Datalog *H60*

Communication hub

EN

www.socomec.com/en/
datalog-H60

socomec
Innovative Power Solutions

1. GLOSSARY

Name	Description
APN	**Access Point Name** The name of the access point allowing the datalogger to connect to the Internet via mobile linkup.
FTP	**File Transfer Protocol** Communication protocol designed for sharing files electronically over a TCP/IP network.
GPRS	**General Packet Radio Service** Mobile telephony standard derived from GSM allowing higher data throughput. Also called 2.5G. Download: max. 86 Kbps Upload: max. 43 Kbps
GSM	**Global System for Mobile Communications** Public switched mobile phone network.
HTTP	**HyperText Transfer Protocol** Client/server communications protocol developed for the web.
IP	**Internet Protocol** Messages protocol for addressing and transmitting TCP bundles on the network.
PSSTT	**Public Switched Telephone Network** See PSTN.
Rail DIN	Standardized 35-mm metal rail used in Europe in the industrial control equipment in racks.
PSTN	**Public Switched Telephone Network** Public switched landline network.
TCP	**Transmission Control Protocol** Connection-based online protocol that offers data package segmentation services that the IP protocol transfers on the network. This protocol provides a reliable data transfer service. See also IP.
TCP/IP	**Transmission Control Protocol / Internet Protocol** Set of network protocols that provide interconnection services between different hardware architecture computers and operating systems. TCP/IP includes standards for communication between computers and conventions for interconnecting networks and routing.
CIS	**Client information system** Output of digital information from ERDF counters that continuously transfer the managed contractual parameters, as well as the consumption volumes measured by the device.

2. NOTES ON THIS MANUAL

This guide describes how to set up, install and start up the device. Please keep these instructions at hand at all times.

2.1. Field of application

This technical description applies to Datalog H60 dataloggers.

2.2. Models

There are several product models, depending on the type of GSM modem and radio card.

SOCOMEC product model:

• - Datalog H60 -WirelessMbus-868MHz-3G

• - RFID radio card Modem 2G – RFID

• - Accessories common to all products:
  - 12v external power supply unit
  - 24V Rail Din power supply
  - GPRS dual-band antenna delivered with a 5-m cable
  - GPRS dual-band antenna delivered with a 10-m cable
  - GPRS dual-band antenna delivered with a 20-m cable

2.3. Target group

This guide is intended for installation engineers.

2.4. General operating principle

The H60 datalogger is a wireless network hub. The basic functionalities of the DATALOG H60 datalogger are to collect data from counters and sensors and to control I/Os.
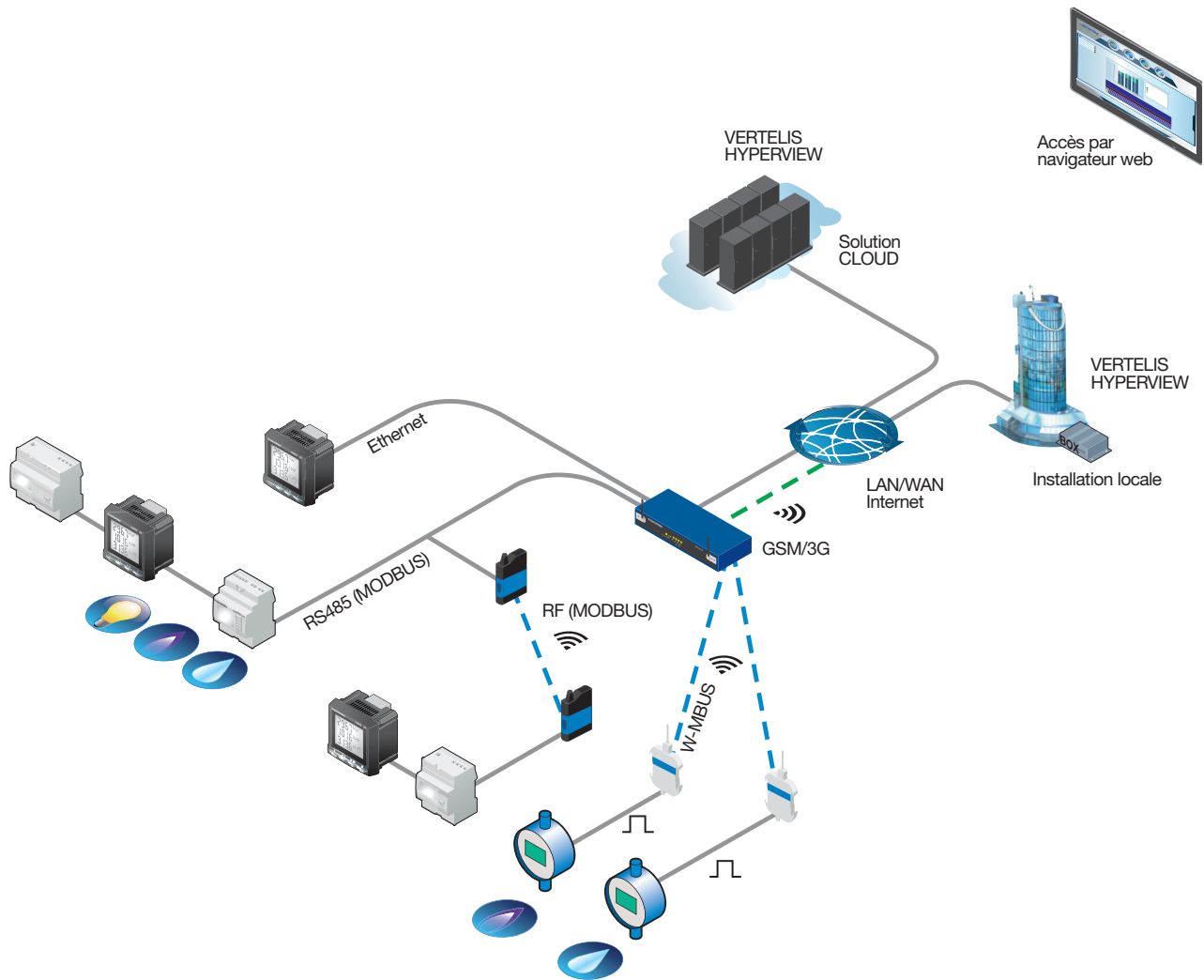
2.5. Technical specifications

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

3. SAFETY INSTRUCTIONS

Please observe all the safety instructions contained in this guide.
Any failure to observe these instructions may cause damage to the equipment and represent a danger to humans.

⚠ CAUTION	The Datalog H60 datalogger may be damaged by electrostatic discharge (ESD). • Datalog H60 installation work must be performed by an authorised electrician. • Avoid contact with component terminals and connector contacts.
⚠ CAUTION	The Datalog H60 datalogger contains a Lithium battery. There is a risk of explosion if the battery is replaced with a battery of the incorrect type.

4. UNPACKING

4.1. Delivery contents

Check that the delivery is complete and that the device is not damaged. Contact Socomec if the delivery is incomplete or damaged.

DATALOG H60	2 angled antennae (RF/Modem)	12v external power supply unit	RailDin and wall-mount assembly kits

Optional:

GPRS dual-band antenna delivered with a 5 or 20-m cable

4.2. Identifying the datalogger

4.2.1. ID label
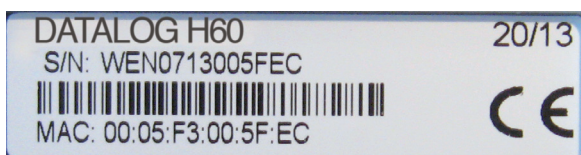
You can identify the Datalog H60 datalogger from its ID label on front of the box.
- DATALOG H60: Product name.
- WEN071300614B: Serial number
- 20/13: Week/year of production.
- MAC: Product's Mac address.
- The barcode corresponds to the product's serial number.

4.2.2. Software release

You can fine the software version on the web interface of the datalogger. The software version is shown on the Overview homepage.

5. SETTING UP

5.1. Choosing the setup environment

Please respect the following environment conditions.

- Operating temperature: -10°C to +45 °C.
  Battery performance is reduced at temperatures below 0°C and above 45°C.

- Storage temperature range: -20°C to +45°C (less than one month). -20°C to +35°C (more than one month)

- Ambient humidity must be between 10 and 90% RH.

- Protect the Datalog H60 against dust, humidity, aggressive substances and moisture.

- The distance between the Datalog H60 and the Modbus equipment must not exceed the maximum distance allowed for its interface type (RS485 or RS232).

- When using the cell link, be sure that reception is optimal when setting up. You can find the details in the relevant sections of this manual.

The Datalog H60 datalogger should automatically send your installation data to the remote server and at regular intervals. Please make sure you do the following:

- To send the installation data via Ethernet, the network must include a router allowing Internet access.

- To send installation data via the GSM connection, there must be sufficient network coverage.

- If you are using the optional GSM remote antenna, please refer to the section about installing the remote antenna before setting up the datalogger.

⚠ CAUTION	To optimize GSM reception, you must leave a clear gap of 20 cm around the antenna.

5.2. Setting up the Datalog H60 – upright position

The Datalog H60 can stand upright on a flat surface.

5.3. Setting up the Datalog H60 – DIN rail

The Datalog H60 datalogger can be mounted on a standard 35-mm metal rail (DIN RAIL).

To mount the unit on the rail you need the following accessory: KFIX01.

1. Attach the assembly kit to the datalogger:

2. Hook the Datalog H60 to the rail using the two brackets.
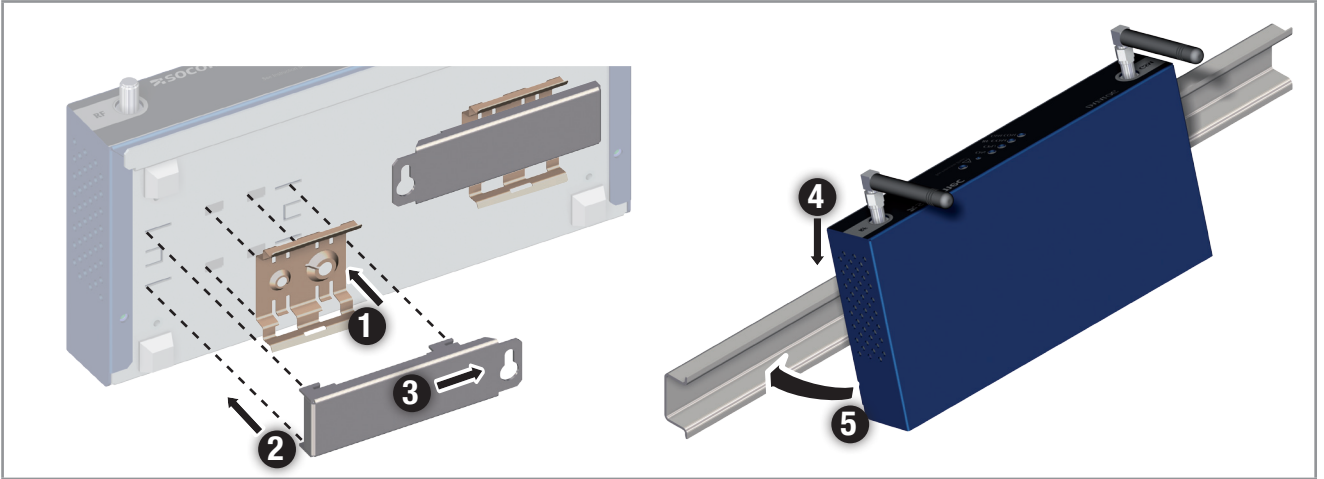
5.4. Wall-mounting the Datalog H60

You can mount the Datalog H60 to the wall.

To mount the unit to the wall you need the following accessory: KFIX01.

1. Attach the assembly kit to the datalogger:

2. Lift the 2 metal strips to secure the brackets:
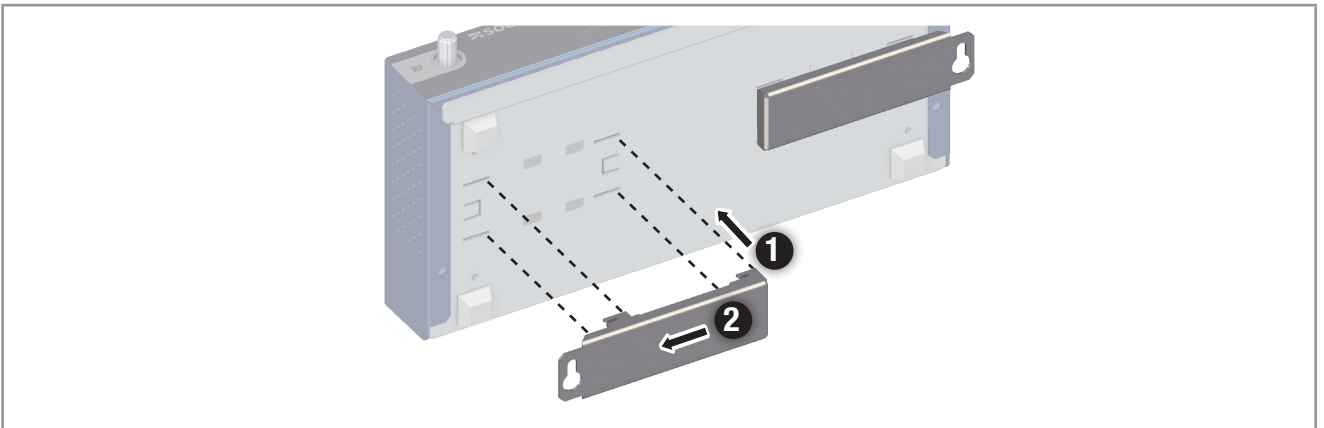
3. Decide where you want to position the datalogger and mark the drill holes with a pencil.

4. Drill the holes (diameter: 6 mm) at the marked positions and insert the wall plugs.

5. Screw the Datalog H60 to the wall.
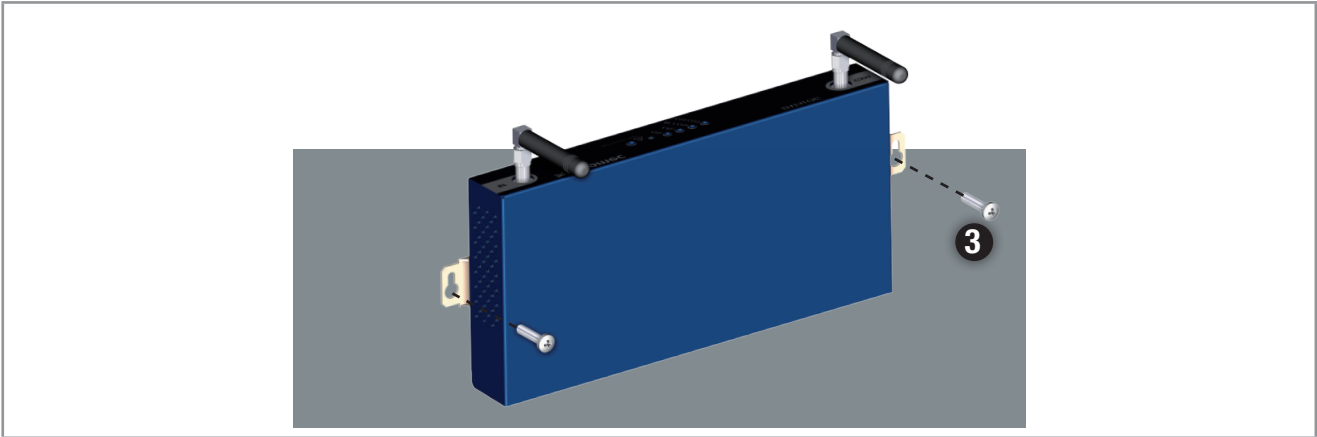
5.5. Setting up the standard angled GPRS/GSM antenna

The datalogger comes with two angled antennae as standard.
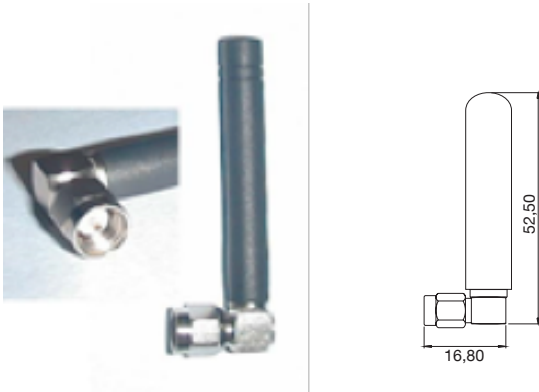
5.5.1. Specifications

Frequency/MHZ	900/1800
Yield	0db
VSWR	< 1: 2.0
Impedance	50 Ohms
Patten	Omni direction
Polarization	Vertical
Connector	SMA R/A male

Screw the GSM antenna cable onto the antenna socket marked "GSM" on the front of the datalogger.

⚠ CAUTION	To optimize GSM reception, you must leave a clear gap of 20 cm around the antenna.

5.6. Setting up the GSM/GPRS remote antenna (optional)

Only if you have control as an optional extra: Remote antenna kit.
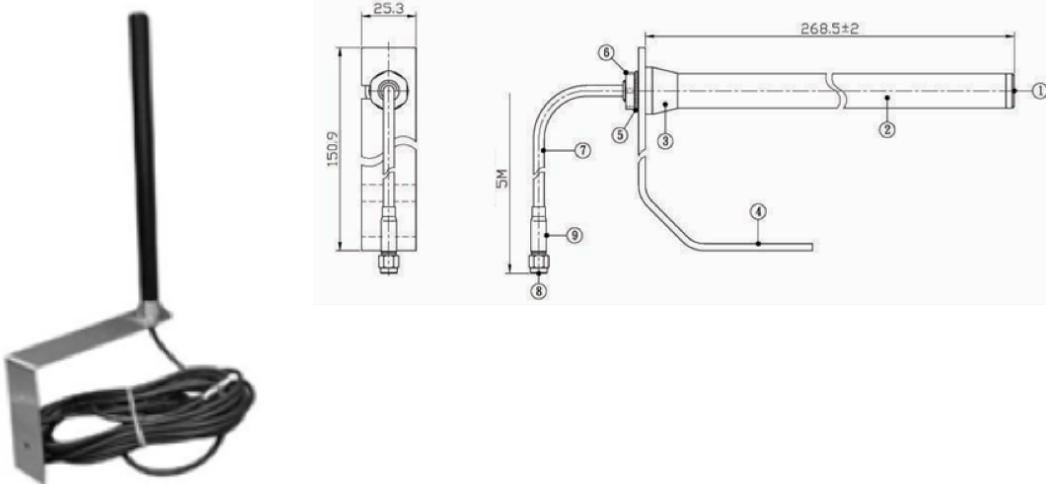
Contents of the kit: 1 antenna and 1 antenna bracket with antenna cable.

Decide on the best place to mount the antenna bracket.
1. Position the antenna by holding the bracket against the wall and marking the drilling holes with a pencil.
2. Drill holes (diameter: 6 mm) at the marked positions and insert the wall plugs.
3. Screw the antenna bracket to the wall.
4. Screw the antenna on the antenna cable.
5. Screw the cable to the antenna jack.

Cable: RG-58.
Frequencies: 824-960 MHz ; 1710-1990 MHz.

6. ELECTRICAL CONNECTIONS

⚠ CAUTION	• All wiring work must be performed by a qualified electrician. • Before installation, all devices connected to the corresponding communication bus must be disconnected on both sides (DC and AC). • Please observe all the safety instructions contained in the inverter documentation.

6.1. Overview of external interfaces and LEDs

6.1.1. Front panel

Antennae

ANT RF	Antenna for RF connection
ANT GSM	Antenna for GPRS or 3G connection

Buttons

REQUEST	Press briefly => Start connection Press and hold (3 seconds) => Show GSM signal strength (see GSM LED) Press and hold 3x => Reset to factory settings
RESET	Restart datalogger hardware.

LEDs

POWER	Lights up when the device is on
CPU	Lights up if there is CPU activity
RF	Off by default and flashes to show RF traffic or when a computer is connected in transparent mode.
GSM	Lights up during GSM connection
	Lights up for 1 second when receiving an SMS
	If you press and hold the Request button it gives you the RSSI reception signal strength by flashing a certain number of times (0 to 5 times)
	0 – signal strength ≤ -112 dBm 1 – signal strength between -111 dBm and -96 dBm 2 – signal strength between -96 dBm and -81 dBm 3 – signal strength between -81 dBm and -66 dBm 4 – signal strength between -66 dBm and -51 dBm 5 - signal strength > -51 dBm

6.1.2. Rear panel
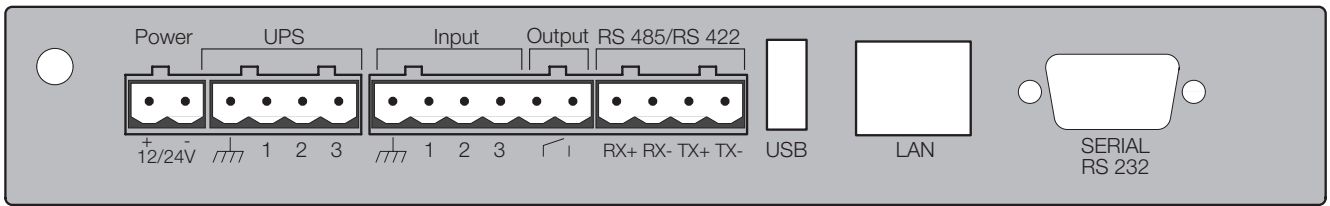
12/24V	Datalogger power-in.
UPS	Inverter status inputs.
Input	3 On/Off inputs.
Output	1 On/Off output.
RS485/RS422	1 RS485/RS422 port.
USB	USB socket.
LAN	LAN socket.
Serial RS232	RS232 port, type DB9.

6.2. Connecting the DATALOG H60 datalogger to the Internet so it can communicate with the remote server

This section describes how to connect the Datalog H60 to the Internet so it can communicate with the remote server. This communication can take place via a choice of three different media:

• Ethernet with an ADSL line.

• 3G (other connections are possible, please contact SOCOMEC)

6.2.1. Connecting to the Internet with an ADSL line

Connect the Datalog H60 datalogger to the ADSL network using the Ethernet port marked "LAN" on the front panel.

| Datalog H60 | ADSL modem/router | Internet | FTP/HTTP server |
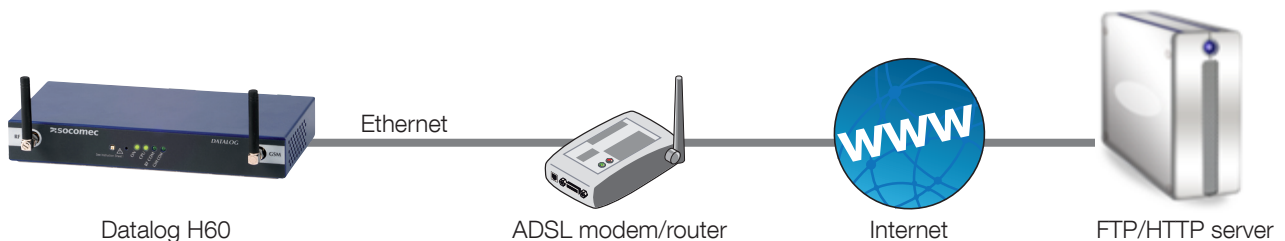
The length and quality of your cable can affect the quality of the signal.

Use an Ethernet cable and follow these tips:

• We recommend using a straight cable.

• Use a high-quality cable, at least a shielded twisted pair cable, cat. 5 (STP Cat 5) or higher.

• The max. permissible length for an Ethernet cable is 100 m per segment.

⚠ WARNING	**The Datalog H60 datalogger's default settings are as follows:** IP address: 192.168.1.12 Subnet mask: 255.255.255.0 DHCP: Disabled

⚠ WARNING	**Network administrator** If your local network is managed by a network administrator, contact them before adding the Datalog H60 datalogger to your network.

6.2.2. Connecting to the Internet using GPRS or 3G

To use the GPRS or 3G connection so the DATALOG H60 datalogger can communicate with the remote server, you need to insert a SIM card into the pull-out slot inside the datalogger.
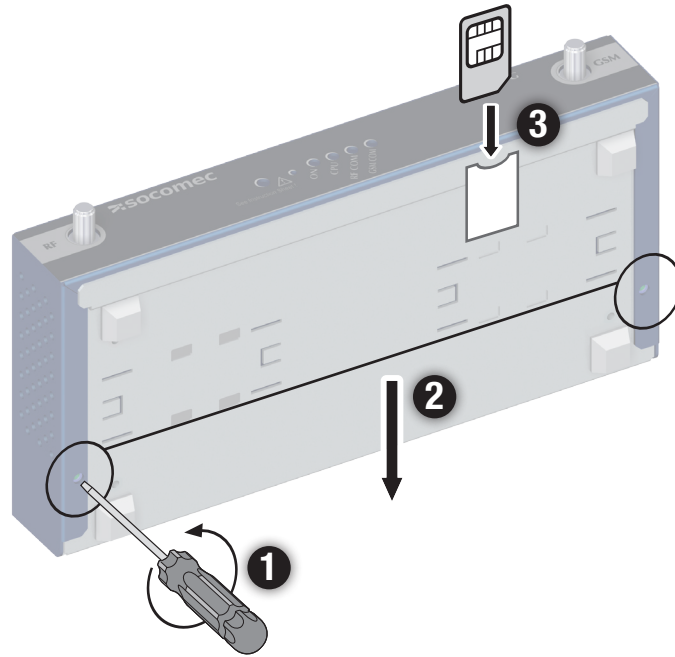
To use the Datalog H60 datalogger in 3G or other mode, you must insert a SIM card with the following properties:
- Can receive and send SMSs
- Suitable communication method with a relevant subscription.

Insert the SIM card into the device.

By default, the datalogger's PIN Mode is Off (see Configuring the Webserver).

There are three possible options:
- The PIN code is disabled: modem communication is active.
- The PIN code is enabled as 0000: modem communication is active.
- The PIN code is enabled and not 0000: modem communication is not active.

⚠ WARNING	If, when you first start up the datalogger, the SIM card has an active PIN code that is not 0000, it is blocked after 3 attempts. You can unlock it on your cell phone using the PUK code supplied by your operator.

To communicate with the remote server via GSM, the datalogger must have optimum GSM coverage.

To find out the signal quality received by the datalogger, press and hold down the REQUEST button (3 seconds).

The GSM (RSSI) signal strength is indicated on the GSM LED by a number of flashes (0 to 5 times).

The number of flashes indicates the signal strength, as follows:
- 0 – signal strength ≤ -112 dBm
- 1 – signal strength between -111 dBm and -96 dBm
- 2 – signal strength between -96 dBm and -81 dBm
- 3 – signal strength between -81 dBm and -66 dBm
- 4 – signal strength between -66 dBm and -51 dBm
- 5 - signal strength > -51 dBm

⚠ WARNING	The signal strength received must be at least equal to -96 dBm or 2 flashes.

6.3. Connecting the datalogger's On/Off output

The Datalog H60 features 1 On/Off output.
This can be used as a digital output.

6.4. Connecting the datalogger's On/Off inputs

The Datalog H60 features 3 On/Off inputs.
These can be used as digital inputs to trigger alarms or as a pulse counter (see the user guide for more details).

⚠️ WARNING	To avoid damage to the datalogger, do not apply current or voltage to the on/off inputs.

In pulse counter mode, pulses must last at least 10 ms and no more than 1,000 ms to be recognised.

6.5. Connecting the datalogger to Modbus devices via the RS485 connection

This section describes how to connect the Datalog H60 datalogger to the RS485 bus communication port so it can communicate with Modbus devices.

Use the RS485 interface labelled RS485/RS422.
This interface is a 4 or 2-wire port.
The software automatically chooses the 4-wire or the 2-wire port.
The datalogger can be at the RS485 bus communication end or near the bus.
To ensure the RS485 data bus works, it must be terminated at both ends with a 120-ohm end resistor.
Depending on where the datalogger is on the bus, this end resistor must be placed outside the box.

See the documentation supplied with the Modbus devices for their RS485 connection and wiring.

Once the RS485 cable is laid next to the datalogger:
- 1. Strip the RS485 communication cable sleeve by approx. 4 cm.
- 3. Shorten the shielding up to the cable sleeve.
- 4. Strip the wires by approx. 6 mm.
- 5. Connect the conductors to the port labelled RS485/RS422 following the assignments in your RS485 bus communication.

Set up with 4-wire RS485:

RX+ RX- TX+ TX-

TX+ TX- RX+ RX-
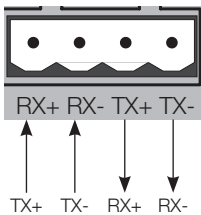
Set up with 2-wire RS485:

RX+ RX- TX+ TX-

+   -
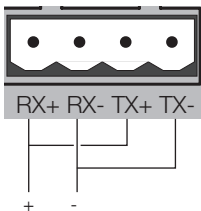
6.6. Connecting the datalogger to the equipment using the RS232 connection

You can use the RS232 port to collect data from an M-Bus device. Connect an M-Bus transceiver to the RS232 port and switch the port to M-Bus mode.

The M-Bus devices must be configured on the bus with a unique address.

Scan the bus from the web interface. The M-Bus devices discovered during this scan are prompted each time the assigned scheduler occurs. Start a new scan when devices are removed from or added to the bus so the datalogger can take the changes into account.

7. CONNECTING TO A LOCAL PC VIA ETHERNET TO CONFIGURE THE DATALOGGER

The first time you start up the Datalog H60 datalogger it must be configured on a PC (mobile or desktop) via the Ethernet network.

You can either connect the PC directly to the datalogger or on the local network, especially if the ADSL connection is to communicate with the remote server.

⚠ WARNING	Ethernet cable recommendations: The length and quality of your cable can affect the performance of the signal: • You need a straight cable. • Use a high-quality cable, at least a shielded twisted pair cable, cat. 5 (FTP Cat 5) or higher. • The max. permissible length for an Ethernet cable is 100 m per segment.

This section describes both ways of connecting a PC to the datalogger.

7.1. Connecting the PC directly to the datalogger

Ethernet

Datalog H60

Connect the Datalog H60 datalogger directly to the PC using the Ethernet port on the box and an Ethernet cable (crossover cable). To find your PC's Ethernet port, please refer to the documentation supplied with the PC.

7.2. Connecting the PC to the datalogger on a local network

Ethernet

Datalog H60   Access point / hub
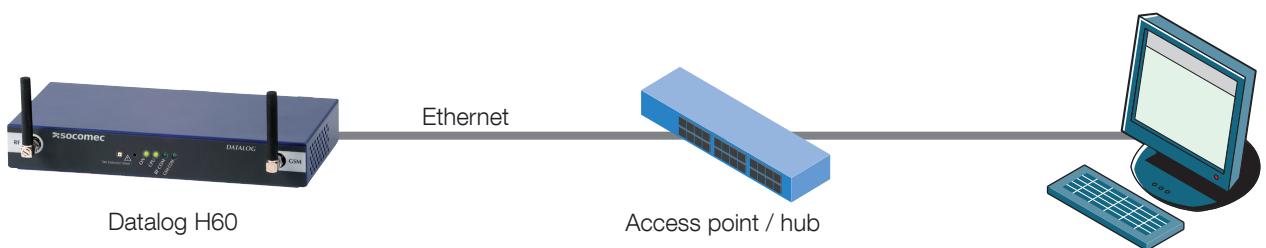
Connect the Datalog H60 datalogger to the local network using the Ethernet port on the box and an Ethernet cable (straight cable). To find the Ethernet port on your access point/hub, please refer to the documentation supplied with it.

7.3. Configuring the PC network to access the datalogger

Whether connecting the PC to the datalogger directly or over the local network, you need to configure a fixed IP address on the PC in the same IP address range and in the same subnet as the Datalog H60 datalogger.

⚠ WARNING	**The Datalog H60 datalogger's default settings are as follows:** IP address: 192.168.1.12 Subnet mask: 255.255.255.0 DHCP: Disabled

⚠ WARNING	**Network administrator** If your local network is managed by a network administrator, contact them before adding the Datalog H60 datalogger to your network.

In the next step you can configure a PC's network address to access the Datalog H60 datalogger:

Configuring a second IP address on the PC:
- 1. On Windows (2000 or XP), click Start/Settings/Network Connections then proxy settings. The Network Connection and Proxy Settings window appears.
- 2. Right-click on Local Area Network then click Properties.
- 3. Select Internet Protocol (TCP/IPv4) then click Properties.
- 4. Click Advanced.
- 5. In the IP Address pane, click Add.
- 6. Enter the IP address 192.168.1.xxx (xxx between 1 and 254 and not 12) and subnet mask 255. 255. 255.0.
- 7. Click Add.
- 8. To confirm your settings, click OK in each of the three windows.
- 9. Close the Network Connection and Proxy Settings window.

8. CONNECTING THE POWER SUPPLY

Connect the power supply only once all the other devices are connected to the Datalog H60 datalogger.

Make sure the following connections (if used) are correct before powering the datalogger:

• On/Off output.

• On/off inputs.

• RS485 input used to connect devices.

• RS232 input used to connect devices.

1-Connect the cable on the 12/24V input to the datalogger's screw-out connector, making sure the polarity is correct.

2-Connect the power supply.

⚠ WARNING	**Start-up phase of the Datalog H60 datalogger:** When you connect the power unit, the POWER LED lights up. The datalogger takes around 20 seconds to start up. When the CPU LED flashes every 0.5 seconds, it is ready.

⚠ WARNING	**About the power supply to use:** The power supply should deliver between 12 and 24V at 15W.

9. STARTING UP THE DATALOGGER

Set up the Datalog H60 datalogger using the on-board web interface.

9.1. Connecting to the on-board web interface

1- Open your web browser. The web interface is compatible with the latest versions of these browsers: Firefox, Chrome and Internet Explorer. Older versions (e.g. IE 7) may work but are no longer supported.

2- Type in the URL of the Datalog H60 datalogger homepage in your browser: http ://192.168.1.12.

3-The following window appears:
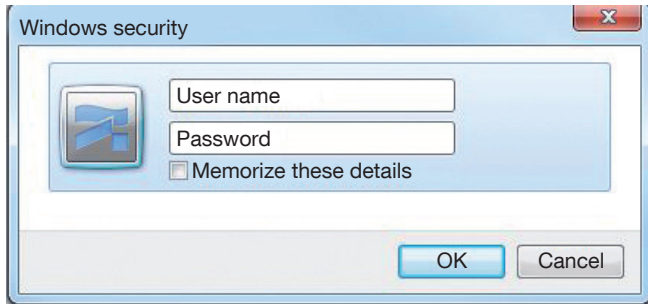
⚠ CAUTION	The Datalog H60 datalogger's default settings are as follows: User name: admin Password: high

4. The home page appears, as shown here:

If the datalogger is not yet ready for use the following message appears:

⚠ CAUTION	If you try to access the web pages during the datalogger's initialization phase, an alert message is displayed at the top of the page, e.g: "The datalogger is initializing. ". Wait until the datalogger is completely initialized to access the web pages.

9.2. Configuring the datalogger's connectivity

The Connectivity page lets you configure the datalogger so it can communicate with the remote server.

9.2.1. Configuring the modem connection

Modem

PIN Mode:	Off
PIN Code:	0000
Call Number:	*99***1#
APN:	
Login:	
Password:	
Mode:	OnDemand
Disconnect delay:	60

——————— Caller ID Whitelist ———————

Caller ID:	

Parameters	Description
PIN Mode	Off: The PIN code of the SIM card must be disabled
	Manual: Enter the PIN code of the SIM card if PIN Code is On
	Automatic: not in use
PIN Code	Enter the SIM card PIN code if Manual is selected under PIN Mode
Call Number	GPRS phone number. The default number is usually *99***1#.
APN	APN name of your mobile operator
Login	APN user name of your mobile operator
Password	APN password of your mobile operator
Mode	OnDemand: The datalogger only establishes the connection if it can communicate with the remote server. It cuts the connection if the data transfer is ended after the time set under Disconnect delay.
	AlwaysOn: not in use
	AlwaysOff: Use this mode if you are connecting to the remote server via Ethernet. This connection is never done via the modem, but with a valid SIM card the datalogger can receive incoming calls or SMSs and send SMSs.
	Off: The modem is always off. Use this mode if you are connecting to the remote server via Ethernet without the option to receive incoming calls or SMSs / send SMSs.
Disconnect delay	Value in seconds of the timeout mode OnDemand between the end of the data exchange and the disconnection.
Caller ID	List of authorised phone numbers for incoming calls and SMSs.

⚠ WARNING	Ask your mobile operator for the data (APN, login, password) relating to your particular SIM card.

9.2.2. Configuring the Ethernet connection

Ethernet

IP:	192	•	168	•	1	•	12
Netmask:	255	•	255	•	255	•	0
Gateway:		•		•		•	

☐ Use DHCP

—————— DNS ——————

DNS servers: [ ] • [ ] • [ ] • [ ]

Parameters	Description
IP	Enter the IP address at which the Datalog H60 can be accessed.
Netmask	Enter the subnet mask of your network. This mask limits the Ethernet network and the defined IP addresses to separate the network ranges from the one another.
Gateway	Enter the gateway address of your network. The gateway address is the IP address of the device that establishes the Internet connection. This is usually the same as your ADSL router's address.
Use DHCP	You can get the Ethernet settings automatically if permitted by the network infrastructure. In this case select dynamic mode and refer to your DHCP server configuration for your gateway's IP address.
DNS servers	The DNS (Domain Name System) server translates the logical Internet addresses (e.g. www. SOCOMEC.com) into corresponding IP addresses. Enter the addresses of the DNS servers that you received from your ISP. You can also enter your router's IP address.

9.2.3. Configuring the remote FTP server

This field tells you the remote FTP server settings for exchanging data.

FTP

Address: _____
Login: _____
Password: _____
Mode: Passive ▼
☐ Use FTPS
Root: /
WS notification: None ▼

Parameters	Description
Address	IP address or name of the remote FTP server
Login	User name used by the datalogger to connect to the remote FTP server
Password	Password used by the datalogger to connect to the remote FTP server
Mode	Active or passive
Use FTPS	Select if the remote server is an FTPS server
Root	Root path on the remote FTP server
WS notification	None: No Web Services notification
	Put: Web Services notification when the datalogger puts data on the FTP server
	Get: Web Services notification when the datalogger downloads data from the FTP server
	Both: Put+Get

9.2.4. Timezone settings

Time

Timezone: _____
Alarm threshold (s): 0
─────────── NTP ───────────
NTP servers: _____

Parameters	Description
Timezone	Text field to enter the timezone. Respect the standard TZ format, see:
Alarm threshold(s)	Difference in seconds between the datalogger time and the NTP synchronisation time, from which an alarm is triggered
NTP servers	IP addresses of NTP servers used to synchronise the datalogger's clock

9.2.5. Configuring data transfers

The datalogger can use 4 types of data on the remote server:
• Datalogger's configuration data.
• Datalogger's supervision data.
• Alarms
• RF data, Modbus, Index.

The datalogger can use the FTP method for each type of data.

Upload

Configuration

Method: FTP

Supervision data

Method: FTP

Alarms

Method: FTP

Data

Method: FTP

Format: CSV

Schedule:

The data format is XML.

Configure data logging by entering the scheduler's unique identifier.

9.2.6. Request button

Request button

Connection request: ☑

Upload status: ☑

SMS Status recipient:

By default, pressing the "Request" button on the front of the device starts a connection to the remote server, downloads the recorded data and files a status file. You can turn these options off.

The system can also send a status SMS to a specified recipient.

9.3. System configuration

9.3.1. Configuring ports

The datalogger's ports can be configured as follows:

Port	Values	Description
rs232	Off Mbus	RS232 disabled RS232 enabled in MBus mode
rs485	Off Modbus	RS485 disabled RS485 enabled in MBus mode
Input #1/mode	Digital input Pulse	Digital input mode Pulse input mode
Input #2/mode	Digital input Pulse	Digital input mode Pulse input mode
Input #3/mode	Digital input Pulse	Digital input mode Pulse input mode

Settings for the RS485 port are:

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

9.4. Configuring alarms

The datalogger can generate alarms:

• System alarms
• On/Off input alarms
• etc.

9.4.1. System alarms

There are 5 types of system alarms:

• Power: alarm generated when power is lost and recovered.

• Modem IP: alarm generated if the datalogger's IP address changes.

• MSISDN: alarm generated if the SIM card in the datalogger changes.

• SW Version: alarm generated if the datalogger's firmware version changes (after an update)

• Default.

Each alarm source can be enabled individually and transferred immediately to a remote server (On) or at the next connection (Delayed).

The datalogger also generates default alarms, which have the following codes:

Code	Description
D_MODEM	**Modem default**
D_ETHERNET	Ethernet interface default
D_INTERNAL_BAT	Internal battery default

The datalogger immediately transfers each default to the remote server.

Select Ignored to list the default codes ignored by the datalogger. If you are entering multiple default codes, separate them with the character ',' (comma).

Select Delayed to list the default codes the datalogger will send at the next connection. If you are entering multiple default codes, separate them with the character ',' (comma).

9.4.2. On/Off input alarms

You can configure multiple alarms for digital inputs.

You can configure an alarm for a digital input as follows:

Name	Description
Index	Digital input index
Label	Alarm name (for reference only)
Mode	On: Immediate transmission Off: Transmission disabled Delayed: Send at next connection
Type	None: Scan disabled Raising: Front scan increasing Falling: Front scan falling Both: Raising+Falling

9.4.3. On/Off output alarms

You can configure an alarm for the digital output.

Name	Description
Label	Alarm name (for reference only)
Mode	On: Immediate transmission Off: Transmission disabled Delayed: Send at next connection
Type	None: Scan disabled Raising: Front scan increasing Falling: Front scan falling Both: Raising+Falling

9.5. Scheduler settings

The scheduler manages all the scheduled tasks.

The scheduler configuration consists of a list of schedules.

Each of these schedules has a unique ID which is used to link a task to a specific schedule. They can be used independently to initiate data collection and downloads.

Each schedule is configured as follows:

Name	Description
Id	Unique schedule identifier
Label	Reference name of the schedule
Type	Daily, Weekly, Monthly, Yearly or Follower: see the description below
Time	Time of first occurrence (not used for Yearly schedules)
Day of Week	Number of the day in the week of the time of first occurrence (1=Monday, 7=Sunday) (used only for Weekly schedules).
Day of Month	Number of the day in the month of the time of first occurrence (used only for Monthly schedules).
Date & Time	Date and time of first occurrence within a given period (used only for Yearly schedules)
Interval	Interval between occurrences (in seconds)
Count	Number of occurrences
Parent	Link to parent schedule for a Follower schedule.

Configuring the various types of schedule:

Daily schedule:
Every day, the first occurrence T_0 is given by the time entered under Time.
The format is as follows: HH:MM:SS. For example, 09:30:00
The following occurrences use the time T_i:

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

Δt is the value in seconds entered under Interval.

Weekly schedule:
Every week, the first occurrence T_0 is given by the day of the week entered under Day of week and the time entered under Time.
The following occurrences use the time T_i:

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

Δt is the value in seconds entered under Interval.

Monthly schedule:
Every month, the first occurrence T_0 is given by the number of the day of the month entered under Day of month and the time entered under Time.

The following occurrences use the time T_i:

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

Δt is the value in seconds entered under Interval.

Yearly schedule:
Every year, the first occurrence T_0 is given by the date entered under Date & Time.
The format is as follows: YYYY-MM-DDTHH:MM:SS.
For example, for a first occurrence on 11 February 2012 at 13H00:
Time = 2012-02-11T13:00:00.

The following occurrences use the time T_i:

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

Δt is the value in seconds entered under Interval.

Follower schedule:

A 'Follower' schedule is produced after the end of each reference schedule occurrence. The Parent schedule cannot be a 'Follower'.

With this type of schedule you can trigger, for example, a data download after the data collection is complete.
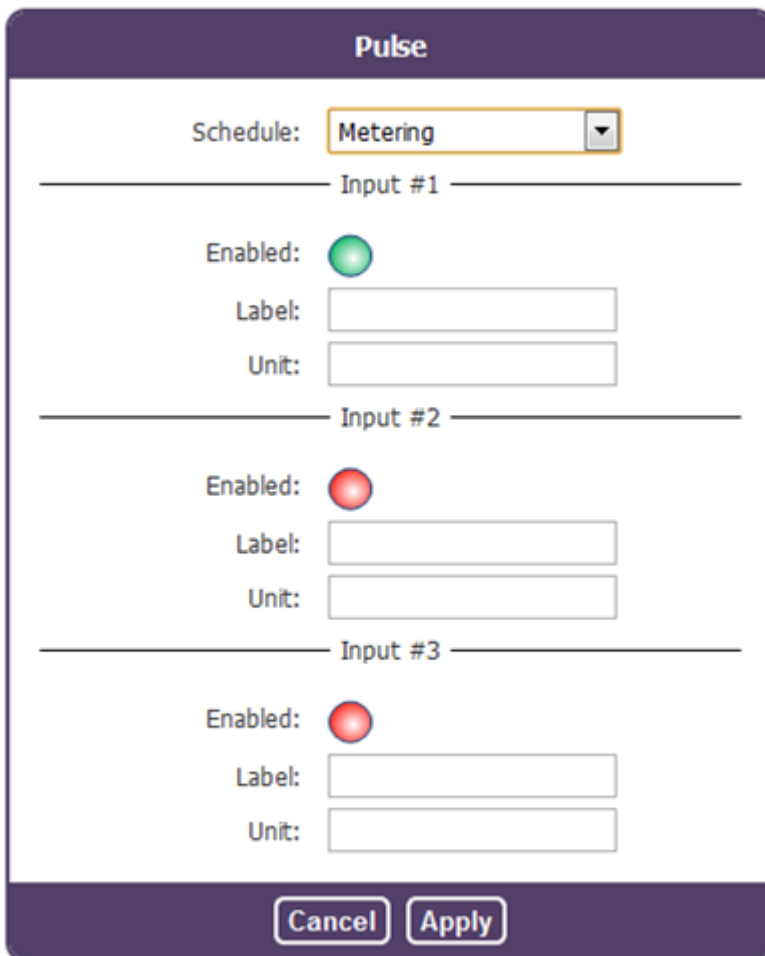
Example:

You want to collect data once a day at midnight and download data just after. You can configure a Daily schedule to collect the data and a first, Follower schedule to download the data.

Examples:

Need	Type	Time	Day of Week	Day of Month	Date & Time	Interval	Count
Every Tuesday at 15:00:00	Weekly	15:00:00	Tuesday			0	1
Every 2nd of the month at 00:00:00	Monthly	00:00:00		2		0	1
Every day at 14:00:00	Daily	14:00:00				0	1
Every hour between 8:00 and 18:00 every Tuesday	Weekly	08:00:00	Tuesday			3600	11
Every 2 hours between 8:00 and 20:00 on 31 December	Yearly				2012-12-31T08:00:00	7200	7

9.6. Metering configuration

9.6.1. Pulse inputs
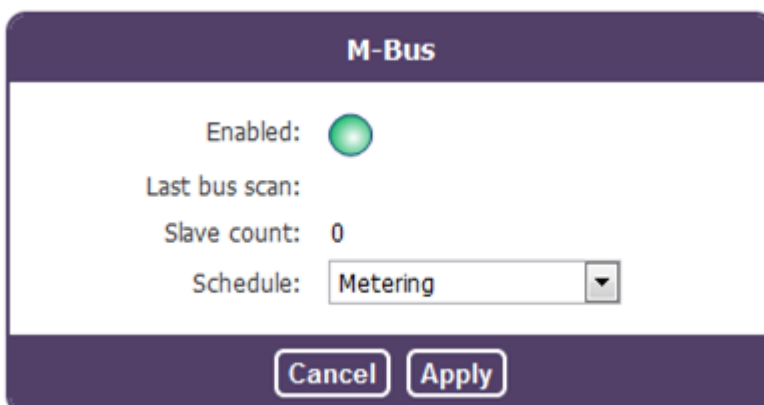
The digital inputs can be configured as pulse counters (see the relevant section). Once the input is activated, an allocated meter will count each pulse of more than 10 ms. The current value is saved for each occurrence of the specified schedule.

The Label and Unit settings are added to the recorded data along with their index value.

9.6.2. M-Bus

If the RS232 port is configured in MBus mode, the acquisition can be activated with the associated schedule.

9.6.3. Wireless Mbus 868MHz.

In Wireless M-Bus radio card version, the Datalog H60 datalogger can receive data from known Wireless M-Bus modules (filtering enabled or not (filtering disabled).
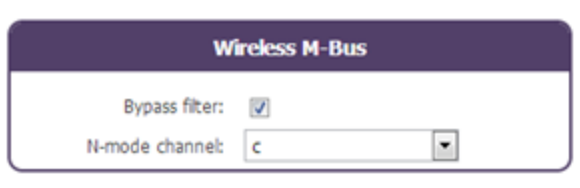
With this interface you can enable or disable module filtering and select the channel mode N.

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.

Extract from the standard EN 13757-4

If filtering is enabled, configure the modules by using the following interface:

Name	Description
Address	Wireless M-Bus address
Label	Module name (for reference only)
Key	Module encryption key (if empty, no encryption)

9.7. Modbus settings

Name	Description
	RTU
Timeout (ms)	Timeout of the Modbus/RTU response in ms
Turnaround (ms)	Time the Modbus/RTU takes to turn around in ms
	TCP
Timeout (ms)	Timeout of the Modbus/TCP response in ms

As well as these settings, you must set the port to "modbus"

Configuring a dataset involves setting the following parameters:

Name	Description
Id	Unique identifier for Modbus dataset
Label	Name of the dataset (for reference only)
Polling	Polling continue (true or false)

Each variable is defined by the following parameters:

Name	Description
Name	Name of the variable (for reference only)
Type	Variable type (S0, S1, S3, S4)
Address	Extended registry address 16-bit
Size	Size in bits for discrete input and coil, in octets for registries
Format	Raw, boolean, integer, float or ascii
Flags	cmd_only, little_endian, no_opt, is_status or is_alarm
Threshold low	Low threshold
Threshold high	High threshold
Threshold hysteresis	Hysteresis applied to two thresholds

See the user guide for details on how to configure data.
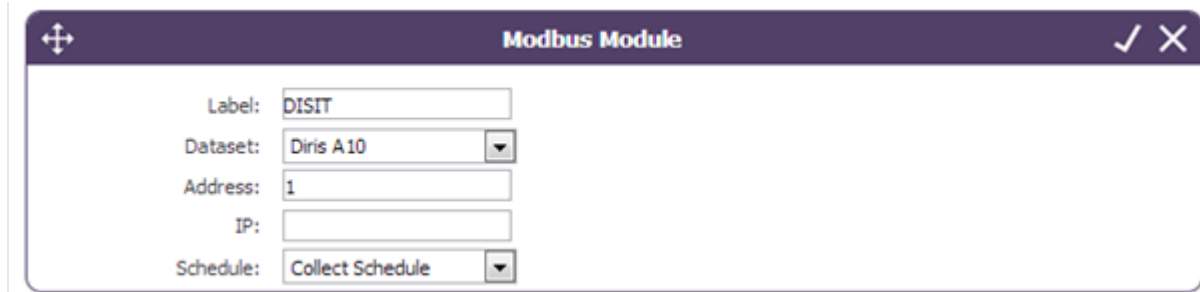
Configuring a Modbus module includes the following parameters:

Name	Description
Label	Name for reference only
Dataset	Dataset identifier
Address	Modbus address (1-247)
IP	IP address (empty for RTU devices)
Schedule	Schedule identifier

Press [✓] to check the Modbus module is correctly configured:

Modbus Module

Label: DISIT
Dataset: Diris A10
Address: 1
IP:
Schedule: Collect Schedule
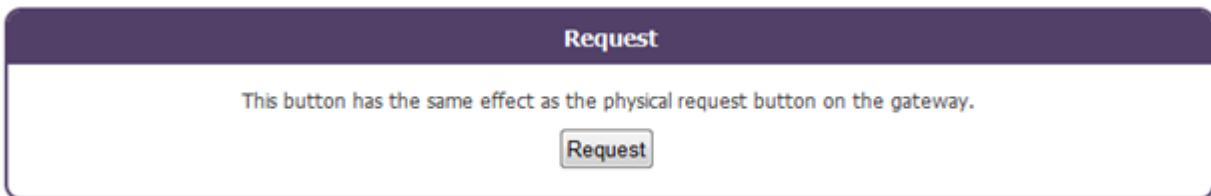
The results of the module Modbus query appear in a separate pane:

```
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: U31, address: 0xc556, type: S4, value: 0
```

[Close]

9.8. Possible actions

9.8.1. Request connection to remote server

Request

This button has the same effect as the physical request button on the gateway.

[Request]

This button does the same as the physical button on the front of the device. A popup appears showing all the stages of connection (including NTP synchronisation and checking the Inbox directory) and listing all downloaded files.

9.8.2. Request M-Bus scan

Use this button to start scanning for M-Bus devices. The M-Bus devices discovered during this scan are prompted each time by the assigned scheduler. Start a new scan when devices are removed from or added to the bus so the datalogger can take the changes into account.

9.8.3. Request a reboot
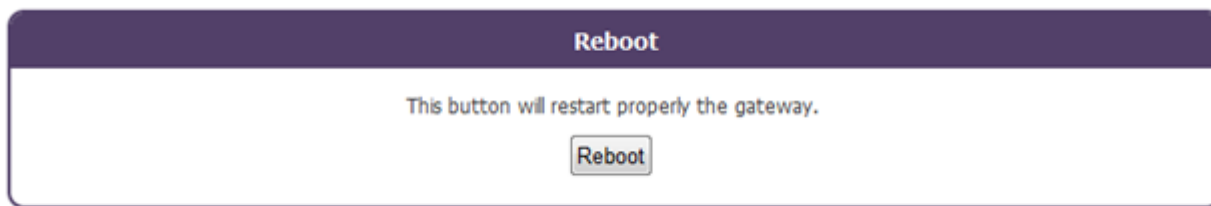
Use this button to reboot the datalogger properly:

9.8.4. Download system files
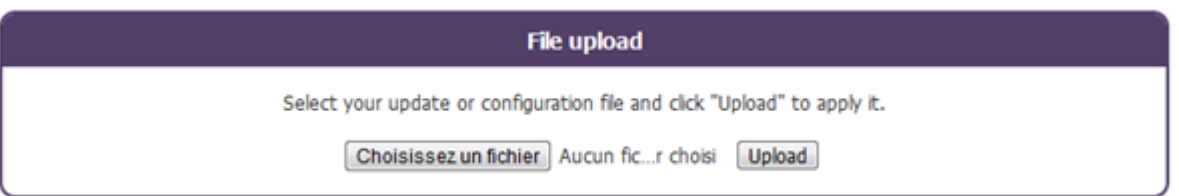
In this window you can download the configuration files or a new firmware locally onto the datalogger.

9.9. Request connection to remote server

Once you have made your settings, you can prompt a connection from the datalogger to the remote server by quickly pressing the REQUEST button.

The GSM LED lights up during the GSM connection.

10. CHECKING THE DATALOGGER IS WORKING PROPERLY

To check the datalogger and the radio mapping are working correctly, you can trigger a scan of the modules and how a Supervision file is registered.

There are two ways to do this:

- If you can access the remote server, save a scan command to the INBOX directory.

See below for an example of an xml scan:

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

This command actuates:

- A report on the clock, battery status, RSSI level and immediate data of each module.
- A supervision file directory containing such information (except immediate data).
- A data file directory containing immediate data.

Once you have saved this command file on the remote server in the INBOX directory, prompt a connection from the datalogger to the remote server by quickly pressing the REQUEST button.

- If you have no access to the remote server to trigger this operation, you can do it by SMS.

SMS to send to the datalogger:

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

The data requested (except for immediate data) is uploaded to the remote server as supervision data using the corresponding upload method you set. The immediate data (scan data) is uploaded to the remote server as data using the corresponding upload method you set.

Once the datalogger has this data, you can check the data extracted.

11. FTP LOCAL SERVER

The datalogger comes with a local FTP server, which acts as a local INBOX. You can store just one file on this server at a time.

It accepts configuration files and binary update files.

12. STOPPING THE DATALOG H60

Disconnect the power cable of the 12/24V connector.

The Datalog H60 datalogger features a backup battery. If power is lost it sends an alert to the server and continues to work until the battery runs empty.

13. COMMUNICATION WITH THE REMOTE SERVER

The datalog H60 datalogger communicates with a remote server via an FTP server and/or a Web Services server. Administration for Web Services server not in use

Whenever its settings are changed, the datalogger can either upload them to an FTP server or send them to a Web Services server (WS).

Alarms and collected data can also be uploaded to an FTP server or sent to a Web Services server.

If using the FTP transfer, the datalogger can also inform a Web Services server about any FTP upload.

The server can also run actions on the datalogger by placing command files in an INBOX directory on the FTP server or by sending it commands when the datalogger prompts the Web Services INBOX. The commands can also be sent to the datalogger by SMS.

13.1. Connection modes

The remote server connection can be established via Ethernet or a cellular network (GPRS or 3G depending on the hardware configuration). Exchanges between the datalogger and remote server are always initiated by the datalogger, but there are various ways in which the remote server can trigger an exchange.

The datalogger can be configured to use a modem in one of these four modes:

• **On demand**: In this mode, the PPP link is created when the datalogger needs to communicate with the remote server. The link is closed after communication with the remote server is established.

• **Always On**: In this mode, the PPP link is kept permanently on, in an independent state. In this mode, you can enable a keepalive mechanism to make sure the link is working. **Mode Always On not in use**

• **Always Off**: In this mode, the PPP link is never made. All communication with the server takes place via the Ethernet interface. However, the modem (if a valid SIM is inserted) is connected to the cellular network, ready to receive incoming calls and/or SMSs.

• **Off**: In this mode, the modem is off.

13.1.1. Managing the SIM card PIN code

You can configure the datalogger to use a SIM card:

• Without PIN code: /com/modem/pin/mode=off

• With PIN code: /com/modem/pin/mode=manual and /com/modem/pin/code=<CODE PIN>

• With automatic PIN code: Automatic mode not in use

13.2. Remote FTP server

The datalogger uses the following files on the FTP server:

Name	Description
CONFIG/<uid>.xml	Current configuration of the datalogger. The datalogger sends this file each time its configuration changes. Any change to these files has no effect on the datalogger. The datalogger just clears it the next time its configuration changes (see INBOX below).
DATA/<uid>-<timestamp>.<format>.gz	Data files uploaded by the datalogger onto the remote server.
ALARM/<uid>-<timestamp>.xml.gz	Alarm files uploaded by the datalogger onto the remote server.
SUPERVISION/<uid>-<timestamp>.xml.gz	Supervision files uploaded by the datalogger onto the remote server (scan statuses and results).
SUPERVISION/<uid>-<timestamp>.log.gz	Log files uploaded by the datalogger onto the remote server by request.
INBOX/<uid>/*.xml	The datalogger supervises this directory. Every file placed here is uploaded and handled by the datalogger.
BIN/<firmware>	This directory contains the datalogger's firmware to update the system

In the table above, <uid>, <timestamp> and <format> must be replaced by the datalogger's unique identifier, the upload's timestamp and the format (xml) respectively.

The timestamp format is "YYYYMMDD-HHMMSS" with an alphabetical sorting of the directory giving the chronological order.
Files with extension ".gz" are compressed.

The datalogger always uploads the files in a 2-step process:
• The file is uploaded with the extension ".tmp".
• The file is renamed without the extension ".tmp".

With this process, the remote server can easily distinguish the files being uploaded from those already uploaded.

⚠ WARNING	You can find the XML schema and an example configuration file in the appendix: - Alarm XML - Supervision XML - Command XML
⚠ WARNING	The XML schemas – specifying the format of the various XML files used by the datalogger – can evolve in future versions when new features are added. These changes will be made so that the old XML files are compatible with the new XML schemas. Similarly, as the XML files generated by the datalogger can contain additional elements, they should be handled in such a way that the new elements can be ignored.

13.3. Remote Web Services server

Not in use

13.4. Connecting to the remote server

The following events prompt a connection to the remote server:
• Connection scheduler
• An alarm
• Changes to the configuration
• A connection-request SMS
• REQUEST button (via the web pages or the button on the front panel)

Whatever the event that prompts the connection, the process is the same:
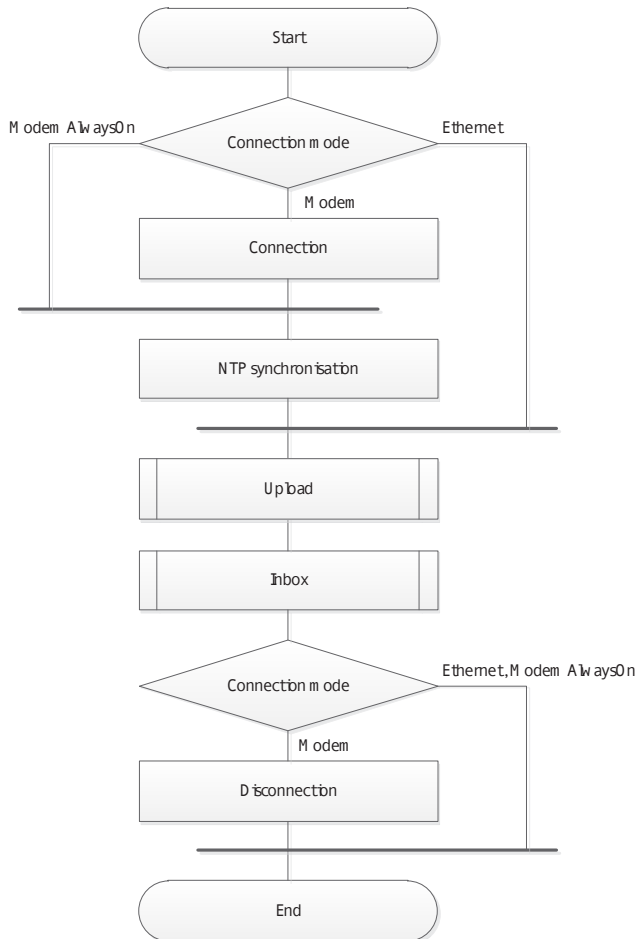
If the GPRS connection fails, a new connection is attempted an hour later, unless a new connection is launched in the meantime (by explicit request or triggered by an alarm or scheduled transmission). This is illustrated below, specifying the disconnection time:
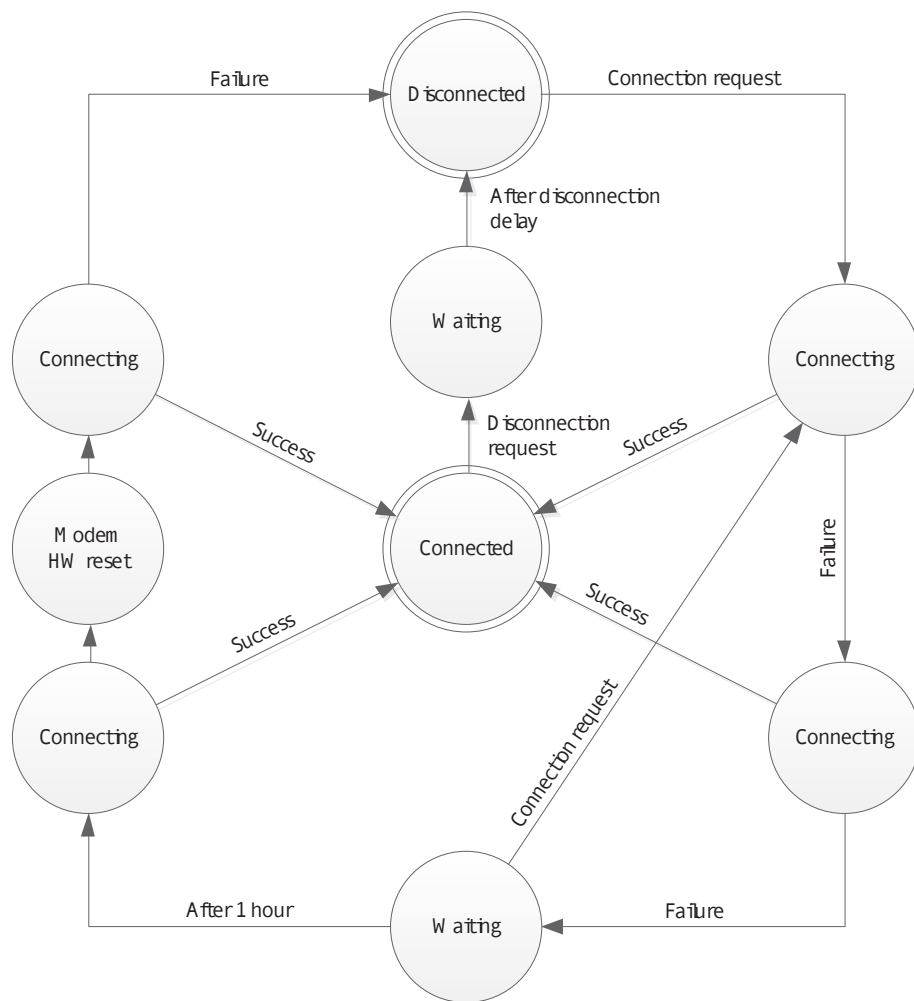
13.4.1. Uploading to the remote server

Configuration, alarms, supervision and data are independently uploaded to the remote server by the datalogger, as described in the schema below.

In this diagram, X represents the type of upload (configuration, alarm, supervision or data).

After a configuration upload is complete, the associated flag is deleted.

After a alarm/supervision/data upload is complete, the associated data is deleted.

```
                        ┌─────────────┐
                        │    Start     │
                        └──────┬──────┘
                               │
                          ◇ Upload X pending ? ◇ ───── no ──────────────┐
                               │                                         │
                              yes                                        │
                               │                                         │
              FTP ──── ◇ X upload method? ◇ ──── WS                      │
               │                               │                         │
        ┌──────────────┐              ┌──────────────┐                   │
        │  FTP upload   │              │   WS call     │                   │
        └──────┬───────┘              └──────┬───────┘                   │
               │                             │                           │
        ◇ WS Notification ? ◇ ── false ──┐   │                           │
               │                         │   │                           │
             true                        │   │                           │
        ┌──────────────┐                 │   │                           │
        │ WS notification │               │   │                           │
        └──────┬───────┘                 │   │                           │
               └─────────────────────────┴───┘                           │
                               │                                         │
                          ◇ X uploaded ? ◇ ──── no ──┐                    │
                               │                     │                    │
                              yes                    │                    │
        ┌────────────────────────────┐              │                    │
        │ Clear X flag or associated data │           │                    │
        └──────┬─────────────────────┘              │                    │
               └──────────────────────────────────────┘                  │
                               │                                         │
                        ┌─────────────┐                                  │
                        │     End      │──────────────────────────────────┘
                        └─────────────┘
```
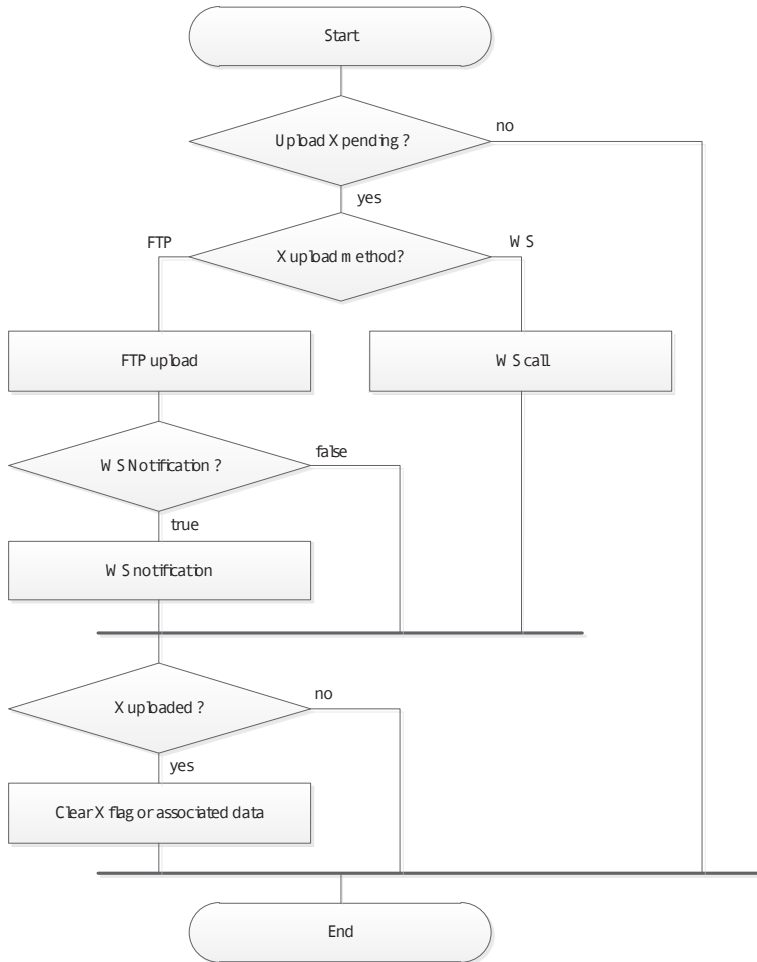
Note: WS mode administration is not implemented

13.4.2. Inbox

The datalogger checks the queuing actions as follows:

```
                        ┌──────────────────┐
                        │      Start       │
                        └────────┬─────────┘
                                 │
                    FTP      ◇ Data Mode ?      W S
              ┌───────────────◇        ◇───────────────┐
              │                 ◇    ◇                  │
    ┌─────────┴─────────┐              ┌────────────────┴──┐
    │  Check FTP Inbox  │              │  Check W S Inbox  │
    └─────────┬─────────┘              └─────────┬─────────┘
              │                     no            │
         ◇ Pending command ? ◇──────────     ◇ Pending commands ? ◇
              │  yes                               │  yes
    ┌─────────┴─────────┐                          │
    │      FTP Get      │                          │
    └─────────┬─────────┘                          │
              │                                     │
         ◇ W S notification ? ◇──── false           │
              │  true                               │
    ┌─────────┴─────────┐                          │
    │  W S notification │                          │
    └─────────┬─────────┘                          │
              └──────────────┬────────────────────┘
                    ┌────────┴─────────┐
                    │  Process commands │
                    └────────┬─────────┘
                    ┌────────┴─────────┐
                    │       End        │
                    └──────────────────┘
```
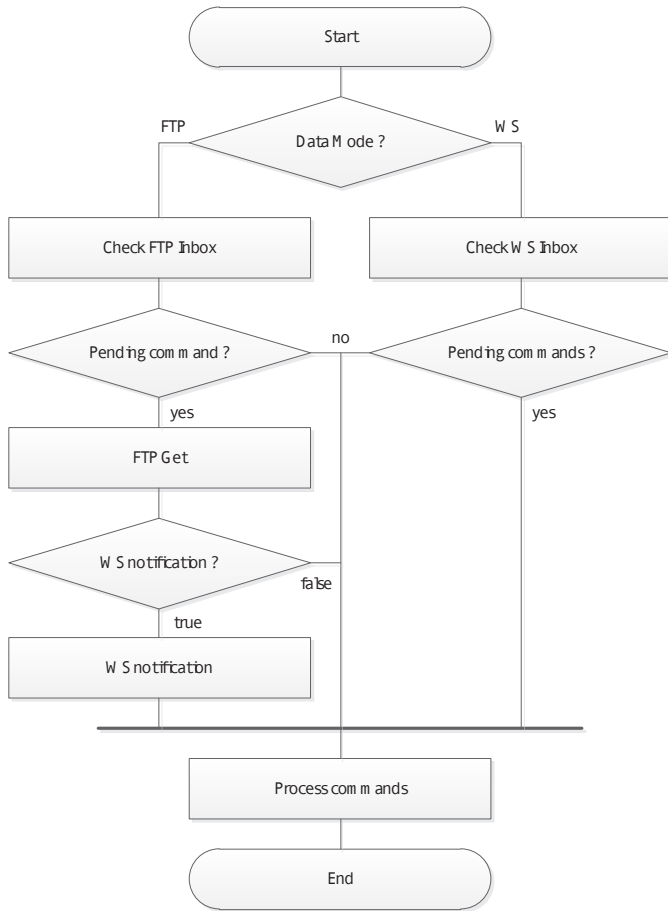
13.5. Request button

By default, pressing the "Request" button starts a connection to the remote server and uploads the datalogger's status as well as the queuing data. These actions can be disabled individually in the configuration settings:

/com/request/upload
and
/com/request/include_status.

The system can also send a status SMS to a specified recipient under the setting:
/com/request/sms_status_recipient
If the field is empty, no SMS will be sent.

14. COMMANDS

Commands can be sent via the remote server (FTP or WS) or via SMS.

When the datalogger receives an SMS, it checks the whitelist of authorised phone numbers (caller_id). If authorised, the contents of the SMS are handled.

Command	Description	Restart
reboot	Restarts the datalogger.	None
factory	Restores the datalogger's default settings	None
update	Starts a firmware update for the datalogger	Alarm (SW)
scan	Starts a scan of the instantaneous values of Radio W-Mbus modules, their RSSI levels, their battery levels and/or their RTC clocks	Supervision data (not scan data)
timesync	Updates the radio module RTC clocks	Supervision data
status	Requests datalogger status	Supervision data
log	Requests log data	Log data
config	Changes datalogger settings (SMS only)	Saves configuration
connect	Connects to the remote server (SMS only)	Implicit (connection)

Commands are not acknowledged on receipt. All commands are recorded and an invalid command triggers an alarm, which is uploaded to the remote server.

All commands accept two optional parameters "uid" and "cid":
• uid: datalogger's unique identifier
• cid: command's identifier

A command is rejected if the included uid parameter does not match the datalogger's uid.
The cid can be freely chosen by the issuer of the command. It is included with all associated uploads.

You can find the schema and an example XML of the commands in the appendix.

14.1. Specific Modbus commands

Specific Modbus commands can be sent to a known Modbus module by using the command "modbus".

Here is a list of supported commands:

Sub command	Description
Write	Write a value to a Modbus device

Format the variable address as explained in Section 11.4.

14.2. Status command

Example: Request gateway status:

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

SMS:
cmd=status
cid=C_1238
```

The following information is sent back to the requester:

XML name	SMS name	Description
-	**uid**	Datalogger's unique identifier
/app/version	**version**	Datalogger's software version
/app/kernel	**kernel**	Linux kernel version.
/system/power	**power**	External power source present (boolean)
/system/defaults	**defaults**	Default codes separated by commas
/com/modem/model	**m_model**	Name of modem model
/com/modem/firmware	**m_version**	Name of modem firmware
/com/modem/imei	**imei**	International Mobile Equipment Identity
/com/modem/msisdn	**msisdn**	Mobile Subscriber ISDN Number (if available)
/com/modem/rssi	**rssi**	Strength of signal received in dBm
/com/modem/csq	**csq**	Signal quality (CSQ, BER)
/com/modem/ip	**m_ip**	Datalogger's IP address on the modem interface (or last allocated address).
/com/ethernet/ip	**e_ip**	Datalogger's IP address on the Ethernet interface
/com/upload/last	**u_last**	Date of last successful connection to the remote server (scheduled or prompted)
/com/upload/next	**u_next**	Date of next scheduled connection to the remote server

If the status command is sent by SMS, the status is returned in a multiple SMS with one variable per line (name = value).

If the status command comes from the INBOX (FTP or WS), the XML file is downloaded in the form of a supervision xml file.

14.3. Update command

Update command for the datalogger firmware:

```
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
```

The firmware file must be in the BIN directory on the FTP server. The checksum corresponds to the checksum md5 of the file.

14.4. Dry contact command (digital output)

Example: Open a dry contact on the datalogger (digital output):

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

15. DATALOGGER SETTINGS

15.1. Parameters

The datalogger parameters are handled in a structured manner. The configuration can be exported to an XML file.
A new configuration and changes to the current configuration are installed using an XML file with the same format.
The XML configuration schema is shown in Section 14.
An example XML file is shown in Section 15.

The main parameters of the datalogger are listed below (factory default values are in bold):

Name	Value	Description
/uid	**6 last digits in the Ethernet MAC address**	Datalogger's unique identifier
/name	**MAC address prefixed with "WGRF_"**	Datalogger name (for reference only)
/enable_local_config	false, **true**	Activation/deactivation of local configuration not in use
/com/modem/pin/mode	**off**, manual, automatic	Administration mode for the SIM card PIN code (see Section 3.2) automatic not in use
/com/modem/pin/code	**0000**	SIM card PIN code
/com/modem/call_number	***99***1**#	GPRS call number
/com/modem/apn		APN
/com/modem/login		APN login
/com/modem/password		APN password
/com/modem/mode	**ondemand**, alwayson, alwaysoff, off	See the description of these modes in Section 3.1). alwayson not in use
/com/modem/delay	**60**	Delay in seconds before disconnected in ondemand connection mode not in use
/com/modem/whitelist/caller_id		Whitelist of authorised caller numbers to receive command SMSs. If empty, no verification
/com/ethernet/use_dhcp	**false**, true	Activation/deactivation of DHCP client
/com/ethernet/ip	**192.168.1.12**	Datalog H60 IP address
/com/ethernet/netmask	**255.255.255.0**	IP network mask
/com/ethernet/gateway		Network gateway IP address
/com/ethernet/dns/server		DNS server IP address
/com/keepalive/method	icmp, tcp, **off**	Keepalive mode (not in use)
/com/keepalive/address		not in use
/com/keepalive/port	**5000**	not in use
/com/keepalive/period	**600**	not in use
/com/keepalive/timeout	**30**	not in use
/com/request/upload	false, **true**	Connect to remote server by pressing Request button
/com/request/include_status	false, **true**	Load the datalogger's status on the remote server by pressing the Request button
/com/request/sms_status_recipient		SMS recipient of status sent by pressing Request
/com/time/ntp/server		NTP server address
/com/time/timezone		Local timezone (uses the standard zoneinfo name, e.g. "Europe/Paris")
/com/time/alarm_threshold	0	Desynchronisation alarm threshold in seconds (0=off).
/com/ftp/address		FTP server address

Name	Value	Description
/com/ftp/login		FTP login
/com/ftp/password		FTP password
/com/ftp/mode	**passive**, active	FTP mode
/com/ftp/secured	**false**, true	Activation/Deactivation of FTP secure protocol FTPS not in use
/com/ftp/root_path	/	Root path on the FTP server
/com/ftp/ws_notification	**none**, put, get, both	WS notification mode for uploading FTP files
/com/ws/address		WS address - *ws not in use*
/com/ws/login		WS login- *ws not in use*
/com/ws/password		WS password - *ws not in use*
/com/ws/secured	**false**, true	SSL/TLS (HTTPS) activation/deactivation for WS - *ws not in use*
/upload/config/method	**ftp**|ws|none	Method used to upload configuration - *ws not in use*
/upload/alarm/method	**ftp**|ws	Method used to upload alarms - *ws not in use*
/upload/supervision/ method	**ftp**|ws	Method used to upload supervision data - *ws not in use*
/upload/data/method	**ftp**|ws	Method used to upload data
/upload/data/format	xml|**csv**	Format used to upload data
/upload/data/schedule		ID of the schedule used to upload data
/alarm/*		Alarm engine settings
/scheduler/*		Scheduler settings
/metering/*		Metering settings
/rfid/*		Active RFID settings
/system/log/level	7	Log level
/system/password/ admin	high	Password to access HTTP services and local FTPs.
/system/password/ install	medium	
/system/password/data	low	
/system/ports/*		

The configuration can be modified locally or remotely. Any change prompts the new configuration to be uploaded onto the server.

15.2. About telephone numbers

/com/modem/whitelist
      If the list is empty, all numbers are considered valid.
/com/modem/whitelist/caller_id
/com/request/sms_status_recipient

Phone numbers must be written out in international format.
They must start with + followed by the country code.

15.3. Configuring by SMS

The initial configuration of the datalogger can be made by SMS, especially the connection settings. Once this initial configuration is completed, the remote server can complete its configuration of the gateway.
The first line of the SMS should contain the command "CMD=config".
The next lines must have the format "SHORTNAME=value". The SHORTNAME is formed of the first letters of each element of the parameter name:

For example, the SHORTNAME of "com/modem/login" is LMC.

The SMS content is subject to the following rules:

• Spaces at the end of a line are ignored.

• SHORTNAMES are not case sensitive.

• The Boolean values, true and false, may be replaced by 0 and 1 respectively.

• The carriage return may be replaced by a semicolon, but they cannot both feature in the same SMS.

• If using the character ';' as separator between variables, you cannot use this character in the variable values.

The length of an SMS message is limited to 160 characters.

You can change only the main settings and communication settings by SMS:

• /uid

• /name

• /enable_local_config

• /com/

Example:

To complete the initial configuration of a datalogger in the following context:

• APN "m2minternet" not requiring an identifier/password.

• Communication with the FTP remote server (168.112.23.123) in passive mode.

You can send the following SMS:

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

Note: All parameters using their default values have been omitted.

On receipt of this SMS, the datalogger applies the settings and connects to the remote server to upload the resulting configuration file. From this point, the datalogger can be configured remotely as described below.

To build a list (e.g. /com/modem/whitelist/caller_id), the variable must be repeated in the SMS. If the variable appears at least once, the current list is replaced. If it appears only once and without value, the current list is cleared.

Examples:

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

After handling this SMS, the gateway uses the server DNS 1.2.3.4.

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

After handling this SMS, the gateway uses the 2 servers, DNS 1.2.3.4 and 1.2.3.5.

```
CMD=config
CTNS=
```

After handling this SMS, the datalogger will not use any DNS server.

15.4. Local configuration

You can configure the datalogger locally via a Web interface:

15.5. Remote configuration

The remote server can change the configuration by placing an XML configuration file in the INBOX directory on the FTP server or with a Web service notification. The format XML is used in both cases.

WS not in use

The XML file is treated as a new configuration if the factory XML attribute is present and equal to true.

⚠ **WARNING**   The partial attribute is still supported but obsolete. The factory attribute equal to true is equivalent to the partial attribute to false.
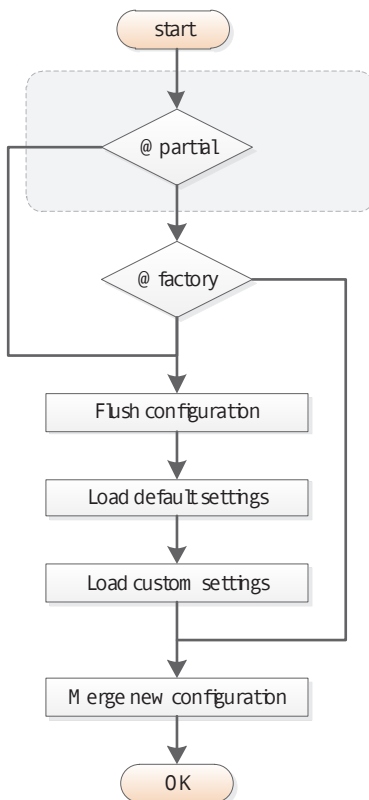
If factory is not present or equal to false, only the values of the configuration settings in the new configuration file are updated.
If a list is present in the new configuration file, the entire list is replaced. This is especially the case for schedules.

15.6. Controlling local access

Access to HTTP services and local FTPs is login/password protected. Any attempt at using these services is recorded.
There are three access levels: "admin", "install" and "data".
The administrator (admin) has read/write access rights to all configuration settings, read rights for the datalogger status information and can trigger actions.
The installer (install) has read/write access rights to the configuration settings of the final modules.
The data user (data) only has read access rights to the datalogger status information.

	Datalogger set-tings	Datalogger status	LAN settings	Actions
admin	R/W	R	R/W	Yes
install		R	R/W	Yes
data		R		No

Access level passwords are configured in /system/password. They may be modified only by a configuration file from the remote server file or locally by the administrator.

⚠️ **WARNING**   We strongly recommend changing your password before use.

Password restriction: must not contain the characters: «, &, ', <, >, ?, `.

Configuration files uploaded to the local FTP by the installer (install) are rejected if they contain non-LAN configuration parameters.

15.7. Configuring ports

The following settings in /config/system/ports are used to configure the ports' operation modes:

Name	Value	Description
rs232/mode	**off**, mbus	Mode RS232
rs485/mode	**off**	Mode RS485 (see below)
input_1/mode	**d_input**, pulse	Digital input mode
input_2/mode	**d_input**, pulse	Digital input mode
input_3/mode	**d_input**, pulse	Digital input mode

The RS485 port settings in /config/system/ports/rs485/ are:

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

Note: Modbus specification stipulate that if no parity bit is used, 2 stop bits must be used.

16. PULSES

The digital inputs can be configured as pulse meters (see the relevant section).

In pulse meter mode, an allocated counter counts each pulse of more than 10 ms. The current value is saved for each occurrence of the specified schedule. The following settings are configured under /config/metering/pulse:

Name	Description
Schedule	ID of the schedule for getting pulses
input_1/label	Input name (for reference only)
input_1/unit	Unit (and weight) of pulse (for reference only)
input_2/label	Input name (for reference only)
input_2/unit	Unit (and weight) of pulse (for reference only)
input_3/label	Input name (for reference only)
input_3/unit	Unit (and weight) of pulse (for reference only)

The "Label" and "Unit" settings are added to the recorded data along with their index value.

17. WIRED M-BUS

You can collect data from an M-Bus device. To do this connect an M-Bus transceiver to the RS232 port and switch the port to M-Bus mode.
The M-Bus devices must be configured on the bus with a unique address.
Start a bus scan from the web interface. The M-Bus devices discovered during this scan are prompted each time by the assigned scheduler. Start a new scan when devices are removed from or added to the bus so the datalogger can take the changes into account.

The following settings are configured under /config/metering/mbus:

Name	Description
Schedule	Schedule ID for M-Bus data collection

18. WIRELESS M-BUS

In Wireless M-Bus radio card version, the Datalog H60 datalogger can receive data from known Wireless M-Bus modules in S1 or T1 modes.

The following settings are configured under /config/metering/wmbus:

Name	Value	Description
mode	S1, **T1**	Mode Wireless M-Bus
long_preamble	true, **false**	Length of Radio preamble (ignored in T mode)
modules/*		List of Wireless M-Bus modules

If OMS encryption is enabled, the number of modules is restricted to 64: Modules without encryption keys are ignored.

Each module is configured as follows:

Name	Description
module/address	Wireless M-Bus address
module/label	Module name (for reference only)
module/key	Module encryption key

19. ACTIVE RFID TAGS

In RFID radio card mode, the gateway can:
• Monitor the presence of active RFID tags in its range of coverage.
• Recover data from active RFID tags.

The receiver is compatible with active ELA Innovation tags: http://www.rfid-ela.eu

The tags must be configured in 24-bit mode with 16-bit radio checksum.
The datalogger is sent all active RFID tags on a regular basis.
An optional CRC prompt can be configured to filter all tags that are not configured with the same prompt.
Any ID received with an RSSI above the configured threshold is ignored. This threshold reduces the coverage area.

Note: The RSSI range is between 110 and 200. If the threshold is set to a value less than 110, no data is received. If the threshold is set to a value greater than 200, all data received is handled.

If the tag is an ID tag (the heavy bit of the ID is equal to 0), the next 3 bits are handled as alarm indicators and are not therefore considered as part of the ID.
If the tag is an ID+DATA tag (the heavy bit of the ID is equal to 1), the first 12 bits are used as the ID and the next 12 bits are considered as data.

The datalogger maintains a list of tags in its coverage area. A tag is added to this list only if its ID is received regularly during the "entering" period of detection. A label is removed from this list if its ID is not received within the "leaving" period of detection.

If an ID+DATA tag is seen as being in the coverage area, its data is recorded and uploaded to the remote server.

Alarms can be sent immediately or at the next login for the following events: tag input, tag output, flag set in the ID of an ID tag.

Whenever the gateway connects to the remote server, it sends:
• All the data collected from ID+DATA tags.
• A list of all tags in the coverage area.
• All the various alarms.

The following settings are available under /config/rfid:

Name	Value	Description
rssi_threshold	0-**255**	RSSI level of tag filtering
Crc	**0**	Optional CRC prompt
detection_delay/entering	**600**	Time after which a tag is considered as in range
detection_delay/leaving	**500**	Time after which a tag is considered as out of range
alarm/sources/entering	**on**, off, delayed	Alarm when a tag enters the area
alarm/sources/leaving	**on**, off, delayed	Alarm when a tag leaves the area
alarm/sources/id_flags	on, off, **delayed**	Alarm when an ID tag has a flag

Note: The data received from ID+DATA tags is saved as raw data. It is not converted to temperature/humidity/ movement because the datalogger is not aware of this type of information. Furthermore, specific 'end of battery life' values are not recognised (as this value depends on its type, 0x7FF for T/HR tags and 0xFFF for MOV tags).

20. MODBUS

The datalogger can function as a master Modbus device. This feature allows the system to read/write in the registers on RTU and TCP Modbus slave modules.
The datalogger's Modbus configuration consists of a list of datasets and a list of modules. A dataset is a list of records for a given type of Modbus slave device. The list of modules links a Modbus slave module to a dataset and a scheduler.
In polling mode, the value of all variables is continuously updated. These values can be monitored to detect changes or compare them to thresholds.

The current values are saved if:

• The value of a monitored data changes or crosses a certain threshold

• The assigned schedule occurs.

In instantaneous mode (i.e. not in polling mode), the values of all variables are updated and saved if and only if the assigned schedule occurs.
Regardless of the process of collecting Modbus data, it is possible to write in the registers of some slave modules using the Modbus command.

20.1. Configuration

The Modbus configuration under /config/modbus contains the following settings:

Name	Value	Description
tcp/timeout	1000	Timeout of the Modbus/TCP response in ms
rtu/timeout	1000	Timeout of the Modbus/RTU response in ms
rtu/turnaround	100	Time the Modbus/RTU takes to turn around in ms
datasets/*		List of datasets
modules/*		List of modules

In addition to these settings, the settings /config/system/ports/rs485/ must be correctly configured. In particular, the setting /config/system/ports/rs485/mode must be set equal to "modbus".

20.2. Modbus datasets

Configuring a dataset (/config/modbus/datasets/dataset) involves setting the following parameters:

Name	Description
id	Unique identifier for Modbus dataset
label	Name of the dataset (for reference only)
vars/*	List of variables
boundaries/*	List of restrictions
polling	Polling continue (true or false)

20.2.1. Variables

All variables are defined under /config/modbus/datasets/dataset/vars/var by the following settings:

Name	Description
name	Name of the variable (for reference only)
type	Variable type (S0, S1, S3, S4)
address	Extended registry address 16-bit
size	Size in bits for discrete input and coil, in octets for registries
format	See list below
flags	List of flags separated by commas (see the flag definition below)
threshold/low	Low threshold (see below)

Name	Description
threshold/high	High threshold (see below)
threshold/hysteresis	Hysteresis applied to two thresholds

20.2.1.1. "Type" parameter

A variable type is one of the four types of Modbus register.

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

In the table below, the read/write function codes are given as indications only. Modbus requests are not part of the configuration, but are taken into account. In particular, multiple Read function codes are used wherever this reduces communication costs.

20.2.1.2. "Address" parameter

This document always refers to the register Modbus address (starting with 0), never the register number of the Modbus (starting with 1).

20.2.1.3. "Format" parameter

Format	Description	Coil	Register
raw	The data is represented as a binary string for discrete inputs and coils and as a hexadecimal string for registers	•	•
boolean	True or false	•	
integer	16 or 32-bit integer, unsigned		•
float	16 or 32-bit with floating comma (IEEE 754)		•
ascii	ASCII character sequence		•

20.2.1.4. "Flag" parameter

Format	Description
cmd_only	This variable is not read from the Modbus module but can be written
little_endian	Interprets the two 16-bit registers into one 32-bit value in little-endian
no_opt	A dedicated Modbus request is used to read this variable.
is_status	In polling mode, any change to this variable prompts the system to read the dataset.
is_alarm	Similar to is_status but also starts a connection request.

20.2.1.5. "Alarm" parameter

For the floating and integer variables, you can define two thresholds (alarm/low and alarm/high) with a hysteresis value. Whenever the variable is updated, its value is compared to this level to determine an assigned status (low, normal, high), as shown below:

When at least one threshold level is set, the is_status and is_alarm flags apply to the resulting status. For example, if is_alarm is set to true, the connection request is triggered only when the variable status changes, not whenever its value changes.

Supplementary data in polling mode:

In polling mode, additional data is kept for integer and floating values: The values min/max/average and the number of samples since the last data record.

20.2.2. Boundaries

Not used in V2.x.

20.3. Modbus slaves

A module is an instance of a dataset for a given Modbus address. Configuring a Modbus module (/config/modbus/modules/module) includes the following parameters:

Name	Description
label	Name for reference only
dataset	Dataset identifier
address	Modbus address (1-247)
ip	IP address (empty for RTU devices)
schedule	Schedule identifier

Note: The Modbus/TCP devices must be configured to understand the Modbus TCP port by default (502).

20.4. Variable addresses

The Modbus command uses addresses formatted as shown below:

Modbus/RTU

> <modbus_address>/<register_type>@<register_address>

> Example:
> > Input register at address 0x0056 on Modbus device 45.
> > => 45/S3@0x0056

Modbus/TCP

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

> Example:
> > Input register at address 0x0056 on Modbus device 223 or IP address 192.168.0.17.
> > => 192.168.0.17:223/S3@0x0F56

Note: The modbus_address and register_address can be decimal or hexadecimal. If hexadecimal, preceded the address with "0x".

21. SCHEDULER

The scheduler manages all the recurring tasks.
The scheduler configuration consists of a list of schedules.
Each of these schedules has a unique ID (integer, positive) which is used to link a task to a specific schedule.

They can be used independently to initiate data collection and downloads.

Name	Description
/scheduler/schedules/schedule	Configure each schedule (see below) here.

Each schedule is configured as follows:

Name	Description
schedule/id	Unique schedule identifier (integer)
schedule/label	Reference name of the schedule
schedule/type	Daily, Weekly, Monthly, Yearly or Follower: see the description below
schedule/parent	Link to parent schedule for a Follower schedule.
schedule/start/time	Time of first occurrence (not used for Yearly schedules)
schedule/start/datetime	Date and time of first occurrence within a given period (used only for Yearly schedules)
schedule/start/dayofweek	Number of the day in the week of the time of first occurrence (1=Monday, 7=Sunday) (used only for Weekly schedules).
schedule/start/dayofmonth	Number of the day in the month of the time of first occurrence (used only for Monthly schedules).
schedule/interval	Interval between occurrences (in seconds)
schedule/count	Number of occurrences

Configuring the various types of schedule:

Daily schedule:

Every day, the first occurrence T0 is given by the time entered under Time.
The variable time is formatted as follows: HH:MM:SS
For example, 09:30:00
The following occurrences use the time Ti:

Δt is the value in seconds entered under Interval.

Weekly schedule:

Every week, the first occurrence T0 is given by the day of the week entered under dayofweek and the time entered under time.
The variable time is formatted as follows: HH:MM:SS
For example, 09:30:00
The variable dayofweek is an integer between 1 and 7 (1=Monday and 7=Sunday)
The following occurrences use the time Ti:

Δt is the value in seconds entered under Interval.

Monthly schedule:

Every month, the first occurrence T0 is given by the number of the day of the month entered under dayofmonth and the time entered under time.
The variable time is formatted as follows: HH:MM:SS
For example, 09:30:00
The following occurrences use the time Ti:

Δt is the value in seconds entered under Interval.

Yearly schedule:

Every year, the first occurrence T0 is given by the date entered under datetime.
The variable datetime is formatted as follows: YYYY-MM-DDTHH:MM:SS
For example, for a first occurrence on 11 February 2012 at 13H00:
datetime = 2012-02-11T13:00:00.
The following occurrences use the time Ti:

Δt is the value in seconds entered under Interval.

Follower schedule:

A 'Follower' schedule is produced after the end of each reference schedule occurrence. The parent schedule cannot be a 'Follower'.

With this type of schedule you can trigger, for example, a data download after the data collection is complete.

Example:

You want to collect data from all the modules once a day at midnight and download data just after. You can configure a "Daily" schedule to collect the data and a first, "Follower" schedule to download the data.


```
<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>
```

Examples:

Need	type	time	dayofweek	dayofmonth	datetime	interval	count
Every Tuesday at 15:00:00	week	15:00:00	2			0	1
Every 2nd of the month at 00:00:00	month	00:00:00		2		0	1
Every day at 14:00:00	day	14:00:00				0	1
Every day between 8H00 and 18H00 every Tuesday	week	08:00:00	2			3600	11
Every 2 hours between 8H00 and 20H00 on 31 December	year				2012-12-31T08:00:00	7200	7

21.1. Alarm engine

The alarm engine generates alarms based on internal events.

Each alarm source can be enabled individually and uploaded immediately to a remote server (on) or at the next connection (delayed).

Name	Values	Description
sources/power	on, off, delayed	Main power source status change
sources/modem_ip	on, off, delayed	Change in IP address
sources/msisdn	on, off, delayed	Change in MSISDN
sources/sw_version	on, off, delayed	Change in software version
sources/defaults/ignored	<empty>	List of default codes ignored by the gateway (separated by commas)
sources/defaults/delayed	<empty>	List of default codes sent at the next connection (separated by commas)
sources/d_inputs/*		Change in On/Off input status (see below).
sources/d_output		Change in On/Off output status (see below).

The gateway also generates default alarms, which have the following codes:

Code	Description
D_MODEM	Modem error
D_MODEM_PUK	SIM card blocked
D_ETHERNET	Ethernet interface error
D_WAVENIS	Wavenis radio default
D_RFID	RFID receiver default
D_INTERNAL_BAT	Internal battery error

With the "power" alarm enabled, the alarm engine sends an alarm when its power is lost or recovered.

With the "modem ip" alarm enabled, the alarm engine sends an alarm containing the IP address of the gateway every time it changes.

With the "msisdn" alarm enabled, the alarm engine sends an alarm when the SIM card is changed.

With the "sw version" alarm enabled, the alarm engine sends an alarm containing the software version of the gateway after an update.

You can configure multiple alarms for digital inputs.

You can configure an alarm for a digital input as follows:

Name	Value	Description
d_input/index		Digital input index
d_input/label		Alarm name (for reference only)
d_input/mode	on, off, delayed	On: Immediate transmission
Off: Transmission disabled		
Delayed: Send at next connection		
d_input/type	none, raising, falling, both	None: Scan disabled
Raising: Front scan increasing		
Falling: Front scan falling		
Both: Raising+Falling		

You can configure an alarm for the digital output, as follows:

Name	Value	Description
d_output/label		Alarm name (for reference only)
d_output/mode	on, off, delayed	On: Immediate transmission
Off: Transmission disabled		
Delayed: Send at next connection		
d_output/type	none, raising, falling, both	None: Scan disabled
Raising: Front scan increasing		
Falling: Front scan falling		
Both: Raising+Falling		

The lifting front corresponds to the closing of the gateway's dry contact.

The alarm is uploaded to the remote server in XML as specified by the XML alarm schema (see Section 16).

22. LOG FILES

The datalogger records key events in log files. The size of the log file is limited to 200 KB.

You can have past and current log files concatenated and uploaded to the remote server as supervision data. The log file is a text file with one log entry per line.

Each line is formatted as follows: [TIMESTAMP][LEVEL][SOURCE] EVENT

This is a Unix timestamp (time elapsed since the EPOC) in seconds followed by a point and the microseconds.

The following events are logged:

Events	Formats
Schedules	Schedule X occurred
Defaults	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
Commands	Processing command X
Clock	Clock synchronization delta=X
	NTP connection failure
Internal	Reboot
	Reboot modem

This list is not exhaustive and each line can contain additional information after that specified here. For example, when a schedule occurs, the log file may contain "Schedule X occurred, next is Y at Z".

You can set the logs' level of detail under /system/log/level. The value must be between 8 (no log) and 1 (more detailed):

Level	Name	Description
8	None	Nothing logged
7	Critical	Critical info only
6	Error	
5	App	Default level
4	Warning	
3	Notice	
2	Info	
1	Debug	Most detailed logs

The log level setting can also be defined by the source. This gives the format: default_level,source:level, source:level,…

23. SYNCHRONISING THE INTERNAL CLOCK

The gateway synchronises the system clock using the NTP Protocol.

It keeps the time in UTC (universal time, also known as GMT) and computes a local time based on the timezone you enter. The system supports daylight saving time (DST).

If using a modem connection, synchronisation occurs at the start of each connection to the server, but no more than once per day.

An alarm is triggered when the difference between the two clocks is more than a configurable value.

If using an Ethernet connection, an NTP client is enabled on the gateway. This client sets the speed of the system clock, to keep it synchronised with the NTP server's clock.

24. UPDATING THE GATEWAY FIRMWARE

An update can be done remotely. The new firmware should be made available on the FTP server in the dedicated BIN directory (see Section 3.2). An update command must then be sent to the gateway.

Example: Update command for the datalogger's firmware with the wrf_wavenis_v101.bin firmware in the BIN directory.

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

The checksum corresponds to the checksum md5 of the file.

25. SUPPORT

If you experience any technical problems related to our products, please contact SOCOMEC support

email: support@Socomec.com - http://www.Socomec.com

We will need the following:
• Datalogger's site number
• Datalogger's hardware and software versions

APPENDIX I. XSD SCHEMA – CONFIGURATION

config.xsd

```xml
<?xml version=»1.0»?>
<xsd:schema xmlns:xsd=»http://www.w3.org/2001/XMLSchema»
    xmlns:wd=»http://www.SOCOMEC.com/WRF_config_20110606»
    targetNamespace=»http://www.SOCOMEC.com/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>
```

```xml
                <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 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:element name=»alarm_threshold» type=»xsd:integer» minOccurs=»0»/>
```

```xml
            </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»>
```

```xml
            <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»/>
```

```xml
        </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>
                    <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>
```

```xml
                    <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»>
```

```xml
                <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>
```

```xml
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
    </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: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» >
```

```xml
                    <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: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»/>
```

```xml
                <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>
```

APPENDIX II. XSD SCHEMA – CONFIGURATION

config.xml

```xml
<?xml version=»1.0»?>
<config
    xmlns=»http://www.SOCOMEC.com/WRF_config_20110606»
    xmlns:xsi=»http://www.w3.org/2001/XMLSchema-instance»
    xsi:schemaLocation=»http://www.SOCOMEC.com/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>
```

```xml
        <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 dectection</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>
```

```xml
                        <address>012345678901</address>
                        <label>test1</label>
                        <key>00112233445566778899</key>
                    </module>
                </modules>
            </wmbus>
        </metering>
        <rfid>
            <rssi_threshold>255</rssi_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>
```

APPENDIX III. XSD SCHEMA – ALARMS

alarm.xsd

```xml
<?xml version=»1.0»?>
<xsd:schema xmlns:xsd=»http://www.w3.org/2001/XMLSchema»
    xmlns:wd=»http://www.SOCOMEC.com/WRF_alarm_20110606»
    targetNamespace=»http://www.SOCOMEC.com/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>
```

```xml
            <xsd:element name=»description» type=»xsd:string» minOccurs=»0»/>
        </xsd:sequence>
    </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»>
```

```xml
<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»>
```

```xml
<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'>
```

```xml
		<xsd:sequence>
			<xsd:element name=»input» type=»wd:wavenis_input» minOccurs=»0»/>
			<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
        <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: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>
```

APPENDIX IV. EXAMPLE XML – ALARMS

alarm.xml

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

```xml
</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>
```

```xml
        <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>moduflow-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>
```

APPENDIX V. XSD SCHEMA – SUPERVISION

supervision.xsd

```
<?xml version=»1.0»?>
<xsd:schema xmlns:xsd=»http://www.w3.org/2001/XMLSchema»
    xmlns:wd=»http://www.SOCOMEC.com/WRF_supervision_20110606»
    targetNamespace=»http://www.SOCOMEC.com/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»/>
```

```xml
<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:sequence>
        <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: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:sequence>
        <xsd:attribute name=»cid» type=»xsd:string» use=»optional»/>
        <xsd:attribute name=»date» type=»xsd:dateTime»/>
    </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>
```

APPENDIX VI. EXAMPLE XML – SUPERVISION

supervision.xml

```xml
<?xml version=»1.0»?>
<supervision
    xmlns=»http://www.SOCOMEC.com/WRF_supervision_20110606»
    xmlns:xsi=»http://www.w3.org/2001/XMLSchema-instance»
    xsi:schemaLocation=»http://www.SOCOMEC.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>11.246</firmware>
                <imei>AA-BBBBBB-CCCCCC</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>
```

```xml
                    </module>
                </modules>
            </wavenis>
        </scan>

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

    </supervision>
```

APPENDIX VII. XSD SCHEMA – DATA

data.xsd

```xml
<?xml version=»1.0»?>
<xsd:schema xmlns:xsd=»http://www.w3.org/2001/XMLSchema»
    xmlns:wd=»http://www.SOCOMEC.com/WRF_data_20110606»
    targetNamespace=»http://www.SOCOMEC.com/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>
```

```xml
                            <xsd:extension base=»xsd:string»>
                                <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:sequence>
    <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
                <xsd:enumeration value=»B» />
                <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:schema>
```

APPENDIX VIII. CSV FORMAT – DATA

The CSV format (Comma Separated Values) has no official definition.

However, it does follow these rules:

One line contains one single record
Each record corresponds to one single line
Each line ends in an end return.
Each line contains the same number of fields.
Each field is separated by a comma.

Each line is formatted as follows:

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

With:

Timestamp: DD/MM/YYYY HH:MM
Source: The possible values are:
FL: WaveFlow
TH: WaveTherm
LO: WaveLog
SE: WaveSense
Address: Wavenis address
Label: Wavenis module if configured, otherwise empty.
Input: input ID (A, B, C or D).
Value: Value of the given input.

APPENDIX IX. XSD SCHEMA – COMMANDS

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/SOCOMEC/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/SOCOMEC/Gateway/WG310P/trunk/XML_Schemas/command.xsd?format=txt»
type=»text/plain» title=»Plain Text» /><link rel=»alternate» href=»/trac/export/3915/Projets/Clients/SOCOMEC/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-toggledeleted»).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»><img src=»/trac/chrome/site/SOCOMEC.jpg» alt=»(please configure the
[header_logo] section in trac.ini)» /></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/SOCOMEC/Gateway/WG310P/trunk/
XML_Schemas/command.xsd?rev=3715» title=»Revision 3715»>Previous Revision</a></span></li><li><span class=»missing»>Next
Revision &rarr;</span></li><li><a href=»/trac/browser/Projets/Clients/SOCOMEC/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/SOCOMEC/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/SOCOMEC?order=name» title=»View SOCOMEC»>SOCOMEC</a><span
class=»pathentry sep»>/</span><a class=»pathentry» href=»/trac/browser/Projets/Clients/SOCOMEC/Gateway?order=name» title=»View
Gateway»>Gateway</a><span class=»pathentry sep»>/</span><a class=»pathentry» href=»/trac/browser/Projets/Clients/SOCOMEC/Gateway/
WG310P?order=name» title=»View WG310P»>WG310P</a><span class=»pathentry sep»>/</span><a class=»pathentry» href=»/trac/browser/
Projets/Clients/SOCOMEC/Gateway/WG310P/trunk?order=name» title=»View trunk»>trunk</a><span class=»pathentry sep»>/</span><a
class=»pathentry» href=»/trac/browser/Projets/Clients/SOCOMEC/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/SOCOMEC/
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>,
						<span title=»4047 bytes»>4.0 KB</span>
						checked in by ghofmann, <a class=»timeline» href=»/trac/timeline?from=2012-11-19T18%3A15%3A52%2B01%3A00&amp;precision=s
econd» title=»2012-11-19T18:15:52+01:00 in Timeline»>3 weeks</a> ago
						(<a href=»/trac/changeset/3716/Projets/Clients/SOCOMEC/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»>xmlns: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»>xmlns:wd=</span><span
class=»s»>»http://www.SOCOMEC.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.SOCOMEC.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><span class=»nt»>&gt;</span></td></tr><tr><th id=»L7»><a href=»#L7»>7</a></th><td></td></tr><tr><th
id=»L8»><a href=»#L8»>8</a></th><td><span class=»nt»>&lt;xsd:element</span> <span class=»na»>name=</span><span
class=»s»>»commands»</span><span class=»nt»>&gt;</span></td></tr><tr><th id=»L9»><a href=»#L9»>9</a></th><td>         <span class=»
nt»>&lt;xsd:complexType&gt;</span></td></tr><tr><th id=»L10»><a href=»#L10»>10</a></th><td>         <span class=»nt»>&lt;xsd:seque
nce&gt;</span></td></tr><tr><th id=»L11»><a href=»#L11»>11</a></th><td>         <span class=»nt»>&lt;xsd:element</
span> <span class=»na»>name=</span><span class=»s»>»cmd»</span> <span class=»na»>maxOccurs=</span><span
class=»s»>»unbounded»</span><span class=»nt»>&gt;</span></td></tr><tr><th id=»L12»><a href=»#L12»>12</a></th><td>
<span class=»nt»>&lt;xsd:complexType&gt;</span></td></tr><tr><th id=»L13»><a href=»#L13»>13</a></th><td>
<span class=»nt»>&lt;xsd:sequence&gt;</span></td></tr><tr><th id=»L14»><a href=»#L14»>14</a></th><td>
<span class=»nt»>&lt;xsd:choice&gt;</span></td></tr><tr><th id=»L15»><a href=»#L15»>15</a></th><td>
<span class=»nt»>&lt;xsd:element</span> <span class=»na»>name=</span><span class=»s»>»reboot»</span> <span class=»na»>type=</
span><span class=»s»>»wd:cmd_simple»</span><span class=»nt»>/&gt;</span></td></tr><tr><th id=»L16»><a href=»#L16»>16</a></th><
td>                <span class=»nt»>&lt;xsd:element</span> <span class=»na»>name=</span><span
class=»s»>»factory»</span> <span class=»na»>type=</span><span class=»s»>»wd:cmd_simple»</span><span class=»nt»>/&gt;</span></
td></tr><tr><th id=»L17»><a href=»#L17»>17</a></th><td>                <span class=»nt»>&lt;xsd:element</
span> <span class=»na»>name=</span><span class=»s»>»update»</span> <span class=»na»>type=</span><span class=»s»>»wd:cmd_
update»</span><span class=»nt»>/&gt;</span></td></tr><tr><th id=»L18»><a href=»#L18»>18</a></th><td>
<span class=»nt»>&lt;xsd:element</span> <span class=»na»>name=</span><span class=»s»>»scan»</span> <span class=»na»>type=</
span><span class=»s»>»wd:cmd_scan»</span><span class=»nt»>/&gt;</span></td></tr><tr><th id=»L19»><a href=»#L19»>19</a></th><
td>                <span class=»nt»>&lt;xsd:element</span> <span class=»na»>name=</span><span class=»s»>»
timesync»</span> <span class=»na»>type=</span><span class=»s»>»wd:cmd_timesync»</span><span class=»nt»>/&gt;</
span></td></tr><tr><th id=»L20»><a href=»#L20»>20</a></th><td>                <span class=»nt»>&lt;xsd:element</
span> <span class=»na»>name=</span><span class=»s»>»wavenis»</span> <span class=»na»>type=</span><span class=»s»>»wd:cmd_
wavenis»</span><span class=»nt»>/&gt;</span></td></tr><tr><th id=»L21»><a href=»#L21»>21</a></th><td>
<span class=»nt»>&lt;xsd:element</span> <span class=»na»>name=</span><span class=»s»>»modbus»</span> <span class=»na»>type=</
span><span class=»s»>»wd:cmd_modbus»</span><span class=»nt»>/&gt;</span></td></tr><tr><th id=»L22»><a href=»#L22»>22</a></th>
<td>                <span class=»nt»>&lt;xsd:element</span> <span class=»na»>name=</span><span
class=»s»>»status»</span> <span class=»na»>type=</span><span class=»s»>»wd:cmd_simple»</span><span class=»nt»>/&gt;</span></
```

```
23    <xsd:element name="log" type="wd:cmd_simple"/>
24    <xsd:element name="d_output" type="wd:cmd_d_output"/>
25        </xsd:choice>
26      </xsd:sequence>
27    <xsd:attribute name="uid" type="xsd:string" use="optional"/>
28    <xsd:attribute name="cid" type="xsd:string" use="optional"/>
29        </xsd:complexType>
30      </xsd:element>
31    </xsd:sequence>
32      </xsd:complexType>
33  </xsd:element>
34  <xsd:complexType name="cmd_simple">
35  </span>
36      <xsd:complexContent>
37        <xsd:restriction base="xsd:anyType">
38        </xsd:restriction>
39      </xsd:complexContent>
40  </xsd:complexType>
41
42  <xsd:complexType name="cmd_update">
43    <xsd:sequence>
44      <xsd:element name="firmware" type="xsd:string"/>
45    <xsd:element name="checksum" type="xsd:string"/>
46      </xsd:sequence>
47  </xsd:complexType>
48
49  <xsd:complexType name="cmd_scan">
50      <xsd:sequence>
51    <xsd:element name="address" type="wd:wavenis_address"
52      minOccurs="0" maxOccurs="unbounded"/>
53      </xsd:sequence>
54      <xsd:attribute name="mode">
55    <xsd:simpleType>
56      <xsd:list itemType="wd:scan_modes"/>
57      </xsd:simpleType>
58    <xsd:attribute>
59  </xsd:complexType>
60  </td>
61  <xsd:simpleType name="scan_modes">
62      <xsd:restriction base="xsd:string">
63    <xsd:enumeration value="data"/>
64      <xsd:enumeration value="rssi"/>
65      <xsd:enumeration value="life-counter"/>
66    <xsd:enumeration value="rtc"/>
67      <xsd:enumeration value="tic"/>
68      </xsd:restriction>
69  <xsd:simpleType>
70
71  <xsd:complexType name="cmd_timesync">
72      <xsd:sequence>
73    <xsd:element name="address" type="wd:wavenis_address"/>
74          minOccurs="0" maxOccurs="unbounded"/>
75      </xsd:sequence>
76  </xsd:complexType>
77
78  <xsd:complexType name="cmd_wavenis">
79      <xsd:sequence>
80    <xsd:element name="address" type="wd:wavenis_address"
81      minOccurs="0" maxOccurs="unbounded"/>
82      </xsd:sequence>
83    <xsd:attribute name="subcmd" type="xsd:string"/>
84      <xsd:attribute name="data" type="wd:hex_string" use="optional"/>
85  </xsd:complexType>
86
87  <xsd:complexType name="cmd_d_output">
88    <xsd:attribute name="subcmd" type="wd:d_output_cmds"/>
89  </xsd:complexType>
90
91  <span
```

```
class=»nt»>&lt;xsd:simpleType</span> <span class=»na»>name=</span><span class=»s»>»d_output_cmds»</span><span class=»nt»>&gt;</
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»>&gt;</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»>/&gt;</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»>/&gt;</span></
td></tr><tr><th id=»L95»><a href=»#L95»>95</a></th><td>        <span class=»nt»>&lt;/xsd:restriction&gt;</span></td></tr><tr><th
id=»L96»><a href=»#L96»>96</a></th><td><span class=»nt»>&lt;/xsd:simpleType&gt;</span></td></tr><tr><th id=»L97»><a
href=»#L97»>97</a></th><td></td></tr><tr><th id=»L98»><a href=»#L98»>98</a></th><td><span class=»nt»>&lt;xsd:complexType</
span> <span class=»na»>name=</span><span class=»s»>»cmd_modbus»</span><span class=»nt»>&gt;</span></td></tr><tr><th
id=»L99»><a href=»#L99»>99</a></th><td>        <span class=»nt»>&lt;xsd:sequence&gt;</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»>/&gt;</span></td></tr><tr><th id=»L101»><a
href=»#L101»>101</a></th><td>        <span class=»nt»>&lt;/xsd:sequence&gt;</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»>/&gt;</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»>»wd:hex_string»</span><span class=»nt»>/&gt;</span></td></tr><tr><th
id=»L104»><a href=»#L104»>104</a></th><td><span class=»nt»>&lt;/xsd:complexType&gt;</span></td></tr><tr><th id=»L105»><a
href=»#L105»>105</a></th><td></td></tr><tr><th id=»L106»><a href=»#L106»>106</a></th><td><span class=»nt»>&lt;xsd:simpleType</
span> <span class=»na»>name=</span><span class=»s»>'modbus_cmd_string'</span><span class=»nt»>&gt;</span></td></tr><tr><th
id=»L107»><a href=»#L107»>107</a></th><td>        <span class=»nt»>&lt;xsd:restriction</span> <span class=»na»>base=</span><span
class=»s»>'xsd:string'</span><span class=»nt»>&gt;</span></td></tr><tr><th id=»L108»><a href=»#L108»>108</a></th><td>
<span class=»nt»>&lt;xsd:pattern</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»>/&gt;</span></td></tr><tr><th id=»L109»><a href=»#L109»>109</a></
th><td>        <span class=»nt»>&lt;/xsd:restriction&gt;</span></td></tr><tr><th id=»L110»><a href=»#L110»>110</a></th><td><span
class=»nt»>&lt;/xsd:simpleType&gt;</span></td></tr><tr><th id=»L111»><a href=»#L111»>111</a></th><td></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»>&gt;</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»>&gt;</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»>/&gt;</span></td></tr><tr><th id=»L115»><a
href=»#L115»>115</a></th><td>        <span class=»nt»>&lt;/xsd:restriction&gt;</span></td></tr><tr><th id=»L116»><a href=»#L116»>116</
a></th><td><span class=»nt»>&lt;/xsd:simpleType&gt;</span></td></tr><tr><th id=»L117»><a href=»#L117»>117</a></th><td></td></
tr><tr><th id=»L118»><a href=»#L118»>118</a></th><td><span class=»nt»>&lt;xsd:simpleType</span> <span class=»na»>name=</
span><span class=»s»>'wavenis_address'</span><span class=»nt»>&gt;</span></td></tr><tr><th id=»L119»><a href=»#L119»>119</a></
th><td>        <span class=»nt»>&lt;xsd:restriction</span> <span class=»na»>base=</span><span class=»s»>'xsd:string'</span><span
class=»nt»>&gt;</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»>/&gt;</
span></td></tr><tr><th id=»L121»><a href=»#L121»>121</a></th><td>        <span class=»nt»>&lt;/xsd:restriction&gt;</span></td></
tr><tr><th id=»L122»><a href=»#L122»>122</a></th><td><span class=»nt»>&lt;/xsd:simpleType&gt;</span></td></tr><tr><th id=»L123»><a
href=»#L123»>123</a></th><td></td></tr><tr><th id=»L124»><a href=»#L124»>124</a></th><td><span class=»nt»>&lt;/xsd:schema&gt;</
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/SOCOMEC/Gateway/WG310P/trunk/XML_Schemas/command.xsd»
/>
          <input type=»hidden» name=»old_path» value=»/Projets/Clients/SOCOMEC/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=»altlinks»>
       <h3>Download in other formats:</h3>
       <ul>
        <li class=»first»>
         <a rel=»nofollow» href=»/trac/browser/Projets/Clients/SOCOMEC/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/SOCOMEC/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>
```

APPENDIX X. EXAMPLE XML – COMMANDS

command.xml

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

socomec
Innovative Power Solutions