler est en charge de toutes les tâches périodiques.

La configuration du scheduler consiste en une liste de schedules.

Chacun de ces schedules possède un identifiant unique qui est utilisé pour lier une tâche à un schedule précis. Ils peuvent être utilisés indépendamment pour déclencher la collecte de données et télécharger des données.

The image shows a 'Schedule' configuration window with the following fields:

- Id:**
- Label:**
- Type:**
- Time:**
- Interval:**
- Count:**

Buttons:

Add new schedule

[Click here to add a new schedule ...](#)

Chaque schedule est configuré comme suit :

Nom	Description
Id	Identifiant unique de schedule
Label	Nom uniquement informatif du schedule
Type	Daily, Weekly, Monthly, Yearly ou Follower : voir description ci-dessous
Time	Heure de la première occurrence (non utilisé pour les schedules de type Yearly)
Day of Week	Numéro du jour dans la semaine de la première occurrence (1=Lundi, 7=Dimanche) (utilisé uniquement pour les schedules de type Weekly).
Day of Month	Numéro du jour dans le mois de la première occurrence (utilisé uniquement pour les schedules de type Monthly).
Date & Time	Date et heure de la première occurrence dans une période donnée (utilisé uniquement pour les schedules de type Yearly).
Interval	Interval entre les occurrences (en secondes)
Count	Nombre d'occurrences
Parent	Référence au schedule parent pour un schedule de type Follower.

Configuration des différents types de schedules :

Schedule de type Daily :

Chaque jour, la première occurrence T_0 est donnée par l'heure renseignée dans Time.

Le format est le suivant : HH:MM:SS. Par exemple 09:30:00

Les occurrences suivantes interviendront à l'heure T_i :

$$T_i = T_0 + i \times \Delta t \quad \begin{cases} i < count \\ \forall i \text{ jour}(T_i) = \text{jour}(T_0) \end{cases}$$

Δt correspond à la valeur en secondes renseignée dans Interval.

Schedule de type Weekly :

Chaque semaine, la première occurrence T_0 est donnée par le jour de la semaine renseigné dans Day of week et l'heure renseignée dans Time.

Les occurrences suivantes interviendront à l'heure T_i :

$$T_i = T_0 + i \times \Delta t \quad \begin{cases} i < count \\ \forall i \text{ semaine}(T_i) = \text{semaine}(T_0) \end{cases}$$

Δt correspond à la valeur en secondes renseignée dans Interval.

Schedule de type Monthly :

Chaque mois, la première occurrence T_0 est donnée par le numéro de jour du mois renseigné dans Day of month et l'heure renseignée dans Time.

Les occurrences suivantes interviendront à l'heure T_i :

$$T_i = T_0 + i \times \Delta t \quad \begin{cases} i < count \\ \forall i \text{ mois}(T_i) = \text{mois}(T_0) \end{cases}$$

Δt correspond à la valeur en secondes renseignée dans Interval.

Schedule de type Yearly :

Chaque année, la première occurrence T_0 est donnée par la date renseignée dans Date & Time.

Le format est le suivant : AAAA-MM-JJTHH:MM:SS.

Par exemple, pour une première occurrence le 11 février 2012 à 13H00 :

Time = 2012-02-11T13:00:00.

Les occurrences suivantes interviendront à l'heure T_i :

$$T_i = T_0 + i \times \Delta t \quad \begin{cases} i < count \\ \forall i \text{ année}(T_i) = \text{année}(T_0) \end{cases}$$

Δt correspond à la valeur en secondes renseignée dans Interval.

Schedule de type Follower :

Un schedule de type «Follower» sera produit après la fin de chaque occurrence du schedule de référence. Le schedule Parent ne peut pas être de type «Follower».

Ce type permet de déclencher par exemple un téléchargement des données après l'achèvement d'une collecte de données prévue.

Exemple :

Vous souhaitez collecter les données une fois par jour à minuit et télécharger les données, juste après. Vous pouvez configurer un schedule de type Daily pour la collecte de données et un schedule de type Follower du premier schedule pour le téléchargement des données.

Exemples :

Besoin	Type	Time	Day of Week	Day of Month	Date & Time	Interval	Count
Tous les mardi à 15:00:00	Weekly	15:00:00	Tuesday			0	1
Tous les 2 ^e jour du mois à 00:00:00	Monthly	00:00:00		2		0	1
Toutes les jours à 14:00:00	Daily	14:00:00				0	1
Toutes les heures entre 8H00 et 18H00 tous les mardis	Weekly	08:00:00	Tuesday			3600	11
Toutes les 2 heures entre 8H00 et 20H00 le 31 décembre	Yearly				2012-12-31T08:00:00	7200	7

9.6. Configuration du metering

9.6.1. Entrées impulsionnelles

Pulse

Schedule:

Input #1

Enabled:

Label:

Unit:

Input #2

Enabled:

Label:

Unit:

Input #3

Enabled:

Label:

Unit:

Les entrées numériques peuvent être sélectivement configurées en compteurs d'impulsions (voir chapitre associé). Une fois l'entrée activée, un compteur associé va s'incrémenter après chaque impulsion de plus de 10ms. La valeur courante sera sauvegardée pour chaque occurrence du schedule spécifié.

Les paramètres label et unit sont ajoutés dans les données enregistrées avec la valeur d'index.

9.6.2. M-Bus

M-Bus

Enabled:

Last bus scan:

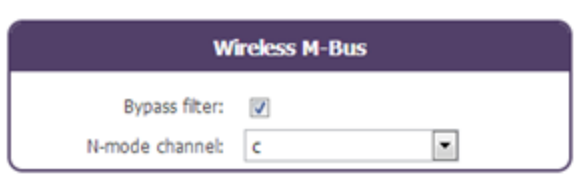
Slave count: 0

Schedule:

Si le port RS232 est configuré en mode MBus , l'acquisition peut être activée avec le schedule associé.

9.6.3. Wireless Mbus 868MHz.

En version de carte radio Wireless M-Bus le datalogger Datalog H60 peut recevoir des données provenant de modules Wireless M-Bus connus (filtrage activé, ou non (filtrage désactivé)).



L'interface permet d'activer ou de désactiver le filtrage des modules et de sélectionner le canal mode N utilisé.

8.2 Mode N - Paramètres de liaison physique

Les sous-modes doivent être affectés aux canaux et aux fréquences comme décrit Tableau 18.

Tableau 18 — Mode N - Fréquences

Sous-mode	Canal ^b	Fréquence centrale [MHz]	Séparation des canaux [kHz]	GFSK [kb/s]	GMSK [kb/s]	4GFSK [kb/s]	Tolérance de fréquence [± kHz]
N1a, N2a	1a ^c	169,406250	12,5		4,8		1,5
N1b, N2b	1b	169,418750	12,5		4,8		1,5
N1c, N2c	2a	169,431250	12,5	2,4			2,0
N1d, N2d	2b	169,443750	12,5	2,4			2,0
N1e, N2e	3a	169,456250	12,5		4,8		1,5
N1f, N2f	3b ^c	169,468750	12,5		4,8		1,5
N2g	0 ^d	169,437500	50			38,4	4,25
a	1	169,412500	25				
a	2	169,437500	25				
a	3	169,462500	25				

^a Ces canaux sont facultatifs et réservés pour utilisation future ou usage spécifique au pays.

^b Désignation des canaux conformément à la décision de la commission de l'UE 2005/928/EC.

^c L'utilisation de ces canaux doit être privilégiée en cas de nécessité de retransmettre les données transmises par le compteur.

^d Ce canal peut être utilisé pour la retransmission par bonds des données de comptage, comme spécifié dans l'EN 13757-5. Le facteur d'utilisation pour l'émission en provenance du compteur doit être limité à 0,02 % sur ce canal.

Extraits de la norme EN 13757-4

Si le filtrage est activé, la configuration des modules est faite en utilisant l'interface suivante :

Nom	Description
Address	Adresse Wireless M-Bus
Label	Nom du module (uniquement informatif)
Key	Clé de cryptage du module (si vide, pas de cryptage)

9.7. Configuration Modbus

Nom	Description
RTU	
Timeout (ms)	Timeout de réponse Modbus/RTU en ms
Turnaround (ms)	Délais de turnaround Modbus/RTU en ms
TCP	
Timeout (ms)	Timeout de réponse Modbus/TCP en ms

En complément de ces paramètres, le port doit être configuré en “modbus”

Modbus Dataset

Id:

Label:

Polling:

Variables

Name	Type	Address	Size	Format	Flags	Threshold low	Threshold high	Threshold hysteresis
<input type="text"/>	INVALID 0	<input type="text"/>	<input type="text"/>	INVALID 0	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Cancel
Apply

La configuration d'un ensemble de données consiste à configurer les paramètres suivants :

Nom	Description
Id	Identifiant unique de l'ensemble de données Modbus
Label	Nom de l'ensemble de données (uniquement informatif)
Polling	Polling continue (true ou false)

Chaque variable est définie par les paramètres suivants :

Nom	Description
Name	Nom de la variable (uniquement informatif)
Type	Type de variable (S0, S1, S3, S4)
Address	Adresse de registre étendue 16-bit
Size	Taille en bits pour discrete input et coil, en octets pour les registres
Format	Raw, boolean, integer, float ou ascii
Flags	cmd_only, little_endian, no_opt, is_status ou is_alarm
Threshold low	Niveau de seuil bas
Threshold high	Niveau de seuil haut
Threshold hysteresis	Hystérésis appliquée aux deux seuils

Voir le manuel d'exploitation pour avoir les détails de configuration.

Modbus Module

Label:

Dataset:

Address:

IP:

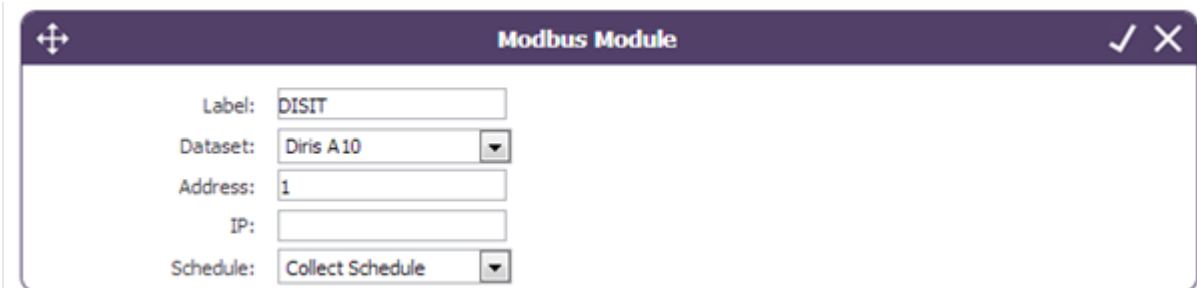
Schedule: INVALID 0

Cancel
Apply

La configuration d'un module Modbus comprend les paramètres suivants:

Nom	Description
Label	Nom uniquement informatif
Dataset	Identifiant de l'ensemble de données
Address	Adresse Modbus (1-247)
IP	Adresse IP (vide pour les équipements RTU)
Schedule	Identifiant du schedule

Afin de vérifier la bonne configuration d'un module Modbus, vous pouvez appuyer sur  :



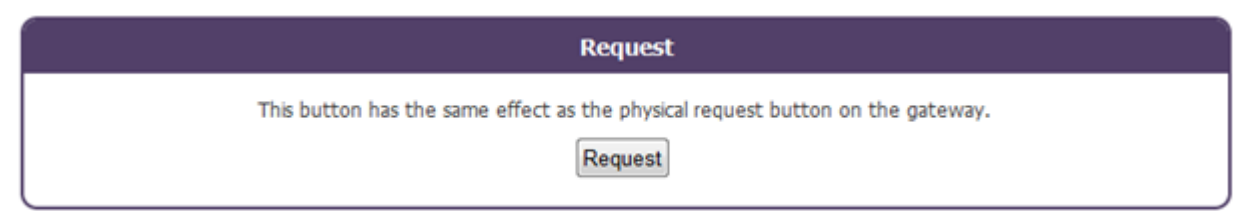
Le résultat de l'interrogation du module Modbus s'affiche dans une fenêtre :

```

name: Power Factor phase 3 -: leading and + : lagging : PF3, address: 0xc586, type: S4, value: 1000
name: Power Factor phase 2 -: leading and + : lagging : PF2, address: 0xc584, type: S4, value: 1000
name: Power Factor phase 1 -: leading and + : lagging : PF1, address: 0xc582, type: S4, value: 1000
Variable Apparent power phase 3 : S3 failed. Error: Illegal data address
name: Apparent power phase 2 : S2, address: 0xc57e, type: S4, value: 0
name: Apparent power phase 1 : S1, address: 0xc57c, type: S4, value: 0
name: Reactive Power phase 3 +/- : Q3, address: 0xc57a, type: S4, value: 0
name: Reactive Power phase 2 +/- : Q2, address: 0xc578, type: S4, value: 0
name: Reactive Power phase 1 +/- : Q1, address: 0xc576, type: S4, value: 0
name: Active Power phase 3 +/- : P3, address: 0xc574, type: S4, value: 0
name: Active Power phase 2 +/- : P2, address: 0xc572, type: S4, value: 0
name: Active Power phase 1 +/- : P1, address: 0xc570, type: S4, value: 0
name: Σ power factor : -: leading et + : lagging : PF, address: 0xc56e, type: S4, value: 1000
name: Σ apparent power : S, address: 0xc56c, type: S4, value: 0
name: Σ reactive Power +/- : Q, address: 0xc56a, type: S4, value: 0
name: Σ active Power +/- : P, address: 0xc568, type: S4, value: 0
name: Neutral Current : In, address: 0xc566, type: S4, value: 0
name: Current : I3, address: 0xc564, type: S4, value: 0
name: Current : I2, address: 0xc562, type: S4, value: 0
name: Current : I1, address: 0xc560, type: S4, value: 0
name: Frequency : F, address: 0xc55e, type: S4, value: 4997
name: Simple voltage : V3, address: 0xc55c, type: S4, value: 0
name: Simple voltage : V2, address: 0xc55a, type: S4, value: 0
name: Simple voltage : V1, address: 0xc558, type: S4, value: 22876
name: Phase to Phase Voltage : I131, address: 0xc556, type: S4, value: 0
    
```

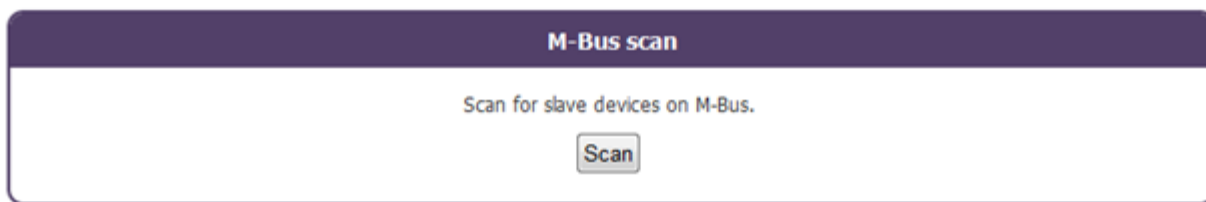
9.8. Actions exécutables

9.8.1. Demande de connexion au serveur distant



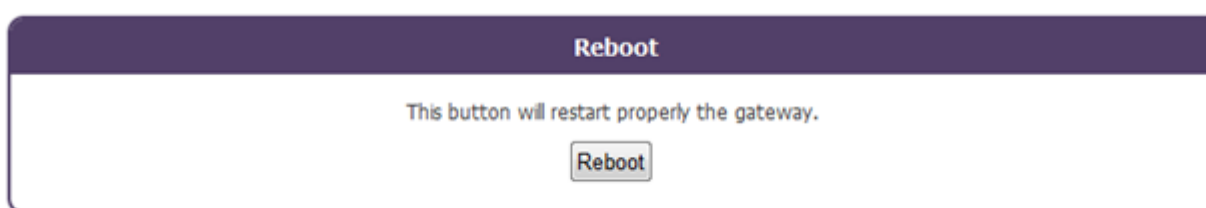
Ce bouton a le même effet que le bouton physique présent en face avant du produit. Une fenêtre popup apparaît affichant toutes les étapes de connexion notamment la synchronisation NTP, la vérification de répertoire Inbox et indique tous les fichiers téléchargés.

9.8.2. Demande de scan M-Bus



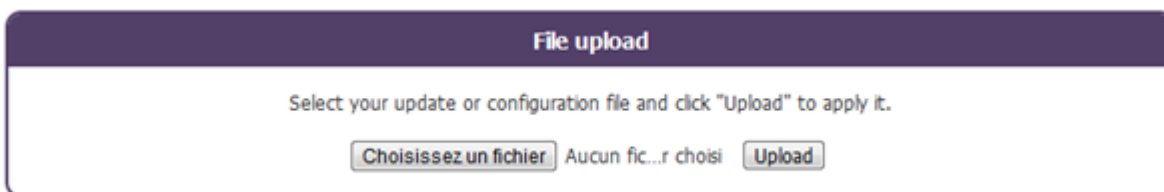
Un scan des équipements M-Bus peut être lancé par ce bouton. Les équipements M-Bus découverts durant ce scan seront interrogés à chaque occurrence du scheduler associé. Si des équipements sont enlevés ou ajoutés sur le bus, un nouveau scan doit être initié afin que le datalogger prenne en compte la modification.

9.8.3. Demande de reboot



Ce bouton permet de redémarrer correctement le datalogger.

9.8.4. Téléchargement de fichiers système



Cette fenêtre permet de télécharger en local sur le datalogger des fichiers de configuration ou un nouveau firmware.

9.9. Demande de connexion au serveur distant

Une fois la configuration effectuée, vous devez provoquer une connexion du datalogger au serveur distant par un appui court sur le bouton REQUEST.

La LED GSM s'allume en cours de connexion GSM.

10. VÉRIFICATION DU BON FONCTIONNEMENT DU DATALOGGER

Afin de vérifier le bon fonctionnement du datalogger et le mapping radio, vous pouvez déclencher un scan des modules et le dépôt d'un fichier de type Supervision.

Pour cela, il existe deux solutions :

- Si vous avez accès au serveur distant, déposer dans le répertoire INBOX une commande de type scan.

Ci-dessous un exemple de commande xml scan:

```
<commands>
  <cmd cid='Install'>
    <scan mode='rtc life-counter rssi data' />
  </cmd>
</commands>
```

Cette commande, déclenche :

- Un relevé de l'horloge, de l'état de la batterie, du niveau RSSI et des données immédiates de chaque module configuré.
- Un dépôt du fichier de supervision contenant ces informations (sauf données immédiates).
- Un dépôt du fichier de données contenant les données immédiates.

Une fois ce fichier de commande déposé sur le serveur distant dans le répertoire INBOX, vous devez provoquer une connexion du datalogger au serveur distant par un appui court sur le bouton REQUEST.

- Si vous n'avez pas accès au serveur distant pour déclencher cette opération, vous pouvez la réaliser via SMS.

SMS à envoyer au datalogger :

```
cmd=scan
cid=Install
mode=rtc,life-counter,rssi,data
```

Les données demandées (sauf pour les données immédiates) seront téléchargées sur le serveur distant en tant que données de supervision utilisant la méthode de téléchargement configurée correspondante. Les données immédiates (données scan) seront téléchargées sur le serveur distant en tant que données en utilisant la méthode de téléchargement configuré correspondante.

Une fois que le datalogger a déposé ces données, vous pouvez vérifier les informations remontées.

11. SERVEUR FTP LOCAL

Le datalogger embarque un serveur FTP local. Il fonctionne comme un INBOX local. Un seul fichier peut être déposé sur ce serveur à la fois.

Il accepte des fichiers de configuration et des fichiers binaires de mise à jour.

12. ARRÊT DU DATALOG H60

Débranchez le câble d'alimentation du connecteur 12/24V.

Le datalogger Datalog H60 est équipée d'une batterie de secours. Sur perte d'alimentation elle envoie une alerte au serveur puis continue à fonctionner jusqu'à ce que la batterie soit vide.

13. COMMUNICATION AVEC LE SERVEUR DISTANT

Le datalogger datalog H60 communique avec un serveur distant via un serveur FTP et/ou un serveur Web Services. Gestion du serveur Web Services non-implémenté

Chaque fois que sa configuration est modifiée, le datalogger peut soit la télécharger sur un serveur FTP soit l'envoyer vers un serveur Web Services (WS).

De même, les alarmes et les données collectées peuvent être téléchargés sur un serveur FTP ou envoyées à un serveur Web Services.

Lorsque vous utilisez le transfert FTP, le datalogger peut également informer un serveur Web Services de tout téléchargement FTP.

Le serveur peut aussi lancer des actions sur le datalogger en plaçant des fichiers de commande dans un répertoire INBOX sur le serveur FTP ou en lui envoyant des commandes lorsque le datalogger interroge le Web Services INBOX. Les commandes peuvent également être envoyées au datalogger par SMS.

13.1. Modes de connexion

La connexion au serveur distant peut être établie via Ethernet ou un réseau cellulaire (GPRS ou 3G en fonction de la configuration matérielle). Les échanges entre le datalogger et le serveur distant sont toujours initiés par le datalogger, mais différentes méthodes sont disponibles pour le serveur distant pour déclencher un échange.

Le datalogger peut être configurée pour utiliser le modem dans un des quatre modes suivants :

- **On demand** : Dans ce mode, le lien PPP est créé lorsque le datalogger a besoin de communiquer avec le serveur distant. Le lien sera fermé après l'achèvement de la communication avec le serveur distant.
- **Always On** : Dans ce mode, le lien PPP sera maintenue en permanence de façon indépendante. Dans ce mode, un mécanisme de keepalive peut être activé pour s'assurer que le lien est fonctionnel. **Mode Always On non-implémenté**
- **Always Off** : Dans ce mode, le lien PPP n'est jamais créé. Toutes les communications avec le serveur passent par l'interface Ethernet. Le modem (si une carte SIM valide est présente) est cependant connecté au réseau cellulaire, prêt à recevoir les appels entrants et / ou SMS.
- **Off** : Dans ce mode, le modem est hors tension.

13.1.1. Gestion du code PIN de la carte SIM

Le datalogger peut être configuré pour utiliser une carte SIM:

- Sans code PIN : `/com/modem/pin/mode=off`
- Avec le code PIN : `/com/modem/pin/mode>manual` et `/com/modem/pin/code=<CODE PIN>`
- Avec un code PIN automatique : Mode automatique non-implémenté

13.2. Serveur FTP distant

Le datalogger utilise les fichiers suivants sur le serveur FTP :

Nom	Description
CONFIG/<uid>.xml	Configuration actuelle du datalogger. Ce fichier est transféré par celui-ci après chaque modification de sa configuration. La modification de ce fichier n'a pas d'effet sur le datalogger. Le datalogger va simplement l'écraser la prochaine fois que sa configuration est modifiée (voir ci-dessous INBOX).
DATA/<uid>-<timestamp>.<format>.gz	Fichiers de données téléchargés par le datalogger sur le serveur distant.
ALARM/<uid>-<timestamp>.xml.gz	Fichiers d'alarme téléchargés par le datalogger sur le serveur distant.
SUPERVISION/<uid>-<timestamp>.xml.gz	Fichiers de supervision téléchargés par le datalogger sur le serveur distant (états et résultats de scan).
SUPERVISION/<uid>-<timestamp>.log.gz	Fichiers de logs téléchargés par le datalogger sur le serveur distant sur requête.
INBOX/<uid>/*.xml	Le datalogger surveille ce répertoire. Tout fichier placé dans celui-ci sera téléchargé et traité par le datalogger.
BIN/<firmware>	Ce répertoire contient le firmware du datalogger pour réaliser une mise à jour

Dans le tableau ci-dessus, <uid>, <timestamp> et <format> doivent être remplacés respectivement par l'identifiant unique du datalogger, l'horodatage du téléchargement et le format (xml).

Le format d'horodatage est «AAAAMMJJ-HHMMSS» de sorte qu'un tri alphabétique du répertoire donne l'ordre chronologique.

Les fichiers avec l'extension « .gz » sont compressés.

Le datalogger télécharge toujours les fichiers en suivant un processus en 2 étapes :

- Le fichier est téléchargé avec une extension supplémentaire « .tmp ».
- Le fichier est renommé en supprimant l'extension « .tmp ».

Ce processus permet au serveur distant de distinguer facilement les fichiers en cours de téléchargement des fichiers complètement téléchargés.



Le schéma XML et un exemple de fichier de configuration de configuration est présenté dans l'annexe :

- XML des alarmes
- XML de supervision
- XML des commandes



Les schémas XML spécifiant le format des différents fichiers XML utilisés par le datalogger peut évoluer dans les futures versions lorsque de nouvelles fonctionnalités seront ajoutées. Ces changements seront apportés afin que les anciens fichiers XML restent compatibles avec les nouveaux schémas XML. De même, comme les fichiers XML générés par le datalogger peuvent contenir des éléments supplémentaires, leur traitement doit être mis en œuvre afin que les nouveaux éléments soient ignorés.

13.3. Serveur Web Services distant

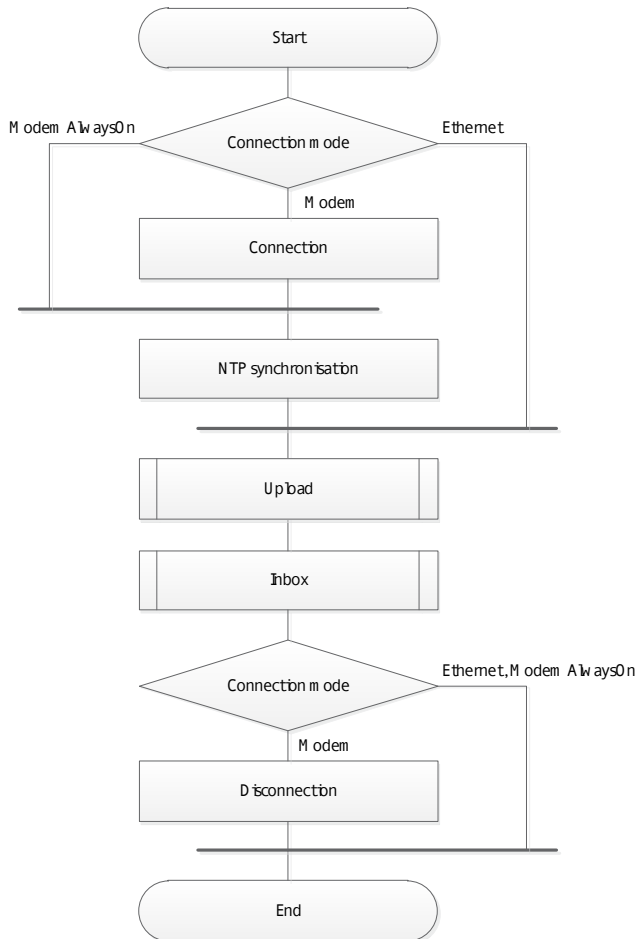
Non-implémenté

13.4. Connexion au serveur distant

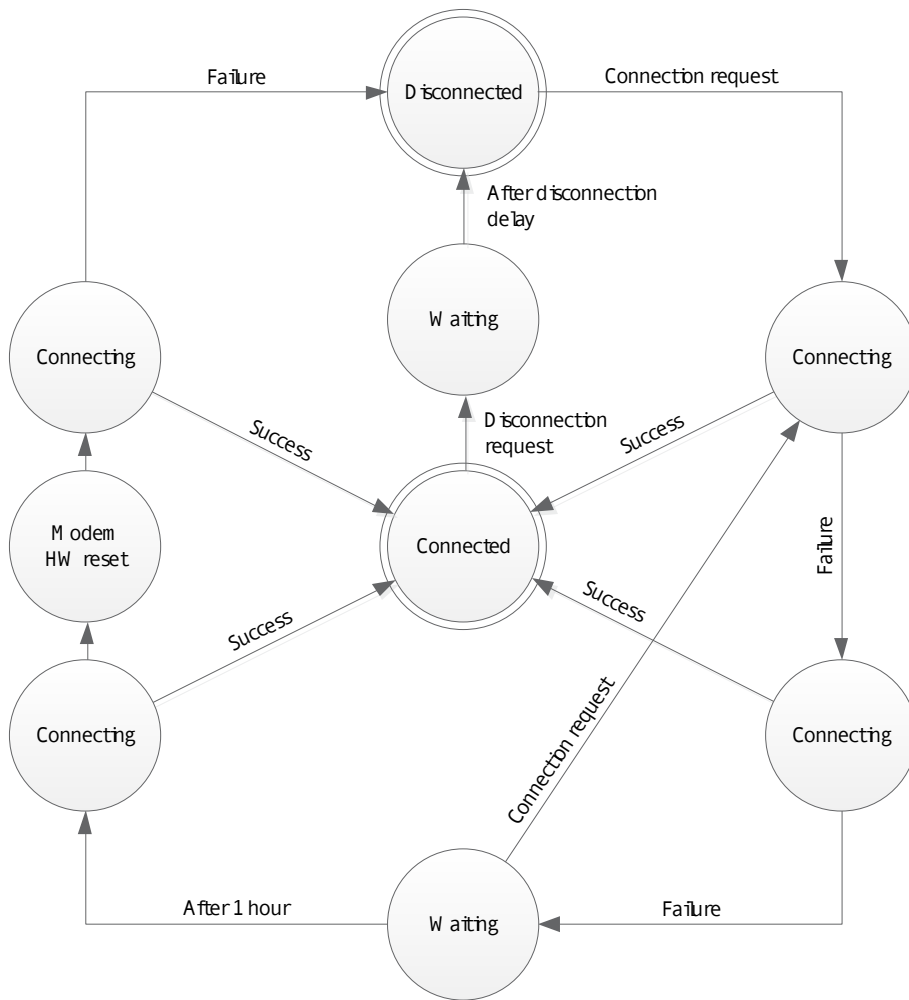
Une connexion au serveur distant peut être initiée par l'un des événements suivants :

- Scheduler de connexion
- Emission d'une alarme
- Modification de la configuration
- SMS de demande de connexion
- Bouton REQUEST (via les pages web ou le bouton en face avant du boîtier)

Indépendamment de l'événement déclencheur, le processus suivant est exécuté :



En cas d'échec de la connexion GPRS, une nouvelle connexion va être tentée une heure plus tard, à moins qu'une nouvelle connexion n'ait été lancée dans l'intervalle (sur demande explicite, ou déclenchée par une alarme ou par un envoi périodique). Ceci est illustré ci-dessous, précisant le délai de déconnexion :



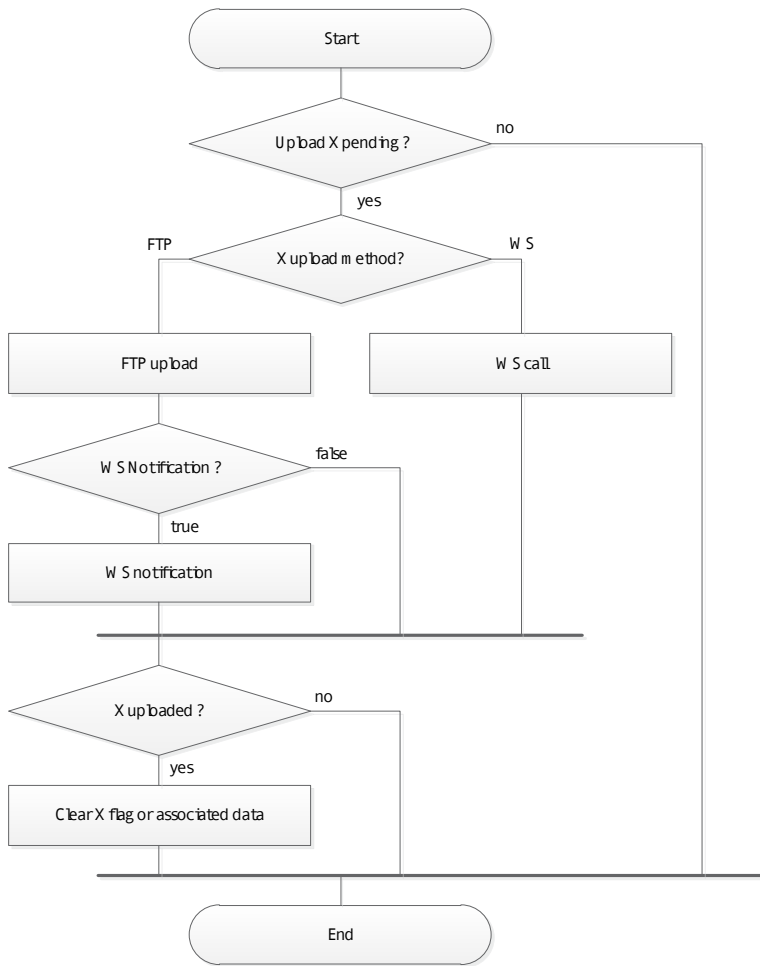
13.4.1. Process de téléchargement sur le serveur distant (Upload)

Configuration, alarme, supervision et données sont téléchargées sur le serveur distant par le datalogger de façon indépendante comme décrit dans le schéma ci-dessous.

Dans ce diagramme, X représente le type de téléchargement (configuration, alarme, supervision ou données).

Après l'achèvement d'un téléchargement d'une configuration, le drapeau associé est effacé.

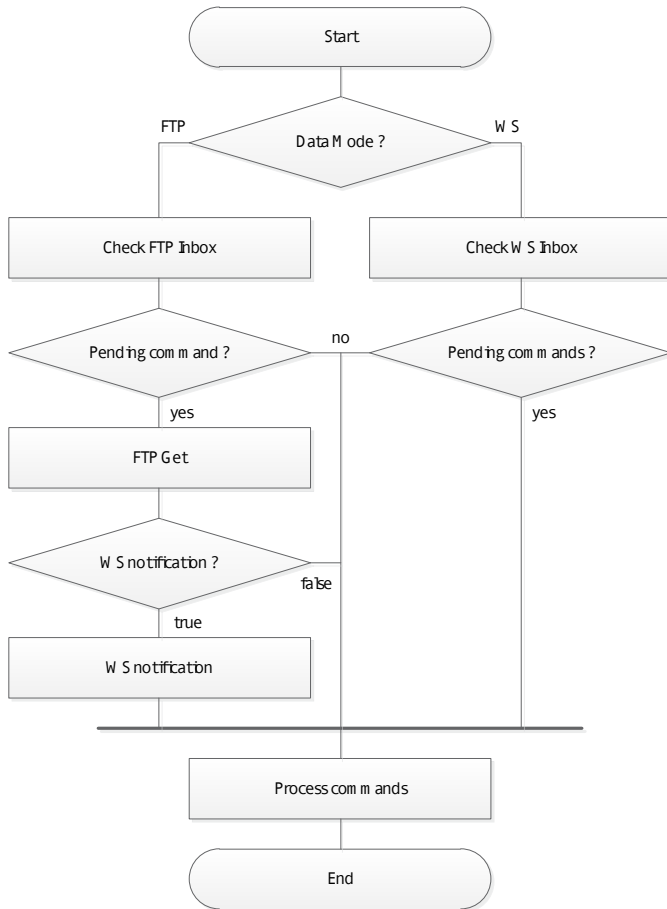
Après l'achèvement d'un téléchargement d'une alarme/supervision/données, les données associées sont effacées.



Remarque : La gestion du mode WS n'est pas implémentée

13.4.2. Inbox

Le datalogger vérifie les actions en attente comme suit:



13.5. Bouton Request

Par défaut, appuyer sur le bouton «request» va déclencher une connexion au serveur distant et le téléchargement, en plus des données en attente, du status du datalogger. Les deux peuvent être sélectivement désactivés avec les paramètres de configuration :

`/com/request/upload`

et

`/com/request/include_status.`

Un SMS d'état peut-être également envoyé au destinataire spécifié le paramètre suivant :

`/com/request/sms_status_recipient`

Dans le cas où ce champ est vide, aucun SMS ne sera envoyé.

14. COMMANDES

Les commandes peuvent être envoyées via le serveur distant (FTP ou WS) ou via SMS.

Lorsque le datalogger reçoit un SMS, elle vérifie la liste blanche des numéros de téléphone autorisés (caller_id). S'il est autorisé, le contenu du SMS est traité.

Commande	Description	Retour
reboot	Redémarre le datalogger	Aucun
factory	Restaure les paramètres usine du datalogger	Aucun
update	Lance la mise à jour du firmware du datalogger	Alarme (SW)
scan	Lance un scan des valeurs instantanées de modules Radio W-Mbus, de leurs niveaux RSSI, de leurs niveaux de batterie, et/ou de leurs horloges RTC	Données Supervision (sauf données scan)
timesync	Lance la mise à jour des horloges RTC de modules radios	Données Supervision
status	Demande d'envoi du status du datalogger	Données Supervision
log	Demande d'envoi des données de Log	Données Log
config	Modification de la configuration du datalogger (seulement SMS)	Dépôt de la configuration
connect	Déclenche une connexion au serveur distant (seulement SMS)	Implicite (connexion)

Les commandes ne sont pas acquittées lorsqu'elles sont reçues. Toutes les commandes sont enregistrées et une commande non valide déclenche une alarme, qui est téléchargée vers le serveur distant.

Toutes les commandes acceptent deux paramètres facultatifs «uid» et «cid»:

- uid : identifiant unique du datalogger
- cid : identifiant de commande

Une commande sera rejetée si le paramètre uid inclus et ne correspond pas à l'uid du datalogger. Le cid peut être librement choisi par l'émetteur de la commande. Il sera inclus avec tout téléchargement associé.

Le schéma et un exemple XML des commandes sont présentés en annexe.

14.1. Commandes spécifiques Modbus

Les commandes spécifiques Modbus peuvent être envoyées à un module Modbus module connu en utilisant la commande "modbus".

La liste des commandes supportées est :

Sous-commande	Description
Write	Ecrire une valeur sur un équipement Modbus

Les adresses des variables doivent être formatées tel que expliqué au §11.4.

14.2. Commande Status

Exemple : Demande d'envoi du status de la passerelle :

```
XML:
<cmd cid='C_1237'>
  <status/>
</cmd>
```

```
SMS:
cmd=status
cid=C_1238
```

Les informations suivantes sont renvoyées au demandeur :

Nom XML	Nom SMS	Description
-	uid	Identifiant unique du datalogger
/app/version	version	Version logicielle du datalogger
/app/kernel	kernel	Linux kernel version.
/system/power	power	Présence de l'alimentation externe (booléen)
/system/defaults	defaults	Codes défauts séparés par des virgules
/com/modem/model	m_model	Nom du model de modem
/com/modem/firmware	m_version	Version du firmware du modem
/com/modem/imei	imei	International Mobile Equipment Identity
/com/modem/msisdn	msisdn	Mobile Subscriber ISDN Number (si disponible)
/com/modem/rssi	rssi	Puissance du signal reçu en dBm
/com/modem/csq	csq	Qualité de signal (CSQ, BER)
/com/modem/ip	m_ip	Adresse IP du datalogger sur l'interface modem (ou dernière adresse attribuée).
/com/ethernet/ip	e_ip	Adresse IP du datalogger sur l'interface Ethernet
/com/upload/last	u_last	Date de la dernière connexion au serveur distant réussie (périodique ou déclenchée)
/com/upload/next	u_next	Date de la prochaine connexion périodique au serveur distant

Lorsque la commande d'état a été envoyée par SMS, le statut est renvoyé dans un SMS multiple avec une variable par ligne (nom = valeur).

Lorsque la commande d'état vient de INBOX (FTP ou WS), le fichier XML est téléchargé sous forme d'un fichier xml de supervision.

14.3. Commande de mise à jour

Commande de mise à jour du firmware du datalogger :

```
XML:
<cmd cid='C_1238'>
  <update>
    <firmware>wrf_w-mbus_v101.bin</firmware>
    <checksum>c1fb7d81f3d53a8b7bf94098115249d3</checksum>
  </update>
</cmd>
```

```
SMS:
cmd=update
cid=C_1237
firmware=wrf_w-mbus_v101.bin
checksum=c1fb7d81f3d53a8b7bf94098115249d3
```

Le fichier du firmware doit être disponible dans le répertoire BIN sur le serveur FTP. Le checksum correspond au checksum md5 du fichier.

14.4. Commande d'un contact sec (sortie numérique)

Exemple: Ouvrir un contact sec du datalogger (sortie numérique) :

```
XML:  
<cmd cid='C_1239'>  
  <d_output subcmd='open' />  
</cmd>  
SMS:  
cmd=d_output  
cid=C_1239  
subcmd=open
```

15. CONFIGURATION DU DATALOGGER

15.1. Paramètres

Les paramètres du datalogger sont traités de manière structurée. La configuration peut être exportée dans un fichier XML. L'installation d'une nouvelle configuration et la modification de la configuration actuelle sont réalisées en utilisant un fichier XML avec le même format.

Le schéma XML de configuration est présenté dans le paragraphe 14.

Un exemple de fichier XML de configuration est présenté dans le paragraphe 15.

Les principaux paramètres du datalogger sont énumérés ci-dessous (en gras sont indiqués les valeurs par défaut configurées en usine) :

Nom	Valeur	Description
/uid	6 derniers digits de l'adresse MAC Ethernet	Identifiant unique du datalogger
/name	Adresse MAC préfixé de « WGRF_ »	Nom du datalogger (uniquement informatif)
/enable_local_config	false, true	Activation/désactivation de la configuration locale non-implémenté
/com/modem/pin/mode	off , manual, automatic	Mode de gestion du code PIN de la carte SIM (voir §3.2) automatic non-implémenté
/com/modem/pin/code	0000	Code PIN de la carte SIM
/com/modem/call_number	*99***1#	Numéro d'appel de la connexion GPRS
/com/modem/apn		APN
/com/modem/login		Login APN
/com/modem/password		Mot de passe APN
/com/modem/mode	ondemand , alwayson, alwaysoff, off	Voir la description des modes §3.1). alwayson non-implémenté
/com/modem/delay	60	Délais en secondes avant déconnexion dans le mode de connexion ondemand non-implémenté
/com/modem/whitelist/caller_id		Liste blanche de numéros d'appelants autorisés pour la réception de SMS de commande. Si vide, pas de vérification
/com/ethernet/use_dhcp	false , true	Activation/désactivation du client DHCP
/com/ethernet/ip	192.168.1.12	Adresse IP du Datalog H60
/com/ethernet/netmask	255.255.255.0	Masque de réseau IP
/com/ethernet/gateway		Adresse IP de la passerelle réseau
/com/ethernet/dns/server		Adresse IP du serveur DNS
/com/keepalive/method	icmp, tcp, off	Méthode Keepalive (non-implémenté)
/com/keepalive/address		non-implémenté
/com/keepalive/port	5000	non-implémenté
/com/keepalive/period	600	non-implémenté
/com/keepalive/timeout	30	non-implémenté
/com/request/upload	false, true	Connexion au serveur distant sur appui du bouton Request
/com/request/include_status	false, true	Téléchargement du status du datalogger sur le serveur distant sur appui du bouton Request
/com/request/sms_status_recipient		Destinataire du SMS de status envoyé sur appui du bouton Request
/com/time/ntp/server		Adresse du serveur NTP
/com/time/timezone		Fuseau horaire local (utilise le nom zoneinfo standard tel que "Europe/Paris")

Nom	Valeur	Description
/com/time/alarm_threshold	0	Seuil d'alarme de désynchronisation en secondes (0=off).
/com/ftp/address		Adresse du serveur FTP
/com/ftp/login		Identifiant FTP
/com/ftp/password		Mot de passe FTP
/com/ftp/mode	passive , active	Mode FTP
/com/ftp/secured	false , true	Activation/Désactivation du protocole FTP sécurisé FTPS non implémenté
/com/ftp/root_path	/	Répertoire racine sur le serveur FTP
/com/ftp/ws_notification	none , put, get, both	Mode notification WS de téléchargement FTP de fichier
/com/ws/address		Adresse WS - <i>ws non implémenté</i>
/com/ws/login		Identifiant WS - <i>ws non implémenté</i>
/com/ws/password		Mot de passe WS - <i>ws non implémenté</i>
/com/ws/secured	false , true	Activation/Désactivation SSL/TLS (HTTPS) pour WS - <i>ws non implémenté</i>
/upload/config/method	ftp ws none	Méthode utilisée pour le téléchargement de la configuration - <i>ws non implémenté</i>
/upload/alarm/method	ftp ws	Méthode utilisée pour le téléchargement des alarmes - <i>ws non implémenté</i>
/upload/supervision/method	ftp ws	Méthode utilisée pour le téléchargement des données de supervision - <i>ws non implémenté</i>
/upload/data/method	ftp ws	Méthode utilisée pour le téléchargement des données
/upload/data/format	xml csv	Format utilisé pour les données téléchargées
/upload/data/schedule		ID du schedule utilisé pour le téléchargement des données
/alarm/*		Configuration du moteur d'alarme
/scheduler/*		Configuration des schedules
/metering/*		Configuration Metering
/rfid/*		Configuration Active RFID
/system/log/level	7	Niveau de log
/system/password/admin	high	Mots de passe pour l'accès aux services HTTP et FTP locaux.
/system/password/install	medium	
/system/password/data	low	
/system/ports/*		

La configuration peut être modifiée localement ou à distance. Toute modification déclenche le téléchargement de la nouvelle configuration sur le serveur.

15.2. Précisions sur les numéros de téléphone

/com/modem/whitelist

Si la liste est vide, tous les numéros sont considérés comme valides.

/com/modem/whitelist/caller_id

/com/request/sms_status_recipient

Les numéros de téléphone doivent être écrit en format international.
Ils doivent commencer par + et le code du pays.

15.3. Configuration par SMS

La configuration initiale du datalogger peut être faite par SMS. Notamment, les paramètres de connexion peuvent être envoyés par SMS. Une fois cette configuration initiale terminée, le serveur distant peut compléter la configuration de la passerelle.

La première ligne du SMS doit contenir la commande «CMD=config».

Les lignes suivantes doivent avoir le format «SHORTNAME=VALEUR». Le SHORTNAME est constitué des premières lettres de chaque élément composant le nom du paramètre :

Par exemple, le SHORTNAME de «com/modem/login» est LMC.

Le contenu SMS est soumis aux règles suivantes:

- Les caractères espace en fin de ligne sont ignorés.
- Les SHORTNAME ne sont pas sensibles à la casse.
- Les valeurs booléennes, true et false, peuvent être remplacés respectivement par 0 et 1.
- Le retour chariot peut être remplacé par un point-virgule, mais les deux ne peuvent pas être mélangés dans un même SMS.
- Dans le cas de l'utilisation du caractère ';' comme séparateur de variables, il n'est pas possible d'utiliser ce caractère dans les valeurs des variables.

La longueur d'un SMS est limitée à 160 caractères.

Seuls les paramètres principaux et de communication peuvent être modifiés par SMS :

- /uid
- /name
- /enable_local_config
- /com/

Exemple :

Pour réaliser la configuration initiale d'un datalogger avec le contexte suivant:

- APN « m2minternet » ne nécessitant pas un identifiant / mot de passe.
- Communication avec le serveur distant de type FTP (168.112.23.123) en mode passif.

Vous pouvez envoyer le SMS suivant:

```
CMD=config
CMA=m2minternet
CFA=168.112.23.123
CFL=login
CFP=password
```

Remarque : Tous les paramètres utilisant leur valeur par défaut ont été omis.

Sur réception de ce SMS, le datalogger applique les paramètres et se connecte au serveur distant pour déposer le fichier de configuration qui en résulte. A partir de là, le datalogger peut être configurée à distance comme décrit ci-dessous.

Pour construire une liste (par exemple /com/modem/whitelist/caller_id), la variable concernée doit être répétée dans le SMS. Si la variable apparaît au moins une fois, la liste actuelle est remplacée. Si elle apparaît qu'une seule fois et sans valeur, la liste actuelle est effacée.

Exemples :

```
CMD=config
CTNS=1.2.3.4
```

Après le traitement de ce SMS, la passerelle va utiliser le serveur DNS 1.2.3.4.

```
CMD=config
CTNS=1.2.3.4
CTNS=1.2.3.5
```

Après le traitement de ce SMS, le datalogger va utiliser 2 serveurs DNS 1.2.3.4 et 1.2.3.5.

```
CMD=config
CTNS=
```

Après le traitement de ce SMS, le datalogger ne va plus utiliser aucun serveur DNS.

15.4. Configuration locale

Le datalogger peut être configuré localement à travers une interface Web.

15.5. Configuration distante

Le serveur distant peut modifier la configuration en plaçant un fichier de configuration XML dans le répertoire INBOX sur le serveur FTP ou par signification à travers le service Web. Le format XML est utilisé dans les deux cas.

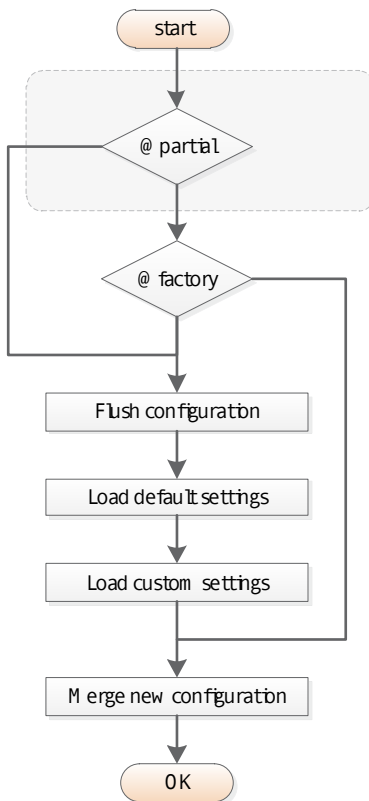
WS non implémenté

Le fichier XML est traité comme une nouvelle configuration si l'attribut XML factory est présent et égale la valeur true.



AVERTISSEMENT

L'attribut partial est toujours supporté mais obsolète. L'attribut factory égal à true est équivalent à l'attribut partial égal à false.



Lorsque factory n'est pas présent ou égale à false, seuls les valeurs des paramètres de configuration présents dans le nouveau fichier de configuration sont mises à jour.

Lorsqu'une liste est présente dans le nouveau fichier de configuration, la liste entière est remplacée. C'est notamment le cas pour les schedules.

15.6. Contrôle de l'accès local

L'accès aux services HTTP et FTP locaux sont protégés par login/mot de passe. Toute tentative d'utilisation de ces services est enregistrée.

Trois niveaux d'accès sont définis : « admin », « install » et « data ».

L'administrateur (admin) possède les droits d'accès en lecture/écriture à tous les paramètres de configuration, en lecture pour les informations d'état du datalogger et peut déclencher des actions.

L'installateur (install) possède les droits d'accès en lecture/écriture aux paramètres de configuration des modules finaux.

L'utilisateur des données (data) possède seulement le droit d'accès en lecture aux informations d'état du datalogger.

	Configuration datalogger	Etat datalogger	Configuration LAN	Actions
admin	R/W	R	R/W	Yes
install		R	R/W	Yes
data		R		No

Le mot de passe associé aux niveaux d'accès sont configurés dans /system/password. Ils peuvent être modifiés uniquement par un fichier de configuration depuis le serveur distant ou localement par l'administrateur.



Il est fortement recommandé de modifier ces mots de passe avant déploiement.

Restriction sur les mots de passe : ils ne doivent pas contenir les caractères « , & , ' , < , > , ? , ` ».

Lorsqu'un fichier de configuration est téléchargé sur le FTP local par l'installateur (install), il sera rejeté s'il contient des paramètres n'appartenant pas à la configuration LAN.

15.7. Configuration des ports

Les paramètres suivants dans /config/system/ports sont utilisés pour configurer le mode de fonctionnement des ports :

Nom	Valeur	Description
rs232/mode	off , mbus	Mode RS232
rs485/mode	off	Mode RS485 (voir ci-dessous)
input_1/mode	d_input , pulse	Mode entrée numérique
input_2/mode	d_input , pulse	Mode entrée numérique
input_3/mode	d_input , pulse	Mode entrée numérique

Les paramètres du port RS485, dans /config/system/ports/rs485/, sont :

Nom	Valeur
Mode	off , modbus
baudrate	4800, 9600, 19200 , 38400, 57600, 115200
Data	8
parity	odd, even , none
stop_bit	1 , 2

Remarque : La spécification Modbus précise que si aucun bit de parité n'est utilisé, 2 bits de stop bits doivent être utilisés.

16. IMPULSIONS

Les entrées numériques peuvent être sélectivement configurées en compteurs d'impulsions (voir paragraphe associé).

En mode compteur d'impulsions, un compteur associé va s'incrémenter après chaque impulsion de plus de 10ms. La valeur courante sera sauvegardée pour chaque occurrence du schedule spécifié. Les paramètres suivants sont configurés dans /config/metering/pulse :

Nom	Description
Schedule	ID du schedule pour l'acquisition des impulsions
input_1/label	Nom de l'entrée (uniquement informatif)
input_1/unit	Unité (et poids) d'impulsion (uniquement informatif)
input_2/label	Nom de l'entrée (uniquement informatif)
input_2/unit	Unité (et poids) d'impulsion (uniquement informatif)
input_3/label	Nom de l'entrée (uniquement informatif)
input_3/unit	Unité (et poids) d'impulsion (uniquement informatif)

Les paramètres « label » et « unit » sont ajoutés dans les données enregistrées avec la valeur d'index.

17. WIRED M-BUS

Des données peuvent être collectées d'un équipement M-Bus. Pour cela, un transceiver M-Bus doit être connecté au port RS232 et ce port doit être configuré en mode M-Bus .

Les équipements M-Bus doivent être configurés avec une adresse unique sur le bus.

Un scan du bus doit être initié depuis l'interface web . Les équipements M-Bus découverts durant ce scan seront interrogés à chaque occurrence du scheduler associé. Si des équipements sont enlevés ou ajoutés sur le bus, un nouveau scan doit être initié afin que le datalogger prenne en compte la modification.

Les paramètres suivants sont configurés dans /config/metering/mbus :

Nom	Description
Schedule	Schedule ID for M-Bus data collection

18. WIRELESS M-BUS

En version de carte radio Wireless M-Bus le datalogger Datalog H60 peut recevoir des données provenant de modules Wireless M-Bus connus en mode S1 ou T1.

Les paramètres suivants sont configurés dans /config/metering/wmbus :

Nom	Valeur	Description
mode	S1, T1	Mode Wireless M-Bus
long_preamble	true, false	Longueur du préambule Radio (ignoré en mode T)
modules/*		Liste des modules Wireless M-Bus

Si le cryptage OMS est activé, le nombre de modules est limité à 64. Les modules sans les clés de chiffrement seront ignorés.

Chaque module est configuré comme suit :

Nom	Description
module/address	Adresse Wireless M-Bus
module/label	Nom du module (uniquement informatif)
module/key	Clé de cryptage du module

19. TAGS RFID ACTIFS

En version de carte radio RFID, la passerelle peut:

- Surveiller la présence de tag RFID actifs dans sa zone de couverture.
- Recueillir des données provenant de tag RFID actifs.

Le récepteur est compatible avec les tags actifs de ELA Innovation. <http://www.ela.fr>

Les tags doivent être configurés en mode 24 bits avec checksum radio de 16 bits.

Le datalogger reçoit la transmission périodique de l'ensemble des tags RFID actifs.

Un décalage de CRC optionnel peut être configuré pour filtrer tous les tags qui ne sont pas configurés avec ce même décalage.

Tous les ID reçus avec un RSSI au-dessus du seuil configuré sont ignorés. Ce seuil permet de réduire la zone de couverture.

Remarque : La gamme RSSI est de 110 à 200. Si le seuil est fixé à une valeur inférieure à 110, aucune donnée ne sera reçue. Si le seuil est fixé à une valeur supérieure à 200, toutes les données reçues sont traitées.

Si le tag est de type ID (le bit de poids fort de l'ID est égal à 0), les 3 bits suivants sont traités comme des indicateurs d'alarme et ne sont donc pas considérés comme faisant partie de l'ID.

Si le tag est de type ID+DATA (le bit de poids fort de l'ID est égal à 1), les 12 premiers bits sont utilisés comme l'ID et les 12 bits suivants sont considérés comme des données.

Le datalogger gère une liste de tags dans sa zone de couverture. Un tag entrera dans cette liste seulement si son ID a été reçu régulièrement pendant le délai de détection «entering». Une étiquette sera retirée de cette liste dès que son ID n'a pas été reçu pendant le délai de détection de «leaving».

Lorsqu'un tag ID+DATA est considéré comme étant dans la zone de couverture, ses données seront enregistrées et téléchargées sur le serveur distant.

Les alarmes peuvent être envoyées immédiatement ou lors de la prochaine connexion pour les événements suivants : entrer d'un tag, sortie d'un tag, un indicateur est défini dans l'ID d'un tag de type ID.

Chaque fois que la passerelle se connecte au serveur distant, elle envoie :

- Toutes les données recueillies à partir de tags ID+DATA.
- Une liste de tous les tags dans la zone de couverture.
- Toutes les alarmes différées.

Les paramètres suivants sont disponibles dans /config /rfid :

Nom	Valeur	Description
rssi_threshold	0- 255	Niveau RSSI de filtrage des tags
Crc	0	Décalage de CRC optionnel
detection_delay/entering	600	Délais avant qu'un tag soit considéré dans la zone
detection_delay/leaving	500	Délais avant qu'un tag soit considéré hors zone
alarm/sources/entering	on , off, delayed	Alarme lorsqu'un tag entre dans la zone
alarm/sources/leaving	on , off, delayed	Alarme lorsqu'un tag sort de la zone
alarm/sources/id_flags	on, off, delayed	Alarme lorsqu'un tag ID présente un indicateur est défini

Remarque : Les données reçues de tags ID+DATA sont enregistrés sous forme de données brutes. Elles ne sont pas converties en température / humidité / mouvement car le datalogger n'est pas au courant de ce type d'information. En outre, les valeurs spécifiques de type «end of battery life» ne sont pas reconnues (puisque cette valeur dépend de son type, 0x7FF pour les tags T/HR et 0xFFF pour les tags MOV).

20. MODBUS

Le datalogger peut fonctionner comme un dispositif maître Modbus. Cette fonctionnalité permet de lire / écrire dans les registres sur des modules Modbus RTU et TCP esclaves.

La configuration Modbus du datalogger consiste en une liste d'ensembles de données et une liste de modules. Un ensemble de données est une liste de registres pour un type donné d'équipement esclave Modbus. La liste des modules associe un module esclave Modbus à un ensemble de données et un scheduler.

En mode polling, la valeur de toutes les variables sont continuellement mises à jour. Ces valeurs peuvent être surveillées pour détecter des changements ou les comparer à des seuils.

Les valeurs courantes seront enregistrées lorsque :

- La valeur d'une donnée surveillée change ou franchit un certain seuil,
- Le schedule associé se produit.

En mode d'instantané (i.e. pas en mode polling), les valeurs de toutes les variables sont mises à jour et enregistrées lorsque et seulement lorsque le schedule associé se produit.

Indépendamment du processus de collecte des données Modbus, il est possible d'écrire dans les registres de certains modules esclaves à l'aide de la commande Modbus.

20.1. Configuration

La configuration Modbus dans /config/modbus contient les paramètres suivants :

Nom	Valeur	Description
tcp/timeout	1000	Timeout de réponse Modbus/TCP en ms
rtu/timeout	1000	Timeout de réponse Modbus/RTU en ms
rtu/turnaround	100	Délais de turnaround Modbus/RTU en ms
datasets/*		Liste d'ensembles de données
modules/*		List de modules

En complément de ces paramètres, les paramètres /config/system/ports/rs485/ doivent être correctement configurés. En particulier, le paramètre /config/system/ports/rs485/mode doit être configuré égale à "modbus".

20.2. Ensembles de données Modbus

La configuration d'un ensemble de données (/config/modbus/datasets/dataset) consiste à configurer les paramètres suivants :

Nom	Description
id	Identifiant unique de l'ensemble de données Modbus
label	Nom de l'ensemble de données (uniquement informatif)
vars/*	Liste de variables
boundaries/*	Liste des registres limites
polling	Polling continue (true or false)

20.2.1. Variables

Chaque variable est définie dans /config/modbus/datasets/dataset/vars/var par les paramètres suivants :

Nom	Description
name	Nom de la variable (uniquement informatif)
type	Type de variable (S0, S1, S3, S4)
address	Adresse de registre étendue 16-bit
size	Taille en bits pour discrete input et coil, en octets pour les registres
format	Voir liste ci-dessous
flags	Liste de flags séparés par des virgules (voir la définition du flag ci-dessous)

Nom	Description
threshold/low	Niveau de seuil bas (voir ci-dessous)
threshold/high	Niveau de seuil haut (voir ci-dessous)
threshold/hysteresis	Hystérésis appliquée aux deux seuils

20.2.1.1. Paramètre "type"

Le type d'une variable est l'un des quatre types de registres Modbus.

Type	Description	Read (multiple)	Write (single)	Write (multiple)
S1	Discrete input	0x02	-	-
S0	Coil	0x01	0x05	0x0F
S3	Input register	0x04	-	-
S4	Holding register	0x03	0x06	0x10

Dans le tableau ci-dessous, les codes fonction de lecture / écriture sont donnés à titre indicatif. Les requêtes Modbus ne font pas partie de la configuration, mais seront déduites. En particulier plusieurs codes de fonction Read seront utilisés partout où cela réduit les coûts de communication.

20.2.1.2. Paramètre "address"

Ce document se réfère toujours aux adresses Modbus de registre (commençant à 0) et jamais au numéro de registre Modbus (commençant à 1).

20.2.1.3. Paramètre "format"

Format	Description	Coil	Register
raw	Les données seront représentées sous forme d'une chaîne binaire pour les discrete inputs et les coils et d'une chaîne hexadécimale pour les registers	•	•
boolean	Vrai ou faux	•	
integer	16 ou 32-bit entier non-signé		•
float	16 ou 32-bit à virgule flottante (IEEE 754)		•
ascii	Chaîne de caractères ASCII		•

20.2.1.4. Paramètre "flag"

Format	Description
cmd_only	La variable ne sera pas lue à partir du module Modbus, mais peut être écrite
little_endian	Interprète les deux registres de 16 bits d'une valeur de 32 bits en little-endian
no_opt	Une requête Modbus dédié sera utilisée pour lire cette variable.
is_status	En mode polling, tout changement de cette variable va déclencher une lecture de l'ensemble de données.
is_alarm	Comme is_status mais aussi déclenche une demande de connexion.

20.2.1.5. Paramètre “alarm”

Pour les variables flottantes et entières, deux seuils peuvent être définis (alarm/low et alarm/high) avec une valeur d’hystérésis. Chaque fois que la variable est mise à jour, sa valeur est comparée à ce niveau afin de déterminer un statut associé (low, normal, high), comme indiqué ci-dessous:

Lorsqu’au moins un niveau de seuil est défini, les flags is_status et is_alarm s’appliquent à l’état qui en résulte. Par exemple, si is_alarm est définie à true, la demande de connexion sera déclenchée seulement lorsque l’état de la variable change et pas à chaque fois que sa valeur change.

Données supplémentaires dans le mode polling :

En mode polling, des données supplémentaires seront maintenues pour les valeurs entières et flottante : Les valeurs min / max / moyennes et le nombre d’échantillons depuis le dernier enregistrement de données.

20.2.2. Boundaries

Non implémenté en V2.x.

20.3. Esclaves Modbus

Un module est une instance d’un ensemble de données pour une adresse Modbus donnée. La configuration d’un module Modbus (/config/modbus/modules/module) comprend les paramètres suivants:

Nom	Description
label	Nom uniquement informatif
dataset	Identifiant de l’ensemble de données
address	Adresse Modbus (1-247)
ip	Adresse IP (vide pour les équipements RTU)
schedule	Identifiant du schedule

Remarque : Les équipements Modbus/TCP doivent être configurés pour écouter le port Modbus TCP par défaut (502).

20.4. Adresse des variables

La commande Modbus utilise des adresses formatées comme expliqué ci-dessous :

Modbus/RTU

<modbus_address>/<register_type>@<register_address>

Exemple :

Registre d’entrée à l’adresse 0x0056 sur l’équipement Modbus 45.
=> 45/S3@0x0056

Modbus/TCP

<device_ip>:<modbus_address>/<register_type>@<register_address>

Exemple:

Registre d’entrée à l’adresse 0x0F0C sur l’équipement Modbus 223 on
à l’adresse IP 192.168.0.17.
=> 192.168.0.17:223/S3@0x0F56

Remarque : La modbus_address et le register_address peuvent être soit sous forme décimale ou hexadécimale. Ce dernier doit être précédé de «0x».

21. SCHEDULER

Le scheduler est en charge de toutes les tâches périodiques.

La configuration du scheduler consiste en une liste de schedules.

Chacun de ces schedules possède un identifiant (entier positive) unique qui est utilisé pour lier une tâche à un schedule spécifique.

Les schedules peuvent être utilisés indépendamment pour déclencher la collecte de données et télécharger des données recueillies.

Name	Description
/scheduler/schedules/schedule	La configuration de chaque schedule (voir ci-dessous) sera enregistrée sous cet élément.

Chaque schedule est configuré comme suit :

Nom	Description
schedule/id	Identifiant unique de schedule (Entier)
schedule/label	Nom uniquement informatif du schedule
schedule/type	Daily, Weekly, Monthly, Yearly ou Follower : voir description ci-dessous
schedule/parent	Référence au schedule parent pour un schedule de type Follower.
schedule/start/time	Heure de la première occurrence (non utilisé pour les schedules de type Yearly)
schedule/start/datetime	Date et heure de la première occurrence dans une période donnée (utilisé uniquement pour les schedules de type Yearly).
schedule/start/dayofweek	Numéro du jour dans la semaine de la première occurrence (1=Lundi, 7=Dimanche) (utilisé uniquement pour les schedules de type Weekly).
schedule/start/dayofmonth	Numéro du jour dans le mois de la première occurrence (utilisé uniquement pour les schedules de type Monthly).
schedule/interval	Interval entre les occurrences (en secondes)
schedule/count	Nombre d'occurrences

Configuration des différents types de schedules :

Schedule de type Daily :

Chaque jour, la première occurrence T0 est donnée par l'heure renseignée dans time.

Le format de la variable time est le suivant : HH:MM:SS

Par exemple 09:30:00

Les occurrences suivantes interviendront à l'heure Ti :

Δt correspond à la valeur en secondes renseignée dans interval.

Schedule de type Weekly :

Chaque semaine, la première occurrence T0 est donnée par le jour de la semaine renseignée dans dayofweek et l'heure renseignée dans time.

Le format de la variable time est le suivant : HH:MM:SS

Par exemple 09:30:00

La variable dayofweek est un entier entre 1 et 7 (1=Lundi et 7=Dimanche)

Les occurrences suivantes interviendront à l'heure Ti :

Δt correspond à la valeur en secondes renseignée dans interval.

Schedule de type Monthly :

Chaque mois, la première occurrence T0 est donnée par le numéro de jour du mois renseigné dans dayofmonth et l'heure renseignée dans time.

Le format de la variable time est le suivant : HH:MM:SS

Par exemple 09:30:00

Les occurrences suivantes interviendront à l'heure Ti :

Δt correspond à la valeur en secondes renseignée dans interval.

Schedule de type Yearly :

Chaque année, la première occurrence T0 est donnée par la date renseignée dans datetime.

Le format de la variable datetime est le suivant : AAAA-MM-JJTHH:MM:SS

Par exemple, pour une première occurrence le 11 février 2012 à 13H00 :

datetime = 2012-02-11T13:00:00.

Les occurrences suivantes interviendront à l'heure Ti :

Δt correspond à la valeur en secondes renseignée dans interval.

Schedule de type Follower :

Un schedule de type «Follower» sera produit après la fin de chaque occurrence du schedule de référence. Le schedule parent ne peut pas être de type «Follower».

Ce type permet de déclencher par exemple un téléchargement des données après l'achèvement d'une collecte de données prévue.

Exemple :

Vous souhaitez collecter les données de tous les modules une fois par jour à minuit et télécharger les données, juste après. Vous pouvez configurer un schedule de type « Daily » pour la collecte de données et un schedule de type « Follower » du premier schedule pour le téléchargement des données.

```
<schedules>
  <schedule>
    <id>1</id>
    <label>Data collect</label>
    <type>day</type>
    <start>
      <time>00:00:00</time>
    </start>
  </schedule>
  <schedule>
    <id>2</id>
    <label>Data upload</label>
    <type>follow</type>
    <parent>1</parent>
  </schedule>
</schedules>
```

Exemples :

Besoin	type	time	dayofweek	dayofmonth	datetime	interval	count
Tous les mardi à 15:00:00	week	15:00:00	2			0	1
Tous les 2ème jour du mois à 00:00:00	month	00:00:00		2		0	1
Tous les jours à 14:00:00	day	14:00:00				0	1
Toutes les heures entre 8H00 et 18H00 tous les mardis	week	08:00:00	2			3600	11
Toutes les 2 heures entre 8H00 et 20H00 le 31 décembre	year				2012-12-31T08:00:00	7200	7

21.1. Moteur d'alarme

Le moteur d'alarme génère des alarmes basées sur des événements internes.

Chaque source d'alarme peut être activée individuellement et sera téléchargée immédiatement au serveur distant (on) ou à la connexion suivante (delayed).

Nom	Valeurs	Description
sources/power	on, off, delayed	Changement d'état de l'alimentation principale
sources/modem_ip	on, off, delayed	Changement d'adresse IP
sources/msisdn	on, off, delayed	Changement MSISDN
sources/sw_version	on, off, delayed	Changement de la version logicielle
sources/defaults/ignored	<empty>	Liste des codes défauts ignorés par la passerelle (séparés par des virgules)
sources/defaults/delayed	<empty>	Liste des codes défauts transférés à la connexion suivante (séparés par des virgules)
sources/d_inputs/*		Changement d'état des entrées TOR (voir ci-dessous).
sources/d_output		Changement d'état des sorties TOR (voir ci-dessous).

La passerelle génère également des alarmes de type défaut, dont les codes sont reportés ci-dessous :

Code	Description
D_MODEM	Défaut du modem
D_MODEM_PUK	Carte SIM bloquée
D_ETHERNET	Défaut de l'interface Ethernet
D_WAVENIS	Défaut radio Wavenis
D_RFID	Défaut du récepteur RFID
D_INTERNAL_BAT	Défaut batterie interne

Lorsque l'alarme «power» est activée, le moteur d'alarme enverra une alarme sur la perte et la récupération de son alimentation.

Lorsque l'alarme «modem_ip» est activée, le moteur d'alarme enverra une alarme contenant l'adresse IP de la passerelle chaque fois qu'elle change.

Lorsque l'alarme «msisdn» est activée, le moteur d'alarme enverra une alarme sur changement de la carte SIM.

Lorsque l'alarme «sw_version» est activée, le moteur d'alarme enverra une alarme contenant la version logicielle de la passerelle, après une mise à jour.

Des alarmes multiples peuvent être configurées pour les entrées numériques.

Une alarme pour une entrée numérique peut être configurée comme suit:

Nom	Valeur	Description
d_input/index		Index de l'entrée numérique
d_input/label		Nom de l'alarme (uniquement informatif)
d_input/mode	on, off, delayed	On : Envoi immédiat
Off : Envoi désactivé		
Delayed : Envoi à la prochaine connexion		
d_input/type	none, raising, falling, both	None : Détection désactivée
Raising : Détection sur front montant		
Falling : Détection sur front descendant		
Both : Raising+Falling		

Une alarme peut être configurée pour la sortie numérique comme suit :

Nom	Valeur	Description
d_output/label		Nom de l'alarme (uniquement informatif)
d_output/mode	on, off, delayed	On : Envoi immédiat
Off : Envoi désactivé		
Delayed : Envoi à la prochaine connexion		
d_output/type	none, raising, falling, both	None : Détection désactivée
Raising : Détection sur front montant		
Falling : Détection sur front descendant		
Both : Raising+Falling		

Le front montant correspond à la fermeture du contact sec de la passerelle.

L'alarme est téléchargée vers le serveur distant en XML comme spécifié par le schéma d'alarme XML (voir paragraphe 16).

22. FICHIERS DE LOG

Le datalogger enregistre les principaux événements dans des fichiers de log. La taille du fichier de log est limitée à 200Ko.

Sur demande, les fichiers de log précédents et actuels sont concaténés et téléchargés vers le serveur distant en tant que données de supervision.

Le fichier de log est un fichier texte avec une entrée log par ligne.

Chaque ligne est formatée comme suit: [TIMESTAMP][LEVEL][SOURCE] EVENT

L'horodatage est un timestamp Unix (temps écoulé depuis l'EPOC) en secondes suivi par un point et les microsecondes.

Les événements suivants sont enregistrés:

Evènements	Formats
Schedules	Schedule X occurred
Défauts	Default X detected
	Default X cleared
PPP	PPP connecting
	PPP connected IP=X
	PPP connection failure
	PPP disconnect
FTP	FTP connecting
	FTP connected
	FTP connection failure
	FTP get X
	FTP put X
	FTP disconnect
SMS	SMS received from X
Commandes	Processing command X
Horloge	Clock synchronization delta=X
	NTP connection failure
Internes	Reboot
	Reboot modem

La liste n'est pas exhaustive et chaque ligne peut contenir des informations supplémentaires après celle spécifiée ci-dessus. Par exemple, lorsque survient un schedule, le fichier de log peut contenir «Schedule X occurred, next is Y at Z».

Le niveau de détail des logs peut être configuré par le paramètre /system/log/level. La valeur doit être comprise entre 8 (pas de log) à 1 (plus détaillé):

Level	Name	Description
8	None	Rien n'est loggé
7	Critical	Info critiques seulement
6	Error	
5	App	Niveau par défaut
4	Warning	
3	Notice	
2	Info	
1	Debug	Logs le plus détaillé

Le paramètre de niveau de log peut également être défini par source. Le format est alors : default_level,source:level, source:level,...

23. SYNCHRONISATION DE L'HORLOGE INTERNE

La passerelle synchronise l'horloge système en utilisant le protocole NTP.

Elle conserve l'heure en UTC (Temps Universel, aussi connu comme GMT) et calcule un temps local basé sur le fuseau horaire configuré. Elle gère l'heure d'été (DST).

Lorsque la connexion utilise le modem, la synchronisation se fait au début de chaque connexion au serveur, mais pas plus d'une fois par jour.

Une alarme est déclenchée lorsque la différence entre les deux horloges est plus grande qu'une valeur configurable.

Lorsque la connexion utilise Ethernet, un client NTP est activé sur la passerelle. Ce client va régler la vitesse de l'horloge système, afin de la maintenir synchronisée avec l'horloge du serveur NTP.

24. MISE À JOUR DU FIRMWARE DE LA PASSERELLE

Une mise à jour peut s'effectuer à distance. Le nouveau firmware doit être mis à disposition sur le serveur FTP dans le répertoire BIN dédié (voir paragraphe 3.2). Une commande mise à jour doit ensuite être envoyée à la passerelle.

Exemple : Commande de mise à jour du firmware du datalogger avec le firmware wrf_wavenis_v101.bin présent dans le répertoire BIN.

XML:

```
<cmd cid='C_1238'>  
  <update>  
    <firmware>wrf_wavenis_v101.bin</firmware>  
    <checksum>c1fb7d81f3d53a8b7bf94098115249d3</checksum>  
  </update>  
</cmd>
```

SMS:

cmd=update

cid=C_1237

firmware=wrf_wavenis_v101.bin

checksum=c1fb7d81f3d53a8b7bf94098115249d3

Le checksum correspond au checksum md5 du fichier.

25. SUPPORT

En cas de problèmes techniques relatifs à nos produits, contactez le support de SOCOMEC

Mail : support@Socomec.com - <http://www.Socomec.com>

Nous aurons besoin des éléments suivants :

- Numéro de série du datalogger.
- Version de matériel et de logiciel du datalogger.

ANNEXE I. SCHÉMA XSD – CONFIGURATION



config.xsd

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:wd="http://www.SOCOMECCOM/WRF_config_20110606"
  targetNamespace="http://www.SOCOMECCOM/WRF_config_20110606"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xsd:element name="config">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="uid" type="xsd:string" minOccurs="0"/>
        <xsd:element name="name" type="xsd:string" minOccurs="0"/>
        <xsd:element name="enable_local_config" type="xsd:boolean" minOccurs="0"/>
        <xsd:element name="com" type="wd:com" minOccurs="0"/>
        <xsd:element name="upload" type="wd:upload" minOccurs="0"/>
        <xsd:element name="alarm" type="wd:alarm" minOccurs="0"/>
        <xsd:element name="scheduler" type="wd:scheduler" minOccurs="0"/>
        <xsd:element name="wavenis" type="wd:wavenis" minOccurs="0"/>
        <xsd:element name="metering" type="wd:metering" minOccurs="0"/>
        <xsd:element name="rfid" type="wd:rfid" minOccurs="0"/>
        <xsd:element name="modbus" type="wd:modbus" minOccurs="0"/>
        <xsd:element name="system" type="wd:system" minOccurs="0"/>
      </xsd:sequence>
      <!-- @partial deprecated by @factory -->
      <xsd:attribute name="partial" type="xsd:boolean" default="true" use="optional"/>
      <xsd:attribute name="factory" type="xsd:boolean" default="false" use="optional"/>
    </xsd:complexType>
    <xsd:key name="schedule_id">
      <xsd:selector xpath="."/scheduler/schedules/schedule/>
      <xsd:field xpath="id"/>
    </xsd:key>
    <xsd:keyref name="upload_schedule_id_ref" refer="wd:schedule_id">
      <xsd:selector xpath="."/upload/data/>
      <xsd:field xpath="schedule"/>
    </xsd:keyref>
    <xsd:keyref name="wavenis_schedule_id_ref" refer="wd:schedule_id">
      <xsd:selector xpath="."/wavenis/modules/module/>
      <xsd:field xpath="schedule"/>
    </xsd:keyref>
    <xsd:keyref name="follow_schedule_id_ref" refer="wd:schedule_id">
      <xsd:selector xpath="."/scheduler/schedules/schedule/>
      <xsd:field xpath="parent"/>
    </xsd:keyref>
    <xsd:keyref name="pulse_schedule_id_ref" refer="wd:schedule_id">
      <xsd:selector xpath="."/metering/pulse/>
      <xsd:field xpath="schedule"/>
    </xsd:keyref>
    <xsd:keyref name="mbus_schedule_id_ref" refer="wd:schedule_id">
      <xsd:selector xpath="."/metering/mbus/>
      <xsd:field xpath="schedule"/>
    </xsd:keyref>
  </xsd:element>

  <xsd:complexType name="com">
    <xsd:sequence>
      <xsd:element name="modem" type="wd:modem" minOccurs="0"/>
      <xsd:element name="ethernet" type="wd:ethernet" minOccurs="0"/>
      <xsd:element name="keepalive" type="wd:keepalive" minOccurs="0"/>
      <xsd:element name="request" type="wd:request" minOccurs="0"/>
      <xsd:element name="time" type="wd:time" minOccurs="0"/>
      <xsd:element name="ftp" type="wd:ftp" minOccurs="0"/>
      <xsd:element name="ws" type="wd:ws" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="modem">
    <xsd:sequence>
      <xsd:element name="pin" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="mode">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="off"/>
                  <xsd:enumeration value="manual"/>
                  <xsd:enumeration value="automatic"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```



```

        </xsd:element>
        <xsd:element name=>code< type=>wd:pin_code< minOccurs=>0</>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name=>call_number< type=>xsd:string< minOccurs=>0</>
<xsd:element name=>apn< type=>xsd:string< minOccurs=>0</>
<xsd:element name=>login< type=>xsd:string< minOccurs=>0</>
<xsd:element name=>password< type=>xsd:string< minOccurs=>0</>
<xsd:element name=>mode< minOccurs=>0<
    <xsd:simpleType>
        <xsd:restriction base=>xsd:string>
            <xsd:enumeration value=>ondemand</>
            <xsd:enumeration value=>alwayson</>
            <xsd:enumeration value=>alwaysoff</>
            <xsd:enumeration value=>off</>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name=>delay< type=>xsd:integer< minOccurs=>0</>
<xsd:element name=>whitelist< minOccurs=>0>
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name=>caller_id<
                minOccurs=>0 maxOccurs=>unbounded< type=>xsd:string</>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name=>'ethernet'>
    <xsd:sequence>
        <xsd:element name=>use_dhcp< type=>xsd:boolean< minOccurs=>0</>
        <xsd:element name=>ip< type=>wd:ipv4< minOccurs=>0</>
        <xsd:element name=>netmask< type=>wd:ipv4< minOccurs=>0</>
        <xsd:element name=>gateway< minOccurs=>0>
            <xsd:simpleType>
                <xsd:union memberTypes=>wd:ipv4 wd:empty</>
            </xsd:simpleType>
        </xsd:element>
        <xsd:element name=>dns< minOccurs=>0>
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name=>server<
                        minOccurs=>0 maxOccurs=>unbounded< type=>wd:ipv4</>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:complexType>
<xsd:complexType name=>'keepalive'>
    <xsd:sequence>
        <xsd:element name=>method< minOccurs=>0>
            <xsd:simpleType>
                <xsd:restriction base=>xsd:string>
                    <xsd:enumeration value=>icmp</>
                    <xsd:enumeration value=>tcp</>
                    <xsd:enumeration value=>off</>
                </xsd:restriction>
            </xsd:simpleType>
        </xsd:element>
        <xsd:element name=>address< type=>xsd:string< minOccurs=>0</>
        <xsd:element name=>port< type=>wd:port< minOccurs=>0</>
        <xsd:element name=>period< type=>xsd:positiveInteger< minOccurs=>0</>
        <xsd:element name=>timeout< type=>xsd:positiveInteger< minOccurs=>0</>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name=>'request'>
    <xsd:sequence>
        <xsd:element name=>upload< type=>xsd:boolean< minOccurs=>0</>
        <xsd:element name=>include_status< type=>xsd:boolean< minOccurs=>0</>
        <xsd:element name=>sms_status_recipient< type=>xsd:string< minOccurs=>0</>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name=>'time'>
    <xsd:sequence>
        <xsd:element name=>ntp< minOccurs=>0>
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name=>server<
                        minOccurs=>0 maxOccurs=>unbounded< type=>xsd:string</>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name=>timezone< type=>xsd:string< minOccurs=>0</>
        </xsd:sequence>
    </xsd:complexType>

```

```

    <xsd:element name="alarm_threshold" type="xsd:integer" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='ftp'>
  <xsd:sequence>
    <xsd:element name="address" type="xsd:string" minOccurs="0"/>
    <xsd:element name="login" type="xsd:string" minOccurs="0"/>
    <xsd:element name="password" type="xsd:string" minOccurs="0"/>
    <xsd:element name="mode" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="passive"/>
          <xsd:enumeration value="active"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="secured" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="root_path" type="xsd:string" minOccurs="0"/>
    <xsd:element name="ws_notification" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="none"/>
          <xsd:enumeration value="put"/>
          <xsd:enumeration value="get"/>
          <xsd:enumeration value="both"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='ws'>
  <xsd:sequence>
    <xsd:element name="address" type="xsd:string" minOccurs="0"/>
    <xsd:element name="login" type="xsd:string" minOccurs="0"/>
    <xsd:element name="password" type="xsd:string" minOccurs="0"/>
    <xsd:element name="secured" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='upload'>
  <xsd:sequence>
    <xsd:element name="config" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="method" type="wd:upload_method_none" minOccurs="0"/>
          <xsd:element name="omit_password" type="xsd:boolean" minOccurs="0"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="supervision" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="method" type="wd:upload_method" minOccurs="0"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="alarm" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="method" type="wd:upload_method" minOccurs="0"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="data" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="method" type="wd:upload_method" minOccurs="0"/>
          <xsd:element name="format" type="wd:upload_format" minOccurs="0"/>
          <xsd:element name="schedule" minOccurs="0">
            <xsd:simpleType>
              <xsd:union memberTypes="xsd:positiveInteger wd:empty"/>
            </xsd:simpleType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="common" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="size_limit" type="xsd:positiveInteger" minOccurs="0"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="upload_method">

```

```

<xsd:restriction base=»xsd:string»>
  <xsd:enumeration value=»ftp»/>
  <xsd:enumeration value=»ws»/>
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name=»upload_method_none»>
  <xsd:restriction base=»xsd:string»>
    <xsd:enumeration value=»none»/>
    <xsd:enumeration value=»ftp»/>
    <xsd:enumeration value=»ws»/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name=»upload_format»>
  <xsd:restriction base=»xsd:string»>
    <xsd:enumeration value=»xml»/>
    <xsd:enumeration value=»csv»/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name='alarm'>
  <xsd:sequence>
    <xsd:element name=»sources» minOccurs=»0»>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name=»power» type=»wd:alarm_mode»
            minOccurs=»0»/>
          <xsd:element name=»modem_ip» type=»wd:alarm_mode»
            minOccurs=»0»/>
          <xsd:element name=»msisdn» type=»wd:alarm_mode»
            minOccurs=»0»/>
          <xsd:element name=»sw_version» type=»wd:alarm_mode»
            minOccurs=»0»/>
          <xsd:element name=»defaults»>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name=»ignored» type=»xsd:string» minOccurs=»0»/>
                <xsd:element name=»delayed» type=»xsd:string» minOccurs=»0»/>
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
          <xsd:element name=»d_inputs» type=»wd:d_inputs»
            minOccurs=»0»/>
          <xsd:element name=»d_output» type=»wd:d_output»
            minOccurs=»0»/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name=»alarm_mode»>
  <xsd:restriction base=»xsd:string»>
    <xsd:enumeration value=»on»/>
    <xsd:enumeration value=»off»/>
    <xsd:enumeration value=»delayed»/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name=»io_alarm_type»>
  <xsd:restriction base=»xsd:string»>
    <xsd:enumeration value=»none»/>
    <xsd:enumeration value=»raising»/>
    <xsd:enumeration value=»falling»/>
    <xsd:enumeration value=»both»/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name='d_inputs'>
  <xsd:sequence>
    <xsd:element name=»d_input» minOccurs=»0» maxOccurs=»unbounded»>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name=»index» type=»xsd:integer»/>
          <xsd:element name=»label» type=»xsd:string» minOccurs=»0»/>
          <xsd:element name=»mode» type=»wd:alarm_mode» minOccurs=»0»/>
          <xsd:element name=»type» type=»wd:io_alarm_type» minOccurs=»0»/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='d_output'>
  <xsd:sequence>
    <xsd:element name=»label» type=»xsd:string» minOccurs=»0»/>
    <xsd:element name=»mode» type=»wd:alarm_mode» minOccurs=»0»/>
    <xsd:element name=»type» type=»wd:io_alarm_type» minOccurs=»0»/>
  </xsd:sequence>
</xsd:complexType>

```

```

</xsd:sequence>
</xsd:complexType>

<xsd:complexType name='scheduler'>
  <xsd:sequence>
    <xsd:element name='schedules' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='schedule' type='wd:schedule'
            minOccurs='0' maxOccurs='unbounded'/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='schedule'>
  <xsd:sequence>
    <xsd:element name='id' type='xsd:positiveInteger'/>
    <xsd:element name='label' type='xsd:string'/>
    <xsd:element name='type'>
      <xsd:simpleType>
        <xsd:restriction base='xsd:string'>
          <xsd:enumeration value='day'/>
          <xsd:enumeration value='week'/>
          <xsd:enumeration value='month'/>
          <xsd:enumeration value='year'/>
          <xsd:enumeration value='follow'/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name='parent' minOccurs='0'>
      <xsd:simpleType>
        <xsd:union memberTypes='xsd:positiveInteger wd:empty'/>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name='start' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='time' minOccurs='0'>
            <xsd:simpleType>
              <xsd:union memberTypes='xsd:time wd:empty'/>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name='datetime' minOccurs='0'>
            <xsd:simpleType>
              <xsd:union memberTypes='xsd:dateTime wd:empty'/>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name='dayofweek' minOccurs='0'>
            <xsd:simpleType>
              <xsd:union memberTypes='wd:schedule_dayofweek wd:empty'/>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name='dayofmonth' minOccurs='0'>
            <xsd:simpleType>
              <xsd:union memberTypes='wd:schedule_dayofmonth wd:empty'/>
            </xsd:simpleType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name='interval' minOccurs='0'>
      <xsd:simpleType>
        <xsd:union memberTypes='xsd:positiveInteger wd:empty'/>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name='count' minOccurs='0'>
      <xsd:simpleType>
        <xsd:union memberTypes='xsd:positiveInteger wd:empty'/>
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name='schedule_dayofweek'>
  <xsd:restriction base='xsd:integer'>
    <xsd:minInclusive value='1'/>
    <xsd:maxInclusive value='7'/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='schedule_dayofmonth'>
  <xsd:restriction base='xsd:integer'>
    <xsd:minInclusive value='1'/>
    <xsd:maxInclusive value='31'/>
  </xsd:restriction>
</xsd:simpleType>

```

```

<xsd:complexType name='wavenis'>
  <xsd:sequence>
    <xsd:element name='bridge' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='enabled' type='xsd:boolean' minOccurs='0'>
            <xsd:element name='port' type='wd:port' minOccurs='0'>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    <xsd:element name='time' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='mode' minOccurs='0'>
            <xsd:simpleType>
              <xsd:restriction base='xsd:string'>
                <xsd:enumeration value='utc'>
                <xsd:enumeration value='local'>
                <xsd:enumeration value='nodst'>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name='alarm' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='mode' minOccurs='0'>
            <xsd:simpleType>
              <xsd:restriction base='xsd:string'>
                <xsd:enumeration value='basic'>
                <xsd:enumeration value='extended'>
              </xsd:restriction>
            </xsd:simpleType>
          <xsd:element name='sources' minOccurs='0'>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name='unknown'
                  minOccurs='0' type='wd:alarm_mode'>
                <xsd:element name='route'
                  minOccurs='0' type='wd:alarm_mode'>
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name='modules' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='module' type='wd:wavenis_module' minOccurs='0'
            maxOccurs='unbounded'>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='wavenis_module'>
  <xsd:sequence>
    <xsd:element name='address' type='wd:wavenis_address'>
    <xsd:element name='label' type='xsd:string'>
    <xsd:element name='type'>
      <xsd:simpleType>
        <xsd:restriction base='xsd:string'>
          <xsd:enumeration value='waveflow'>
          <xsd:enumeration value='wavetherm_dallas'>
          <xsd:enumeration value='wavetherm_pt100'>
          <xsd:enumeration value='wavetherm_pt1000'>
          <xsd:enumeration value='wavesense_4-20ma'>
          <xsd:enumeration value='wavesense_0-5v'>
          <xsd:enumeration value='wavelog'>
          <xsd:enumeration value='wavetic'>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name='repeaters' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='repeater'
            minOccurs='0' maxOccurs='3'
            type='wd:wavenis_address'>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name='mode'>
      <xsd:simpleType>

```

```

        <xsd:restriction base=»xsd:string»>
          <xsd:enumeration value=»immediate»/>
          <xsd:enumeration value=»datalog»/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  <xsd:element name=»nbinput»>
    <xsd:simpleType>
      <xsd:restriction base=»xsd:integer»>
        <xsd:minInclusive value=»1»/>
        <xsd:maxInclusive value=»4»/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name=»schedule» type=»xsd:positiveInteger»/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name='metering'>
  <xsd:sequence>
    <xsd:element name=»pulse» type=»wd:pulse_config» minOccurs=»0»/>
    <xsd:element name=»mbus» type=»wd:mbus_config» minOccurs=»0»/>
    <xsd:element name=»wmbus» type=»wd:wmbus_config» minOccurs=»0»/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='pulse_config'>
  <xsd:sequence>
    <xsd:element name=»schedule» minOccurs=»0»>
      <xsd:simpleType>
        <xsd:union memberTypes=»xsd:positiveInteger wd:empty»/>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name=»input_1» type=»wd:pulse_input» minOccurs=»0»/>
    <xsd:element name=»input_2» type=»wd:pulse_input» minOccurs=»0»/>
    <xsd:element name=»input_3» type=»wd:pulse_input» minOccurs=»0»/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='pulse_input'>
  <xsd:sequence>
    <xsd:element name=»label» type=»xsd:string»/>
    <xsd:element name=»unit» type=»xsd:string»/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='mbus_config'>
  <xsd:sequence>
    <xsd:element name=»schedule»>
      <xsd:simpleType>
        <xsd:union memberTypes=»xsd:positiveInteger wd:empty»/>
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='wmbus_config'>
  <xsd:sequence>
    <xsd:element name=»mode» minOccurs=»0»>
      <xsd:simpleType>
        <xsd:restriction base=»xsd:string»>
          <xsd:enumeration value=»T1»/>
          <xsd:enumeration value=»S1»/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name=»long_preamble» type=»xsd:boolean» minOccurs=»0»/>
    <xsd:element name=»modules» minOccurs=»0»>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name=»module» type=»wd:wmbus_module» minOccurs=»0»
            maxOccurs=»unbounded» />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='wmbus_module'>
  <xsd:sequence>
    <xsd:element name=»address» type=»xsd:string»/>
    <xsd:element name=»label» type=»xsd:string»/>
    <xsd:element name=»key» type=»xsd:string»/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='system'>
  <xsd:sequence>
    <xsd:element name=»log» minOccurs=»0»>

```

```

<xsd:complexType>
  <xsd:sequence>
    <xsd:element name="level" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="password" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="admin" type="wd:password" minOccurs="0"/>
      <xsd:element name="install" type="wd:password" minOccurs="0"/>
      <xsd:element name="data" type="wd:password" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ports" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="rs232" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="mode">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="off"/>
                  <xsd:enumeration value="mbus"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="rs485" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="mode" minOccurs="0">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="off"/>
                  <xsd:enumeration value="modbus"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
            <xsd:element name="baudrate" type="wd:uart_baudrate" minOccurs="0"/>
            <xsd:element name="data" type="wd:uart_data_bits" minOccurs="0"/>
            <xsd:element name="parity" type="wd:uart_parity" minOccurs="0"/>
            <xsd:element name="stop_bit" type="wd:uart_stop_bits" minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="input_1" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="mode">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="d_input"/>
                  <xsd:enumeration value="pulse"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="input_2" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="mode">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="d_input"/>
                  <xsd:enumeration value="pulse"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="input_3" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="mode">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="d_input"/>
                  <xsd:enumeration value="pulse"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

```

        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name='rfid'>
  <xsd:sequence>
    <xsd:element name='rssi_threshold' type='xsd:integer' minOccurs='0' />
    <xsd:element name='crc' type='xsd:integer' minOccurs='0' />
    <xsd:element name='detection_delay' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='entering' type='xsd:integer' minOccurs='0' />
          <xsd:element name='leaving' type='xsd:integer' minOccurs='0' />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name='alarm' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='sources' minOccurs='0'>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name='entering' type='wd:alarm_mode' minOccurs='0' />
                <xsd:element name='leaving' type='wd:alarm_mode' minOccurs='0' />
                <xsd:element name='id_flags' type='wd:alarm_mode' minOccurs='0' />
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name='modbus'>
  <xsd:sequence>
    <xsd:element name='tcp' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='timeout' type='xsd:integer' minOccurs='0' />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name='rtu' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='timeout' type='xsd:integer' minOccurs='0' />
          <xsd:element name='turnaround' type='xsd:integer' minOccurs='0' />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name='datasets' type='wd:modbus_datasets' minOccurs='0' />
    <xsd:element name='modules' type='wd:modbus_modules' minOccurs='0' />
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='modbus_datasets'>
  <xsd:sequence>
    <xsd:element name='dataset' minOccurs='0' maxOccurs='unbounded'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='id' type='xsd:integer' />
          <xsd:element name='label' type='xsd:string' minOccurs='0' />
          <xsd:element name='vars' type='wd:modbus_vars' minOccurs='0' />
          <xsd:element name='boundaries' type='wd:modbus_boundaries' minOccurs='0' />
          <xsd:element name='polling' type='xsd:boolean' minOccurs='0' />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='modbus_vars'>
  <xsd:sequence>
    <xsd:element name='var' minOccurs='0' maxOccurs='unbounded'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='name' type='xsd:string' />
          <xsd:element name='type' type='wd:modbus_type' />
          <xsd:element name='address' type='wd:hex_string' />
          <xsd:element name='size' type='xsd:integer' minOccurs='0' />
          <xsd:element name='format' type='wd:modbus_format' />
          <xsd:element name='flags' type='wd:modbus_flags' minOccurs='0' />
          <xsd:element name='threshold' minOccurs='0' />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

```



```

<xsd:complexType>
  <xsd:sequence>
    <xsd:element name='low'>
      <xsd:simpleType>
        <xsd:union memberTypes='xsd:string wd:empty'>
        </xsd:simpleType>
      </xsd:element>
    <xsd:element name='high'>
      <xsd:simpleType>
        <xsd:union memberTypes='xsd:string wd:empty'>
        </xsd:simpleType>
      </xsd:element>
    <xsd:element name='hysteresis'>
      <xsd:simpleType>
        <xsd:union memberTypes='xsd:string wd:empty'>
        </xsd:simpleType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

```

```

<xsd:complexType name='modbus_boundaries'>
  <xsd:sequence>
    <xsd:element name='boundary' minOccurs='0' maxOccurs='unbounded'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='type' type='wd:modbus_type'>
          <xsd:element name='address' type='wd:hex_string'>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

```

```

<xsd:complexType name='modbus_modules'>
  <xsd:sequence>
    <xsd:element name='module' minOccurs='0' maxOccurs='unbounded'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='label' type='xsd:string' minOccurs='0'>
          <xsd:element name='dataset' type='xsd:integer'>
          <xsd:element name='address' type='wd:modbus_addr' minOccurs='0'>
          <xsd:element name='ip' minOccurs='0'>
            <xsd:simpleType>
              <xsd:union memberTypes='wd:ipv4 wd:empty'>
              </xsd:simpleType>
          </xsd:element>
          <xsd:element name='schedule' type='xsd:integer' minOccurs='0'>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

```

```

<xsd:simpleType name='modbus_addr'>
  <xsd:restriction base='xsd:integer'>
    <xsd:minInclusive value='1'>
    <xsd:maxInclusive value='247'>
  </xsd:restriction>
</xsd:simpleType>

```

```

<xsd:simpleType name='modbus_type'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='S[0,1,3,4]'>
  </xsd:restriction>
</xsd:simpleType>

```

```

<xsd:simpleType name='modbus_format'>
  <xsd:restriction base='xsd:string'>
    <xsd:enumeration value='raw'>
    <xsd:enumeration value='boolean'>
    <xsd:enumeration value='integer'>
    <xsd:enumeration value='float'>
    <xsd:enumeration value='ascii'>
  </xsd:restriction>
</xsd:simpleType>

```

```

<xsd:simpleType name='modbus_flags'>
  <xsd:restriction base='xsd:string'>
<!--

```

```

  <xsd:enumeration value='cmd_only'>
  <xsd:enumeration value='little_endian'>

```

```

        <xsd:enumeration value=>no_opt</>
        <xsd:enumeration value=>is_status</>
        <xsd:enumeration value=>is_alarm</>
        <xsd:enumeration value=>signed</>
        <xsd:enumeration value=></>
-->
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='wavenis_address'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='([0-9a-zA-Z]{12})([0-9]{5}-?[0-9]{2}-?[0-9]{8})'/?>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='pin_code'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[0-9]{4,6}'/?>
  </xsd:restriction>
</xsd:simpleType>

<!--
password:
  ASCII characters from 0x21 to 0x7E
  Except: « (0x22), & (0x26), ' (0x27), < (0x3C), > (0x3E), ? (0x3F), ` (0x60)
-->
<xsd:simpleType name='password'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[&#x0021;-&#x007E;~&#x0022;&#x0026;&#x0027;&#x003C;&#x003E;&#x003F;&#x0060;][1,120]'/?>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name=>empty<>
  <xsd:restriction base=>xsd:string<>
    <xsd:length value=>0</>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='ipv4'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}'/?>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='port'>
  <xsd:restriction base=>xsd:integer<>
    <xsd:minInclusive value=>1</>
    <xsd:maxInclusive value=>65535</>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='uart_baudrate'>
  <xsd:restriction base='xsd:integer'>
    <xsd:pattern value='1200|2400|4800|9600|19200|38400|57600|115200'/?>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='uart_data_bits'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[5-9]'/?>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name=>uart_parity<>
  <xsd:restriction base=>xsd:string<>
    <xsd:enumeration value=>odd</>
    <xsd:enumeration value=>even</>
    <xsd:enumeration value=>none</>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='uart_stop_bits'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[1-2]'/?>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='hex_string'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[0-9a-zA-Z]*'/?>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

ANNEXE II. SCHÉMA XSD – CONFIGURATION



config.xml

```
<?xml version="1.0"?>
<config
  xmlns="http://www.SOCOMECCOM/WRF_config_20110606"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.SOCOMECCOM/WRF_config_20110606 config.xsd">
  <uid>07F38D</uid>
  <name>WGRF_07F38D</name>
  <enable_local_config>true</enable_local_config>
  <com>
    <modem>
      <pin>
        <mode>manual</mode>
        <code>1234</code>
      </pin>
      <call_number>*99***1#</call_number>
      <apn>m2minternet</apn>
      <login>login</login>
      <password>password</password>
      <mode>ondemand</mode>
      <whitelist>
        <caller_id>+33123242526</caller_id>
        <caller_id>+33123242527</caller_id>
      </whitelist>
    </modem>
    <ethernet>
      <use_dhcp>>false</use_dhcp>
      <ip>192.168.10.10</ip>
      <netmask>255.255.255.0</netmask>
      <gateway>192.168.10.254</gateway>
      <dns>
        <server>192.168.10.254</server>
        <server>8.8.8.8</server>
        <server>8.8.4.4</server>
      </dns>
    </ethernet>
    <keepalive>
      <method>tcp</method>
      <address>12.13.14.15</address>
      <port>8003</port>
      <period>1800</period>
      <timeout>10</timeout>
    </keepalive>
    <request>
      <upload>true</upload>
      <include_status>true</include_status>
      <sms_status_recipient/>
    </request>
    <time>
      <ntp>
        <server>pool.ntp.org</server>
      </ntp>
      <timezone>Europe/Paris</timezone>
      <alarm_threshold>300</alarm_threshold>
    </time>
    <ftp>
      <address>12.13.14.16</address>
      <login>login</login>
      <password>password</password>
      <mode>passive</mode>
      <secured>>false</secured>
      <root_path>/</root_path>
      <ws_notification>none</ws_notification>
    </ftp>
    <ws>
      <address>12.13.14.16</address>
      <login>login</login>
      <password>password</password>
      <secured>>false</secured>
    </ws>
  </com>
  <upload>
    <config>
      <method>ftp</method>
    </config>
    <supervision>
      <method>ftp</method>
    </supervision>
    <alarm>
      <method>ws</method>
    </alarm>
  </upload>
</config>
```

```

</alarm>
<data>
  <method>ftp</method>
  <format>csv</format>
  <schedule>1</schedule>
</data>
</upload>
<alarm>
  <sources>
    <defaults>
      <ignored/>
      <delayed/>
    </defaults>
    <d_inputs>
      <d_input>
        <index>0</index>
        <label>Intrusion detection</label>
        <type>both</type>
      </d_input>
    </d_inputs>
  </sources>
</alarm>
<scheduler>
  <schedules>
    <schedule>
      <id>1</id>
      <label>upload schedule</label>
      <type>week</type>
      <start>
        <time>01:00:00</time>
        <dayofweek>7</dayofweek>
      </start>
      <interval>1</interval>
      <count>1</count>
    </schedule>
    <schedule>
      <id>2</id>
      <label>collect schedule</label>
      <type>day</type>
      <start>
        <time>04:00:00</time>
      </start>
      <interval>43200</interval>
      <count>2</count>
    </schedule>
  </schedules>
</scheduler>
<wavenis>
  <bridge>
    <enabled>true</enabled>
    <port>4000</port>
  </bridge>
  <modules>
    <module>
      <address>011A1030A5D4</address>
      <label>main</label>
      <type>waveflow</type>
      <repeaters>
        <repeater>011A1030A7D3</repeater>
      </repeaters>
      <mode>datalog</mode>
      <nbinput>4</nbinput>
      <schedule>2</schedule>
    </module>
  </modules>
</wavenis>
<metering>
  <pulse>
    <schedule/>
    <input_1>
      <label/>
      <unit/>
    </input_1>
    <input_2>
      <label/>
      <unit/>
    </input_2>
    <input_3>
      <label/>
      <unit/>
    </input_3>
  </pulse>
  <mbus>
    <schedule/>
  </mbus>
  <wmbus>
    <mode>T1</mode>
    <long_preamble>false</long_preamble>
  </modules>

```

```

    <module>
      <address>012345678901</address>
      <label>test1</label>
      <key>00112233445566778899</key>
    </module>
  </modules>
</wmbus>
</metering>
<rfid>
  <rssj_threshold>255</rssj_threshold>
  <crc>0</crc>
  <detection_delay>
    <entering>600</entering>
    <leaving>500</leaving>
  </detection_delay>
  <alarm>
    <sources>
      <entering>on</entering>
      <leaving>on</leaving>
      <id_flags>on</id_flags>
    </sources>
  </alarm>
</rfid>
<modbus>
  <rtu>
    <timeout>500</timeout>
    <turnaround>20</turnaround>
  </rtu>
  <datasets>
    <dataset>
      <id>1</id>
      <label>dataset_1</label>
      <vars>
        <var>
          <name>var1</name>
          <type>S4</type>
          <address>0x1234</address>
          <size>2</size>
          <format>integer</format>
          <flags>is_alarm</flags>
        </var>
        <var>
          <name>var1</name>
          <type>S4</type>
          <address>0x1234</address>
          <size>2</size>
          <format>integer</format>
          <flags></flags>
        </var>
        <var>
          <name>var1</name>
          <type>S4</type>
          <address>0x1234</address>
          <size>2</size>
          <format>integer</format>
          <flags>is_alarm,is_status</flags>
        </var>
        <var>
          <name>var2</name>
          <type>S0</type>
          <address>1234</address>
          <size>1</size>
          <format>boolean</format>
          <flags>is_status</flags>
          <threshold>
            <low>20</low>
            <high>40</high>
            <hysteresis>2</hysteresis>
          </threshold>
        </var>
      </vars>
      <boundaries>
        <boundary>
          <type>S0</type>
          <address>1234</address>
        </boundary>
      </boundaries>
      <polling>true</polling>
    </dataset>
  </datasets>
</modules>
<module>
  <label>my_module</label>
  <dataset>1</dataset>
  <address>26</address>
  <schedule>1</schedule>
</module>
</modules>
</modbus>

```

</config>

ANNEXE III. SCHÉMA XSD – ALARMES



alarm.xsd

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:wd="http://www.SOCOMECCOM/WRF_alarm_20110606"
  targetNamespace="http://www.SOCOMECCOM/WRF_alarm_20110606"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xsd:element name="alarms">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="uid" type="xsd:string"/>
        <xsd:choice maxOccurs="unbounded">
          <xsd:element name="command" type="wd:cmd_alarm"/>
          <xsd:element name="ntp" type="wd:ntp_alarm"/>
          <xsd:element name="sw_version" type="wd:version_alarm"/>
          <xsd:element name="msisdn" type="wd:msisdn_alarm"/>
          <xsd:element name="power" type="wd:power_alarm"/>
          <xsd:element name="modem_ip" type="wd:ip_alarm"/>
          <xsd:element name="default" type="wd:default_alarm"/>
          <xsd:element name="d_input" type="wd:d_input_alarm"/>
          <xsd:element name="d_output" type="wd:d_output_alarm"/>
          <xsd:element name="wavenis_unknown" type="wd:wavenis_topo_alarm"/>
          <xsd:element name="wavenis_route" type="wd:wavenis_topo_alarm"/>
          <xsd:element name="wavenis_cmd" type="wd:wavenis_cmd_alarm"/>
          <xsd:element name="waveflow" type="wd:waveflow_alarm"/>
          <xsd:element name="wavetherm" type="wd:wavethermsense_alarm"/>
          <xsd:element name="wavesense" type="wd:wavethermsense_alarm"/>
          <xsd:element name="wavelog" type="wd:wavelog_alarm"/>
          <xsd:element name="wavetic" type="wd:wavetic_alarm"/>
          <xsd:element name="wavetalk" type="wd:wavetalk_alarm"/>
          <xsd:element name="modbus_cmd" type="wd:modbus_cmd_alarm"/>
          <xsd:element name="modbus" type="wd:modbus"/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="alarm" abstract="true">
    <xsd:sequence>
      <xsd:element name="date" type="xsd:dateTime"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="cmd_alarm_base" abstract="true">
    <xsd:complexContent>
      <xsd:extension base="wd:alarm">
        <xsd:sequence>
          <xsd:element name="cid" type="xsd:string" minOccurs="0"/>
          <xsd:element name="source">
            <xsd:simpleType>
              <xsd:restriction base="xsd:string">
                <xsd:enumeration value="sms"/>
                <xsd:enumeration value="ws"/>
                <xsd:enumeration value="ftp"/>
                <xsd:enumeration value="local"/>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:element>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name="cmd_alarm">
    <xsd:complexContent>
      <xsd:extension base="wd:cmd_alarm_base">
        <xsd:sequence>
          <xsd:element name="error">
            <xsd:simpleType>
              <xsd:restriction base="xsd:string">
                <xsd:enumeration value="invalid_uid"/>
                <xsd:enumeration value="malformed"/>
                <xsd:enumeration value="other"/>
                <xsd:enumeration value="none"/>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name="description" type="xsd:string" minOccurs="0"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:schema>
```

```

    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="ntp_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">
      <xsd:sequence>
        <xsd:element name="before" type="xsd:dateTime"/>
        <xsd:element name="after" type="xsd:dateTime"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="power_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">
      <xsd:sequence>
        <xsd:element name="state">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="on"/>
              <xsd:enumeration value="off"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="version_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">
      <xsd:sequence>
        <xsd:element name="version" type="xsd:string"/>
        <xsd:element name="kernel" type="xsd:string"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="msisdn_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">
      <xsd:sequence>
        <xsd:element name="msisdn" type="xsd:string"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="ip_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">
      <xsd:sequence>
        <xsd:element name="ip" type="wd:ipv4"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="default_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">
      <xsd:sequence>
        <xsd:element name="name" type="xsd:string"/>
        <xsd:element name="type" type="wd:basic_alarm_type"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="d_input_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">
      <xsd:sequence>
        <xsd:element name="index" type="xsd:integer"/>
        <xsd:element name="label" type="xsd:string"/>
        <xsd:element name="type" type="wd:basic_alarm_type"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="d_output_alarm">
  <xsd:complexContent>
    <xsd:extension base="wd:alarm">

```



```

        <xsd:sequence>
          <xsd:element name=>label<> type=>xsd:string</>
          <xsd:element name=>type<> type=>wd:basic_alarm_type</>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:simpleType name='basic_alarm_type'>
    <xsd:restriction base=>xsd:string<>
      <xsd:enumeration value=>set</>
      <xsd:enumeration value=>reset</>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name=>wavenis_topo_alarm<>
    <xsd:complexContent>
      <xsd:extension base=>wd:alarm<>
        <xsd:sequence>
          <xsd:element name=>address<> type=>wd:wavenis_address</>
          <xsd:element name=>repeaters<> minOccurs=>0<>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name=>repeater<>
                  minOccurs=>0' maxOccurs=>'3'
                  type=>wd:wavenis_address</>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>

  <xsd:complexType name=>wavenis_cmd_alarm<>
    <xsd:complexContent>
      <xsd:extension base=>wd:cmd_alarm_base<>
        <xsd:sequence>
          <xsd:element name=>subcmd<> type=>xsd:string</>
          <xsd:element name=>address<> type=>wd:wavenis_address</>
          <xsd:element name=>result<>
            <xsd:simpleType>
              <xsd:restriction base=>xsd:string<>
                <xsd:enumeration value=>ok</>
                <xsd:enumeration value=>error</>
                <xsd:enumeration value=>unsupported</>
                <xsd:enumeration value=>open</>
                <xsd:enumeration value=>close</>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name=>request<> type=>wd:hex_string<> minOccurs=>0</>
          <xsd:element name=>response<> type=>wd:hex_string<> minOccurs=>0</>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name='wavetic_alarm'>
    <xsd:complexContent>
      <xsd:extension base=>wd:alarm<>
        <xsd:sequence>
          <xsd:element name=>address<> type=>wd:wavenis_address</>
          <xsd:element name=>eol_battery<> type=>xsd:boolean</>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name='wavetalk_alarm'>
    <xsd:complexContent>
      <xsd:extension base=>wd:alarm<>
        <xsd:sequence>
          <xsd:element name=>address<> type=>wd:wavenis_address</>
          <xsd:element name=>life-counter<> type=>xsd:integer</>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name='wavethermsense_alarm'>
    <xsd:complexContent>
      <xsd:extension base=>wd:alarm<>
        <xsd:sequence>
          <xsd:element name=>address<> type=>wd:wavenis_address</>
          <xsd:element name=>input<> type=>wd:wavenis_input<> minOccurs=>0</>
          <xsd:element name=>threshold<>
            <xsd:complexType>
              <xsd:sequence>

```

```

        <xsd:element name=>high<> type=>xsd:boolean</>
        <xsd:element name=>low<> type=>xsd:boolean</>
        <xsd:element name=>duration<> type=>xsd:integer
            minOccurs=>0</>
        <xsd:element name=>value<> type=>xsd:string
            minOccurs=>0</>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name=>sensor_fault<> type=>xsd:boolean</>
<xsd:element name=>eol_battery<> type=>xsd:boolean</>
</xsd:sequence>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>

<xsd:complexType name='wavelog_alarm'>
    <xsd:complexContent>
        <xsd:extension base=>wd:alarm<>
            <xsd:sequence>
                <xsd:element name=>address<> type=>wd:wavenis_address</>
                <xsd:element name=>app_status<>
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name=>reset_flag<> type=>xsd:boolean</>
                            <xsd:element name=>input_conflict<> type=>xsd:boolean</>
                            <xsd:element name=>discontinuity<> type=>xsd:boolean</>
                            <xsd:element name=>eol_battery<> type=>xsd:boolean</>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name=>io_status<>
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name=>input1<> type=>xsd:boolean</>
                            <xsd:element name=>input2<> type=>xsd:boolean</>
                            <xsd:element name=>input3<> type=>xsd:boolean</>
                            <xsd:element name=>input4<> type=>xsd:boolean</>
                            <xsd:element name=>output1<> type=>xsd:boolean</>
                            <xsd:element name=>output2<> type=>xsd:boolean</>
                            <xsd:element name=>output3<> type=>xsd:boolean</>
                            <xsd:element name=>output4<> type=>xsd:boolean</>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name=>open<> type=>xsd:boolean</>
                <xsd:element name=>close<> type=>xsd:boolean</>
                <xsd:element name=>open_timeout<> type=>xsd:boolean</>
                <xsd:element name=>close_timeout<> type=>xsd:boolean</>
                <xsd:element name=>input1<> type=>xsd:boolean</>
                <xsd:element name=>input2<> type=>xsd:boolean</>
                <xsd:element name=>input3<> type=>xsd:boolean</>
                <xsd:element name=>input4<> type=>xsd:boolean</>
                <xsd:element name=>eol_battery<> type=>xsd:boolean</>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name=>waveflow_alarm<>
    <xsd:complexContent>
        <xsd:extension base=>wd:alarm<>
            <xsd:sequence>
                <xsd:element name=>address<> type=>wd:wavenis_address</>
                <xsd:element name=>type<>
                    <xsd:simpleType>
                        <xsd:restriction base=>xsd:string<>
                            <xsd:enumeration value=>basic</>
                            <xsd:enumeration value=>set</>
                            <xsd:enumeration value=>reset</>
                        </xsd:restriction>
                    </xsd:simpleType>
                </xsd:element>
                <xsd:choice>
                    <xsd:element name=>status<> type=>wd:waveflow_alarm_status</>
                    <xsd:element name=>leak<> type='wd:wavenis_alarm_leak'</>
                    <xsd:element name=>backflow<> type='wd:wavenis_alarm_backflow'</>
                    <xsd:element name=>wire_cut<> type='wd:wavenis_alarm_input'</>
                    <xsd:element name=>reed_fault<> type='wd:wavenis_alarm_input'</>
                    <xsd:element name=>eol_battery<> type='wd:wavenis_alarm_eol'</>
                </xsd:choice>
            </xsd:sequence>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name='waveflow_alarm_status'>
    <xsd:sequence>
        <xsd:element name=>input<> type=>wd:wavenis_input minOccurs=>0</>
    </xsd:sequence>

```

```

<xsd:element name='leak'>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name='high_threshold' type='xsd:boolean'/>
      <xsd:element name='low_threshold' type='xsd:boolean'/>
      <xsd:element name='flow' type='xsd:integer' minOccurs='0'/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name='backflow' type='xsd:boolean'/>
<xsd:element name='wire_cut' type='xsd:boolean'/>
<xsd:element name='reed_fault' type='xsd:boolean'/>
<xsd:element name='eol_battery' type='xsd:boolean'/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name='wavenis_alarm_leak'>
  <xsd:sequence>
    <xsd:element name='threshold'>
      <xsd:simpleType>
        <xsd:restriction base='xsd:string'>
          <xsd:enumeration value='high'/>
          <xsd:enumeration value='low'/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name='input' type='wd:wavenis_input'/>
    <xsd:element name='flow' type='xsd:integer' minOccurs='0'/>
    <xsd:element name='date' type='xsd:dateTime'/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='wavenis_alarm_backflow'>
  <xsd:sequence>
    <xsd:element name='input' type='wd:wavenis_input'/>
    <xsd:choice>
      <xsd:element name='qty_detection'>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name='quantity' type='xsd:integer'/>
            <xsd:element name='start' type='xsd:dateTime'/>
            <xsd:element name='end' type='xsd:dateTime' minOccurs='0'/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name='flow_detection'>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name='flow' type='xsd:integer'/>
            <xsd:element name='duration' type='xsd:integer'/>
            <xsd:element name='end' type='xsd:dateTime' minOccurs='0'/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='wavenis_alarm_input'>
  <xsd:sequence>
    <xsd:element name='input' type='wd:wavenis_input'/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='wavenis_alarm_eol'>
  <xsd:sequence>
    <xsd:element name='date' type='xsd:dateTime'/>
    <xsd:element name='life_counter' type='xsd:integer'/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name='wavenis_input'>
  <xsd:restriction base='xsd:string'>
    <xsd:enumeration value='A'/>
    <xsd:enumeration value='B'/>
    <xsd:enumeration value='C'/>
    <xsd:enumeration value='D'/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='hex_string'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[0-9a-zA-Z]*'/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='wavenis_address'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='([0-9a-zA-Z]{12})([0-9]{5}-?[0-9]{2}-?[0-9]{8})'/>
  </xsd:restriction>
</xsd:simpleType>

```

```

</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='ipv4'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}'/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name='modbus_cmd_alarm'>
  <xsd:complexContent>
    <xsd:extension base='wd:cmd_alarm_base'>
      <xsd:sequence>
        <xsd:element name='subcmd' type='xsd:string' />
        <xsd:element name='result'>
          <xsd:simpleType>
            <xsd:restriction base='xsd:string'>
              <xsd:enumeration value='ok' />
              <xsd:enumeration value='error' />
              <xsd:enumeration value='no_response' />
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name='description' type='xsd:string' minOccurs='0' />
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name='modbus'>
  <xsd:complexContent>
    <xsd:extension base='wd:alarm'>
      <xsd:sequence>
        <xsd:element name='module'>
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name='label' type='xsd:string' />
              <xsd:element name='address' type='wd:modbus_addr' />
              <xsd:element name='ip' type='wd:ipv4' minOccurs='0' />
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name='variable'>
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name='name' type='xsd:string' />
              <xsd:element name='address' type='wd:hex_string' />
              <xsd:element name='type' type='wd:modbus_type' />
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name='value' type='xsd:string' />
        <xsd:element name='status' type='xsd:string' minOccurs='0' />
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:simpleType name='modbus_addr'>
  <xsd:restriction base='xsd:integer'>
    <xsd:minInclusive value='1' />
    <xsd:maxInclusive value='247' />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='modbus_type'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='S[0,1,3,4]' />
  </xsd:restriction>
</xsd:simpleType>

</xsd:schema>

```

ANNEXE IV. EXEMPLE XML – ALARMES



alarm.xml

```
<?xml version="1.0"?>
<alarms
  xmlns="http://www.SOCOMEK.com/WRF_alarm_20110606"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.SOCOMEK.com/WRF_alarm_20110606 alarm.xsd">

  <uid>07F38D</uid>

  <ntp>
    <date>2011-05-26T07:59:10</date>
    <before>2011-05-26T07:52:00</before>
    <after>2011-05-26T07:59:00</after>
  </ntp>

  <d_input>
    <date>2011-05-26T10:00:00</date>
    <index>0</index>
    <label>input_0</label>
    <type>set</type>
  </d_input>

  <d_input>
    <date>2011-05-26T10:10:00</date>
    <index>0</index>
    <label>input_0</label>
    <type>reset</type>
  </d_input>

  <d_output>
    <date>2011-05-26T10:20:00</date>
    <label>output</label>
    <type>set</type>
  </d_output>

  <power>
    <date>2011-05-26T10:30:00</date>
    <state>off</state>
  </power>

  <modem_ip>
    <date>2011-05-26T10:30:00</date>
    <ip>90.84.146.195</ip>
  </modem_ip>

  <sw_version>
    <date>2011-05-26T10:40:00</date>
    <version>v1.01</version>
    <kernel>2.6.35.6</kernel>
  </sw_version>

  <default>
    <date>2011-05-26T10:50:00</date>
    <name>D_ETHERNET</name>
    <type>set</type>
  </default>

  <default>
    <date>2011-05-26T10:55:00</date>
    <name>D_ETHERNET</name>
    <type>reset</type>
  </default>

  <waveflow>
    <date>2011-05-26T11:00:00</date>
    <address>011A1030A5D4</address>
    <type>basic</type>
    <status>
      <input>A</input>
      <leak>
        <high_threshold>true</high_threshold>
        <low_threshold>>false</low_threshold>
        <flow>246</flow>
      </leak>
      <backflow>>false</backflow>
      <wire_cut>>false</wire_cut>
      <reed_fault>>false</reed_fault>
      <eol_battery>>false</eol_battery>
    </status>
  </waveflow>
```

```

<waveflow>
  <date>2011-05-27T13:00:00</date>
  <address>00278-03-03146635</address>
  <type>set</type>
  <leak>
    <threshold>high</threshold>
    <input>A</input>
    <flow>456</flow>
    <date>2011-05-27T12:58:00</date>
  </leak>
</waveflow>

<waveflow>
  <date>2011-05-27T13:10:00</date>
  <address>011A1030A5D4</address>
  <type>reset</type>
  <leak>
    <threshold>high</threshold>
    <input>A</input>
    <date>2011-05-27T13:09:00</date>
  </leak>
</waveflow>

<waveflow>
  <date>2011-05-27T14:00:00</date>
  <address>011A1030A5D4</address>
  <type>set</type>
  <backflow>
    <input>A</input>
    <qty_detection>
      <quantity>345</quantity>
      <start>2011-05-27T13:59:00</start>
    </qty_detection>
  </backflow>
</waveflow>

<waveflow>
  <date>2011-05-27T14:10:00</date>
  <address>011A1030A5D4</address>
  <type>reset</type>
  <backflow>
    <input>A</input>
    <qty_detection>
      <quantity>500</quantity>
      <start>2011-05-27T13:59:00</start>
      <end>2011-05-27T14:09:00</end>
    </qty_detection>
  </backflow>
</waveflow>

<waveflow>
  <date>2011-05-27T15:00:00</date>
  <address>011A1030A5D4</address>
  <type>set</type>
  <backflow>
    <input>A</input>
    <flow_detection>
      <flow>323</flow>
      <duration>3</duration>
    </flow_detection>
  </backflow>
</waveflow>

<waveflow>
  <date>2011-05-27T15:10:00</date>
  <address>011A1030A5D4</address>
  <type>reset</type>
  <backflow>
    <input>A</input>
    <flow_detection>
      <flow>356</flow>
      <duration>13</duration>
      <end>2011-05-27T15:10:00</end>
    </flow_detection>
  </backflow>
</waveflow>

<waveflow>
  <date>2011-05-27T16:00:00</date>
  <address>011A1030A5D4</address>
  <type>set</type>
  <wire_cut>
    <input>B</input>
  </wire_cut>
</waveflow>

<waveflow>
  <date>2011-05-27T16:10:00</date>

```

```

    <address>011A1030A5D4</address>
    <type>reset</type>
    <wire_cut>
      <input>B</input>
    </wire_cut>
  </waveflow>

<waveflow>
  <date>2011-05-27T17:00:00</date>
  <address>011A1030A5D4</address>
  <type>set</type>
  <reed_fault>
    <input>A</input>
  </reed_fault>
</waveflow>

<waveflow>
  <date>2011-05-27T17:10:00</date>
  <address>011A1030A5D4</address>
  <type>reset</type>
  <reed_fault>
    <input>A</input>
  </reed_fault>
</waveflow>

<waveflow>
  <date>2011-05-27T18:00:00</date>
  <address>011A1030A5D4</address>
  <type>set</type>
  <eol_battery>
    <date>2011-05-27T17:59:00</date>
    <life_counter>4000</life_counter>
  </eol_battery>
</waveflow>

<wavenis_cmd>
  <date>2011-05-27T19:00:00</date>
  <cid>C_1239</cid>
  <source>ftp</source>
  <subcmd>modulflow-open</subcmd>
  <address>011A0A30AAA0</address>
  <result>ok</result>
</wavenis_cmd>

<wavenis_cmd>
  <date>2011-05-27T20:00:00</date>
  <cid>C_1240</cid>
  <source>ftp</source>
  <subcmd>raw</subcmd>
  <address>011A0A30AAA0</address>
  <result>ok</result>
  <request>10012001</request>
  <response>9001200106</response>
</wavenis_cmd>

<modbus_cmd>
  <date>2011-05-27T21:00:00</date>
  <cid>C_1240</cid>
  <source>ftp</source>
  <subcmd>write</subcmd>
  <result>ok</result>
</modbus_cmd>

<modbus>
  <date>2011-05-27T22:00:00</date>
  <module>
    <label>module1</label>
    <address>1</address>
    <ip>192.168.1.2</ip>
  </module>
  <variable>
    <name>var1</name>
    <address>2</address>
    <type>S4</type>
  </variable>
  <value>5</value>
  <status>low</status>
</modbus>
</alarms>

```

ANNEXE V. SCHÉMA XSD – SUPERVISION



supervision.xsd

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:wd="http://www.SOCOMECCOM/WRF_supervision_20110606"
  targetNamespace="http://www.SOCOMECCOM/WRF_supervision_20110606"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xsd:element name="supervision">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="uid" type="xsd:string"/>
        <xsd:choice maxOccurs="unbounded">
          <xsd:element name="status" type="wd:status"/>
          <xsd:element name="scan" type="wd:scan"/>
          <xsd:element name="timesync" type="wd:timesync"/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="status">
    <xsd:sequence>
      <xsd:element name="app">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="version" type="xsd:string"/>
            <xsd:element name="kernel" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="system">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="power" type="xsd:boolean"/>
            <xsd:element name="defaults" type="xsd:string"/>
            <xsd:element name="uptime" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="com">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="modem">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="model" type="xsd:string"/>
                  <xsd:element name="firmware" type="xsd:string"/>
                  <xsd:element name="imei" type="xsd:string"/>
                  <xsd:element name="msisdn" type="xsd:string"/>
                  <xsd:element name="rssi" type="xsd:string"/>
                  <xsd:element name="csq" type="xsd:string"/>
                  <xsd:element name="ip" type="wd:ipv4_opt"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="ethernet">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="ip" type="wd:ipv4_opt"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="upload">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="last" type="wd:datetime_opt"/>
                  <xsd:element name="next" type="wd:datetime_follow"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="wavenis">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="address" type="wd:wavenis_address"/>
            <xsd:element name="last" type="wd:datetime_opt"/>
            <xsd:element name="modules">

```



```

        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name='count' type='xsd:integer' />
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name='metering'>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name='mbus'>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name='last' type='wd:datetime_opt' />
            <xsd:element name='last_scan' type='wd:datetime_opt' />
            <xsd:element name='modules'>
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name='count' type='xsd:integer' />
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name='rfid'>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name='count' type='xsd:integer' minOccurs='0' />
      <xsd:element name='tags' minOccurs='0'>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name='tag' type='wd:rfid_tag' minOccurs='0'
              maxOccurs='unbounded' />
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name='modbus'>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name='next' type='wd:empty' />
      <xsd:element name='modules'>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name='count' type='xsd:integer' />
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:attribute name='cid' type='xsd:string' use='optional' />
<xsd:attribute name='date' type='xsd:dateTime' />
</xsd:complexType>

<xsd:complexType name='rfid_tag'>
  <xsd:sequence>
    <xsd:element name='id' type='wd:hex_string' />
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name='scan'>
  <xsd:sequence>
    <xsd:element name='wavenis' minOccurs='0'>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name='modules'>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name='module' type='wd:wavenis_scan_data'
                  minOccurs='0' maxOccurs='unbounded' />
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:attribute name='cid' type='xsd:string' use='optional' />
<xsd:attribute name='date' type='xsd:dateTime' />

```

```

</xsd:complexType>
<xsd:complexType name="wavenis_scan_data">
  <xsd:sequence>
    <xsd:element name="date" type="xsd:dateTime"/>
    <xsd:element name="address" type="wd:wavenis_address"/>
    <xsd:element name="rssi" type="xsd:integer" minOccurs="0"/>
    <xsd:element name="life-counter" type="xsd:integer" minOccurs="0"/>
    <xsd:element name="rtc" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="request_date" type="xsd:dateTime"/>
          <xsd:element name="response_date" type="xsd:dateTime"/>
          <xsd:element name="date" type="xsd:dateTime"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="timesync">
  <xsd:sequence>
    <xsd:element name="wavenis" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="count" type="xsd:integer"/>
          <xsd:element name="skipped" minOccurs="0">
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name="address" type="wd:wavenis_address"
                  minOccurs="0" maxOccurs="unbounded"/>
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:attribute name="cid" type="xsd:string" use="optional"/>
    <xsd:attribute name="date" type="xsd:dateTime"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="wavenis_address">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="([0-9a-zA-Z]{12})|([0-9]{5}-?[0-9]{2}-?[0-9]{8})"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="ipv4">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="empty">
  <xsd:restriction base="xsd:string">
    <xsd:length value="0"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="ipv4_opt">
  <xsd:union memberTypes="wd:ipv4 wd:empty"/>
</xsd:simpleType>
<xsd:simpleType name="datetime_opt">
  <xsd:union memberTypes="xsd:dateTime wd:empty"/>
</xsd:simpleType>
<xsd:simpleType name="follow">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="follow"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="datetime_follow">
  <xsd:union memberTypes="xsd:dateTime wd:follow wd:empty"/>
</xsd:simpleType>
<xsd:simpleType name="hex_string">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9a-zA-Z]*"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

ANNEXE VI. EXEMPLE XML – SUPERVISION



supervision.xml

```
<?xml version="1.0"?>
<supervision
  xmlns="http://www.SOCOMEK.com/WRF_supervision_20110606"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.SOCOMEK.com/WRF_supervision_20110606 supervision.xsd">
  <uid>07F38D</uid>
  <status cid='cmd_123' date='2011-05-26T13:12:00'>
    <app>
      <version>0.2</version>
      <kernel>2.6.35.6</kernel>
    </app>
    <system>
      <power>true</power>
      <defaults/>
      <uptime>33 days</uptime>
    </system>
    <com>
      <modem>
        <model>Cinterion BGS2-W</model>
        <firmware>1.1.246</firmware>
        <imei>AA-BBBBBB-CCCCC</imei>
        <msisdn>380561234567</msisdn>
        <rssi>1</rssi>
        <csq>3</csq>
        <ip>10.0.1.23</ip>
      </modem>
      <ethernet>
        <ip>192.168.0.10</ip>
      </ethernet>
      <upload>
        <last>2011-05-26T10:30:00</last>
        <next>2011-05-27T10:30:00</next>
      </upload>
    </com>
    <wavenis>
      <address>011A1030A6E3</address>
      <last>2011-05-26T13:00:00</last>
      <modules>
        <count>58</count>
      </modules>
    </wavenis>
    <metering>
      <mbus>
        <last>2011-05-26T12:55:00</last>
        <last_scan>2011-05-21T12:05:00</last_scan>
        <modules>
          <count>8</count>
        </modules>
      </mbus>
    </metering>
    <rfid>
      <count>2</count>
      <tags>
        <tag>
          <id>23F005</id>
        </tag>
        <tag>
          <id>23F006</id>
        </tag>
      </tags>
    </rfid>
  </status>
  <scan cid='cmd_124' date='2011-05-26T13:18:00'>
    <wavenis>
      <modules>
        <module>
          <date>2011-05-26T13:18:10</date>
          <address>011A1030A5D4</address>
          <rssi>21</rssi>
          <life-counter>5389</life-counter>
          <rtc>
            <request_date>2011-05-26T13:18:11</request_date>
            <response_date>2011-05-26T13:18:14</response_date>
            <date>2011-05-26T13:13:25</date>
          </rtc>
        </module>
      </modules>
    </wavenis>
  </scan>
</supervision>
```

```
    </modules>
  </wavenis>
</scan>

<timesync cid='cmd_125' date='2011-05-26T13:27:00'>
  <wavenis>
    <count>58</count>
  </wavenis>
</timesync>
</supervision>
```

ANNEXE VII. SCHÉMA XSD – DONNÉES



data.xsd

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:wd="http://www.SOCOMECCOM/WRF_data_20110606"
  targetNamespace="http://www.SOCOMECCOM/WRF_data_20110606"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xsd:element name="data">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="uid" type="xsd:string"/>
        <xsd:choice maxOccurs="unbounded">
          <xsd:element name="pulse" type="wd:pulse"/>
          <xsd:element name="mbus" type="wd:mbus"/>
          <xsd:element name="wavenis" type="wd:wavenis"/>
          <xsd:element name="modbus" type="wd:modbus"/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="pulse">
    <xsd:sequence>
      <xsd:element name="input_1" minOccurs="0">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="xsd:nonNegativeInteger">
              <xsd:attribute name="label" type="xsd:string" use="optional"/>
              <xsd:attribute name="unit" type="xsd:string" use="optional"/>
            </xsd:extension>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="input_2" minOccurs="0">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="xsd:nonNegativeInteger">
              <xsd:attribute name="label" type="xsd:string" use="optional"/>
              <xsd:attribute name="unit" type="xsd:string" use="optional"/>
            </xsd:extension>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="input_3" minOccurs="0">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="xsd:nonNegativeInteger">
              <xsd:attribute name="label" type="xsd:string" use="optional"/>
              <xsd:attribute name="unit" type="xsd:string" use="optional"/>
            </xsd:extension>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="date" type="xsd:dateTime"/>
  </xsd:complexType>

  <xsd:complexType name="mbus">
    <xsd:sequence>
      <xsd:element name="header" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="Id" type="xsd:string"/>
            <xsd:element name="Manufacturer" type="xsd:string" minOccurs="0"/>
            <xsd:element name="Version" type="xsd:string" minOccurs="0"/>
            <xsd:element name="Medium" type="xsd:string"/>
            <xsd:element name="AccessNumber" type="xsd:string"/>
            <xsd:element name="Status" type="xsd:string"/>
            <xsd:element name="Signature" type="xsd:string" minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="records" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="record" minOccurs="0" maxOccurs="unbounded">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```

        <xsd:attribute name=>type>
          <xsd:simpleType>
            <xsd:restriction base=>xsd:string>
              <xsd:enumeration value=>inst/>
              <xsd:enumeration value=>max/>
              <xsd:enumeration value=>min/>
              <xsd:enumeration value=>error_state/>
              <xsd:enumeration value=>manufacturer/>
              <xsd:enumeration value=>actual/>
              <xsd:enumeration value=>stored/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute name=>unit type=>xsd:string/>
        <xsd:attribute name=>storage type=>xsd:nonNegativeInteger
          use=>optional/>
        <xsd:attribute name=>tariff type=>xsd:nonNegativeInteger
          use=>optional/>
        <xsd:attribute name=>sub-unit type=>xsd:nonNegativeInteger
          use=>optional/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name=>date type=>xsd:dateTime/>
</xsd:complexType>

<xsd:complexType name='wavenis'>
  <xsd:sequence>
    <xsd:element name=>header>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name=>address type=>wd:wavenis_address />
          <xsd:element name=>label type=>xsd:string />
          <xsd:element name=>type type=>wd:wavenis_type />
          <xsd:element name=>status type=>xsd:integer />
          <xsd:element name=>io_state type=>xsd:integer minOccurs=>0 />
          <xsd:element name=>retry_count type=>xsd:integer minOccurs=>0 />
          <xsd:element name=>err_status type=>wd:wavenis_err_status minOccurs=>0 />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name=>records minOccurs=>0>
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name=>record maxOccurs=>unbounded>
            <xsd:complexType>
              <xsd:sequence>
                <xsd:element name=>date type=>xsd:dateTime />
                <xsd:element name=>input type=>wd:wavenis_input minOccurs=>0 />
                <xsd:element name=>value type=>xsd:float minOccurs=>0 />
                <xsd:element name=>event_status type=>xsd:integer minOccurs=>0 />
                <xsd:element name=>cause type=>xsd:integer minOccurs=>0 />
                <xsd:element name=>tic_fixed type=>xsd:anyType minOccurs=>0 />
                <xsd:element name=>tic type=>xsd:anyType minOccurs=>0 />
              </xsd:sequence>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:attribute name=>date type=>xsd:dateTime />
  </xsd:complexType>

<xsd:simpleType name=>wavenis_err_status>
  <xsd:restriction base=>xsd:string>
    <xsd:enumeration value=>no_response />
    <xsd:enumeration value=>err_repeater_1 />
    <xsd:enumeration value=>err_repeater_2 />
    <xsd:enumeration value=>err_repeater_3 />
    <xsd:enumeration value=>err_config />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name='wavenis_address'>
  <xsd:restriction base='xsd:string'>
    <xsd:pattern value='([0-9a-zA-Z]{12})|([0-9]{5}-?[0-9]{2}-?[0-9]{8})' />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name=>wavenis_input>
  <xsd:restriction base=>xsd:string>
    <xsd:enumeration value=>A />
    <xsd:enumeration value=>B />
  </xsd:restriction>
</xsd:simpleType>

```

```

        <xsd:enumeration value="C" />
        <xsd:enumeration value="D" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="wavenis_type">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="waveflow" />
        <xsd:enumeration value="wavetherm_dallas" />
        <xsd:enumeration value="wavetherm_pt100" />
        <xsd:enumeration value="wavetherm_pt1000" />
        <xsd:enumeration value="wavesense_4-20ma" />
        <xsd:enumeration value="wavesense_0-5v" />
        <xsd:enumeration value="wavelog" />
        <xsd:enumeration value="wavetic" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="modbus">
    <xsd:sequence>
        <xsd:element name="header">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="label" type="xsd:string" />
                    <xsd:element name="address" type="xsd:integer" />
                    <xsd:element name="ip" type="wd:ipv4" minOccurs="0" />
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="records" minOccurs="0">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="record" maxOccurs="unbounded">
                        <xsd:complexType>
                            <xsd:sequence>
                                <xsd:element name="name" type="xsd:string" />
                                <xsd:element name="address" type="wd:hex_string" />
                                <xsd:element name="type" type="wd:modbus_type" />
                                <xsd:group ref="wd:modbus_interpreted_value" minOccurs="0"/>
                                <xsd:element name="value" type="wd:modbus_value_type"/>
                            </xsd:sequence>
                        </xsd:complexType>
                    </xsd:element>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="date" type="xsd:dateTime" />
</xsd:complexType>

<xsd:group name="modbus_interpreted_value">
    <xsd:sequence>
        <xsd:element name="min" type="xsd:string"/>
        <xsd:element name="max" type="xsd:string"/>
        <xsd:element name="avg" type="wd:modbus_avg_type"/>
    </xsd:sequence>
</xsd:group>

<xsd:simpleType name="modbus_type">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="S[0,1,3,4]"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="modbus_avg_type">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="count" type="xsd:integer"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="modbus_value_type">
    <xsd:simpleContent>
        <xsd:extension base="xsd:string">
            <xsd:attribute name="date" type="xsd:dateTime"/>
        </xsd:extension>
    </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="ipv4">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}"/>
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="hex_string">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="0x[0-9a-zA-Z]*"/>
    </xsd:restriction>
</xsd:simpleType>

```

```
</xsd:restriction>  
</xsd:simpleType>  
</xsd:schema>
```


ANNEXE VIII. FORMAT CSV – DONNÉES

Le format CSV (Comma Separated Values) est un format sans définition formelle.

Néanmoins, il suit les règles suivantes :

Une ligne contient un seul enregistrement
Chaque enregistrement correspond à une seule ligne
Chaque ligne se termine par un retour de ligne.
Chaque ligne contient le même nombre de champs.
Chaque champ est séparé par une virgule.

Chaque ligne est formatée comme suit:

<timestamp>,<Source>,<Address>,,,<Label>,<Input>,<Valeur>

Avec:

Horodatage: JJ/MM/AAAA HH:MM

Source : Les valeurs possibles sont:

FL: WaveFlow

TH: WaveTherm

LO: WaveLog

SE: WaveSense

Adresse : adresse Wavenis

Label : module de Wavenis s'il est configuré, vide sinon.

Entrée: identifiant d'entrée (A, B, C ou D).

Valeur: Valeur de l'entrée donnée.

ANNEXE IX. SCHÉMA XSD – COMMANDES



command.xsd

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" «http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd»>
<html xmlns="http://www.w3.org/1999/xhtml">

<head>
<title>
  command.xsd in Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas
  – SOCOMEC
</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<!--[if IE]><script type="text/javascript">window.location.hash = window.location.hash;</script><![endif-->
<link rel="search" href="/trac/search" />
<link rel="help" href="/trac/wiki/TracGuide" />
<link rel="alternate" href="/trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd?format=txt"
type="text/plain" title="Plain Text" /><link rel="alternate" href="/trac/export/3915/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_
Schemas/command.xsd" type="application/xml; charset=utf-8" title="Original Format" />
<link rel="start" href="/trac/wiki" />
<link rel="stylesheet" href="/trac/chrome/common/css/trac.css" type="text/css" /><link rel="stylesheet" href="/trac/chrome/common/css/
code.css" type="text/css" /><link rel="stylesheet" href="/trac/pygments/trac.css" type="text/css" /><link rel="stylesheet" href="/trac/chrome/
common/css/browser.css" type="text/css" />
title="Revision 3715" />
<link rel="shortcut icon" href="/trac/chrome/common/trac.ico" type="image/x-icon" />
<link rel="icon" href="/trac/chrome/common/trac.ico" type="image/x-icon" />
<link type="application/opensearchdescription+xml" rel="search" href="/trac/search/opensearch" title="Search SOCOMEC" />
<script type="text/javascript" src="/trac/chrome/common/js/jquery.js"></script><script type="text/javascript" src="/trac/chrome/common/
js/babel.js"></script><script type="text/javascript" src="/trac/chrome/common/js/trac.js"></script><script type="text/javascript" src="/trac/
chrome/common/js/search.js"></script><script type="text/javascript" src="/trac/chrome/site/md5.js"></script>
<!--[if lt IE 7]>
<script type="text/javascript" src="/trac/chrome/common/js/ie_pre7_hacks.js"></script>
<![endif-->
<script type="text/javascript" src="/trac/chrome/common/js/folding.js"></script>
<script type="text/javascript">
jQuery(document).ready(function($) {
  $('<trac-toggledelimited>').show().click(function() {
    $(this).siblings().find('<trac-deleted>').toggle();
    return false;
  }).click();
  $('<#jumploc input>').hide();
  $('<#jumploc select>').change(function () {
    this.parentNode.parentNode.submit();
  });
  $('<#preview table.code>').enableCollapsibleColumns($('<#preview table.code thead th.content>'));
});
</script>
</head>
<body>
<div id="banner">
<div id="header">
<a id="logo" href="/trac/wiki/TracIni#header_logo-section"></a>
</div>
<form id="search" action="/trac/search" method="get">
<div>
<label for="proj-search">Search:</label>
<input type="text" id="proj-search" name="q" size="18" value="" />
<input type="submit" value="Search" />
</div>
</form>
<div id="metanav" class="nav">
<ul>
<li class="first"><a href="/trac/login">Login</a></li><li><a href="/trac/prefs">Preferences</a></li><li><a href="/trac/wiki/
TracGuide">Help/Guide</a></li><li><a href="/trac/about">About Trac</a></li><li class="last"><a href="/trac/register">Register</a></li>
</ul>
</div>
</div>
<div id="mainnav" class="nav">
<ul>
<li class="first"><a href="/trac/wiki">Wiki</a></li><li><a href="/trac/timeline">Timeline</a></li><li><a href="/trac/roadmap">Roadmap</
a></li><li class="active"><a href="/trac/browser">Browse Source</a></li><li><a href="/trac/report">View Tickets</a></li><li><a href="/
trac/search">Search</a></li><li><a href="/trac/ticketcalendar">Calendar</a></li><li><a href="/trac/pdashboard">Metrics</a></li><li
class="last"><a href="/trac/ticketgantt">Gantt Ticket</a></li>
</ul>
</div>
<div id="main">
<div id="ctxtnav" class="nav">
<h2>Context Navigation</h2>
<ul>
```

```
<li class=>first><span>&larr; <a class=>prev href=>../trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd?rev=3715 title=>Revision 3715>Previous Revision</a></span></li><li class=>missing>Next Revision &rarr;</span></li><li class=>next><a href=>../trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd?annotate=blame title=>Annotate each line with the last changed revision (this can be time consuming...)>Blame</a></li><li class=>last><a href=>../trac/log/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd>Revision Log</a></li></ul><hr /></div><div id=>content class=>browser><h1><a class=>pathentry first href=>../trac/browser?order=name title=>Go to repository index>source:</a><a class=>pathentry href=>../trac/browser/Projets?order=name title=>View Projets>Projets</a><span class=>pathentry sep></span><a class=>pathentry href=>../trac/browser/Projets/Clients?order=name title=>View Clients>Clients</a><span class=>pathentry sep></span><a class=>pathentry href=>../trac/browser/Projets/Clients/SOCOMECE?order=name title=>View SOCOMECE>SOCOMECE</a><span class=>pathentry sep></span><a class=>pathentry href=>../trac/browser/Projets/Clients/SOCOMECE/Gateway?order=name title=>View Gateway>Gateway</a><span class=>pathentry sep></span><a class=>pathentry href=>../trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P?order=name title=>View WG310P>WG310P</a><span class=>pathentry sep></span><a class=>pathentry href=>../trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk?order=name title=>View trunk>trunk</a><span class=>pathentry sep></span><a class=>pathentry href=>../trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas?order=name title=>View XML_Schemas>XML_Schemas</a><span class=>pathentry sep></span><a class=>pathentry href=>../trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd?order=name title=>View command.xsd>command.xsd</a><span class=>pathentry sep>>@</span><a class=>pathentry href=>../trac/changeset/3915/Projets title=>View changeset 3915>3915</a><br style=>clear: both /></h1><div id=>jumprev><form action=> method=>get><div><label for=>rev>View revision:</label><input type=>text id=>rev name=>rev size=>6 /></div></form></div><table id=>info summary=>Revision info><tr><th scope=>col>Revision <a href=>../trac/changeset/3716/Projets>3716</a>,<br><span title=>4047 bytes>4.0 KB</span> checked in by ghoftmann, <a class=>timeline href=>../trac/timeline?from=2012-11-19T18:30:00&precision=second title=>2012-11-19T18:30:00 in Timeline>3 weeks</a> ago<br><a href=>../trac/changeset/3716/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd>diff</a></th></tr><tr><td class=>message searchable><p>Fix modbus command syntax which do not accept rtu addresses.<br /></p></td></tr></table><div id=>preview class=>searchable><table class=>code><thead><tr><th class=>lineno title=>Line numbers>Line</th><th class=>content></th></tr></thead><tbody><tr><th id=>L1><a href=>#L1>1</a></th><td><span class=>cp>&lt;?xml version=>1.0?&gt;</span></td></tr><tr><th id=>L2><a href=>#L2>2</a></th><td><span class=>nt>&lt;xsd:schema</span><span class=>na>xmns:xsd=</span><span class=>s>http://www.w3.org/2001/XMLSchema</span></td></tr><tr><th id=>L3><a href=>#L3>3</a></th><td><span class=>na>xmns:wd=</span><span class=>s>http://www.SOCOMECE.com/WRF_command_20110606</span></td></tr><tr><th id=>L4><a href=>#L4>4</a></th><td><span class=>na>targetNamespace=</span><span class=>s>http://www.SOCOMECE.com/WRF_command_20110606</span></td></tr><tr><th id=>L5><a href=>#L5>5</a></th><td><span class=>na>elementFormDefault=</span><span class=>s>qualified</span></td></tr><tr><th id=>L6><a href=>#L6>6</a></th><td><span class=>na>attributeFormDefault=</span><span class=>s>unqualified</span></td></tr><tr><th id=>L7><a href=>#L7>7</a></th><td><span class=>na>name=</span><span class=>s>&gt;</span></td></tr><tr><th id=>L8><a href=>#L8>8</a></th><td><span class=>na>element</span><span class=>s>command</span></td></tr><tr><th id=>L9><a href=>#L9>9</a></th><td><span class=>na>name=</span><span class=>s>command</span></td></tr><tr><th id=>L10><a href=>#L10>10</a></th><td><span class=>na>type=</span><span class=>s>complexType</span></td></tr><tr><th id=>L11><a href=>#L11>11</a></th><td><span class=>na>type=</span><span class=>s>element</span></td></tr><tr><th id=>L12><a href=>#L12>12</a></th><td><span class=>na>name=</span><span class=>s>maxOccurs=</span><span class=>s>unbounded</span></td></tr><tr><th id=>L13><a href=>#L13>13</a></th><td><span class=>na>type=</span><span class=>s>complexType</span></td></tr><tr><th id=>L14><a href=>#L14>14</a></th><td><span class=>na>type=</span><span class=>s>sequence</span></td></tr><tr><th id=>L15><a href=>#L15>15</a></th><td><span class=>na>type=</span><span class=>s>choice</span></td></tr><tr><th id=>L16><a href=>#L16>16</a></th><td><span class=>na>name=</span><span class=>s>reboot</span></td></tr><tr><th id=>L17><a href=>#L17>17</a></th><td><span class=>na>type=</span><span class=>s>element</span></td></tr><tr><th id=>L18><a href=>#L18>18</a></th><td><span class=>na>type=</span><span class=>s>update</span></td></tr><tr><th id=>L19><a href=>#L19>19</a></th><td><span class=>na>type=</span><span class=>s>scan</span></td></tr><tr><th id=>L20><a href=>#L20>20</a></th><td><span class=>na>type=</span><span class=>s>wd:cmd_scan</span></td></tr><tr><th id=>L21><a href=>#L21>21</a></th><td><span class=>na>type=</span><span class=>s>wd:cmd_wavenis</span></td></tr><tr><th id=>L22><a href=>#L22>22</a></th><td><span class=>na>type=</span><span class=>s>wd:cmd_modbus</span></td></tr><tr><th id=>L23><a href=>#L23>23</a></th><td><span class=>na>type=</span><span class=>s>wd:cmd_simple</span></td></tr></tbody></table></div>
```



```
span>/td></tr><tr><th id=>L92<><a href=>#L92<>92</a></th><td> <span class=>nt<>&lt;xsd:restriction</span> <span class=>na<>base=</span><span class=>s<>>xsd:string</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L93<><a href=>#L93<>93</a></th><td> <span class=>nt<>&lt;xsd:enumeration</span> <span class=>na<>value=</span><span class=>s<>>open</span><span class=>nt<>&lt;/span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L94<><a href=>#L94<>94</a></th><td> <span class=>nt<>&lt;xsd:enumeration</span> <span class=>na<>value=</span><span class=>s<>>close</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L95<><a href=>#L95<>95</a></th><td> <span class=>nt<>&lt;xsd:restriction</span></td></tr><tr><th id=>L96<><a href=>#L96<>96</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L97<><a href=>#L97<>97</a></th><td><span class=>nt<>&lt;xsd:restriction</span></td></tr><tr><th id=>L98<><a href=>#L98<>98</a></th><td><span class=>nt<>&lt;xsd:complexType</span></td></tr><tr><th id=>L99<><a href=>#L99<>99</a></th><td> <span class=>na<>name=</span><span class=>s<>>cmd_modbus</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L100<><a href=>#L100<>100</a></th><td> <span class=>nt<>&lt;xsd:element</span> <span class=>na<>name=</span><span class=>s<>>address</span> <span class=>na<>type=</span><span class=>s<>>wd:modbus_cmd_string</span> <span class=>na<>maxOccurs=</span><span class=>s<>>unbounded</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L101<><a href=>#L101<>101</a></th><td> <span class=>nt<>&lt;xsd:sequence</span></td></tr><tr><th id=>L102<><a href=>#L102<>102</a></th><td> <span class=>nt<>&lt;xsd:attribute</span> <span class=>na<>name=</span><span class=>s<>>subcmd</span> <span class=>na<>type=</span><span class=>s<>>xsd:string</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L103<><a href=>#L103<>103</a></th><td> <span class=>nt<>&lt;xsd:attribute</span> <span class=>na<>name=</span><span class=>s<>>data</span> <span class=>na<>type=</span><span class=>s<>>xsd:string</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L104<><a href=>#L104<>104</a></th><td><span class=>nt<>&lt;xsd:complexType</span></td></tr><tr><th id=>L105<><a href=>#L105<>105</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L106<><a href=>#L106<>106</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L107<><a href=>#L107<>107</a></th><td> <span class=>na<>name=</span><span class=>s<>>modbus_cmd_string</span> <span class=>nt<>&lt;/span></td></tr><tr><th id=>L108<><a href=>#L108<>108</a></th><td> <span class=>nt<>&lt;xsd:restriction</span> <span class=>na<>base=</span><span class=>s<>>xsd:string</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L109<><a href=>#L109<>109</a></th><td> <span class=>nt<>&lt;xsd:restriction</span> <span class=>na<>value=</span><span class=>s<>>([0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3}.[0-9]{1,3})?[0-9]{1,4}/S[0,1,3,4]@0x[0-9a-zA-Z]{1,4}'</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L110<><a href=>#L110<>110</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L111<><a href=>#L111<>111</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L112<><a href=>#L112<>112</a></th><td><span class=>nt<>&lt;xsd:simpleType</span> <span class=>na<>name=</span><span class=>s<>>hex_string</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L113<><a href=>#L113<>113</a></th><td> <span class=>nt<>&lt;xsd:restriction</span> <span class=>na<>base=</span><span class=>s<>>xsd:string</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L114<><a href=>#L114<>114</a></th><td> <span class=>nt<>&lt;xsd:pattern</span> <span class=>na<>value=</span><span class=>s<>>[0-9a-zA-Z]*</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L115<><a href=>#L115<>115</a></th><td> <span class=>nt<>&lt;xsd:restriction</span></td></tr><tr><th id=>L116<><a href=>#L116<>116</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L117<><a href=>#L117<>117</a></th><td><span class=>nt<>&lt;xsd:simpleType</span> <span class=>na<>name=</span><span class=>s<>>'wavenis_address'</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L118<><a href=>#L118<>118</a></th><td><span class=>nt<>&lt;xsd:restriction</span> <span class=>na<>base=</span><span class=>s<>>'xsd:string'</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L119<><a href=>#L119<>119</a></th><td> <span class=>nt<>&lt;xsd:restriction</span> <span class=>na<>value=</span><span class=>s<>>'([0-9a-zA-Z]{12}){0-9}[5]?[0-9]{2}?[0-9]{8}'</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L120<><a href=>#L120<>120</a></th><td> <span class=>nt<>&lt;xsd:pattern</span> <span class=>na<>value=</span><span class=>s<>>'([0-9a-zA-Z]{12}){0-9}[5]?[0-9]{2}?[0-9]{8}'</span><span class=>nt<>&lt;/span></td></tr><tr><th id=>L121<><a href=>#L121<>121</a></th><td> <span class=>nt<>&lt;xsd:restriction</span></td></tr><tr><th id=>L122<><a href=>#L122<>122</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L123<><a href=>#L123<>123</a></th><td><span class=>nt<>&lt;xsd:simpleType</span></td></tr><tr><th id=>L124<><a href=>#L124<>124</a></th><td><span class=>nt<>&lt;xsd:schema</span></td></tr></tbody></table>
```

```
</div>  
<div id=>help<><strong>Note:</strong> See <a href=>/trac/wiki/TracBrowser<>TracBrowser</a>  
for help on using the repository browser.</div>  
<div id=>anydiff<>  
<form action=>/trac/diff method=>get<>  
<div class=>buttons<>  
<input type=>hidden name=>new_path value=>/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd  
</div>  
<input type=>hidden name=>old_path value=>/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd />  
<input type=>hidden name=>new_rev />  
<input type=>hidden name=>old_rev />  
<input type=>submit value=>View changes... title=>Select paths and revs for Diff />  
</div>  
</form>  
</div>  
</div>  
<div id=>alllinks<>  
<h3>Download in other formats:</h3>  
<ul>  
<li class=>first<>  
<a rel=>nofollow href=>/trac/browser/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd?format=txt>Plain Text</a>  
</li><li class=>last<>  
<a rel=>nofollow href=>/trac/export/3915/Projets/Clients/SOCOMECE/Gateway/WG310P/trunk/XML_Schemas/command.xsd>Original  
Format</a>  
</li>  
</ul>  
</div>  
</div>  
<div id=>footer lang=>en xml:lang=>en<><hr />  
<a id=>tracpowered href=>http://trac.edgewall.org/><img src=>/trac/chrome/common/trac_logo_mini.png height=>30 width=>107<>  
alt=>Trac Powered</a>  
<p class=>left<>Powered by <a href=>/trac/about<><strong>Trac 0.12.3</strong></a><br />  
By <a href=>http://www.edgewall.org/>Edgewall Software</a></p>  
<p class=>right<>Visit the Trac open source project at<br /><a href=>http://trac.edgewall.org/>http://trac.edgewall.org/</a></p>  
</div>  
</body>  
</html>
```

ANNEXE X. EXEMPLE XML – COMMANDES



command.xml

```
<?xml version="1.0"?>
<commands
  xmlns="http://www.SOCOMEK.com/WRF_command_20110606"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.SOCOMEK.com/WRF_command_20110606 command.xsd">

  <cmd cid="C_1234">
    <timesync>
      <address>011A0A30AAA0</address>
      <address>011A0A30AAA1</address>
    </timesync>
  </cmd>

  <cmd cid="C_1235">
    <scan mode="rtc life-counter">
      <address>011A0A30AAA0</address>
      <address>011A0A30AAA1</address>
    </scan>
  </cmd>

  <cmd cid="C_1236">
    <scan mode="data"/>
  </cmd>

  <cmd cid="C_1237">
    <status/>
  </cmd>

  <cmd cid="C_1238">
    <update>
      <firmware>wrf_wavenis_v101.bin</firmware>
      <checksum>c1fb7d81f3d53a8b7bf94098115249d3</checksum>
    </update>
  </cmd>

  <cmd cid="C_1239">
    <wavenis subcmd="moduflow-open">
      <address>011A0A30AAA0</address>
    </wavenis>
  </cmd>

  <cmd cid="C_1240">
    <wavenis subcmd="raw" data="10012001">
      <address>011A0A30AAA0</address>
    </wavenis>
  </cmd>

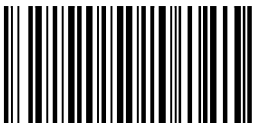
  <cmd cid="C_1241">
    <d_output subcmd="open"/>
  </cmd>

  <cmd cid="C_1242">
    <d_output subcmd="close"/>
  </cmd>

  <cmd cid="C_1243">
    <modbus subcmd="write">
      <address>192.168.0.17:223/S3@0x0F56</address>
    </modbus>
  </cmd>

  <cmd cid="C_1243">
    <modbus subcmd="write">
      <address>223/S3@0x0F56</address>
    </modbus>
  </cmd>

</commands>
```

542 492 A - FR - 10/14

Non contractual document.
Subject to change without notice.