VAISALA

USER'S GUIDE

Vaisala HydroMet™ Data Collection Platform
Volume 3

M210933EN-C

PUBLISHED BY

Vaisala Oyj	Phone (int.):	+358 9 8949 1
P.O. Box 26	Fax:	+358 9 8949 2227
FIN-00421 Helsinki		
Finland		

Visit our Internet pages at http://www.vaisala.com/

Table of Contents

List of Figures

List of Tables

CHAPTER 1

GENERAL INFORMATION

This chapter provides general notes for the product(s) and this manual.

About This Manual

This manual supplements Vaisala HydroMet™ Data Collection Platform User's Guide Volumes 1 and 2, and contains information on configuring sensors and telemetry. Applicable for data logger QML201C and Lizard Setup Software versions 8.00, and AWS Client terminal software version 7.00.

Structure of the Data Collection Platform Documentation

The information in the Vaisala HydroMet™ Data Collection Platform manual set is divided between the different manuals in the documentation set as outlined in Table 1 on page 20.

Table 1 Structure of the DCP Manual Set

Manual	Code	Content
User's Guide, Volume 1	M210784EN	Overview of the data collection platform, the QML logger, and related accessories. Operating instructions for AWS Client software.
User's Guide, Volume 2	M210785EN	Operating instructions for Lizard Setup Software
User's Guide, Volume 3	M210933EN	Telemetry and sensor configuration in Lizard Setup Software
Installation Manual (Field Equipment)	M210786EN	Installation information on the Data Collection Platform with meteorological and/or hydrological sensors

Contents of This Manual

This manual consists of the following chapters:

- Chapter 1, General Information: This chapter provides general notes for the product(s) and this manual.
- Chapter 2, Product Overview: This chapter introduces the communication modules and features on which configuration information is provided in this manual.
- Chapter 3, Configuring Modules and Sensors: This chapter contains additional information on setup management, instructions for configuring optional hardware, as well as configuration and measurement control information for different sensors and serial sensor interfaces.
- Chapter 4, Configuring Telemetry Options: This chapter provides instructions on how to configure different non-TCP/IP telemetry options, that is, modems and satellite transmitters.
- Chapter 5, Configuring TCP/IP-Based Telemetry: This chapter provides instructions for configuring TCP/IP-based telemetry options.
- Chapter 6, Technical Support: This chapter informs you how to contact Vaisala HelpDesk.
- Chapter 7, Technical Data: This chapter provides technical data for the modules included in this manual.
- Appendix A, Configuration Information for Legacy Items: This appendix contains configuration information for devices not necessarily needed in MAWS versions 6.00 or greater.
- Appendix B, GSM 7-Bit Character Set: This appendix contains the 7-bit character set for GSM communications. The character set is provided as an aid for configuring and troubleshooting GSM-based connections.

Version Information

Table 2 Manual Versions

Manual Code	Description
M210933EN-C	This manual. For MAWS system release 8.00 and QML201C.
M210933EN-B	Previous version. For MAWS system release 7.00 and QML201B..
M210933EN-A	First version of this manual.

Related Manuals

Table 3 Related Manuals

Manual Code	Manual Name
M210784EN	Vaisala HydroMet™ Data Collection Platform User’s Guide, Volume 1
M210785EN	Vaisala HydroMet™ Data Collection Platform User’s Guide, Volume 2
M210786EN	Vaisala HydroMet™ Data Collection Platform Installation Manual
M210629EN	Automatic Weather Station MAWS101 User's Guide
M210630EN	Automatic Weather Station MAWS201 User's Guide
M210681EN	Real-time Display Software YourVIEW 2000 YVU2000 User’s Guide
M210743EN	Voice Option for MAWS Technical Reference
M211022EN	Configuring BUFR Reports MAWS Technical Note

Product-Related Safety Precautions

The product has been tested for safety and approved as shipped from the factory. The following safety precautions are not related to any specific procedures and therefore do not appear elsewhere in this manual. They are recommended precautions that personnel must understand and apply during different phases of operation and maintenance.

WARNING	Keep away from live circuits. Operating personnel must observe safety regulations at all times. Component replacement or internal adjustments must be made by qualified maintenance personnel. Do not replace components with the power cable connected. Under certain conditions, dangerous voltages may exist for some time even with the power cable disconnected. To avoid injuries, disconnect power and discharge circuits before touching them.

WARNING	Do not service alone. Under no circumstances should any person reach into parts and assemblies that are mains powered and alive, for the purpose of servicing, except in the presence of someone who is capable of rendering aid.

WARNING	Personnel working with or near high voltages should be familiar with modern methods of resuscitation.

WARNING	Do not service a live system outdoors. Do not open units outdoors when the enclosure used contains line voltage levels.

WARNING	Do not operate in an explosive atmosphere, for example, when flammable gases or fumes are present. Operation of any electrical instrument in such an environment constitutes a definite safety hazard.

WARNING	Do not substitute parts or modify the instrument. Because of the danger of introducing additional hazards, do not install unsuitable parts in the instrument. Contact Vaisala or its authorized representative for repairs to ensure that safety features are maintained.

WARNING	Use only batteries of the same type as originally installed on the system.

CAUTION	Do not make changes to the wiring. Incorrect wiring can damage the device and prevent it from operating correctly.

CAUTION	Risk of damage to the equipment if the battery is replaced with an incorrect type.

ESD Protection

Electrostatic Discharge (ESD) can cause immediate or latent damage to electronic circuits. Vaisala products are adequately protected against ESD for their intended use. However, it is possible to damage the product by delivering electrostatic discharges when touching, removing, or inserting any objects inside the equipment housing.

To make sure you are not delivering high static voltages yourself:

- Handle ESD sensitive components on a properly grounded and protected ESD workbench. When this is not possible, ground yourself with a wrist strap and a resistive connection cord to the equipment chassis before touching the boards. When neither of the above is possible, at least touch a conductive part of the equipment chassis with your other hand before touching the boards.
- Always hold the boards by the edges and avoid touching the component contacts.

Recycling

Recycle all applicable material.

Dispose of batteries and the unit according to statutory regulations. Do not dispose of with regular household refuse.

Regulatory Compliances

The Vaisala HydroMet™ Data Collection Platform complies with the following EU directives:

- Low Voltage Directive (2006/95/EC)
- EMC-Directive (2004/108/EC)

Trademarks

Vaisala HydroMet™ Data Collection Platform is a trademark of Vaisala Oyj.

Windows® is a registered trademark of Microsoft Corporation in the United States and/or other countries.

License Agreement

All rights to any software are held by Vaisala or third parties. The customer is allowed to use the software only to the extent that is provided by the applicable supply contract or Software License Agreement.

Redistribution License Agreement

The QML logger software uses the TCP/IP stack produced by the "lwIP Lightweight TCP/IP stack" -project with the following copyright and license:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Warranty

For certain products Vaisala normally gives a limited one-year warranty. Please observe that any such warranty may not be valid in case of damage due to normal wear and tear, exceptional operating conditions, negligent handling or installation, or unauthorized modifications. Please see the applicable supply contract or Conditions of Sale for details of the warranty for each product.

CHAPTER 2

PRODUCT OVERVIEW

This chapter introduces the communication modules and features on which configuration information is provided in this manual.

PSTN Modem

Figure 1 PSTN Modem

The PSTN modem is a dial-up modem for the weather stations and other systems. The operation mode and settings can be selected with the extended set of AT commands. The PSTN modem is used to connect the weather station to public switched telephone networks (PSTN) or to a leased line point-to-point application.

The PSTN modem unit has a small metal enclosure. The modem is mounted on a DIN rail inside the equipment enclosure. It has been designed for demanding environments. It is rated for -40 to +60 °C operating temperatures. The modem is DC-powered and has low power consumption.

The PSTN modem module has face connectors where the power supply and RS-232 can be connected. There is also a RJ-11 connector for the telephone interface.

The module includes both data compression and data correction functions. The maximum data rate is up to 57.6 Kbits/second. The line input is protected by a telecom PTC fuse, sidactors, and a heavy-duty 350 V discharge tube.

When connecting the PSTN modem to the QML logger, you need to use the terminal strip to provide power for the modem. There is an initial surge arrester to protect the line connection. For the example wiring diagram, see Chapter 7, Technical Data, on page 337.

After the cables are connected, configure the PSTN modem with Lizard Setup Software. Refer to Chapter 4, Configuring Telemetry Options, on page 131.

Leased Line Modem

Figure 2 Leased Line Modem

The leased line modem is intended for weather stations and other systems. The operation mode and settings are selectable with the DIL switch and jumpers. The leased line modem module can be used to connect the weather station to the leased line 2-wire or 4-wire point-to-point or multi-point application.

The module supports standards such as V.21 300/300 bps FSK, V.22 1200/1200 DPSK, and V.23 1200/1200 bps FSK.

The module has a small enclosure with DIN-rail mounting support that can be connected into various enclosures. It has face connectors where the power supply, RS-232, and leased line can be connected.

When connecting a leased line modem to the QML logger, you need to use the terminal strip to provide power for the modem. For the example wiring diagram, see Chapter 7, Technical Data, on page 337.

Cellular Modems

The GSM/GPRS module provides wireless telemetry for remote stations.

MC35 is a GSM Terminal with GPRS (General Packet Radio Service) capability. It offers a wireless TCP/IP connection directly from the QML logger to send data via Internet. In practice, GPRS connectivity means the measuring system is online all the time. Data is available immediately even from a large number of stations. Secondly, data is transferred automatically into files on the user's hard disk, without polling or other data collection functions.

GPRS connectivity can be used, for example, for sending reports and log files to an FTP server or for forming a TCP/IP-based service connection for maintenance purposes.

Figure 3 GSM Terminal MC35

The GSM antenna that is used with the GSM data modems is a directional antenna. The antenna provides high gain, enabling the data communication even in remote and sparsely-populated installation sites where long distances or the terrain cause uncertain connections. In addition to giving a good connection, a strong signal also reduces the power consumption at the station.

The antenna cable is a high-quality coaxial cable. The radio modem is protected against high voltages using a surge arrestor installed on the antenna cable input.

Figure 4 GSM Antenna

When connecting GSM Modem to the QML logger, you need to use the terminal strip to provide power for the modem. For the wiring diagram, see Chapter 7, Technical Data, on page 337.

QML Logger TCP/IP Functionality

Starting from software version 6.00, the QML logger provides the following TCP/IP-based application functionality without using additional protocol providers, such as iConnector for GPRS modems or COM Server Unit DXE421.

- Virtual COM ports for serial port tunneling over TCP/IP
  - Report transmission
  - Service connection
  - Sensor connection
- Report and log file transfer using File Transfer Protocol (FTP)
- Time synchronization using Network Time Protocol (NTP)

Support for a new Ethernet communication module, DSE101, for a 10 Mbps Ethernet connection, is also included. DSE101 is compatible with 10/100 Mbps switches and routers, but the data is always transferred at 10 Mbps.

The QML logger can be connected to different TCP/IP-based networks. The most common ones are listed in Table 4 on page 34.

Table 4 TCP/IP-Based Networks Supported by the QML Logger

Media	QML communication device(s)
Ethernet (LAN)	Ethernet Communication Module DSE101
WAN, for example, ADSL	Ethernet Communication Module DSE101
GSM/GPRS cellular network	GPRS modem with DSU232
PSTN dial-up	PSTN modem with DSU232
RS-232 (point to point)	DSU232

Supported Protocols

Starting from version 6.00, the QML logger supports the following new communications protocols:

- PPP (Point to Point Protocol)
  - Authentication: PAP and CHAP
- Ethernet (using Ethernet Communication Module DSE101)
- IP (Internet Protocol)
- TCP (Transport Control Protocol)
- UDP (User Datagram Protocol)
- DNS (Domain Name Service, client in the QML logger)
- FTP (File Transfer Protocol, client in the QML logger)
- NTP (Network Time Protocol, client in the QML logger)
- DHCP (Dynamic Host Configuration Protocol, client in the QML logger)
- Telnet (limited support: virtual COM ports provide handling for certain escape sequences to enable binary file transfer)

UHF Radio Modem

Figure 5 UHF Radio Modem

The UHF radio modem SATELLINE3AS is a half-duplex radio modem suitable for high-speed data applications. This radio modem offers high-speed data transmission up to 40 km Line-Of-Sight (LOS). It provides the data speeds 19200 bps at 25 kHz and 9600 bps at 12.5 kHz in the air. RS interface data speed is user-selectable from 300 to 38400 bps. The connection between data logger and the radio modem is established by using RS-232.

The Epic model with its 10 W transmitter power, dual receivers with the Diversity Reception operation can have distance up to 60 km LOS. The output power is configurable from 0.1 W up to 10 W.

Figure 6 Radio Modem Satelline 3AS Epic with Optional Display

When using UHF Radio Modem SATELLINE 3AS, the RS-232 communication module should be used to provide an additional RS-232 output for the radio modem, leaving the standard COM port (**COM0**) free for maintenance purposes.

UHF Repeater Function

Message Routing is a built-in feature in the SATELLINE-3AS radio modems, which makes it easier to build up a large radio modem network. Message Routing features a versatile radio protocol, which takes care of routing messages across a radio modem network. Only one radio channel is required even in large networks. Any radio modem in the network can act as a repeater and have a weather station interfaced as well. The repeater can also be chained allowing message transmission through several repeaters/weather stations.

Using the in-built functions in the SATELLINE-3AS radio modems, two different types of repeater operations can be built:

1. The weather station having the radio modem will function as a repeater for a group of other stations.
2. The radio modem alone installed in an enclosure with proper powering can function as independent repeater for a group of weather stations.

UHF Antenna

The CompleTech antenna product range offers optimized antenna solution for every application and installation requirements. The product range includes omnidirectional antennas, dipoles, directional and crosspolarized yagi antennas, and directional and multidirectional stacked arrays to serve demanding telemetry needs.

All the electromechanics in the antenna are molded in polyurethane foam and covered by insulating covers. This construction results in extremely good combination of lightness, durability, and environmental resistance. The effects of water, ice, snow, and pollutants on properties of the antenna are minimized.

Depending on installation site, its function, and LOS distances, two different kinds of antennas are normally used:

1. Directional antenna for stations with longer LOS. The directional antennas range from simple one element yagi to multi-element yagis and stacked yagis with increased gain.

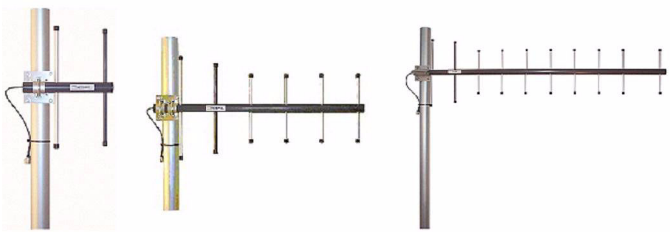

Figure 7 Directional Antennas with Different Gains

2. Omnidirectional antennas for stations function as repeaters or with shorter LOS. Omnidirectional ground plane antennas are robust antennas with high gain and especially suitable for repeater sites.

Figure 8 Omnidirectional Antenna

ORBCOMM Satellite Transceiver Set

Vaisala has integrated the satellite communication media called ORBCOMM with the HydroMet™ systems. Vaisala HydroMet™ systems support operation with the Panasonic KX-67101 transceiver. The ORBCOMM system uses Low-Earth-Orbiting (LEO) satellite, enabling the use of low power and small antenna in the transmitter terminals.

The ORBCOMM transmitters offer a low cost and robust way of transmitting data from remote sites almost in real-time. There is no need to install costly direct ground receiving stations. The local ORBCOMM operators offer a service where the data is transmitted to the users via Internet or via dedicated lines directly from their Gateway Earth Stations.

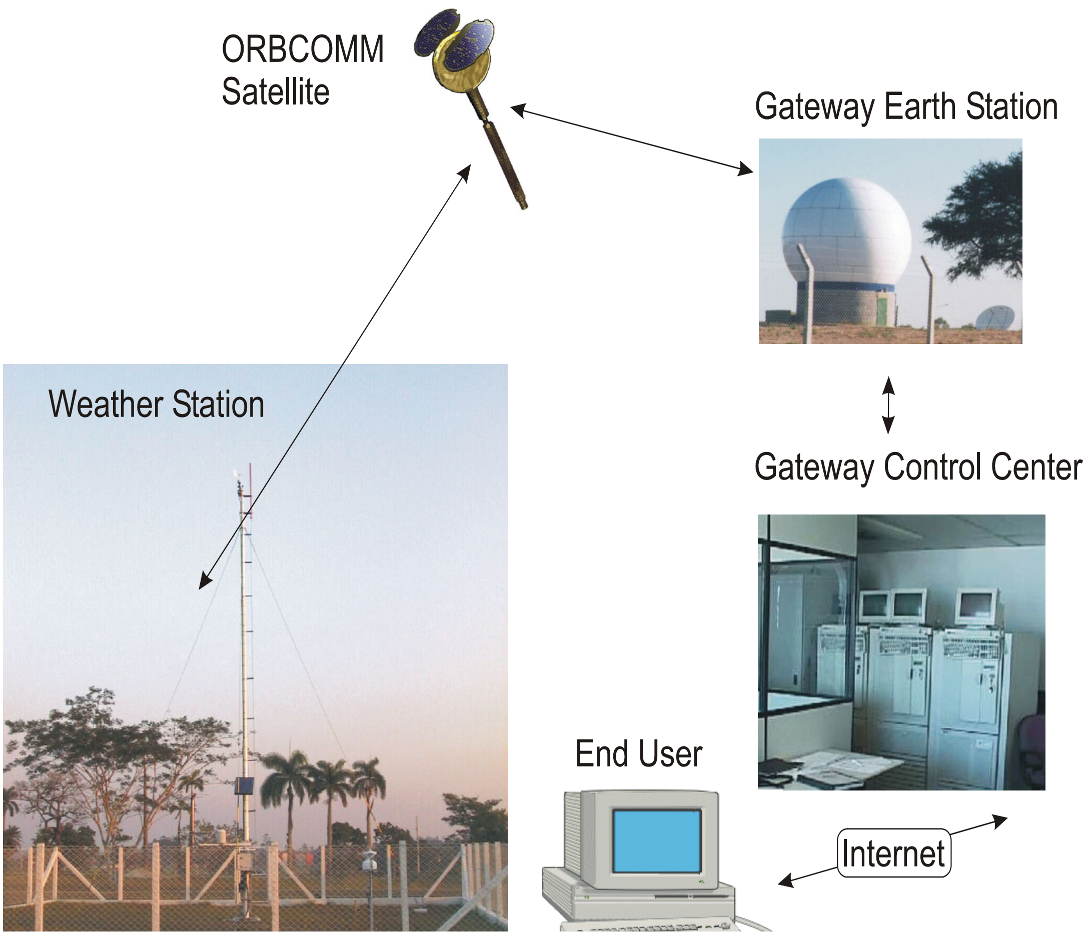

Figure 9 Weather Station System with ORBCOMM

ORBCOMM Data Communication Set includes the necessary cables inside the enclosure, coaxial surge arrester for the RF signal, and all mounting hardware. The antenna can be a standard whip antenna shown in Figure 10 on page 40.

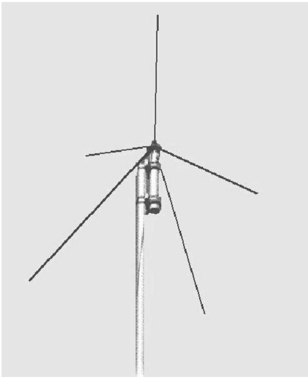

Figure 10 Antenna for the ORBCOMM Satellite Transmitter

The ORBCOMM Satellite Transmitter is connected to an RS-232 communication port. It is recommended that you install the RS-232 communication module on the QML logger to keep the COM0 port free for the service use. By default, the RS-232 communication module is connected to the MOD1 port. For the wiring diagram, see Chapter 7, Technical Data, on page 337.

GOES Satellite Transmitters

Figure 11 GOES Satellite Transmitter

The Satellite Radio Transmitter makes telemetry possible through the GOES, Meteosat, Argos, and SCD satellites. The transmitter has to be mounted inside the enclosure. The satellite transmitter is provided with the necessary cables, antenna, coaxial surge arrester for RF-signal, and all mounting hardware.

The transmitter has Temperature Compensated Crystal Oscillator (TCXO) which provides the long-term frequency stability necessary to keep the transmitter on the assigned frequency. Use of a GPS receiver, either embedded or external, offers additional time management stability. The transmitter is extremely efficient with power and has low power consumption during a transmission, less than 2.75 A.

When connecting GOES Transmitter to the QML logger, you need to use the terminal strip to provide power for the transmitter. For the wiring diagram, see Chapter 7, Technical Data, on page 337.

GOES Crossed Yagi Antenna

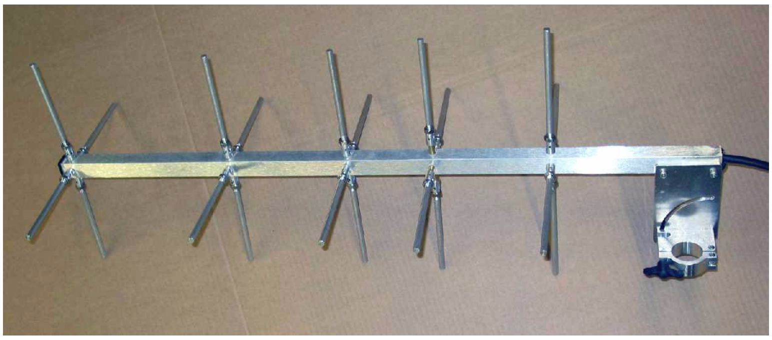

Figure 12 GOES Crossed Yagi Antenna

GOES Crossed Yagi Antenna is a lightweight, all-weather antenna designed to operate in the 401 to 402 MHz frequency range. This antenna radiates right-hand circular polarized energy in a pencil beam pattern with a beam width of approximately 45 degrees measured at the half-power points. The input impedance is 50 ohms, and the VSWR across the operating frequency range of the antenna is better than 2:1. Antenna gain is approximately 10 dB and is designed to provide optimum signal levels at the GOES satellite when connected to the 10 W GOES radio transmitter. The antenna is designed to be conveniently disassembled for transport.

CHAPTER 3
CONFIGURING MODULES AND SENSORS

This chapter contains additional information on setup management, instructions for configuring optional hardware, as well as configuration and measurement control information for different sensors and serial sensor interfaces.

Managing Setups

This section provides additional information on setup management.

Setup Memory Size

The size of the setup memory is 224 kilobytes, allowing you to create also larger setups. The internal logging memory size is 1 638 400 bytes.

Executing a Setup File from CompactFlash Memory Card

You can execute a setup file directly from certain CompactFlash (CF) cards. This applies only to CF cards purchased from Vaisala. Executing a setup from a CF card is especially useful when testing new setups.

NOTE	If the CompactFlash memory card has not been formatted, it must be formatted before use. In MAWS versions 6.00 and later, it is recommended that you format the card in a Windows PC. The file system to use is FAT (not FAT32). Also, do not select the quick format option. To format the CF card in the QML logger, insert it into the CF slot of the QML logger. Give the **EXTFS** *ERASE* command. After the card has been formatted, you can remove it from the slot and carry out the copying of the setup file(s) in your PC.

Typically, a setup is created using Lizard Setup Software and uploaded into the QML logger using the AWS Client software. You can also test new setups in an alternative way. After you have created a new setup, you can copy the setup file (or several files) to the CF memory card to be executed in the QML logger.

This requires that the boot code version of your QML logger is 4.07 or higher. The boot code is independent of the firmware software. In order to check the boot code version, give the **SYSINFO** terminal command. In case your boot code version is lower than 4.07 and you wish to use this new feature, contact Vaisala. You will need to send your logger to Vaisala for the update.

The QML logger supports external memory cards of up to 2 gigabytes. Cards can be read directly in the PC. Several different types of readers are commercially available: internal PCMCIA reader as well as external readers to be connected to USB or parallel port of a PC. It is recommended that you only use cards purchased from Vaisala that have been tested to function in harsh environments.

The setup file can be located in any logger directory but in order not to exceed the maximum path length, Vaisala recommends that you store the file either in the root directory or in the first one below the root.

NOTE	The maximum number of characters allowed in the path name is eight (8). The maximum number of characters allowed in the file name is eight (8), and the maximum number of characters allowed in the file name extension is three (3), as in Lizard Setup Software.

If you want to test several setups, you can copy all of them to the CF card at the same time. However, only one setup file will be executed at a time.

In order to test a new setup by executing the setup file from the CF card, proceed as follows after you have formatted the CF card and copied the setup file:

1. Check that the system includes a boot code version 4.07 or higher. Use the **VER** command or the **SYSINFO** command to check the version.
2. Insert the CompactFlash card into the CF card slot.
3. To check that a setup exists, give the following terminal command. Note that a UNIX/LINUX type file system with only one root is used, and thus the CF card is automatically mounted on the /Ext/ directory path.

   dir /Ext/<*pathname*>

where

pathname	=	Directory used

4. Use the **reset** command to launch a new startup with the setup file to be tested:

   reset [*delay*] /Ext/<*setup file*>

where

delay	=	Period of time (in seconds) before the system resets. The value of the delay parameter can be 0, which causes an immediate reset.
setup file	=	Path and file name without drive identification

For example, if you have copied the setup501.adc setup file to a directory named *test*, give the following command to launch the new startup:

reset 0 /Ext/test/setup501.adc

NOTE

Note that the directory separator (/) is not the same as in the DOS prompt.

Make sure you enter the pathname as presented in the example above (/Ext/).

An example output from the QML logger after selecting a setup file to be executed from the CF card is as follows:

```
/ W> dir /Ext/TEST

 .rwxd 13:11:40 21.03.2005  0
 ..rwxd 13:11:40 21.03.2005  0
 setup501.adcrw-- 12:26:28 21.03.2005  1776

Files shown: 3  Bytes used: 1776


/ W> reset 0 /Ext/TEST/setup501.adc
Resetting device in 0 seconds
COM0: QML201 Startup - Warm Boot
Serial #        : Z402074
Hardware        : Rev B-001
Software        : 5.01     Checksum  : 00000000
Boot sw version : 4.07
System RAM      : 2048kB
Free memory     : 1768kB
Internal temp.  : 23.15'C
Active errors   : NO
Active warnings : NO
Piggyback - 0   : DSU232 rev: B serial no: 008232
Piggyback - 1   : DSU232 rev: B serial no: 003523
NEW flash checksum (02400000 to 025FFFFF) is BBCC1130
Found  setup /Ext/TEST/setup501.adc

Starting to execute setup

Setup running ok
```

NOTE	You can also copy a setup file to the CF card and take it into use by giving only one command. When entering the following command, the QML logger will overwrite the original setup file in the /Cfg directory. The new setup will thus be available after each restart. **reset** [*delay*] /Ext/<*setup file*> copy For example, to copy and to take into use the *setup501.adc* setup file, give the following command: **reset** 0 /Ext/test/setup501.adc copy

Executing an Alternative Setup File from CompactFlash Memory Card

You can execute an alternative setup file from the CompactFlash (CF) memory card. This is especially useful when upgrading new setups remotely.

An alternative setup file will be executed if the primary setup file in the /Cfg directory fails.

NOTE	You must set a specific static parameter to enable this. This also requires that there is a CF memory card in the QML logger.

Proceed as follows:

1. Check that the system includes a boot code version 4.07 or higher. You can use the **VER** command or the **SYSINFO** command to check the version.
2. Insert the CompactFlash card into the CF card slot.
3. To check that a configuration exists, give the following command:

   dir /Ext/<*pathname*>

where

pathname = Directory used

4. To set a specific static parameter to point to an alternative setup file (script), use the **SPSET** command. **Alternativescript** is the name of the static parameter to be used to store the path and the file name.

   SPSET alternativescript /Ext/<setup file>

For example, if you want the setup501.adc setup file in the test directory on the CF card to be executed in case the primary (probably remotely downloaded) setup file fails, use the following command:

SPSET alternativescript /Ext/test/setup501.adc

To verify that the path is correct, give the **SPSET** command. An output from the QML logger is as follows:

```
/ > SPSET

alternativescript = /Ext/test/setup501.adc
```

If an error occurs while executing the primary setup file, the QML logger will check whether the static parameter **alternativescript** is set. If so, the system will be reset. After the reset, the logger will check whether an alternative setup file exists. If the alternative setup file exists and is functional, the logger will execute it. Now you have access to the system again and you can replace the erroneous primary setup file with a new one. After the next reset, the QML logger will try to execute the primary setup file located in the /Cfg directory. An example output from the QML logger when the primary setup file fails is as follows:

```
COM0: QML201 Startup - Cold Boot
Serial #        : Z402074
Hardware        : Rev B-001
Software        : 5.01     Checksum  : 00000000
Boot sw version : 4.07
System RAM      : 2048kB
Free memory     : 1768kB
Internal temp.  : 23.23'C
Active errors   : NO
Active warnings : NO
Piggyback - 0   : DSU232 rev: B serial no: 008232
Piggyback - 1   : DSU232 rev: B serial no: 003523
NEW flash checksum (02400000 to 025FFFFF) is BBCC1130
Found setup /Cfg/failing.adc
Verifying setup file CRC (7DD7036A)...OK
Starting to execute setup
Erroneous setup file
Runtime error [line:16] Parameter error 'DSI486_1_0'
COM0: QML201 Startup - Cold Boot
Serial #        : Z402074
Hardware        : Rev B-001
Software        : 5.01     Checksum  : 00000000
Boot sw version : 4.07
System RAM      : 2048kB
Free memory     : 1768kB
Internal temp.  : 23.23'C
Active errors   : NO
Active warnings : NO
Piggyback - 0   : DSU232 rev: B serial no: 008232
Piggyback - 1   : DSU232 rev: B serial no: 003523
NEW flash checksum (02400000 to 025FFFFF) is BBCC1130
Found () setup /Ext/TEST/setup501.adc
Starting to execute setup
Setup running ok
```

Configuring Optional Hardware

This section contains information needed when configuring optional hardware. The optional hardware described here includes Vaisala digital I/O module and Nokeval 7470 Serial to Analog Converter.

Digital I/O Module

Vaisala digital I/O module is used to extend the I/O capacity of the QML logger. It is equipped with eight inputs and eight outputs, and it interfaces to the logger via the SPI connector. The module is located inside the enclosure with the QML logger, and it conforms to the same environmental immunity and emission standards as the logger.

Configuring Digital I/O Module

1. Add the QMI digital I/O module to the setup in the **Optional hardware** view of Lizard Setup Software. You can also select which outputs are controlled by timers and name the outputs in the **Description for...** text boxes; see Figure 13 on page 49.

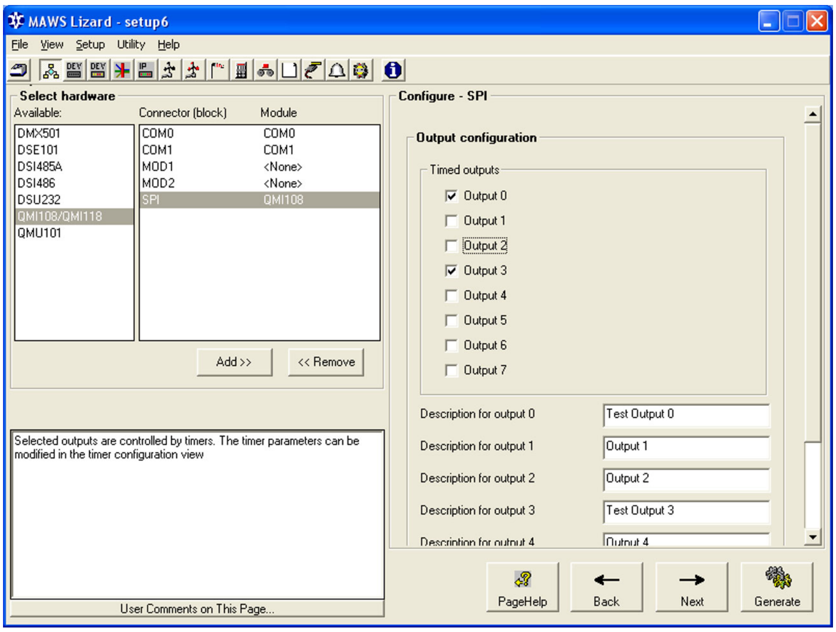

Figure 13 Optional Hardware View: Digital I/O Configuration

2. Proceed to the **Timers** view (see Figure 14 on page 50) to configure the on/off cycle for the selected outputs (0 and 3).

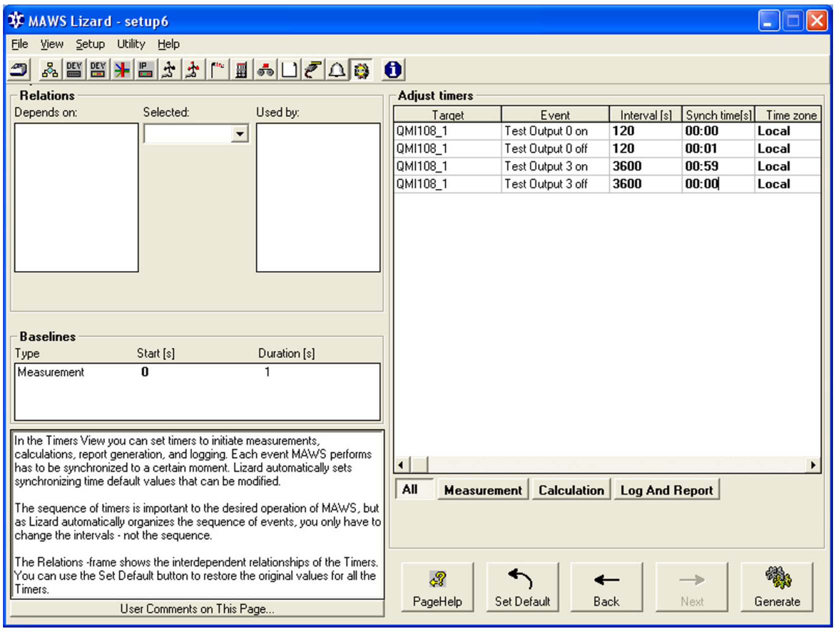

Figure 14 Timers View: Timer Configuration for the Digital Outputs

In the example in Figure 14 on page 50, the configuration is as follows:

- Test output 0 toggles between on and off in a one-minute cycle.
- Test output 3 is on for one minute before each full hour.

Alternatively, you can control the outputs with alarms, for example, by activating the output signal when the alarm condition is met.

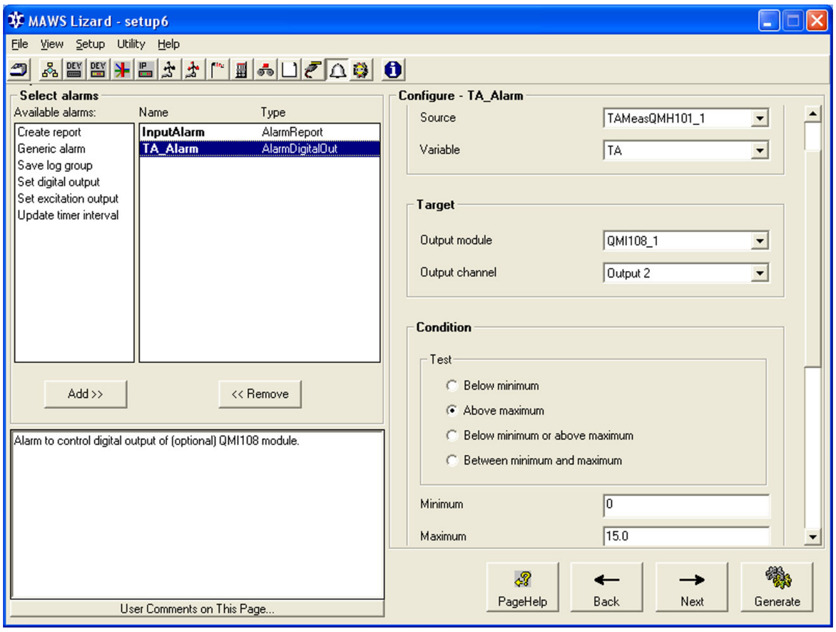

Figure 15 Alarms View: Controlling Digital Output Using an Alarm

In the example, the configuration activates output 2 when the air temperature measured by the QMH101 sensor rises above +15 °C.

Configure the alarm condition in the **Alarms** view; see Figure 15 on page 51 for an example.

The inputs of the digital I/O module can be used like any other application variables, that is, for the following:

- Reporting
- Logging
- As an alarm input

In the **Reports** view, the input statuses are available as individual variables **IN0** through **IN7** and combined as bits into a single status word **ALL**. The input module **status** is also available for the sensors; refer to Figure 16 on page 52.

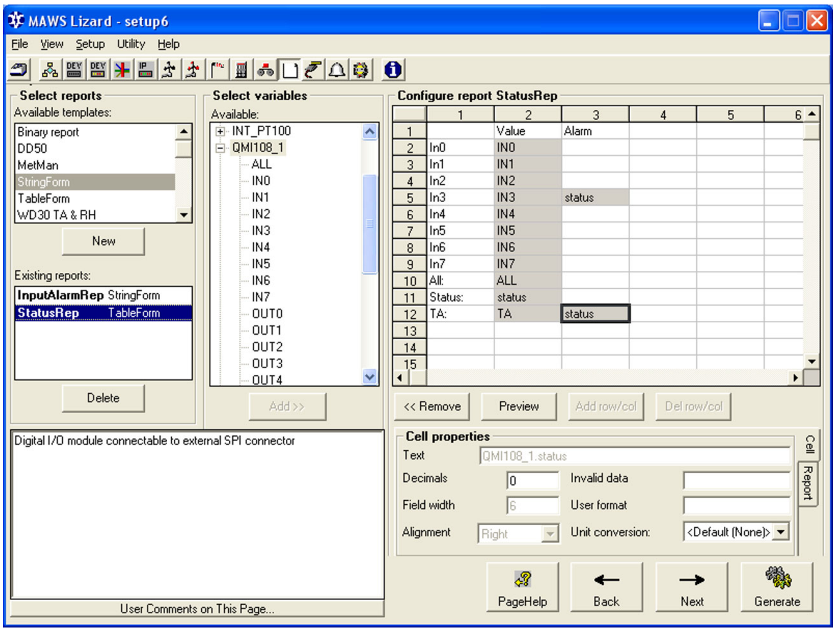

Figure 16 Reports View: Digital Inputs in Report

Reporting Outputs of the Digital I/O Module

You can also include the state of the output pins in the digital I/O module in your reports. To report the output states, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Go to the **Reports** view.
3. Open the available variables in the **Select variables** frame. Select **QMI108_1** to list all the possible variables. Variables **OUT0** ... **OUT7** provide the state of the individual output pin.
4. Add the variables into your report.

Nokeval 7470 Serial to Analog Converter

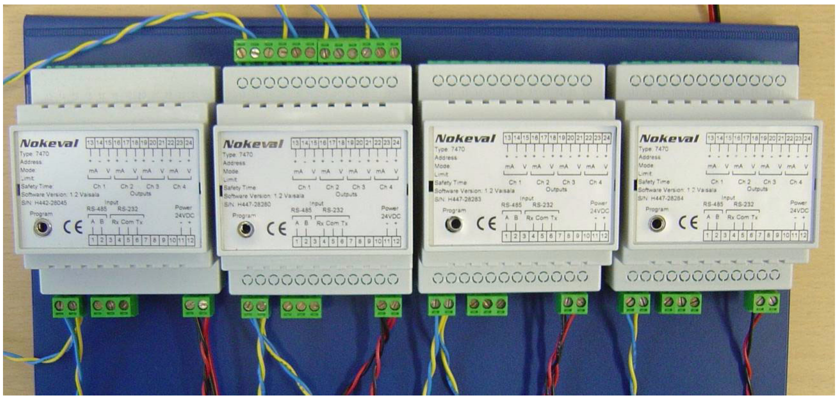

Figure 17 Nokeval Serial to Analog Converters

Nokeval 7470 Serial to Analog Converter is used for converting a predefined ASCII message to one or more voltage or mA outputs. This kind of conversion may be required, for example, when weather information is needed in automation systems where only voltage or current inputs are used instead of a serial data interface. The converter can be controlled through an RS-485 serial line.

As Nokeval converters are fully configurable, knowledge on these devices is required before taking them into use. It is possible to add one or more converters to a single RS-485 serial line with the QML logger.

For further details, see the applicable Nokeval user manual, available at www.nokeval.com.

Configuring Communication Port of Nokeval 7470

One or several Nokeval devices are connected to the QML logger through an RS-485 communication port. The device can be connected to the following RS-485 ports:

- Integrated RS-485 port (COM1)
- Isolated RS-485 port (2-wire or 4-wire connection)

In the **Optional hardware** view of Lizard Setup Software, configure the selected RS-485 port as follows:

1. In the **Line parameters** frame, leave the default values (9600, 8, 1, none), unless the Nokeval device is to be otherwise configured. See Figure 18 on page 54.
2. In the **Wiring options** frame, select the **Two wire connection** type from the list (isolated DSI485A only).
3. Leave all options unselected in the **Handshake options** frame.

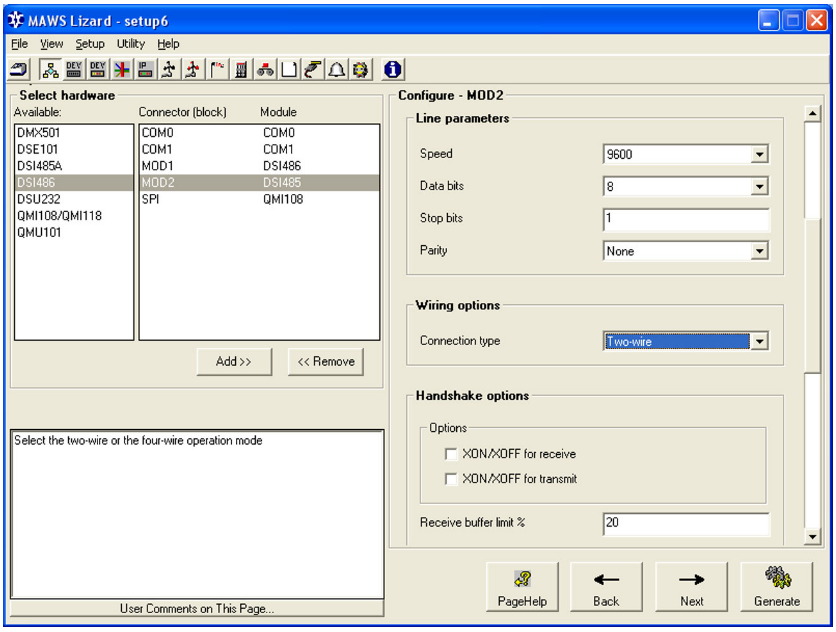

Figure 18 Optional Hardware View: Configuring Communication Port for Nokeval 7470 (1/2)

4. In the **Transmit control** frame, select **Transmit control enabled** and set the timing as follows in the editable text boxes (see Figure 19 on page 55):

   - Start delay (ms): 30

   - End delay (ms): 10

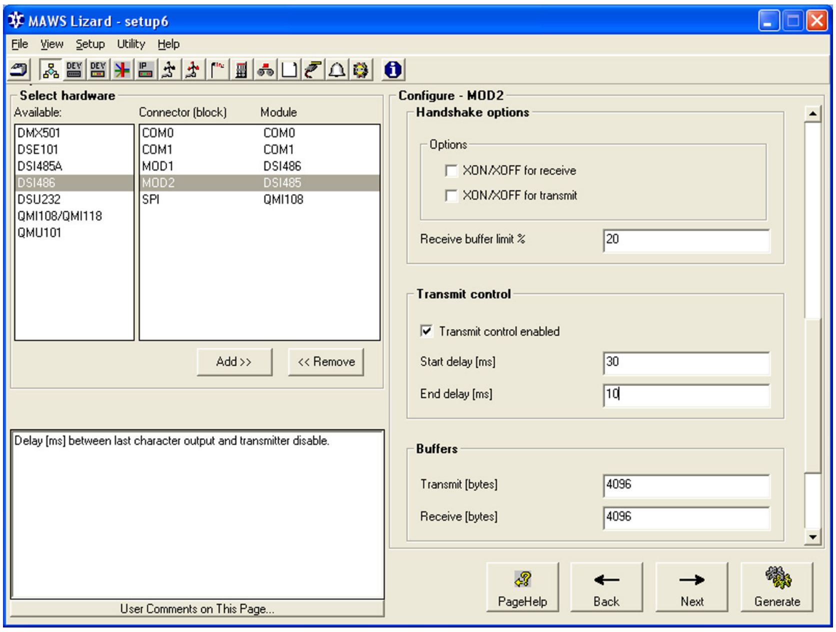

Figure 19 Optional Hardware View: Configuring Communication Port for Nokeval 7470 (2/2)

5. In the **Configure** frame, leave the default values.

6. Proceed to the **Devices** view as shown in Figure 20 on page 56.

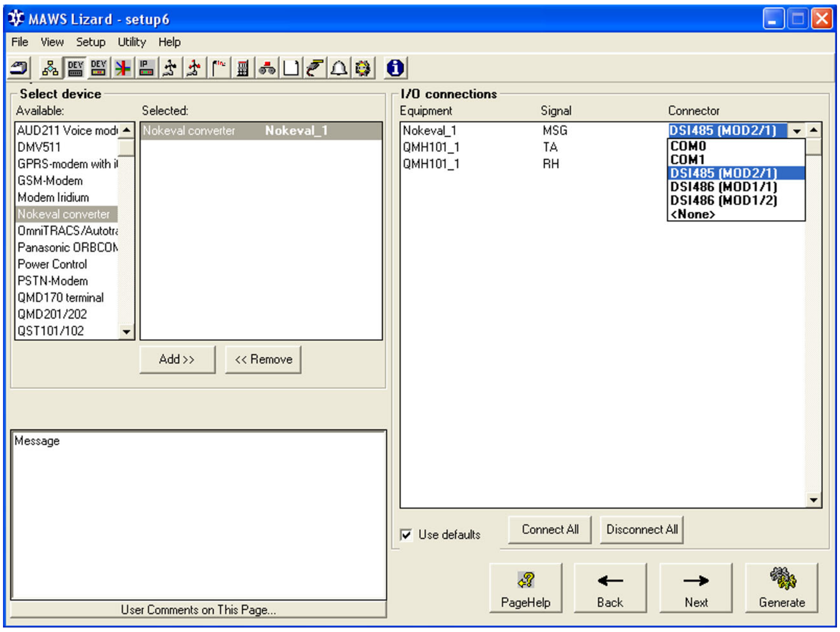

Figure 20 Devices View: Connecting Nokeval 7470 to Serial Port

7. Select *Nokeval converter* from the list and connect it to the selected serial port. Add only one Nokeval converter to the serial port here, even if there are several devices connected to the serial line.

NOTE	Do not connect more than one Nokeval Converter to one serial port in the **Devices** view. If there are several devices connected on the serial bus, they are using the same port and they are addressed in the reports as described in the following sections.

Configuring Nokeval 7470 Data Values

Data is sent to the Nokeval converter through a report containing a valid Nokeval command with data values. In the **Reports** view, create a separate StringForm report for each command to be sent. Proceed as follows:

1. Set a space character as the cell separator of the report.
2. The first cell must contain the address of the Nokeval device to which the command is sent, followed by a valid Nokeval command. For example, **\x80OUT CH 1**. See Figure 21 on page 57.

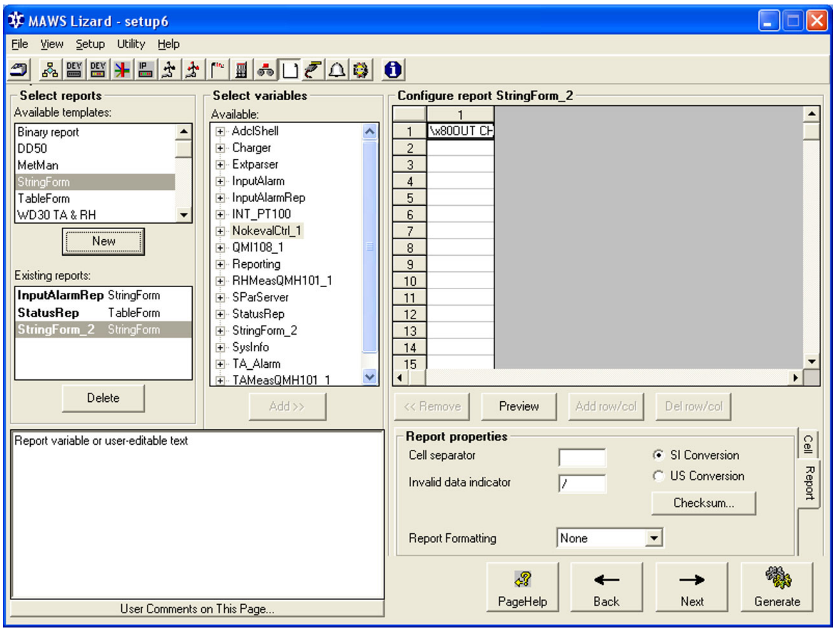

Figure 21 Reports View: Setting Up Report for Nokeval

Give the address as an 8-bit hexadecimal value where the highest bit is always set. The hexadecimal value is given in the report using the special character code **\x**. For example, if the device address is 0, then the first report field should start with **\x80**. If the device address is 2, then the field starts with **\x82**.

3. Give the Nokeval command, excluding the data value, directly after the address. Note that you may not enter a space or any other character between the address and the command. For detailed information about valid commands, refer to the appropriate Nokeval user manual at www.nokeval.com.

   The subsequent field(s) contain the data value(s) of the command. These may be, for example, measurement or calculation results. For details about invalid data handling and scaling of data, see section Handling of Invalid Data on page 59.

   The ETX character and the checksum are automatically added to each command by the QML logger, and therefore they are not allowed to be included in the report.

4. Proceed to the **Communications** view.

5. Link the reports to the Nokeval converter as shown in Figure 22 on page 58.

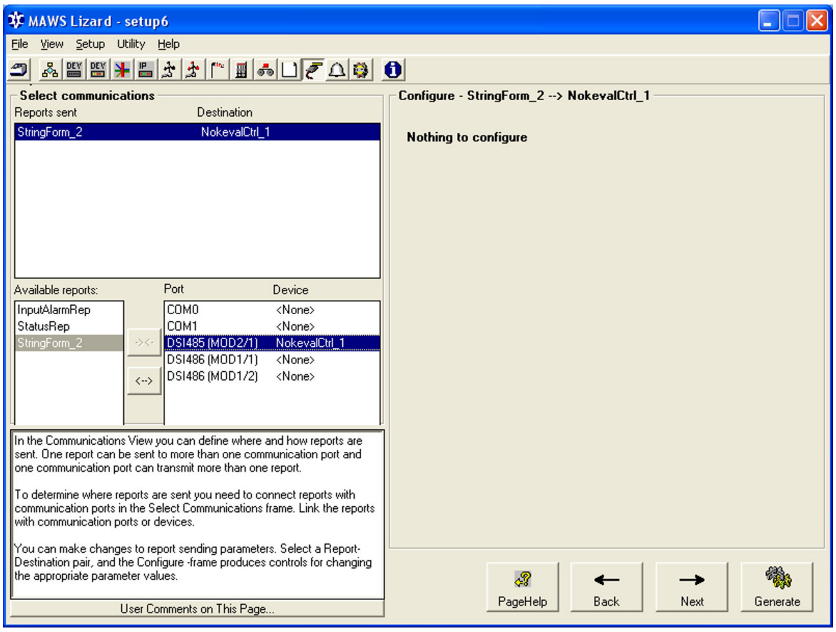

Figure 22 Communications View: Connecting the Report to Nokeval 7470

Scaling of Data

All data values must be properly scaled to fit into the output range of the Nokeval converter outputs. You can carry out the scaling using two alternative methods:

- Configuring the QML logger to scale the data. This is performed using a **User definable conversion**, configured in the Lizard **Calculations** view.
- Configuring the Nokeval converter to scale the data. Refer to the Nokeval user manual for details (at www.nokeval.com).

Handling of Invalid Data

The QML logger will not send a command if the data value is invalid, that is, the report includes the character '/'. The Nokeval converter can be configured to invalidate the output if no command has been received within a specified time (see the appropriate Nokeval user manual for details). Therefore, the system can be configured so that an invalid data value will invalidate the analog output of the converter. In this case, a separate report should be used to control each analog output, that is, the **OUT CH** command should be used, instead of the **OUT SCAN** command.

Alternatively, any specific data reading can be sent when the data value is invalid, by specifying an **invalid data** indicator for the data cell in the report.

Timing of Reports

Up to 32 reports can be generated successively at the same synchronization time. The QML logger will queue the commands and process them one by one.

If more than 32 reports are needed, the synchronization times of the report generation events must be adjusted so that no more than 32 commands will overlap each other.

Error Indicator

There is an available variable indicating the number of errors occurred within the previous queue of Nokeval commands, that is, while processing the queue of reports sent at the same synchronization time. When no errors have occurred, this variable has the value 0; otherwise, it indicates the number of failed commands. This variable of the NokevalCtrl software component is named **fails**. It can be used, for example, to generate an alarm in case an error occurs in communication.

Configuring Sensors

This section provides advanced information on configuration and measurement control for different sensors.

For basic information on setups, refer to Vaisala HydroMet™ Data Collection Platform User's Guide, Volume 2.

Vaisala Weather Transmitter

Vaisala Weather Transmitter is an integrated weather instrument that measures the following variables:

- Wind speed and direction
- Air temperature
- Relative humidity
- Precipitation
- Air pressure

The Weather Transmitter used with the QML logger has the following default configuration:

Table 5 Weather Transmitter Default Configuration

Checksum	ON
Automatic precipitation reset	ON
Mode	Autosend
Precipitation message	Time-based

For more detailed information on the measured parameters, see Weather Transmitter WXT520 User's Guide.

Configuring Vaisala Weather Transmitter

Communications Setup

Vaisala Weather Transmitter is connected to the QML logger through an RS-485 or RS-232 serial line. There are three alternative ways to connect the weather transmitter to the logger:

1. RS-232 autosend mode where the weather transmitter automatically sends measurement data to the QML logger. Only one weather transmitter per serial line is allowed.
2. RS-485 polled 2-wire connection. Several weather transmitter sensor connections are allowed on the same serial line.
3. RS-422 4-wire autosend.

The default data transmission parameters are the following: *9600 bps, 8 data bits, no parity, 1 stop bit*.

For more information on wiring of the RS-485 and RS-232 options, refer to the installation instructions. For more information on configuring serial ports, see Vaisala HydroMet™ Data Collection Platform User’s Guide, Volume 2.

If you need to check/modify the weather transmitter configuration, proceed as follows:

1. Start Lizard.
2. Open an existing setup in the **Setup** view or create a new one.
3. Configure the weather transmitter WXT520 hardware in the **Optional hardware** view as shown in Figure 23 on page 62.

   In this example, COM1 has been selected for use with the WXT520 sensor.

   It is recommended that you keep the default settings in the **Line parameters** frame.

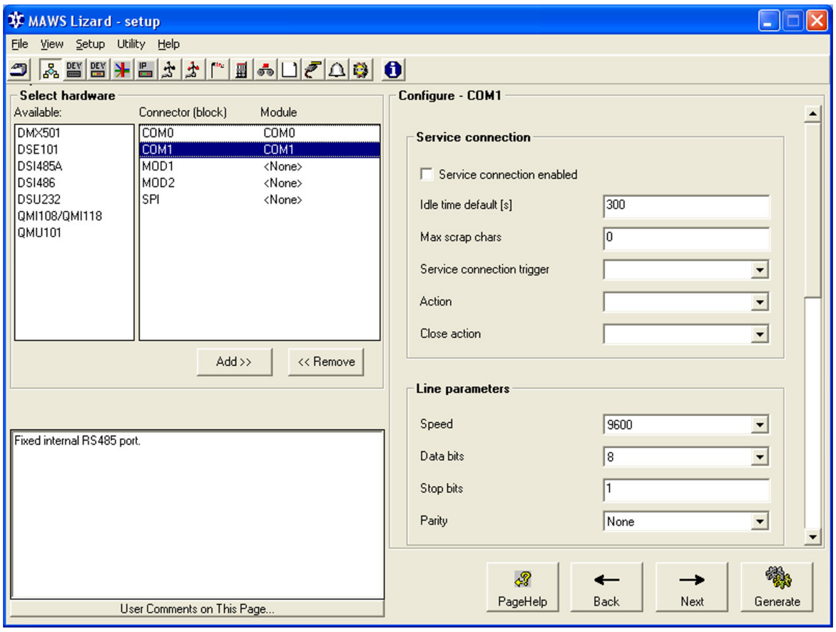

Figure 23 Optional Hardware View: Configuring Weather Transmitter Hardware

4. Proceed to the **Equipment** view as shown in Figure 24 on page 63.
   a. Scroll down the **Available** list box for WXT520.
   b. Click on the **Add** button to select WXT520.
   c. Double-click the **Connector** list for WXT520 in the **I/O Connections** frame in order to select the COM port.

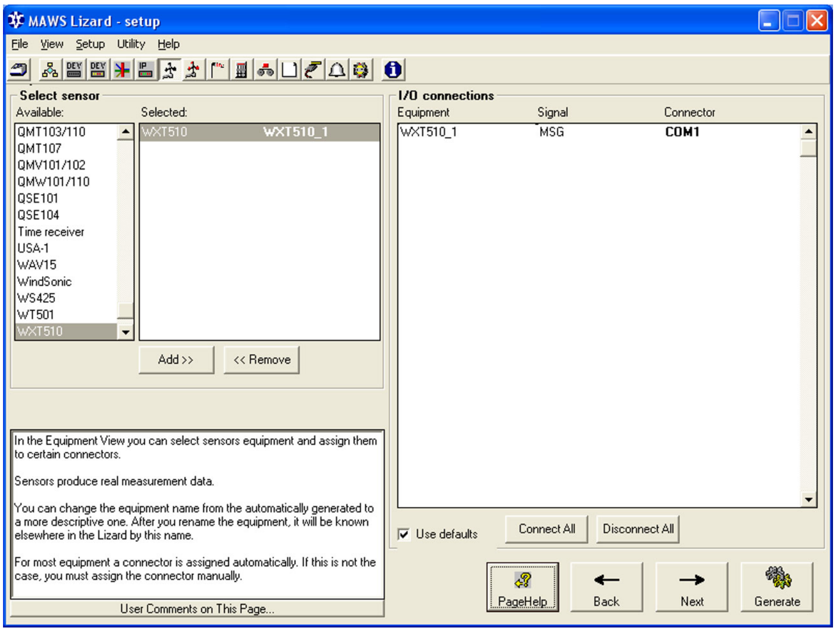

Figure 24 Equipment View: Selecting the I/O Connection for Weather Transmitter

5. Proceed to the **Measurements** view; see Figure 25 on page 63.

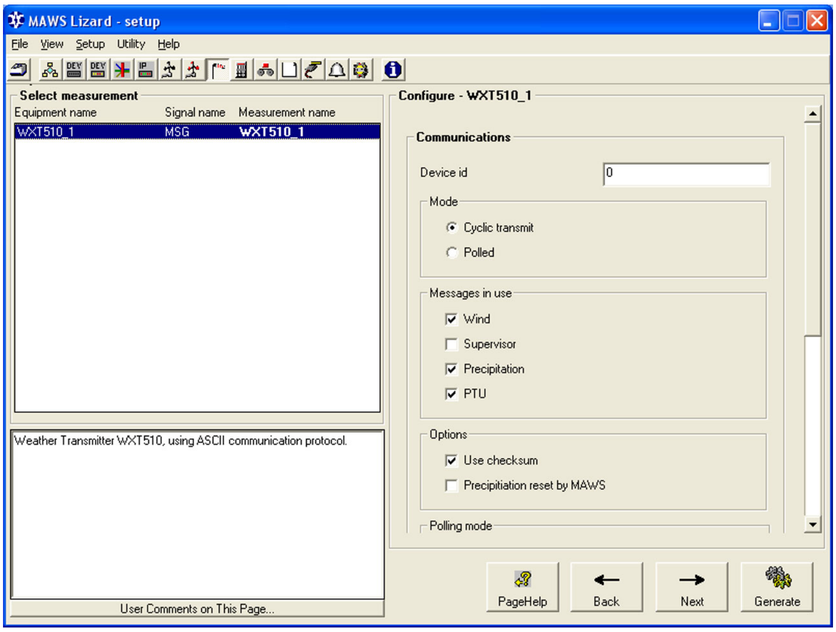

Figure 25 Measurements View: Configuring Weather Transmitter Communications Options

There are several parameter options you can set in the **Communications** frame of the **Measurements** view. Note that in general, the predefined settings are the most suitable for a typical QML logger setup.

In order to change a parameter option, you simply need to select the corresponding check box or remove its selection. The options you select must be enabled in the connected weather transmitter.

The following options are available:

Use the **Device ID** editable text box to distinguish the weather transmitter. For information on how to change the device ID, see Vaisala Weather Transmitter WXT520 User's Guide.

You can select the **Mode** in which the weather transmitter is operated. When you select the **Cyclic transmit mode**, the weather transmitter automatically sends data at predefined intervals. This mode can be used if only one weather transmitter is connected to the same serial line (only one device per serial port is allowed). If you select the **Polled mode**, the QML logger polls the transmitter at predefined intervals to retrieve data. Several devices can be connected to the same serial line.

Select also the **Messages** in use:

- Wind: wind speed and direction are received
- Supervisor: internal data is received from WXT520
- Precipitation: the amount, duration, and intensity of rain and hail are received
- PTU: air pressure, temperature, and humidity are received

NOTE	The options you select must be enabled in the connected weather transmitter.

You can also select the communication timeout in the **Polling mode** and the **Cyclic Transmit mode**; see Figure 26 on page 65.

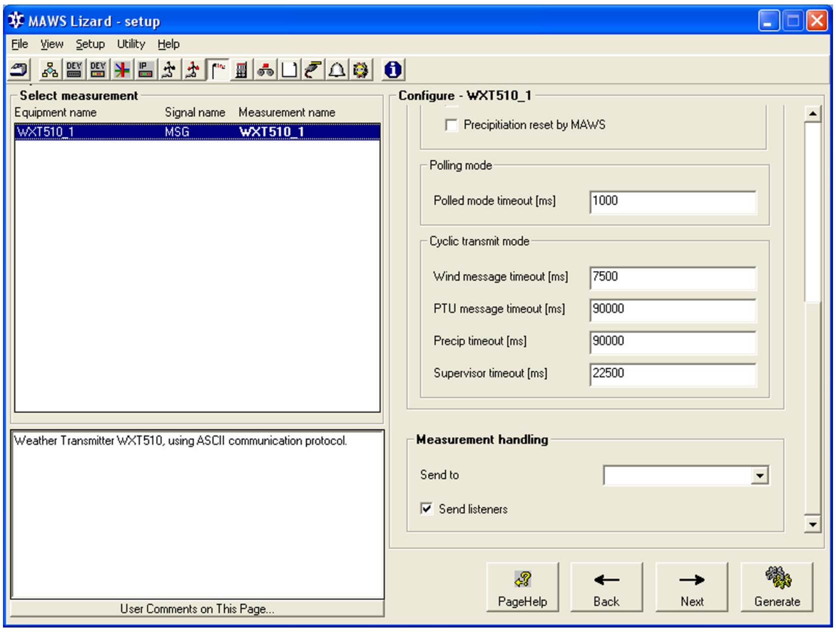

Figure 26 Measurements View: Configuring Weather Transmitter Timeout Parameters

With the **Polled mode timeout** parameter you can define the maximum length of time allowed for the weather transmitter to reply.

With the communication timeout parameters in **Cyclic transmit mode**, you can define a maximum interval between two messages. If the new message from the transmitter has not been received within the given time, the data in that message is then marked as invalid.

Using Sensor Variables in Reports

You can use the variables received from the weather transmitter in Lizard Setup Software like those from any other sensor.

To check which variables are available, proceed as follows:

1. Select the **Reports** view; see Figure 27 on page 66.
2. Create a new report or select an existing one to include the variables for weather transmitter WXT520.
3. Click WXT520_1 in the **Available** list box in the **Select variables** frame.

4. Lizard displays the items that are available for the source in question. The items are grouped under source names like files in directories. For more information on these variables, see Vaisala Weather Transmitter WXT520 User's Guide. When you click on an item in the **Select variables** list box, a brief description also appears in the information pane on the lower left.

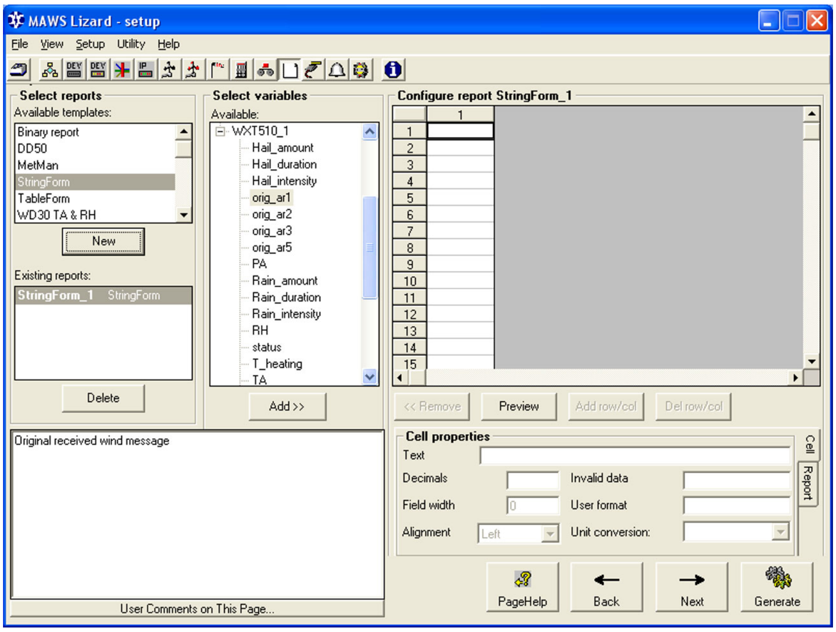

Figure 27 Reports View: Checking Weather Transmitter Variables

Gill WindSonic

To configure Gill WindSonic, proceed as follows:

In the **Equipment** view, select WindSonic from the **Available** list and connect it to the selected serial port. After that, the sensor is ready for use.

NOTE	The WindSonic transmits data continuously, 4 times per second. Therefore it reserves the serial line completely, and the same line cannot be used for other sensors.

For more information, refer to the appropriate Gill WindSonic user manual.

METEK 3D Ultrasonic Anemometer

The Ultrasonic Anemometer, model USA-1, is an ultrasonic wind sensor manufactured by Meteorologische Messtechnik (METEK) GmbH. The sensor measures wind speed and wind direction, or alternatively the three orthogonal wind components **x**, **y**, and **z**.

There are several serial output settings available for this sensor. The implementation is based on a standard version and the QML logger can receive the following message formats:

```
M:x =    10 y =    -3 z =    -1 t =  2203

H:x =    10 y =    -3 z =    -1 t =  2203

D:x =    10 y =    -3 z =    -1 t =  2203
```

The first character of the message indicates the heating status of the sensor:

M	=	Heating is OFF
H	=	Heating is ON
D	=	Heating should be ON but there is a defect

Heating is controlled by the ultrasonic anemometer, and it is based on user-configurable settings.

The message also includes the wind components **x**, **y**, and **z**, as well as a temperature component **t**. The unit of the wind components is cm/s and the unit of the temperature component is Celsius degrees with two decimals.

The example message above is thus interpreted as follows:

x	=	10	X component of wind is 10 cm/s
y	=	-3	Y component of wind is -3 cm/s
z	=	-1	Z component of wind is -1 cm/s
t	=	2203	Temperature is 22.03 Celsius degrees

NOTE	The unit for wind speed is cm/s in the message received from the ultrasonic anemometer. The QML logger automatically converts the unit to m/s, which is the standard unit for measuring wind speed.

The output frequency must be at least one message per second. A typical setting is ten messages per second.

For more detailed information on how to set the parameters for the ultrasonic anemometer, refer to the appropriate METEK user manual.

Configuring Ultrasonic Anemometer USA-1

Communications Setup

The ultrasonic anemometer is connected to the QML logger through an RS-422 or RS-232 serial line. Follow METEK's instructions when selecting cabling for the desired mode.

The default data transmission parameters are the following: *9600 bps, 8 data bits, no parity, 1 stop bit*.

For more information on wiring of the RS-485 and RS-232 options, refer to the delivery-specific wiring diagrams. For more information on configuring serial ports, see Vaisala HydroMet™ Data Collection Platform User’s Guide, Volume 2.

Configuring Ultrasonic Anemometer

Proceed as follows:

1. Configure the ultrasonic anemometer hardware in the **Optional hardware** view; see Figure 28 on page 69.
    - In this example, DSU232 has been selected for use (having an RS-232 interface).
    - It is recommended that you keep the default settings in the **Line parameters** frame.

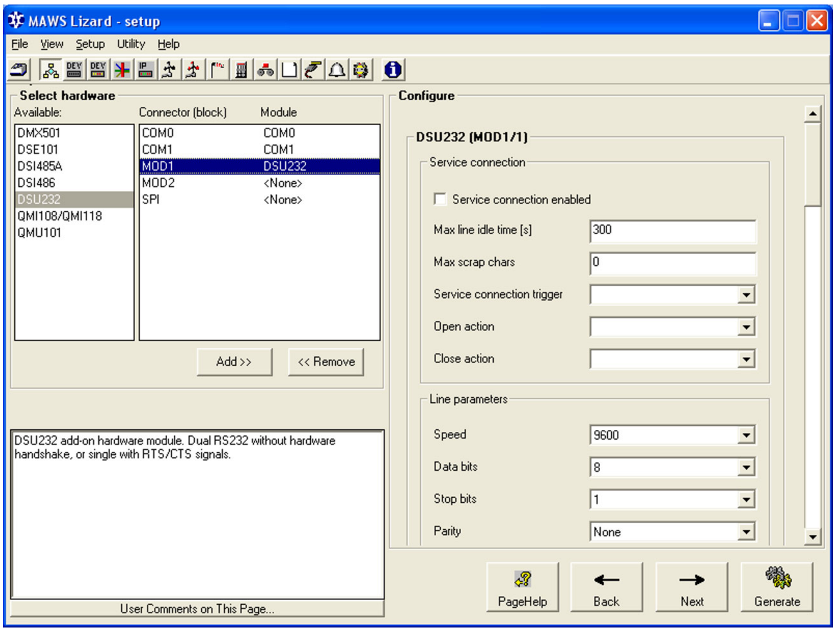

Figure 28 Optional Hardware View: Configuring DSU232 for Use with Ultrasonic Anemometer

2. Proceed to the **Equipment** view; see Figure 29 on page 70.

   a. Scroll down the **Available** list box for USA-1.

   b. Click on the **Add** button to select USA-1.

   c. Double-click the **Connector** list for USA-1 in the **I/O Connections** frame in order to select the COM port.

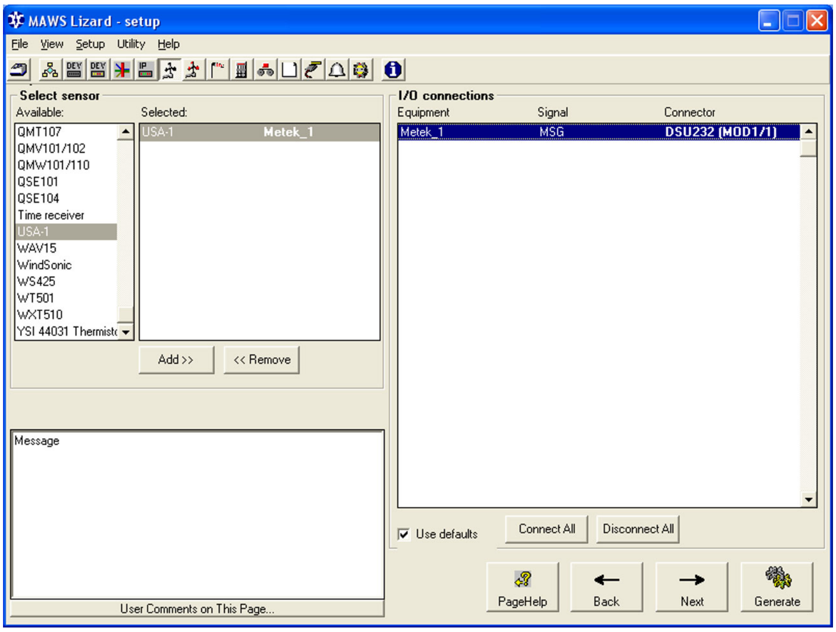

Figure 29 Equipment View: Selecting I/O Connection for Ultrasonic Anemometer

Using Sensor Variables in Reports

You can use the variables received from the ultrasonic anemometer in Lizard Setup Software like those from any other sensor.

To check which variables are available, proceed as follows:

1. Select the **Reports** view; see Figure 30 on page 71.

2. Create a new report or select an existing one to include the variables for the ultrasonic anemometer (Metek_1).

3. Click *Metek_1* in the **Available** list box in the **Select variables** frame.

4. Lizard displays the items that are available for the source in question. The items are grouped under source names like files in directories. When you click on an item in the **Select variables** list box, a brief description also appears in the information pane on the lower left.

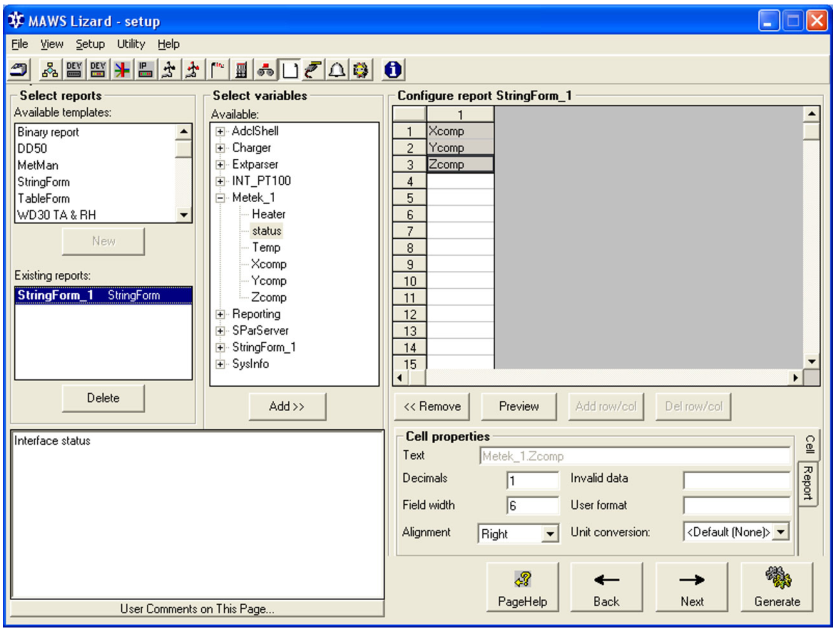

Figure 30 Reports View: Checking Ultrasonic Anemometer Variables

The available variables are as follows:

a. Heater, indicates the status of heating in the ultrasonic anemometer (Metek_1):

0	=	Heating is ON
1	=	Heating is OFF
2	=	Heating defect
3	=	Heating status unknown (for example, no message received from the ultrasonic anemometer)

b. Status, indicates the internal status of the QML logger interface for the ultrasonic anemometer, and the possible values. See the following table for value descriptions:

Value	Description
0	Sensor is working properly (OK)
1	Not measured yet
2	Interface not initialized (no message received after startup)
3	Communication timeout
4	Unknown data received
6	Sensor communication paused because pass-through service connection opened (or no message received after startup when Ignore frames option is selected)
30	Measurement/sensor has been manually disabled

c. Temp, temperature in Celsius degrees

d. Xcomp, x-component of wind in unit of m/s

e. Ycomp, y-component of wind in unit of m/s

f. Zcomp, z-component of wind in unit of m/s

Vaisala WA15 Set Sensors

Wind Vane WAV151 is connected via the digital I/O module to the QML logger. Anemometer WAA151 is connected directly to CHA/F. The following instructions describe the configuration work in detail.

NOTE	Before you can configure the mechanical Vaisala wind vane, you have to install the digital I/O module and connect it to the QML logger.

The wind vane is added to the setup in the **Equipment** view. Simply select **WAV15** from the sensor list. There is no need for other connections because the wind vane is always connected to the digital I/O module.

Typically, Vaisala wind vane is continuously powered from an external 12 VDC power source. In this default mode, there is no need to set or change any default parameters.

However, in low-power systems some power reduction can be implemented. In such case, you can select the **When measuring** option in the **Sensor power on** frame of the **Measurements** view; see Figure 31 on page 73.

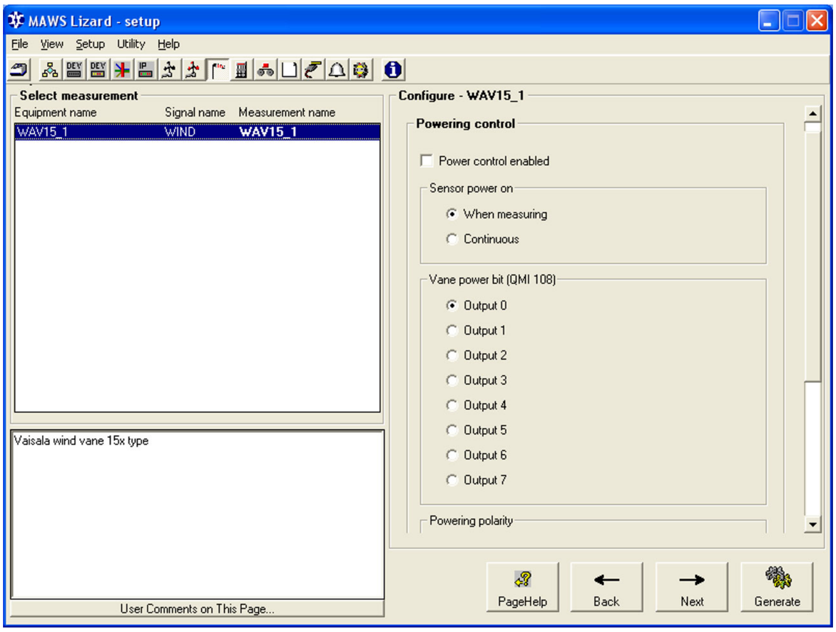

Figure 31 Measurements View: Configuring Powering Control

For the example wiring diagrams, refer to Chapter 7, Technical Data, on page 337.

If you select the **Power control enabled** option, you can choose the following:

- In the **Sensor power on** frame: Powering of the wind vane, either continuous or during measurement only
- In the **Vane power bit** frame: The output pin of the QMI digital I/O module to be used to control sensor excitation in the **When measuring** mode
- In the **Powering polarity** frame: The QMI digital I/O module output, which is typically normal but can be reversed. This depends on the hardware used to control powering.
- The **Pre-measure delay** field: The delay between power on and measurement start [ms]. This option provides stabilization time for sensor before measurement is made. The value should never be less than 60 milliseconds.

Vaisala HUMICAP® Humidity and Temperature Transmitters HMT330 Series

The HMT330 series is a transmitter product family designed for different environments. From the QML logger viewpoint, all versions look identical. The current implementation measures the following variables:

- Air temperature
- Relative humidity

For more detailed information on the measured parameters, see Vaisala HUMICAP® Humidity and Temperature Transmitter Series HMT330 User's Guide.

Configuring Humidity and Temperature Transmitters

The humidity and temperature transmitter is connected to the QML logger through an RS-232 serial line. There are two alternatives to connect the transmitter to the logger:

1. RS-232 autosend mode, where the humidity and temperature transmitter automatically sends measurement data to the QML logger. Only one HMT transmitter per serial line is allowed.
2. RS-232 polled mode, where only one HMT transmitter per serial line is allowed.

The default data transmission parameters are the following: *9600 bps, 8 data bits, no parity,* and *1 stop bit*.

For more information on wiring of the RS-232 options, refer to the delivery-specific wiring diagrams. For more information on configuring serial ports, see Vaisala HydroMet™ Data Collection Platform User’s Guide, Volume 2.

To configure the humidity and temperature transmitter, proceed as follows:

1. Configure the serial transmission hardware in the **Optional hardware** view as shown in Figure 32 on page 75.

   In this example, MOD1/1 on RS-232 module DSU232 is selected for use with the HMT transmitter.

   It is recommended that you keep the default settings in the **Line parameters** frame.

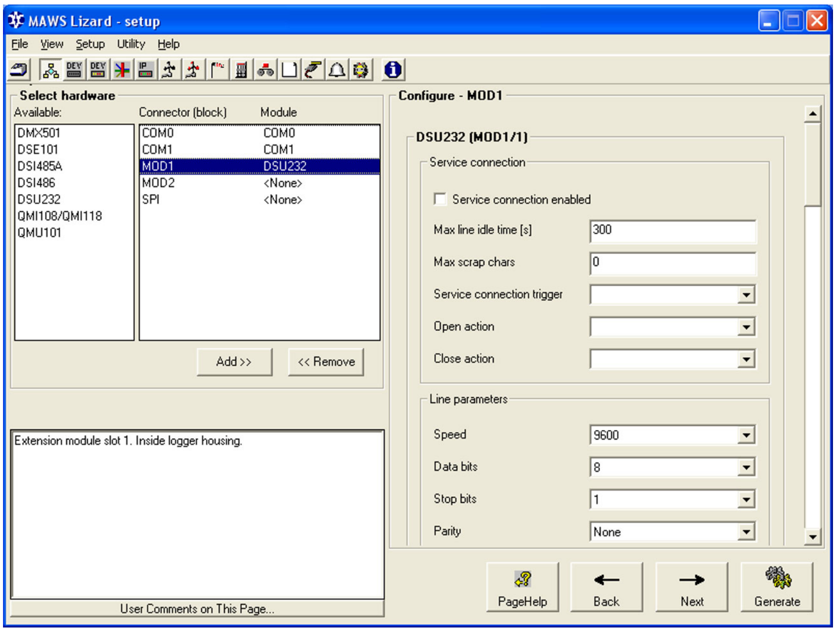

Figure 32 Optional Hardware View: Configuring Serial Transmission Hardware for Humidity and Temperature Transmitter

2. Proceed to the **Equipment** view; see Figure 33 on page 76.
3. Scroll down the **Available** list for HMT330.
4. Click the **Add** button to select HMT330.
5. Double-click the **Connector** list for HMT330 in the **I/O Connections** frame in order to select the COM port.

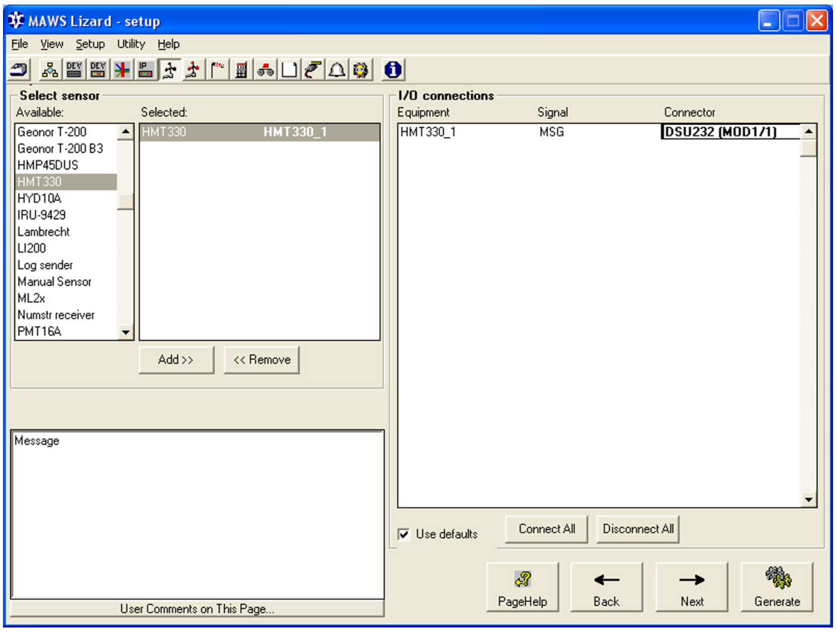

Figure 33 Equipment View: Selecting I/O Connection for Humidity and Temperature Transmitter

6. Proceed to the **Measurements** view; see Figure 34 on page 76.

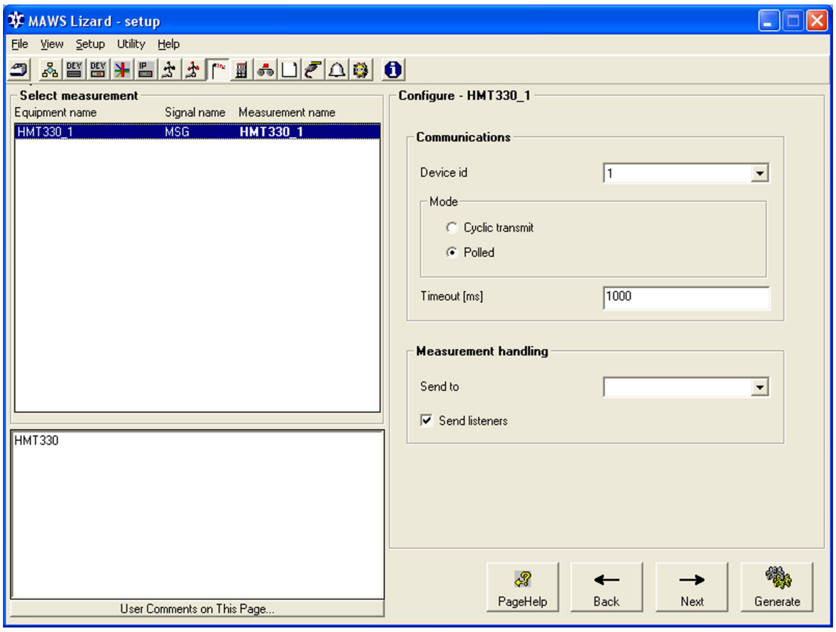

Figure 34 Measurements View: Configuring Humidity and Temperature Transmitter Communications Options

There are several parameter options you can set in the **Communications** frame of the **Measurements** view. Note that in general, the predefined settings are the most suitable for a typical setup.

In order to change a parameter option, you simply need to select or unselect the corresponding check box. The following options are available:

- Use the **Device ID** (editable) text box to distinguish the HMT transmitter. For information on how to change the device ID, see Vaisala HUMICAP® Humidity and Temperature Transmitter Series HMT330 User's Guide.
- Select the Mode in which the HMT transmitter is operated. When you select the **Cyclic transmit** mode, the transmitter automatically sends data at predefined intervals. If you select the **Polled** mode, the QML logger polls the transmitter at predefined intervals to retrieve data.
- In the **Polled** mode, by using the **Timeout** parameter you can define the maximum length of time allowed for the HMT transmitter to reply.
- In the **Cyclic transmit** mode, with the **communication timeout** parameters you can define a maximum interval between two messages. If the new message from the HMT transmitter has not been received within the given time, the data in that message is then marked as invalid.

Using Humidity and Temperature Transmitter Variables in Reports

You can use the variables received from the HMT transmitter in Lizard Setup Software like those from any other sensor.

To check which variables are available, proceed as follows:

1. Go to the **Reports** view.
2. Create a new report or select an existing one to include the variables for the HMT transmitter; see Figure 35 on page 78.
3. Click HMT330_1 in the **Available** list box in the **Select variables** frame.
4. Lizard displays the items that are available for the source in question. For more information on these variables, refer to Vaisala HUMICAP® Humidity and Temperature Transmitter Series HMT330 User's Guide.

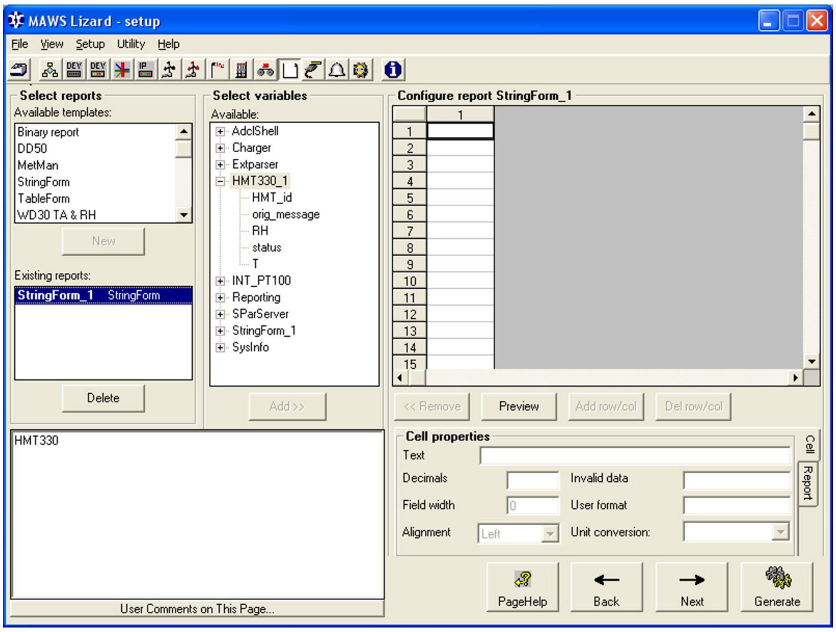

Figure 35 Reports View: Selecting Humidity and Temperature Transmitter Variables

Temperature Measurement PT1000

With the QML logger, you can measure Pt1000 elements. In the **Equipment** view, select the sensor from the **Available** list and connect the sensor to the desired connector, see Figure 36 on page 79.

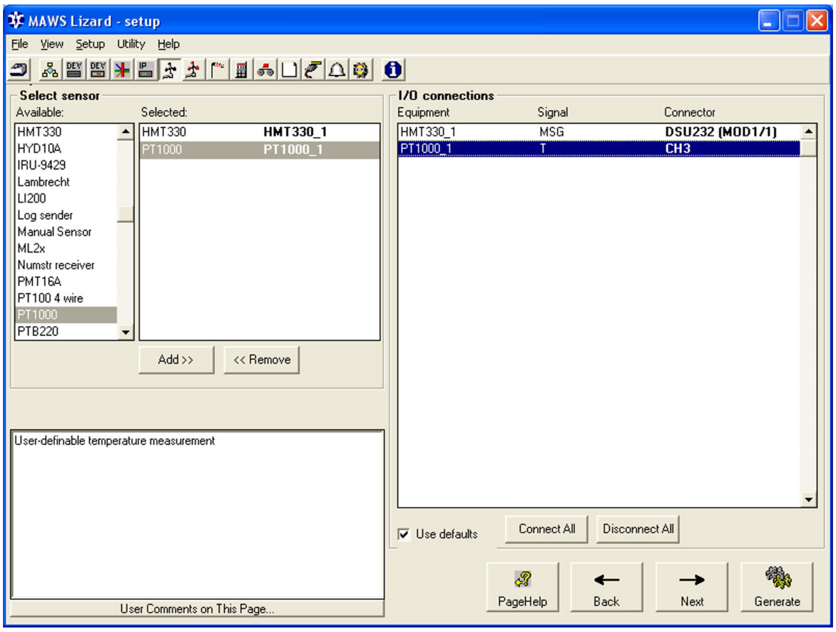

Figure 36 Equipment View: Selecting I/O Connections for PT1000

By default, the Pt1000 measurement is a 3-wire connection but you can change it to a 4-wire connection on the Advanced user level by changing the measurement type. Typically, you do not need to change the predefined Pt1000 parameters. Only the data validation parameters can vary, depending on the installation site. You can see all parameters as an Advanced user.

See Chapter 7, Technical Data, on page 337 for the example wiring diagrams for 3-wire and 4-wire connections of the Pt1000 probe.

Temperature Measurement PT100 in 3-Wire Connection

The Pt100 sensor in a 3-wire connection is rarely used but can be configured with Lizard Setup Software like any other sensor.

In the **Equipment** view, select the predefined Pt100 sensor from the list. Proceed then to the **Measurements** view and select a Pt100 measurement. In order to change to a 3-wire measurement, change the user level to Advanced. The first parameter indicates whether the measurement in question is a 3- wire or a 4-wire one. You do not need to change any other parameter default values.

Absolute Shaft Encoder

Absolute Shaft Encoder measures water level in a stilling well. It is based on a true absolute shaft encoder that uses SSI (Synchronous-Serial Interface) for communication. The QML logger is connected to the encoder via the QMI102 adapter module that reads data from the encoder and sends it as ASCII via the RS-232 or RS-485 line. QMI102 may be connected to COM1 or any other communication port that supports either RS-232 or RS-485. The QMI102 module is installed inside the enclosure.

Figure 37 Absolute Shaft Encoder

Configuring Absolute Shaft Encoder

To add the absolute shaft encoder into your system setup, proceed as follows:

1. Go to the **Equipment** view of Lizard Setup Software and select QSE104 from the **Available** list, see Figure 38 on page 81.
2. Connect QSE104 to an available serial port.

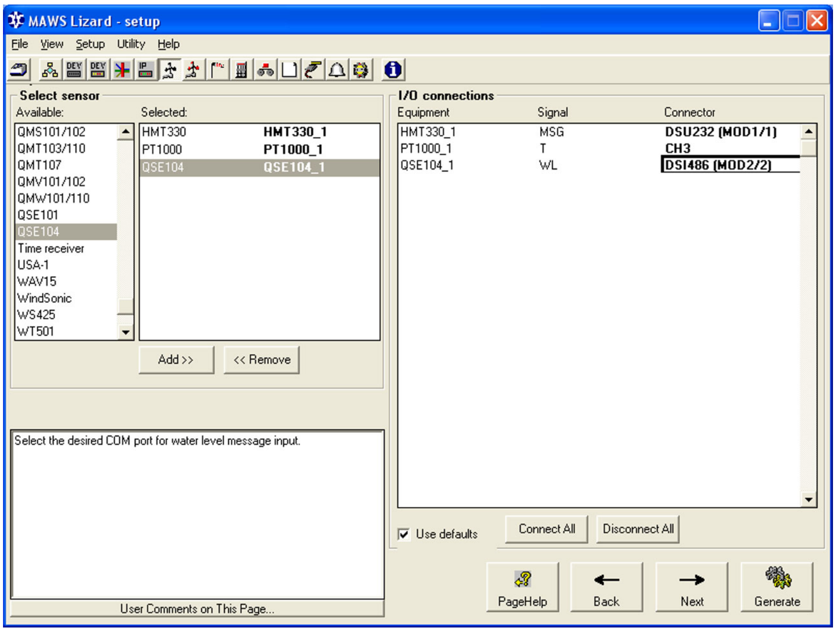

Figure 38 Equipment View: Absolute Shaft Encoder Connected to Optional DSI486 Module

3. Go to the **Measurements** view. All the parameters are predefined. If there is more than one device on the same RS-485 line, the device IDs must be different. See Figure 39 on page 82.

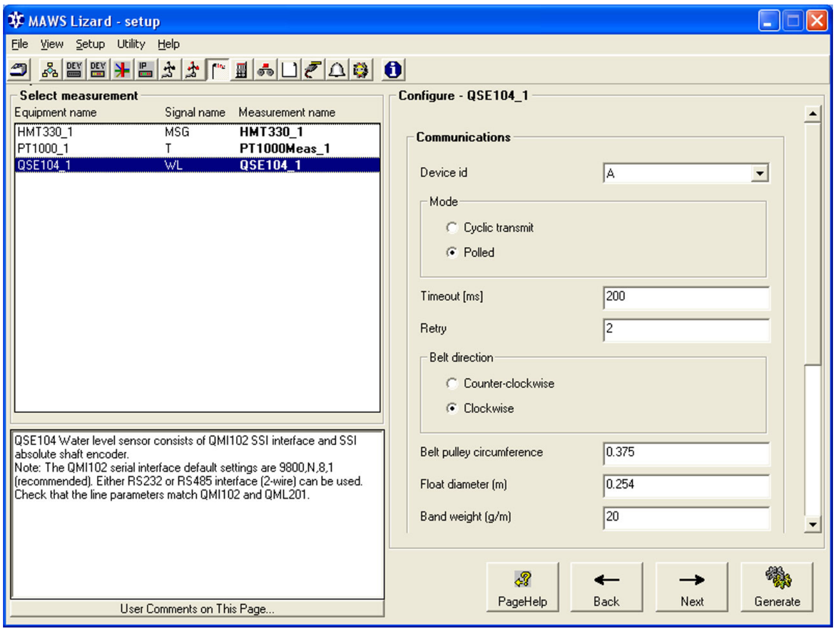

Figure 39 Measurements View: Communications for Absolute Shaft Encoder

Setting Current Water Level for Absolute Shaft Encoder

You can read or set the current water level, that is, the current data value of the shaft encoder, using the following terminal command:

waterlevel [*new value*]

If you enter the parameter new value, the current data value is set as the parameter value. If you do not enter the parameter, then the current data value is displayed.

For example, read the current water level using the following command:

```
/> waterlevel
water level = 12.3400 m
```

To set the current water level, for example, to 5.480 meters, issue the following command:

```
/> waterlevel 5.480
water level = 5.4800 m
```

Setting Current Water Level for Multiple Shaft Encoders

If you have more than one absolute shaft encoder in your system setup, you must define the device number when entering the **waterlevel** command. To see which command is associated with a certain device, click the **waterlevel commands** button in the **Communications** frame of the **Measurements** view in Lizard, as seen in Figure 39 on page 82. A window opens, displaying the QSE devices and the commands associated with them, see Figure 40 on page 83.

Figure 40 Lizard Waterlevel Window

For example, to set the current waterlevel to 6.120 meters for the QSE104_2 sensor, give the following terminal command:

```
/> waterlevel_2 6.120
water level = 6.1200 m
```

Leaf Wetness Sensor

The leaf wetness sensor is configured in the **Equipment** view in MAWS Lizard. To configure the sensor, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Equipment** view.
3. Select the QLW102 sensor from the **Available** list and click **Add** to insert the sensor into the setup.
4. If the connector is used, MAWS Lizard prompts 'Selected HW resource is already in use. Are you sure you want to make this connection'.

    a. Select **No**.

    b. Double-click the text **<None>** in the **Connector** column.

    c. Select an available connector.

Using Leaf Wetness Variables in Reports

Leaf wetness sensor outputs raw data that can be included in reports or stored into log files. Additionally, MAWS Lizard creates an alarm component that reports on/off information about leaf wetness.

To check which variables are available, proceed as follows:

1. Go to the **Reports** view.
2. Create a new report or select an existing one to include the variables for the leaf wetness sensor.
3. Click **LWMeasQLW102_1** and **QLW102Wetness_1** in the **Available** list box in the **Select variables** frame. Lizard displays the items that are available for the source in question.

The LWMeasQLW102_1 variables are outlined in Table 6 on page 84.

Table 6 LWMeasQLW102_1 Variables

Variable	Description
LW	Raw value from 0 to 15: 0 = dry, 15 = wet
Status	The status of the measurement.

QLW102Wetness_1 has one variable: **status**, indicating whether the leaf is wet or dry (1 = wet, 0 = dry). You can set the limit between wet and dry in the **Alarms** view. The default value is **7** (based on the **LWMeasQLW102_1** measurement) and it can be changed in the **Input threshold for indication** field.

Fuel Moisture Sensor

The fuel moisture sensor measures the moisture content of the material on the forest floor or other natural area to help forest managers assess the fire danger. The humidity measurement is based on electrical capacitance. A thermistor, located in the dowel where it fastens to the base, measures the temperature of the dowel giving the estimated temperature on the forest floor. The temperature measurement uses a 10th degree polynomial to convert the NTC resistor value to temperature.

To configure the fuel moisture sensor in Lizard, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Equipment** view.
3. Select the **QFM101** sensor from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. If the connector or connectors are used, MAWS Lizard prompts 'Selected HW resource is already in use. Are you sure you want to make this connection'.

    a. Select **No**.
    b. Double-click the text **<None>** in the **Connector** column.
    c. Select an available connector.

Using Fuel Moisture Variables in Reports

In Lizard, the temperature of the fuel moisture sensor is reported from the NTCpoly component. To check which variables are available, proceed as follows:

1. Go to the **Reports** view.
2. Create a new report or select an existing one to include the variables for the fuel moisture sensor.
3. Click **NTCPoly_1** in the **Available** list box in the **Select variables** frame.
4. Add temperature to the report.

Soil Moisture Sensor EC-5

Soil Moisture Sensor EC-5 uses capacitance to measure the dielectric permittivity of the surrounding medium. The volume of water in the total volume of soil influences most heavily the dielectric permittivity of the soil because the dielectric permittivity of water is much greater than that of the other constituents of the soil. Thus, when the amount of water changes in the soil, the EC-5 probe measures the change in capacitance (from the change in dielectric permittivity) that can be directly correlated with a change in water content. Circuitry inside the EC-5 probe changes the capacitance measurement into a proportional millivolt output.

MAWS software calculates the volumetric water content (VWC, m^3/m^3) values based on the sensor millivolt output. MAWS software uses calibration coefficients provided by the sensor manufacturer. Calibration coefficients are provided for mineral soil, potting soil, and rock wool. VWC of saturated soils is generally 40–60%, which is why MAWS uses the manufacturer's calibration coefficients, purposed for that range. VWC from 0 to 100% needs sensor-specified soil type calibration. MAWS does not support this feature as factory default.

EC-5 factory calibration equations (VWC, m^3/m^3) are as follows:

Mineral soil: 11.9 * 10-4 * mV - 0.401

Potting soil: 0.3 * 10-4 * mV - 0.334

Rock wool: 2.63 * 10-6 * mV2 + 5.07 * 10-4 * mV - 0.0394

The result of the equation is (VWC, m^3/m^3). Millivolt (mV) values of the equations are the sensor raw output values, which are used in the calculation. These coefficients are valid only when the data logger provides 2500 mV excitation voltage for the sensor. These equations are based on the information that VWC of saturated soils is generally 40–60%.

To configure the soil moisture sensor in Lizard, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Sensors** view.
3. Select the EC-5 sensor from the **Available** list and click the **Add** button to insert the sensor into the setup. Lizard automatically suggests channel 1 for the measurement.

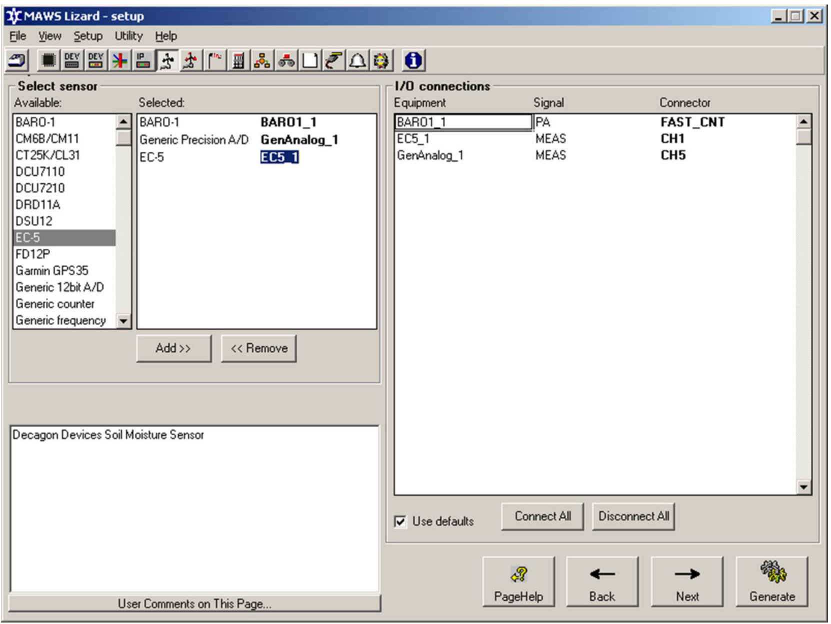

Figure 41 Adding EC-5 Measurement

4. If the connector or connectors are used, Lizard prompts "Selected HW resource is already in use. Are you sure you want to make this connection".
   a. Select **No**.
   b. Double-click the text <None> in the **Connector** column.
   c. Select the available connector. Valid connectors are CH0 ... CH3.

5. Proceed to the **Measurements** view.

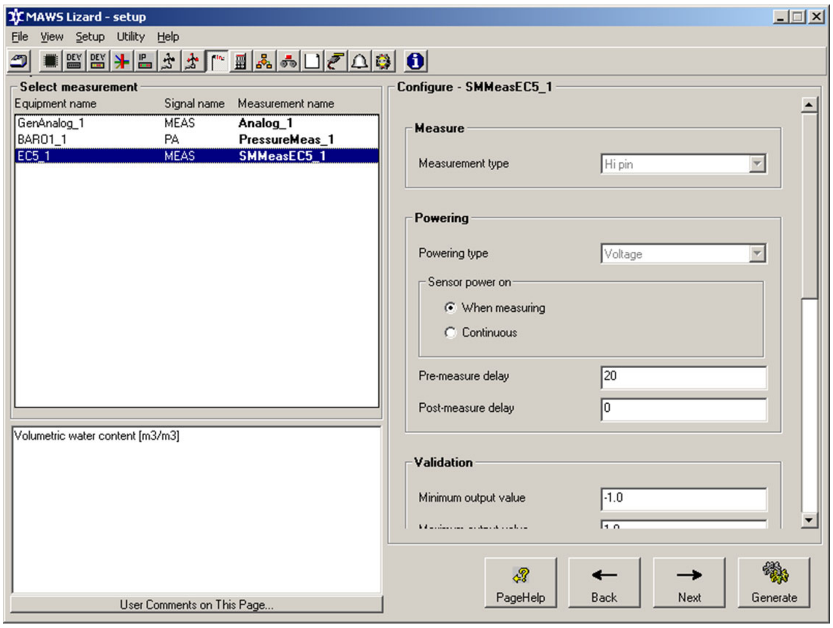

Figure 42 Configuring EC-5 Measurement

6. Change the soil type in the **Conversion** section, if needed. The available soil types are:

   - Mineral soil (default)
   - Potting soil
   - Rock wool

Using Soil Moisture Variables in Reports

To use soil moisture variables in reports, proceed as follows:

1. Go to the **Reports** view.
2. Create a new report or select an existing one to include the variables for the soil moisture sensor.
3. Click SMMeasEC5_1 in the **Available** list box in the **Select variables** frame. Lizard displays the items that are available for the source in question.

Ultrasonic Water/Snow Level Sensor IRU 9429

IRU 9429 is an ultrasonic sensor used in water/snow level measurement. The sensor includes internal temperature compensation, so no compensation component is needed in MAWS.

To configure the sensor in Lizard, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Equipment** view.
3. Select the **IRU 9429** sensor from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. If the connector is used, MAWS Lizard prompts "Selected HW resource is already in use. Are you sure you want to make this connection."

   a. Select **No**.
   b. Double-click the text **<None>** in the **Connector** column.
   c. Select an available connector.

Using Snow/Water Level Variables

To check which variables are available, proceed as follows:

1. Go to the **Reports** view.
2. Create a new report or select an existing one to include the variables for the ultrasonic sensor.
3. Click **SHMeasIRU9429_1** the **Available** list box in the **Select variables** frame.
4. Add snow height to the report.

NMEA GGA Receiver

The NMEA GGA receiver is used to parse a compass message from a GPS device.

To configure the sensor in Lizard, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Additional sensors** view.
3. Select **NMEA_GGA** from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. Select the port in the **Options** frame and set the timeout. The recommended value for the timeout is more than 1.5 times the time between two consecutive messages.

Using GGA Variables

To check which variables are available, proceed as follows:

1. Go to the **Reports** view.
2. Create a new report or select an existing one to include the variables for the compass.
3. Click **NMEA_GGA_1** the **Available** list box in the **Select variables** frame.
4. Add the variables of interest to the report.

For more information about the available variables, see the device manufacturer's documentation or the NMEA-0183 standard. The variables are presented in Table 7 on page 90.

Table 7 GGA Variables

Variable	Description
Ant_height	Altitude in meters
DPGS_el_time	Time in seconds since last DGPS update
DGPS_id	DGPS station ID number
Geoid_height	Height of geoid (mean sea level) above WGS84 ellipsoid

Table 7 GGA Variables (Continued)

Variable	Description
GPS_fix_st	Fix quality: 0 = invalid 1 = GPS fix (SPS) 2 = DGPS fix 3 = PPS fix 4 = Real Time Kinematic 5 = Float RTK 6 = estimated (dead reckoning) (2.3 feature) 7 = Manual input mode 8 = Simulation mode
HDOP	Horizontal dilution of position
Lat_NorS	Latitude (North or South)
Latitude	Latitude in degrees
Lon_EorW	Longitude (East or West)
Longitude	Longitude in degrees
no_of_sat	Number of satellites being tracked
status	Status of sensor interface
UTCHours	Hours of UTC time
UTCMinutes	Minutes of UTC time
UTCSeconds	Seconds of UTC time

Synchronizing QML Logger Clock

It is possible to synchronize the QML logger clock according to the time given in the NMEA GGA message. When adding the NMEA_GGA receiver into the configuration, a static parameter **GPStimeSyncDiff** is automatically generated. The default value is 0, which means that the clock will not be synchronized. The value of **GPStimeSyncDiff** is the tolerance of time difference between the logger time and the time from the NMEA GGA message in seconds. For example, if **GPStimeSyncDiff** is set as 10, and the time difference between the logger clock and the time in the message is 10 seconds or more, the logger clock is synchronized.

NOTE	The time is not synchronized more often than once in a minute.

NOTE	If the number of satellites tracked is zero, the time will not be synchronized.

Vaisala All-Weather Precipitation Gauge VRG101

Vaisala All-Weather Precipitation Gauge VRG101 can be connected to the QML logger through a serial port using Lizard. To configure VRG101, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Additional sensors** view.
3. Select **VRG101** from the **Available** list and click the **Add** button to insert the sensor into the setup.

For more information about VRG101 configuration and the variables it outputs, see section Vaisala All-Weather Precipitation Gauge VRG101 on page 126 and the VRG101 User's Guide.

Vaisala Humidity and Temperature Probe HMP155

Vaisala HUMICAP® Humidity and Temperature Probe HMP155 provides reliable humidity and temperature measurement in a wide range of applications. Humidity measurement is based on the capacitive thin film polymer sensor HUMICAP®180R. Temperature measurement is based on resistive platinum sensors (Pt100). Both the humidity and temperature sensors are located at the tip of the probe, protected by a sintered Teflon filter.

HMP155 comes in active and passive output models, where the relative humidity voltage output is similar, but the temperature output is either active or passive (resistive). Active output version has either two analog outputs or an RS-485 digital output. Passive output version has 4-wire temperature measurement and an analog voltage output for relative humidity measurement.

To configure HMP155 with active output, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Additional sensors** view.
3. Select HMP155 from the **Available** list and click the **Add** button to insert the sensor into the setup.

4. Select the port in the **Options** frame and read carefully the information window in the lower left corner.

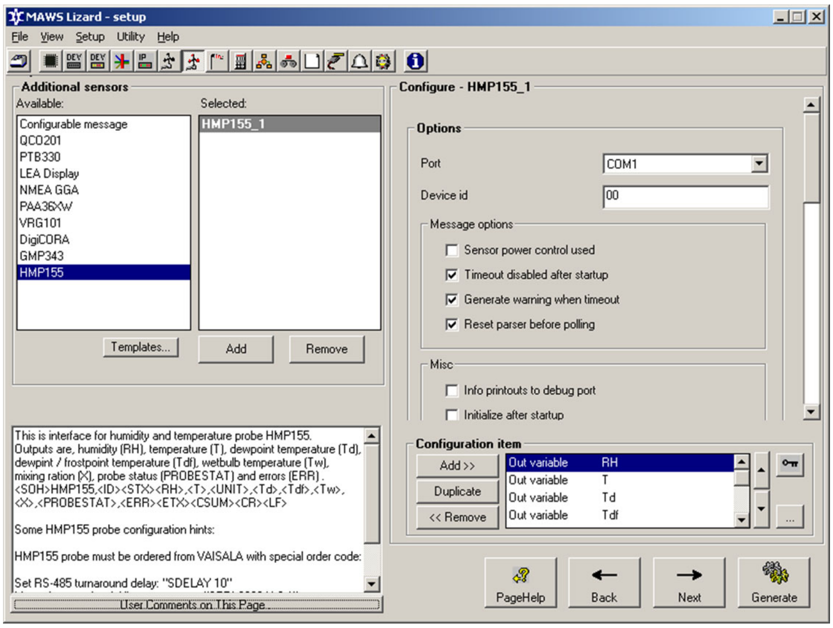

Figure 43 Configuring HTMP155 with Active Output

To configure HMP155 with passive output, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Sensors** view.
3. Select HMP155 PT100 from the **Available** list and click the **Add** button to insert the sensor into the setup. Lizard automatically suggests channels 0 and 7 for the measurements.

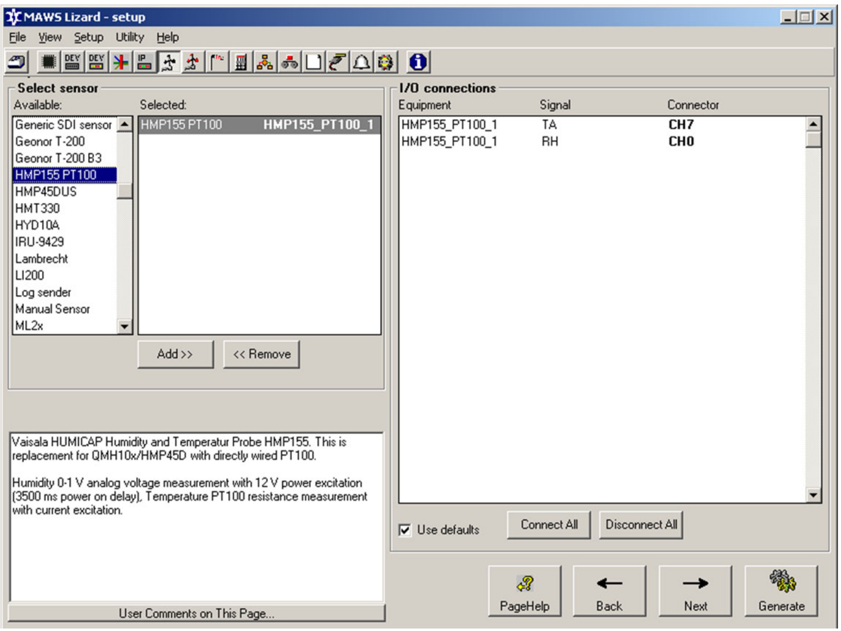

Figure 44 Configuring HMP155 with Passive Output

4. Proceed to the **Measurements** view if you want to make additional fine-tuning to the measurement parameters.

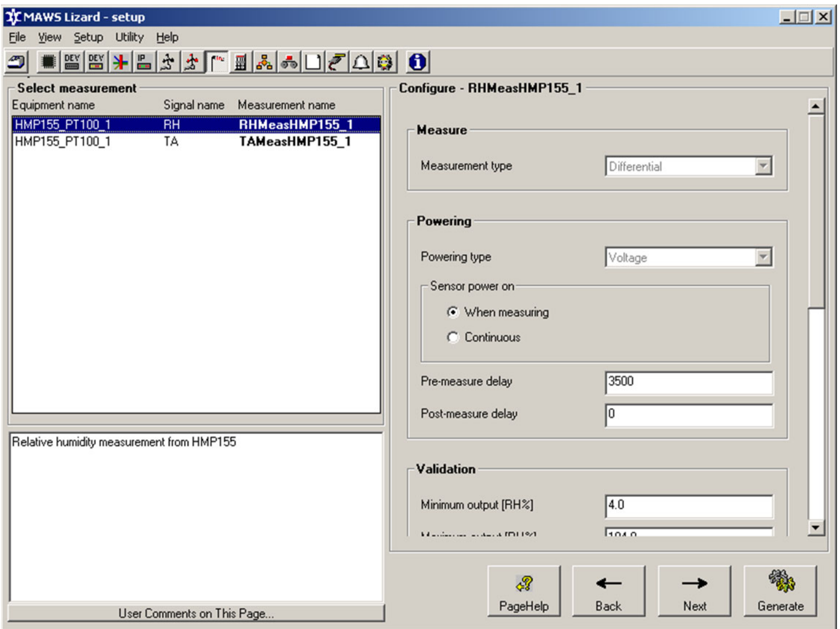

Figure 45 Configuring Relative Humidity Measurement of HMP155 with Passive Output

Digital Barometer PTB330

Vaisala BAROCAP® Digital Barometer PTB330 provides reliable pressure measurement in a wide range of applications. Digital outputs RS-232 (standard) or RS-422/485 (optional) can be selected. Alternatively, analog outputs can be chosen between current and optional voltage signals. A local graphical display is available, as well.

The barometer PTB330 is available with one, two, or three barometer modules. The barometer can be used successfully both in accurate pressure measurement applications at room temperature and in demanding automatic weather station applications.

To configure PTB330, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Additional sensors** view.
3. Select PTB330 from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. Select the port in the **Options** frame and read carefully the information window in the lower left corner.

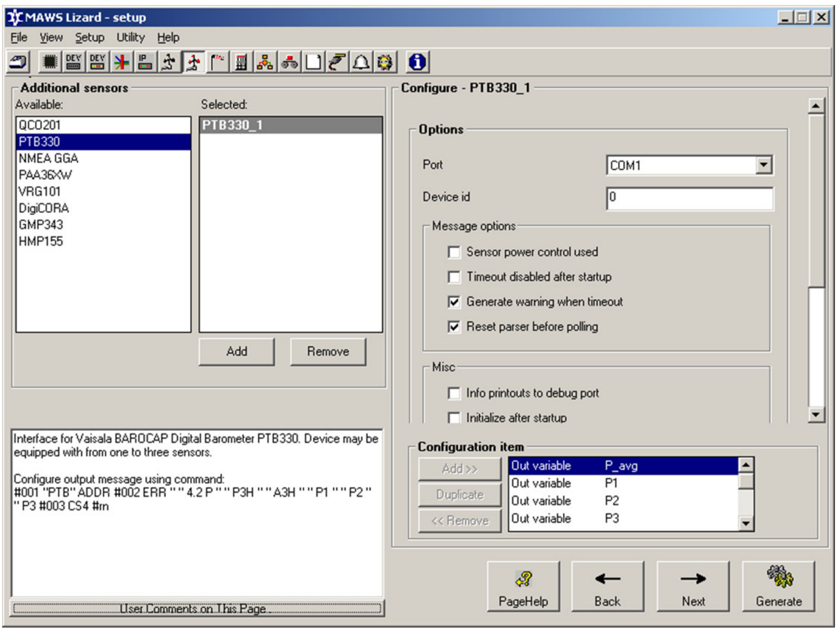

Figure 46 Configuring PTB330

Vaisala Remote Road Surface State Sensor DSC111

Vaisala Remote Road Surface State Sensor DSC111 eliminates disruption to the road surface and to traffic, which was previously associated with the installation of a road weather station. Due to remote installation, there is no need to slot-cut the surface or close the road.

The spectroscopic measuring principle enables accurate measurement of the amounts of water, ice, and snow. In addition, DSC111 provides a measurement of slipperiness and reports the state of the road surface. DSC111 also includes an integrated visibility sensor that provides a very cost-effective solution for detecting low-visibility conditions.

To configure DSC111, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Additional sensors** view.
3. Select DSC111 from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. Select the port in the **Options** frame and check the given default values for parameters.

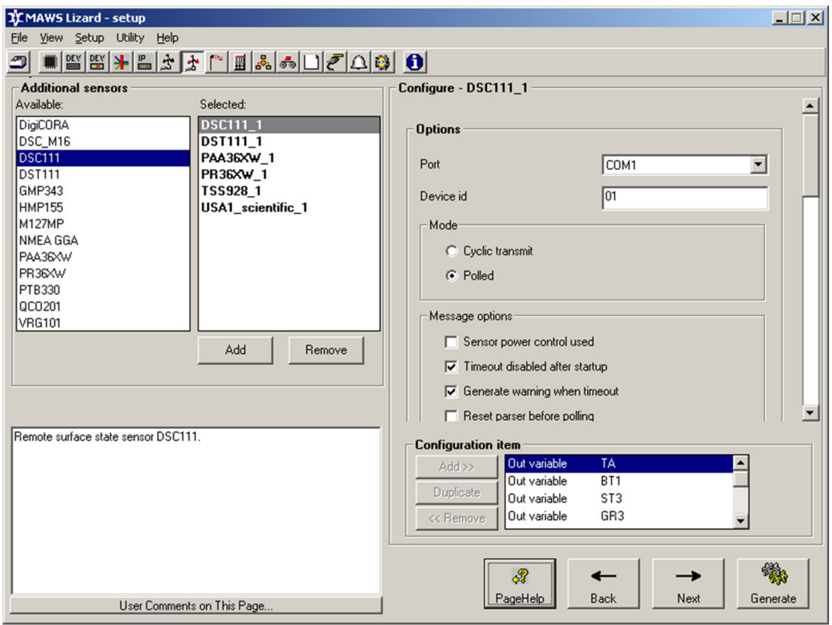

Figure 47 Configuring DSC111

Vaisala Remote Road Surface Temperature Sensor DST111

Vaisala Remote Road Surface Temperature Sensor DST111 provides a remote alternative for measuring road surface temperature. By measuring the infrared radiation emitted by the surface and applying intelligent signal processing, DST111 provides reliable remote surface temperature measurement.

To configure DST111, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one
2. Proceed to the **Additional sensors** view.
3. Select DST111 from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. Select the port in the **Options** frame and check the given default values for parameters.

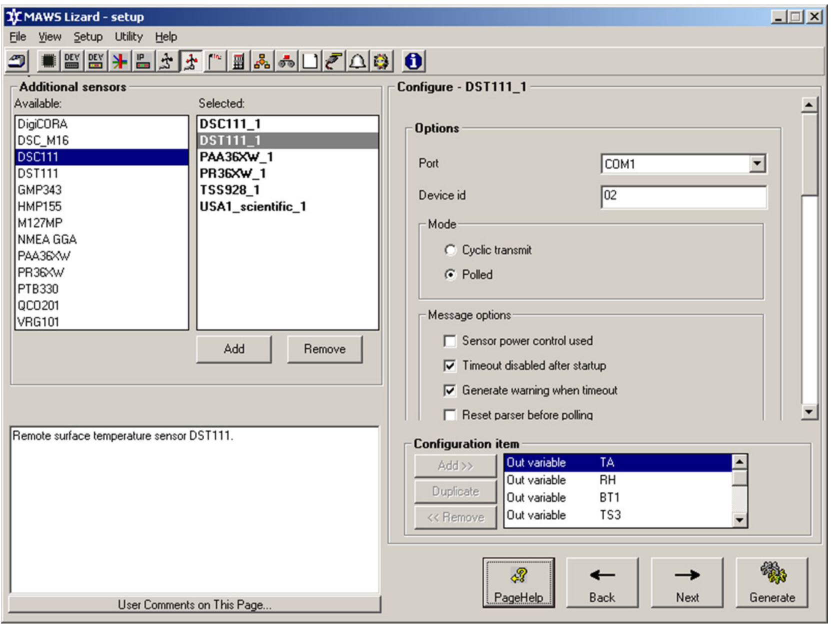

Figure 48 Configuring DST111

Submersible Pressure Sensor PAA-36XW

PAA-36WX is a submersible pressure sensor for water level measurement with RS485 digital interface, absolute pressure version. The sensor is based on the stable, piezoresistive transducer and micro-processor electronics with an integrated 16-bit A/D converter. It is applied when the atmospheric pressure is measured by a separate barometer and when the water level is calculated as the difference between the absolute value and the ambient pressure.

To configure PAA-36XW, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Additional sensors** view.
3. Select PAA-36XW from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. Select the port in the **Options** frame and check the given default values for parameters.

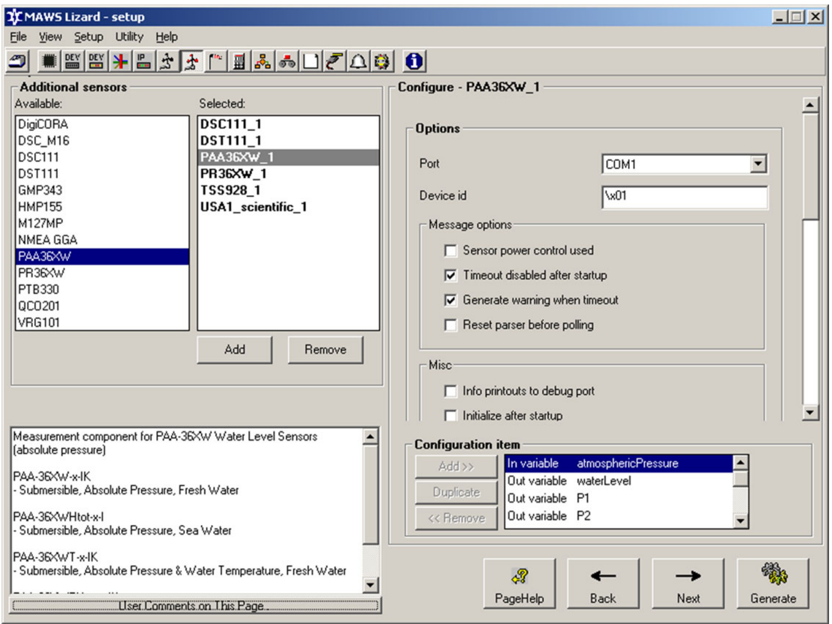

Figure 49 Configuring PAA-36XW

Submersible Pressure Sensor PR-36XW

PAA-36WX is a submersible pressure sensor for water level measurement with RS485 digital interface, relative pressure version. The sensor is based on the stable, piezoresistive transducer and micro-processor electronics with an integrated 16-bit A/D converter. It is fitted with durable cable with an integral vent tube to the atmosphere. These level transmitters can be subject to internal condensation caused by installations in cold water on warm, humid days.

To configure PR-36XW, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Additional sensors** view.
3. Select PR-36XW from the **Available** list and click the **Add** button to insert the sensor into the setup.
4. Select the port in the **Options** frame and check the given default values for parameters.

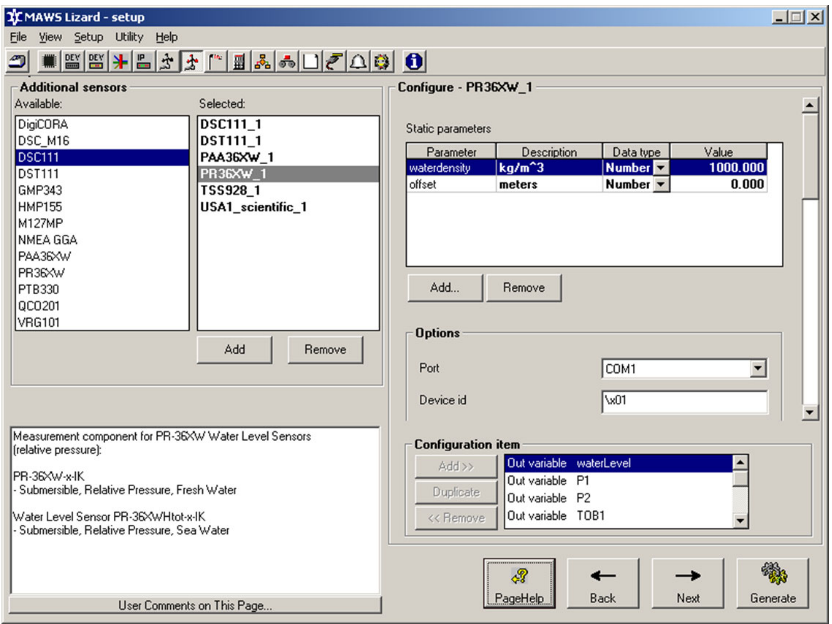

Figure 50 Configuring PR36-XW

Barometer Module BARO-1

BARO-1 is an MCU-based measurement module for determining barometric pressure. The module uses the capacitive Vaisala BAROCAP® sensor and features barometric pressure calculation, additional pressure calculations (HCP, QFE, and QNH), and internal temperature compensation, which ensure accurate, corrected readings for barometric pressure. The module includes a self-diagnostics feature. BARO-1 replaces PMT16A module in data logger QML201C.

To configure BARO-1, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Sensors** view.
3. Select BARO-1 from the **Available** list and click the **Add** button to insert the module into the setup.

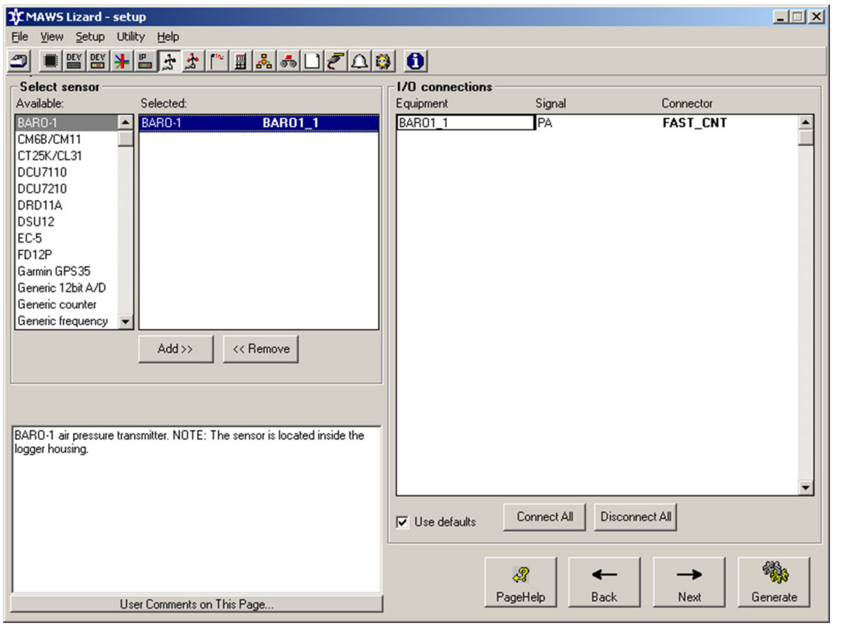

Figure 51 Configuring BARO-1 Module

4. Proceed to the **Measurements** view if you want to make additional fine-tuning to the measurement parameters.

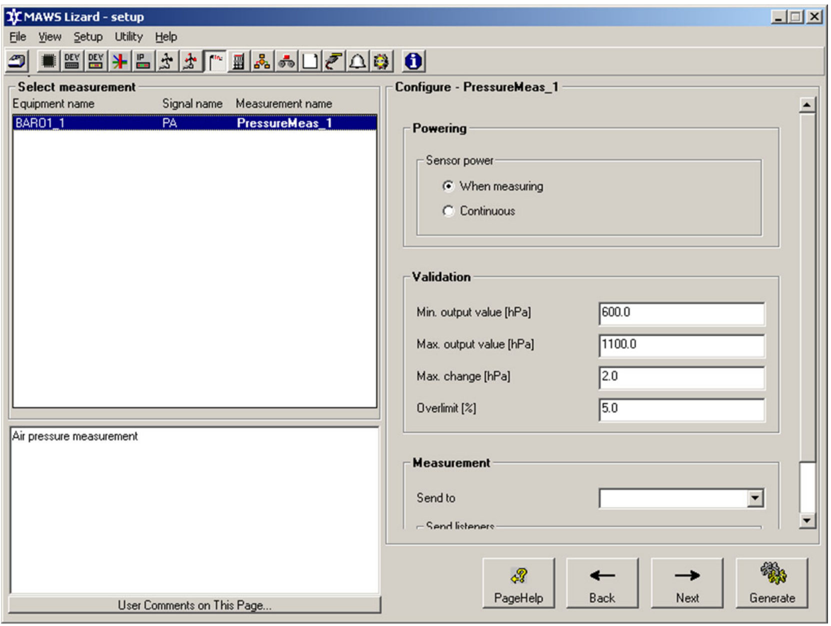

Figure 52 Configuring BARO-1 Measurement

Manual Sensors

Creating Manual Sensor in Lizard Setup Software

In the **Equipment** view, select *Manual Sensor* and click **Add**. *Manual Sensor* is automatically connected to the **Console** connector as shown in Figure 53 on page 102.

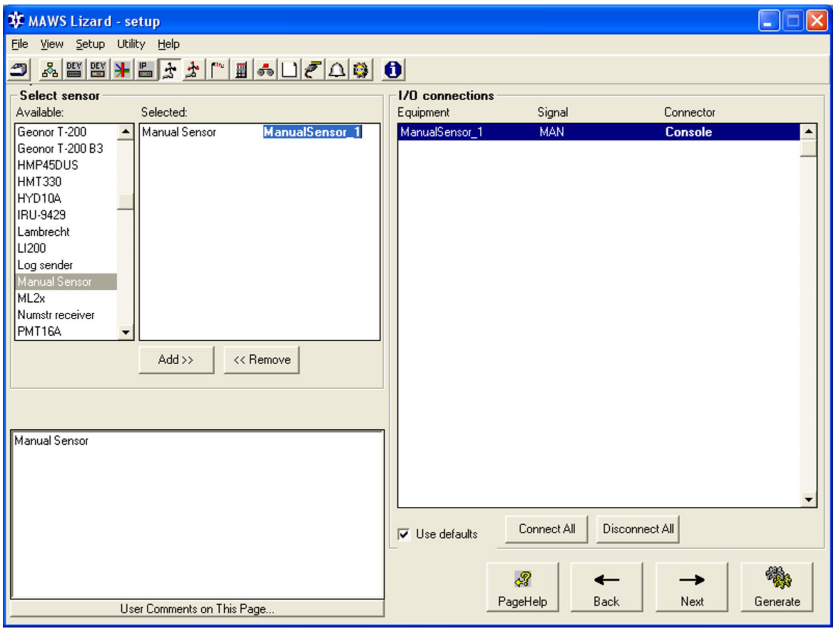

Figure 53 Creating Manual Sensor in Lizard

In the **Measurements** view, configure the manual sensor to use the correct **Datatype**, **Validity period**, and so on; see Figure 54 on page 103.

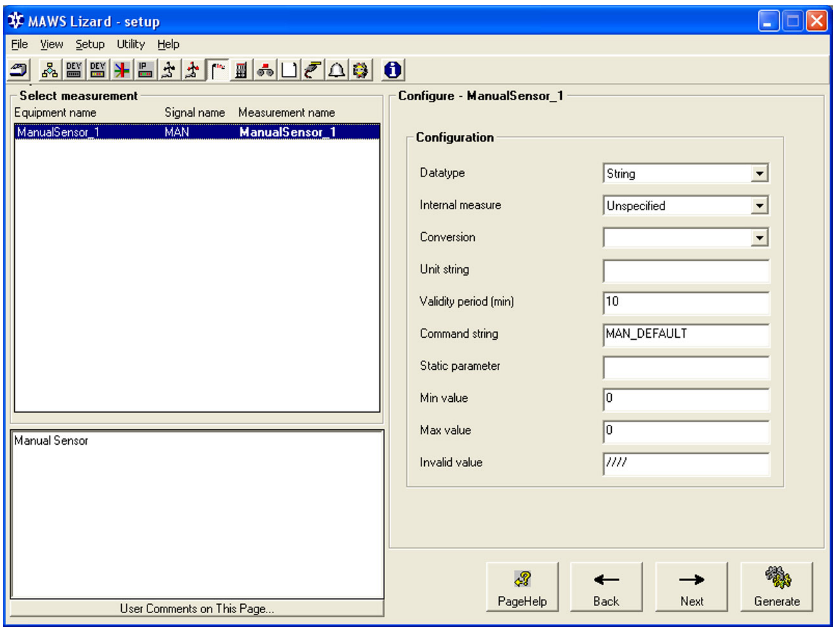

Figure 54 Configuring Manual Sensor in Lizard

Viewing Manual Sensors in AWS Client

For each manual sensor included in your setup, you can view information on the sensor's status, the type of values for which the sensor is used, and the range of valid values for the sensor. To view the manual sensor information, proceed as follows:

1. On the **Settings** menu, select **Manual Sensors**. The list of configured manual sensors is displayed.
2. Select the sensor on which you want to view information from the list.
3. Select **Edit**. The information on the sensor is displayed as shown in Figure 55 on page 104.

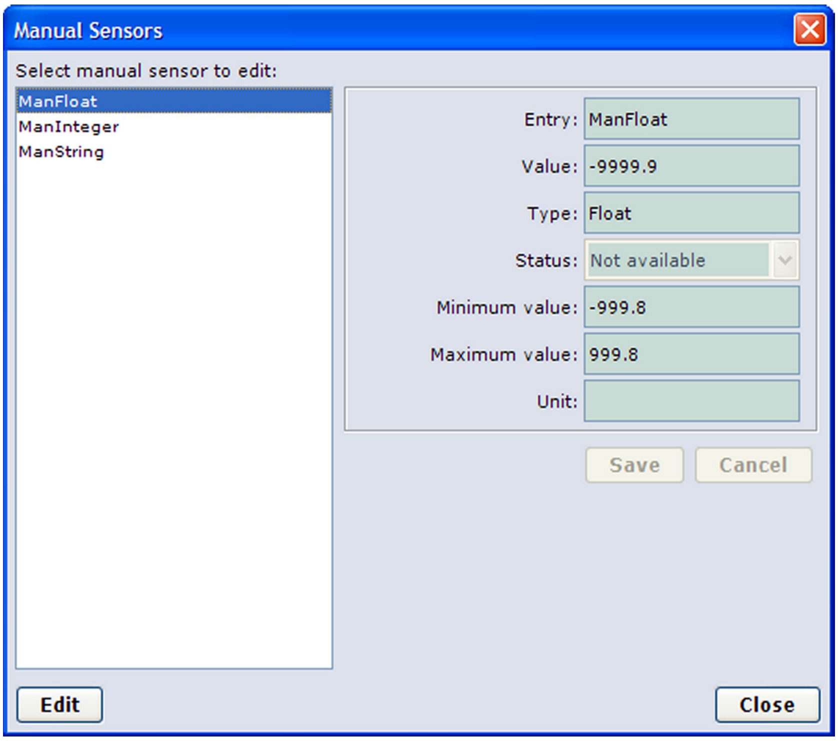

Figure 55 Manual Sensor Details

The fields in the **Manual Sensors** window are described in Table 8 on page 104.

Table 8 Fields in Manual Sensors Window

Field	Description
Entry	Name of the manual sensor as configured in the QML logger setup
Value	Field for entering the value of the manual sensor.
Type	Type of the manual sensor as configured in the QML logger setup. Possible values: **Integer**, **Float**, and **Text**
Status	Status of the manual sensor as reported by the QML logger
Minimum value	Minimum accepted value for the manual sensor reading as configured in the QML logger setup
Maximum value	Maximum accepted value for the manual sensor reading as configured in the QML logger setup
Unit	Unit for the manual sensor value

You can close the manual sensor window by selecting **Close**.

Entering Values for Manual Sensors

With AWS Client, you can enter values for each manual sensor included in your QML logger setup.

NOTE	The values you enter for the manual sensors must match the manual sensor value type: **Integer**, **Float**, or **Text**. The **Save** button is disabled in the **Manual Sensor** window if the entered value is not of the correct type.

To enter values for manual sensors, proceed as follows:

1. On the **Settings** menu, select **Manual Sensors**. The list of configured manual sensors is displayed (see Figure 55 on page 104).
2. Select the sensor for which you want to enter values from the list.
3. Select **Edit**. The information on the sensor is displayed (see Figure 56 on page 105).

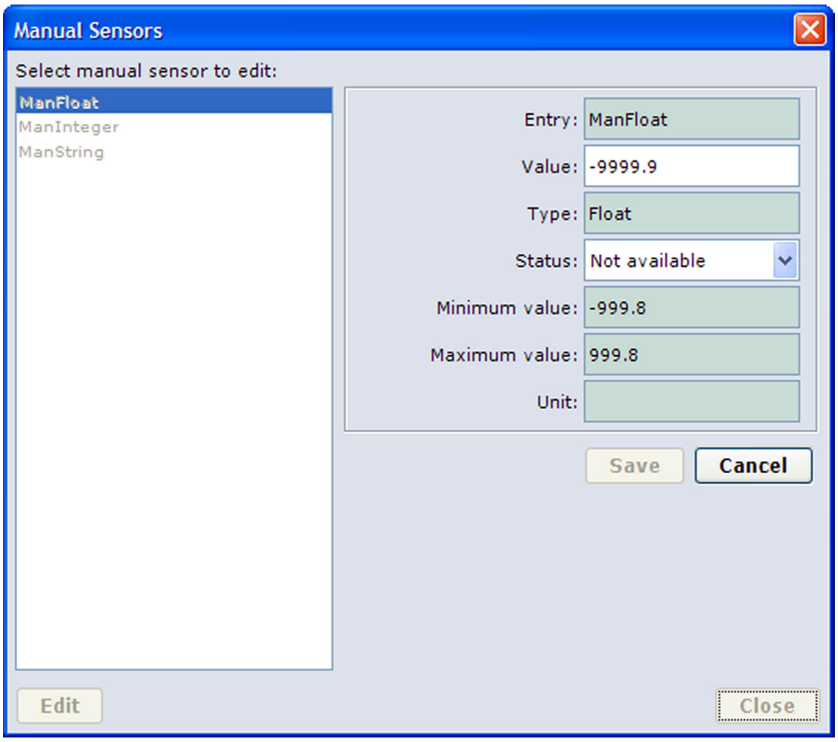

Figure 56 Entering Values for Manual Sensor

4. Enter the value for the manual sensor in the **Value** field.

5. Select the status of the measurement from the **Status** list.
6. Select **Save**. The value stored in the manual sensor is updated. You can cancel your changes by selecting **Cancel**. To close the **Manual Sensors** view, select **Close**.

Vaisala WINDCAP Ultrasonic Wind Sensor WMT700

Vaisala WINDCAP® Ultrasonic Wind Sensor WMT700 series measures wind speed and direction. The advanced, patented Vaisala WINDCAP® wind measurement technology ensures accurate results in all wind directions. The triangle design of the wind sensor provides three measurement paths. Since one of the paths is redundant, the wind speed and direction are always calculated from the two most reliable vectors. The effects of temperature, humidity, and pressure are also fully compensated.

To configure serial WMT700, proceed as follows:

1. Open an existing setup in the **Setup** view or create a new one.
2. Proceed to the **Sensors** view.
3. Select NMEA MWV from the **Available** list and click the **Add** button to insert the sensor into the setup. Select the appropriate serial port from the **Connector** list.

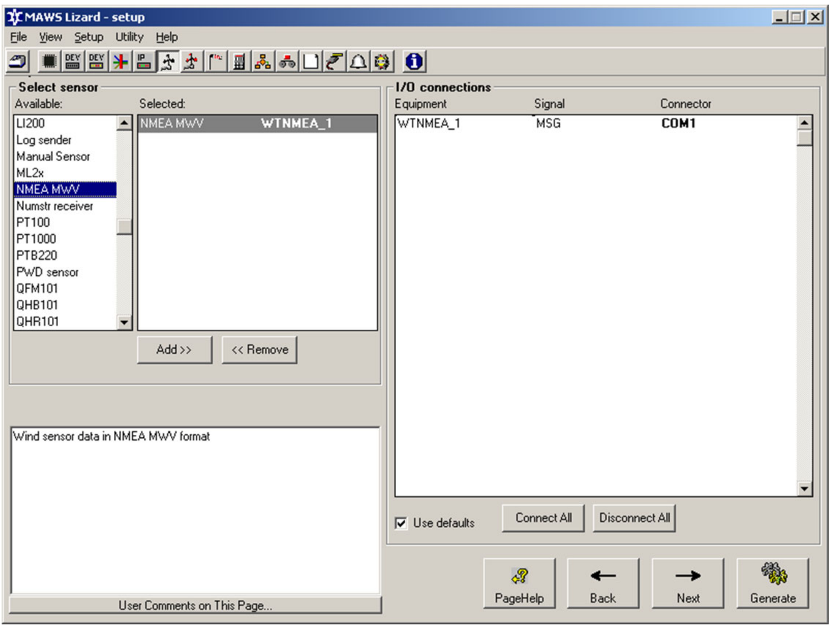

Figure 57 Configuring WMT700

4. Proceed to the **Measurements** view if you want to make additional fine-tuning to the measurement parameters.

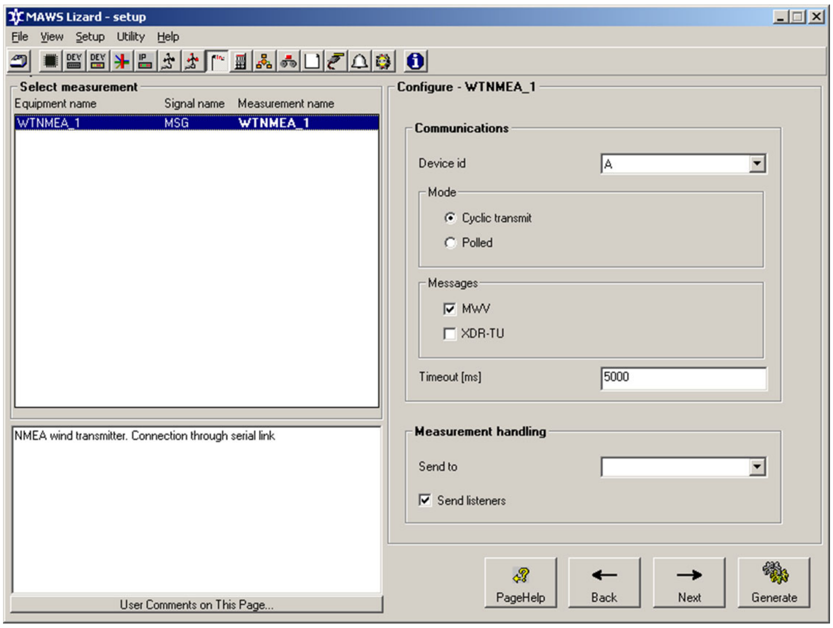

Figure 58 Configuring WMT700 Measurement

To configure WMT700 in SDI12 bus, use Generic SDI Sensor.

Configuring Serial Sensor Interfaces

This section contains information needed when configuring serial sensor interfaces, that is, connecting sensors that use serial communications to the weather station. This section contains only instructions for configuring the interface, not the sensor itself.

Interfaces of the following sensors are covered:

- Present Weather Sensor
- Visibility Sensor
- Ceilometers
- Digital Barometer
- Ultrasonic Wind Sensor
- Wind Transmitters
- Vaisala All-Weather Precipitation Gauge VRG101
- Garmin GPS 35-PC GPS Receiver

NOTE	This chapter contains only instructions for configuring the interface, not the sensor itself. In addition to this document, you will need the appropriate sensor user manual to manage the sensor configuration and hardware setup.

For most of the sensors, wiring depends on the application. You should therefore also have additional delivery-specific wiring instructions and drawings available.

Physical Interface

The QML logger has five different types of optional serial interfaces available for the sensor connections.

The serial communications options are listed in Table 9 on page 109:

Table 9 Serial Communication Options

Interface	Communication Module	Application
RS-232	Dual RS-232 module DSU232 (two RS-232)	Distance <20 m Single sensor Lowest electrical interference tolerance Full duplex communication
RS-485 (4-wire)	Isolated RS-485 module DSI485	Distance up to 1500 m Single sensor Better interference tolerance Full duplex communication
RS-485 (2-wire)	Dual RS-485 modules DSI485, DSI486 (two RS-485)	Distance up to 1500 m Several sensors Better interference tolerance Half duplex communication
Modem	Fixed line modem module DMX501	Long distances or extreme interference conditions
SDI	Dual RS-485 module DSI486	SDI sensors

NOTE The recommended connection interface is 2-wire RS-485 with a dual RS-485 module in the QML logger. This allows an interference-tolerant connection with the possibility to connect several sensors to the same serial line.

NOTE Wiring of the data and power lines depends on the application.

Data Acquisition Methods

In general, there are two alternate methods to connect a serial sensor to the QML logger:

1. Automatic transmission
   - Device transmits the measurement results automatically
   - Only one device per serial port
   - Physical connection can be RS-232 or RS-485
   - Higher data throughput
2. Polled transmission
   - QML logger polls periodically for measurement data
   - Allows multiple sensors per serial port providing that all devices have the RS-485 interface and support multi-device operation
   - Can also be used for a point-to-point connection with RS-232 or RS-485
   - Lower data throughput

Typically, sensors with high data output rate, such as wind sensors, should use automatic transmission with only a single connected device per serial line.

Sensors that produce new measurement data only a few times per minute can use polled transmission, and multiple devices can be connected to the same RS-485 line.

Configuring Serial Line Parameters

NOTE

The QML logger does not perform any automatic configuration of the connected sensors. You have to perform the configuration and store it manually by using the sensor's own service interface.

For sensors without a human-readable command interface, extra QML logger shell commands are provided. For further information, refer to detailed sensor interface descriptions or separate Technical References.

NOTE	When a separate service interface, usually RS-232, is available in the sensor, it is recommended that you use it for the initial configuration. The 2-wire RS-485 which operates in half duplex does not always provide problem-free access to service connection. Also, note that keystrokes are not echoed by the sensor.

In order to have access to the sensor interface, the serial line parameters have to be set to match those on the sensor. The easiest way to do this is to create a setup with Lizard Setup Software. Select the correct interface modules and parameters and run the setup.

Once the serial line parameters are correct, a pass-through access to the connected sensors is allowed. You can open it from the QML logger command shell by issuing the **open** command with the parameters presented in Table 10 on page 111.

Table 10 Parameters for the Open Command

Connector in Lizard	Parameter	Alias
COM0	COM0	COM0
COM1	COM1	COM1
DMX501 (MOD1/1)	DMX501_0	MOD1
DMX501 (MOD2/1)	DMX501_1	MOD2
DSI485A (MOD1/1)	DSI485_0	MOD1
DSI485A (MOD2/1)	DSI485_1	MOD2
DSI486 (MOD1/1)	DSI485_0_0	MOD1_2
DSI486 (MOD1/2)	DSI486_0_1[1]	MOD1_1
DSI486 (MOD2/1)	DSI486_1_0	MOD2_2
DSI486 (MOD2/2)	DSI486_1_1[1]	MOD2_1
DSI486SDI (MOD1/3)	DSI486SDI_0	MOD1_3
DSI486SDI (MOD2/3)	DSI486SDI_1	MOD2_3
DSU232 (MOD1/1)	DSU232_0_0[2]	MOD1_1
DSU232 (MOD1/2)	DSU232_0_1	MOD1_2
DSU232 (MOD2/1)	DSU232_1_0	MOD2_1
DSU232 (MOD2/2)	DSU232_1_1	MOD2_2
DSU232SDI (MOD1/3)	DSU232SDI_0	MOD1_3
DSU232SDI (MOD2/3)	DSU232SDI_1	MOD2_3

1. With the dual RS-485 module, the RS-232 connection is possible only to the channel B on the module, and thus the last number is 1.
2. With the RS-232 and dual RS-485 modules, the number between the underline characters stands for the module place, that is, MOD1 or MOD2, and the last number for the channel on that particular module.

NOTE	Use the **SYSINFO PORTS** command to find out which connectors to use.

The following example describes how to access the PWD sensor with device identifier 1, connected to the M60212 module in Lizard and using a DSI486 module. Note that all **open** or **close** commands are not echoed. In this case, input to PWD has no echo, either.

1. Type **open** to open the QML logger command shell.

   ```
   Service connection opened (COM0)
   ```

   ```
   / >
   ```

2. Type **open** DSI486_1_1 to open a pass-through connection to the serial line.

   ```
   / > open DSI486_1_1
   ```

   ```
   Terminal I/O re-routed to DSI486_1_1
   ```

3. Type **open 1** to open the PWD command shell.

   ```
   1 PWD OPENED FOR OPERATOR COMMANDS
   ```

   ```
   >
   ```

   You can now use the commands described in the PWD sensor user manual to perform configuration and maintenance operations.

4. To close the connection, type **close**.

```
Terminal I/O restored to ADCL shell
Service connection closed
```

NOTE	Some sensors require that you first close their service connection. Include the device identifier, for example, **close 1**. Close the sensor service connection first, and then close the QML logger connection using the **close** command alone.

Configuring Sensor Interface

Serial sensors have and use their own configuration interfaces. In addition to defining the communication parameters, there is typically very little to configure with Lizard Setup Software.

Proceed as follows:

1. Add and configure the communication interface module using the **Optional hardware** view as shown in Figure 59 on page 113.

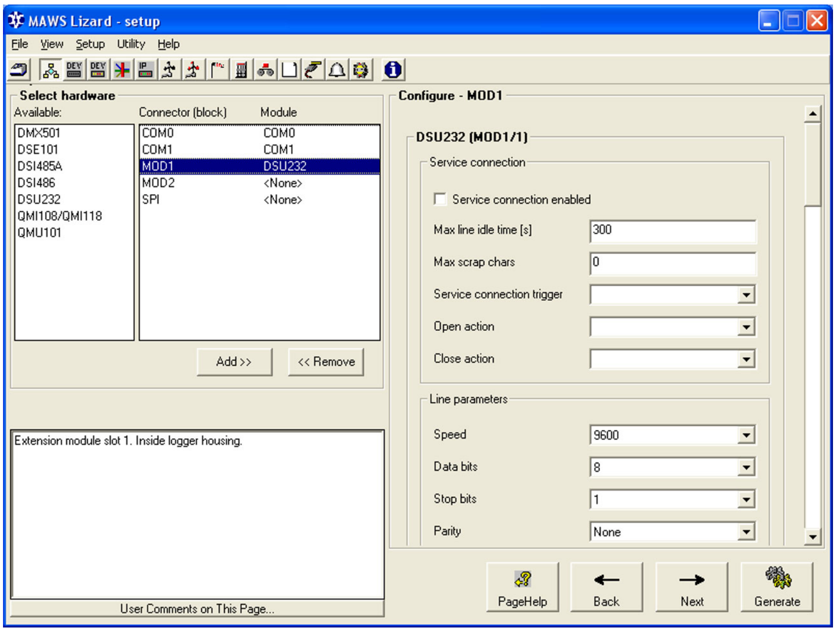

Figure 59 Optional Hardware View: Configuring Communication Module

NOTE	Use the default communication parameters of the sensor. These are not automatically generated by Lizard Setup Software, so you need to enter them manually.

If you are using 2-wire RS-485, make sure that the **Transmit control** option is selected, and that the **Start** and **End delays** are suitable for the connected sensor. Usually, you can leave the default values, but adjustments may be necessary especially if you are connecting multiple sensors to the same serial line. See Figure 60 on page 114 for the options.

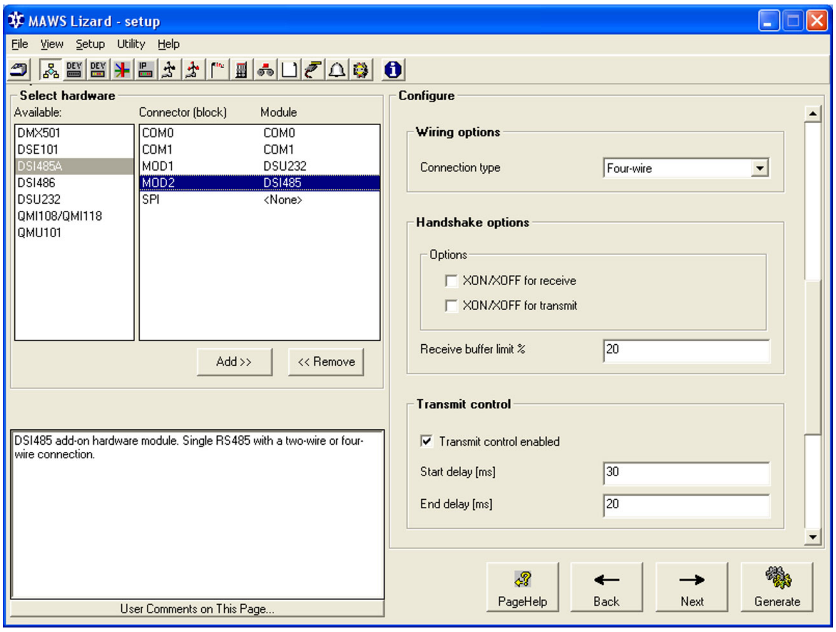

Figure 60 Optional Hardware View: Defining Transmit Control Parameters

You do not need to select the **Service connection enabled** option in order to have service access to the sensor. Selecting this option, in fact, enables service access to the QML logger through the port.

2. Proceed to the **Equipment** or **Additional sensors** view to add and connect the sensor as shown in Figure 61 on page 115.

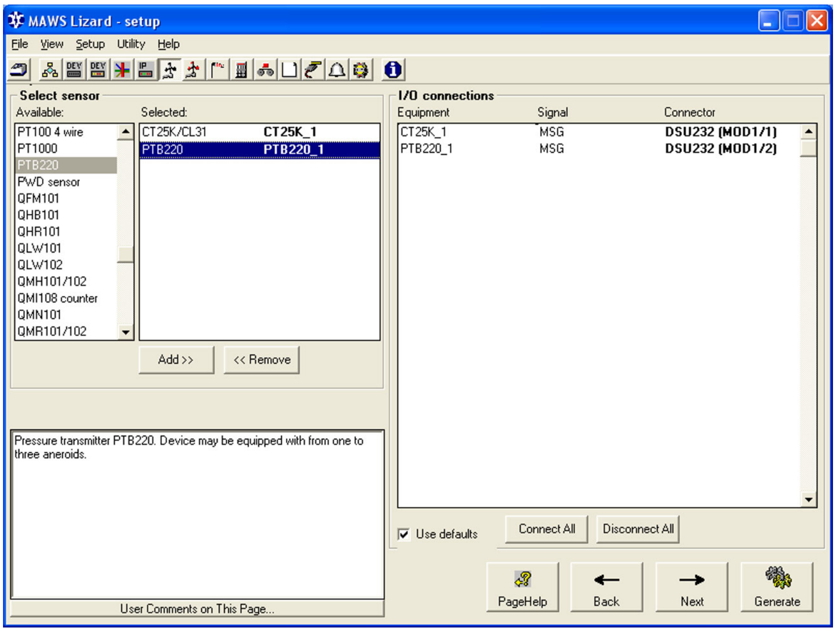

Figure 61 Equipment View: Adding and Connecting Serial Sensor

NOTE	Serial sensors do not have default connectors, so you need to connect the sensor manually to the selected port.

The first number of the connector identifier MODX/X (for example, MOD1/1) indicates the piggyback slot: 1 is the one closer to the battery holder and 2 is the bottom one. The second number is present only if you are using a dual port board such as DSU232 or DSI486. It indicates the channel number, 1 or 2.

3. If you are connecting multiple sensors to the same serial line, the following window opens. Click **Yes**.

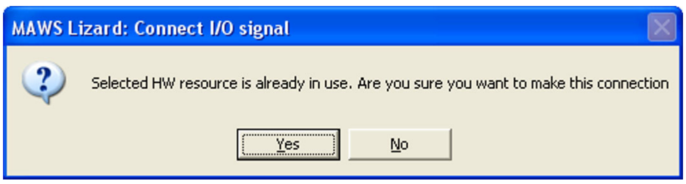

Figure 62 Connect I/O Signal Window

4. Proceed to the **Measurements** view to configure the sensor interface as shown in Figure 63 on page 116.

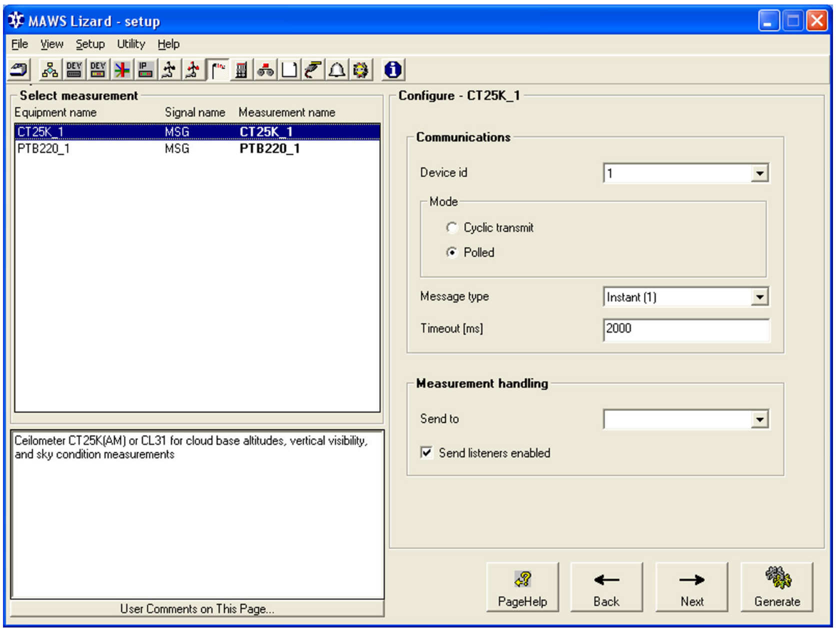

Figure 63 Measurements View: Configuring Serial Sensor Interface

Typical items to configure include the following:

Item	Description/Use
Device ID	Device identifier (must be entered) - Same as the set - Unique for each sensor if multiple devices are connected to the same serial line Used by - Polling and data reception to identify the device - Open commands to identify which device should open its service interface
Mode	Configuration for automatic or polled operations as described in section Data Acquisition Methods on page 110.
Message Type	Many devices support data messages with different contents. This item selects the expected message type.

Item	Description/Use
Timeout	Timeout for data reception. If no new data is received within this period of time, sensor data in reports and logs is marked as invalid until new data is received. Timeout starts from: - Poll message transmit in polled mode Message reception in automatic transmit mode

NOTE	The settings described apply only to the QML logger. You must also configure the sensor to have the matching settings.

Serial Sensor Interfaces

This chapter describes the following options in detail for each sensor:

- Default communication parameters
- Typical sensor interface configuration alternatives
- Data availability with different message types. This refers to the availability of sensor output variables used in the setup for calculation, logging, and reporting
- Sensor-specific issues, such as QML logger shell commands

Present Weather and Visibility Sensors

The present weather and visibility sensors are covered in the same chapter as they produce similar data and share message structures.

The default communication parameters are as follows:

Parameter	Value
Speed	*9600 bps* (PWD sensors)/*300 bps* (FD sensor)
Parity	*even*
Data	*7*
Stop	*1*
Handshake	*none*

The sensor commands for interface configuration are as follows:

Operation	Command(s)	Description
Set polled	**AMES 2 0** or **AMES 7 0**	Disables automatic transmission
Set automatic tx	**AMES 2 30** or **AMES 7 30**	Enables automatic transmission of message 2 or message 7 (PWD21/ PWD22/FD12P only) with 30-second interval
Set device id	**CONF**	Activates configuration dialog

Message availability by sensor type is as follows:

Sensor	Messages
PWD11	2
PWD21	2,7
PWD22	2,7
FD12P	2,7

The available data with FD Present Weather Sensor is as follows:

Lizard variable	Message	Description
pwd_id	2,7	Header text
dev_id	2,7	Device ID
alarms	2,7	Device alarm code
vis_1	2,7	Visibility 1 min average
vis_10	2,7	Visibility 10 min average
pw_nws	2,7	Present weather NWS code
pw_inst	2,7	Instant present weather code
pw_15	2,7	15 min present weather code
pw_1h	2,7	One hour present weather code
water_1h	2,7	Precipitation intensity mm/hour
water_cum	2,7	Cumulative water sum
snow_cum	2,7	Cumulative snow sum
t_xarm	7	Crossarm temperature
lumin	7	Background luminance
metar_i	7	Instant METAR weather code
metar_r	7	Recent METAR weather code
pw_insttxt	2,7	WMO text for instant present weather code
pw_15txt	2,7	WMO text for 15 min present weather code
pw_1htxt	2,7	WMO text for one hour present weather code

Lizard variable	Message	Description
water_inc	2,7	Water sum increment since previous transmission
snow_inc	2,7	Snow sum increment since previous transmission
orig_PWD	2,7	Original FD12P message

The available data with PWD Present Weather Detector is as follows:

Lizard variable	Message	Description
pwd_id	2,7	Header text
dev_id	2,7	Device ID
alarms	2,7	Device alarm code
vis_1	2,7	Visibility 1 min average
vis_10	2,7	Visibility 10 min average
pw_nws	2,7	Present weather NWS code
pw_inst	2,7	Instant present weather code
pw_15	2,7	15 min present weather code
pw_1h	2,7	One hour present weather code
water_1h	2,7	Precipitation intensity mm/hour
water_cum	2,7	Cumulative water sum
snow_cum	2,7	Cumulative snow sum
t_xarm	7	Crossarm temperature
lumin	n/a	Background luminance
metar_i	7	Instant METAR weather code
metar_r	7	Recent METAR weather code
pw_insttxt	2,7	WMO text for instant present weather code
pw_15txt	2,7	WMO text for 15 min present weather code
pw_1htxt	2,7	WMO text for one hour present weather code
water_inc	2,7	Water sum increment since previous transmission
snow_inc	2,7	Snow sum increment since previous transmission
orig_PWD	2,7	Original PWD message

NOTE PWD11 can produce only message 2.

Ceilometers

Ceilometers produce almost equal data using the same data messages.

The default communication parameters for service and data port are as follows:

Parameter	Value
Speed	*2400 bps*
Parity	*even*
Data	*7*
Stop	*1*
Handshake	*none*

The sensor commands for interface configuration are as follows:

Operation	Command(s)	Description
Set polled	**SET MESSAGE TYPE** <*type*> SET MESSAGE MODE POLLING	Sets message type, where <*type*> is MSG1, MSG6 or MSG 61. Disables automatic transmission.
Set auto tx	**SET MESSAGE TYPE** <*type*> SET MESSAGE MODE AUTOSEND	Sets message type, where <*type*> is MSG1, MSG6 or MSG 61. Enables automatic transmission.
Set ID	**SET UNIT_ID** <*ID*>	Sets unit ID, where <*ID*> is a single character 1...Z.
Setinterface type	**SET PORT DATA INTERFACE** <*type*>	Sets interface type for the data connection. <type> is RS-232 or RS-485_2W.

Message availability by sensor type is as follows:

Sensor	Messages
CT25	1,6
CT25KAM	1,61
CL31	CT25 message 1,6

NOTE	In order to have messages 6 or 61 available, you need to purchase the sky condition option separately.

The available data is as follows:

Lizard Variable	Message	Description
Header	1,6,61	Header text (first line data)
GenStat	1,6,61	Status information field
meas_1	1,6,61	Raw measure 1
meas_2	1,6,61	Raw measure 2
meas_3	1,6,61	Raw measure 3
HWStat	1,6,61	Hardware status bits (hex format)
oktas_1	6,61	Octas in layer 1
height_1	6,61	Height of layer 1
oktas_2	6,61	Octas in layer 2
height_2	6,61	Height of layer 2
oktas_3	6,61	Octas in layer 3
height_3	6,61	Height of layer 3
oktas_4	6,61	Octas in layer 4
height_4	6,61	Height of layer 4
oktas_5	61	Octas in layer 5
height_5	61	Height of layer 5
d_stat	1,6,61	Detection status
base_1	1,6,61	Height of cloud base 1 [1]
base_2	1,6,61	Height of cloud base 2 [1]
base_3	1,6,61	Height of cloud base 3 1
v_vis	1,6,61	Vertical visibility [1]
h_sig	1,6,61	Highest signal [1]
i_stat	1,6,61	Hardware status bits (integer format)
orig_CT	1,6,61	Original CT25 message

1. These are meas_1 to meas_3 values processed according to the detection status. Values are exclusive. For example, if cloud bases are shown, vertical visibility and highest signal are missing and vice versa.

Digital Barometer

The default communication parameters for the PTB330 sensor are as follows:

Parameter	Value
Speed	*9600 bps*
Parity	*even*
Data bits	*7*
Stop bits	*1*
Handshake	*XON/XOFF*

The sensor commands for the interface configuration are as follows:

Operation	Command(s)	Description
Set polled	**SMODE POLLED**	Sets sensor to polled operation
Set auto tx	**INTV** *<time> <unit>* **SMODE RUN**	Select transmit interval to *<time> <unit>*, for example, 10 s. Sets sensor to automatic transmit mode.
Set format	**FORM**	See the detailed explanation below.
Set ID	**ADDR** *<id>*	Sets device id for polled operation. *<id>* = 0 ... 99
Activate param	**RESET**	New parameters are taken into use after reset.

To get a suitably formatted data output from the digital barometer, the following message format string must be given using the **FORM** command. The string is almost of maximum length allowed, so do not insert any extra spaces. Proceed as follows:

1. Type FORM <CR>. This command outputs the current format string and prompts `?` for a new one.
2. Copy the string below to the terminal and press <CR>.

```
#001 "PTB" ADDR #002 ERR " " 4.2 P " " P3H " " A3H "
" P1 " " P2 " " P3 #003 CS4 #rn
```

NOTE	After configuring the sensor, enter the **RESET** command in order for the changes to take effect.

The available data is as follows:

Lizard Variable	Description
ptb_id	Device identifier with address, for example, 'PTB2'
err	Error flags
P_avg	Pressure average
trend	Three hour pressure trend
tend	Pressure tendency
P1	Pressure measure #1
P2	Pressure measure #2 [1]
P3	Pressure measure #3 [1]
sum	Message checksum (hex)
orig_str	Original message string

1. All possible data is available with the output format explained previously. Availability of the variables P2 and P3 depends on the number of installed aneroids.

Ultrasonic Wind Sensor

The default communication parameters for the ultrasonic wind sensor are as follows:

Parameter	Value
Speed	*9600 bps*
Parity	*none*
Data bits	*8*
Stop bits	*1*
Handshake	*none*

The sensor has a menu-driven configuration interface, which is shown right after the service connection has been opened with the **open** command.

Menu selections for interface configuration are as follows:

Operation	Menu Selection	Description
Set format	Operation Mode -> Handar RS-232	Select correct output mode.
Set units	Wind speed units -> Meters/second	By default, QML logger assumes wind calculation speed input to be in meters/ second.
Set polled	Output Interval -> 0	Disables automatic transmission.
Store param	Save Configuration	Store modified parameters permanently.

NOTE	The QML logger interface for the ultrasonic wind sensor WS425 currently supports only the **Handar RS-232** mode.

The available data is as follows:

Lizard Variable	Description
type	Message type, always 'W'
avg_time	Averaging time [s]
dev_stat	Device self test status
WD	Wind direction in [deg]
WS	Wind speed [m/s]
unit	Speed unit, always 'T' with QML logger
sum	Eight-bit sum for message, presented in hex ASCII

Wind Transmitters

The WT500 series wind transmitter can be used to interface with different types of wind sensors, and optionally with the humidity probe HMP45 for temperature and humidity measurement.

The default communication parameters are as follows:

Parameter	Value
Speed	*1200 bps*
Parity	*none*
Data bits	*8*
Stop bits	*1*
Handshake	*none*
RS-485 mode	*4-wire*

The sensor commands for interface configuration are as follows:

Operation	Command(s)	Description
Set wind data format	**SETMES 1 TYPE**	Sets wind data format where <*value*> is MWV for autosend operation and MWVQUERY for polled
Set TU data format	**SETMES 2 TYPE XDR_TU**	Sets temperature and humidity data format [1]
Set output port	**SETMES 1 COM 0 SETMES 2 COM 0**	Outputs both messages to the onboard port 0. Another alternative is 1 if there is an additional communications module installed in the WT5XX.
Set autosend intervals	**SETMES 1 INTERVAL 1.0 SETMES 2 INTERVAL 60.0**	Sets wind data interval to 1s. Set temperature and humidity data interval to 60s [1]
Set ID	**SETDEV id** <*value*>	Sets the device identifier, where <*value*> is an uppercase letter A ... J
Set interface mode	**SETCOM 0 WIRES** <*value*>	Select the interface mode for the onboard communications port. Set value 4 to use RS-232 or 4-wire RS-485, and value 2 to use 2-wire RS-485.
Activate parameters	**RESET**	New parameters are taken into use after reset

1. Required only if temperature and humidity measurements are used. If so, you must also enable XDR_TU reception in the Lizard Setup Software Measurements view.

NOTE	MWVQUERY format can only be used alone. If you are using XDR_TU, you must use the MWV format for wind data transfer, and operate WT5XX in an automatic transmission mode.

NOTE	After configuring the device, enter the **RESET** command in order for the changes to take effect.

The available data by message type is as follows:

Lizard Variable	Message	Description
mwv_hdr	MWV(QUERY)	MWV message header
dir	MWV(QUERY)	Wind direction [deg]
ref	MWV(QUERY)	Reference. R = relative, T = true
spd	MWV(QUERY)	Wind speed
uni	MWV(QUERY)	Wind speed unit. K = km/h, M = m/s, N = kt
mwv_sta	MWV(QUERY)	MWV status field and checksum
orig_WT50_mwv	MWV(QUERY)	Original receive string
ta_typ	XDR_TU	Transducer type: C for Ta
ta	XDR_TU	Air temperature
ta_uni	XDR_TU	Ta. unit: C for ºC
ta_id	XDR_TU	Transducer ID for Ta
rh_typ	XDR_TU	Transducer type: H for RH
rh	XDR_TU	Relative humidity
rh_uni	XDR_TU	RH unit: P for %
xdr_sta	XDR_TU	Misc. XDR status
orig_WT50_xdr	XDR_TU	Original receive string

Vaisala All-Weather Precipitation Gauge VRG101

The default communication parameters for the precipitation gauge are as follows:

Parameter	Value
Speed	*9600 bps*
Parity	*none*
Data bits	*8*
Stop bits	*1*
Handshake	*none*

The sensor commands for interface configuration are as follows:

Operation	Command(s)	Description
Open service connection for command mode	**OPEN** [*VRG*] [*device ID*]	If no device ID is set and the precipitation gauge is the only device connected to the QML logger on the line, type **OPEN**. If the devices ID is set as *1*, type **OPEN** *1* or **OPEN** *VRG 1*. If you do not know the device ID, **OPEN** *VRG ** will open the service connection to the precipitation gauge on the line.
Close service connection and exit command mode	**CLOSE**	Automatic message sending and polling are functional in the CLOSE mode.
Set polled	**AMES -1**	Disables automatic message transmission
Set automatic message transmission	**AMES** <*message number*> <*message interval in minutes*>	Sets the message type to be transmitted by the precipitation gauge. For message numbers, 1 = Data message; 2 = Status message.
Set ID	**ID** <*device ID*>	Sets device ID, where <*device ID*> is a two-character identifier 1 ... 99.
Set communication speed	**BAUD** <*bit rate*>	Sets the communication speed; the available speeds are 300, 1200, 2400, 4800, and 9600 bps. The default is 9600 bps. Note that the character frame is fixed: 8 data bits, no parity, 1 stop bit, and cannot be changed.
Change to tipping bucket mode	**CONF** *1 1*	In the tipping bucket mode, the precipitation gauge uses the RS-485 line to give a current pulse, by default, for every 0.1 mm of precipitation.

The available data by message type is as follows:

Lizard Variable	Message	Description
mass	1	Mass measurement from the precipitation gauge in grams
operVoltage	1, 2	Supply voltage to the precipitation gauge
prAccumulation	1	Precipitation accumulation in mm
prIntensity	1	Precipitation intensity in mm/h
statusData	1	Data status: 0 = Data OK, 1 = Failure, 2 = Overflow, 3 = Draining
statusHeater	1	Heater status: 0 = Heater off, 1 = Heater on, 2 = Heater failure
statusHW	1	Precipitation gauge hardware status: 0 = Hardware OK, 1 = Failure, 2 = Warning
temperature	1	Temperature measurement from Pt100 sensor; -99.99 = No sensor
temperatureCPU	1, 2	Temperature of the electronics enclosure

GARMIN GPS35-PC GPS Receiver

The default communication parameters according to NMEA 0183 are as follows:

Parameter	Value
Speed	*4800 bps*
Parity	*none*
Data bits	*8*
Stop bits	*1*
Handshake	*none*

No initial configuration is needed. The device automatically starts outputting GPRMC data at one-second intervals.

The available data is as follows:

Lizard Variable	Description
gp_hdr	GP message header
utc_t	GPS UTC time HHMMSS
gps_status	GPS status A = Valid position V = NAV receiver warning
lat	Latitude, *ddmm.mmmm*
lat_h	Latitude hemisphere, N or S
long	Longitude, *dddmm.mmmm*
long_h	Longitude hemisphere, E or W
utc_d	GPS UTC date DDMMYY
orig_ggps35_gp	Original receive string

CHAPTER 4
CONFIGURING TELEMETRY OPTIONS

This chapter provides instructions on how to configure different non-TCP/IP telemetry options, that is, modems and satellite transmitters.

NOTE	This section only describes the configuration of non-TCP/IP-based telemetry. For configuration instructions for TCP/IP-based communication devices available in Lizard and QML logger software versions 6.00 and later, refer to Chapter 5, Configuring TCP/IP-Based Telemetry, on page 201.

Introduction to Modem Control

Modem control provides means for transferring data reports from a weather station using the following media:

- Public telephone network (PSTN)
- Cellular network. The functionality has been tested with GSM.
- Internet using GPRS as media between a weather station and an Internet service provider (ISP).
- Other devices using standard AT commands. However, the usability for such purposes has to be checked and tested case by case.

Modem control also provides the possibility to remotely control and configure the operation of a weather station.

The functionality of the following devices has been tested with modem control:

- DXM Modem for PSTN
- Siemens MC35 for GSM/GPRS

Minimum System Requirements

The following minimum software versions are required: QML logger version 4.05 and Lizard Setup Software version 4.05.

When updating or adding new functionality to the setups made with earlier versions, you need to remove the existing modem(s) from the setup and reconfigure the setups using the new equipment definitions.

Hardware Configuration

To keep the fixed RS-232 line COM0 free for local service use, it is recommended that the QML logger is equipped with an additional RS-232 module (DSU232) for the modem connection. This module can be added in the **Optional hardware** view of Lizard Setup Software.

Keep the following RS-232 line default parameters:

- Speed: *9600 bps* (see note below)
- Data bits: *8*
- Stop bits: *1*
- Parity: *None*

NOTE	If you are using the old Siemens M20T modem, the communication speed must be set to *19200* bps. Siemens GSM/GPRS modems have an autobaud feature, and you can use the *9600 bps* speed.

NOTE	If you are planning to transfer the log files with the AWS Client software or with another ZModem-capable software over the modem connection, it is highly recommended that you increase the transmit buffer size in the QML logger. Enter **12000** to the **Transmit [bytes]** field under **Buffers** in the **Optional hardware** view.

I/O Connection

Select the **Devices** view in Lizard Setup Software.

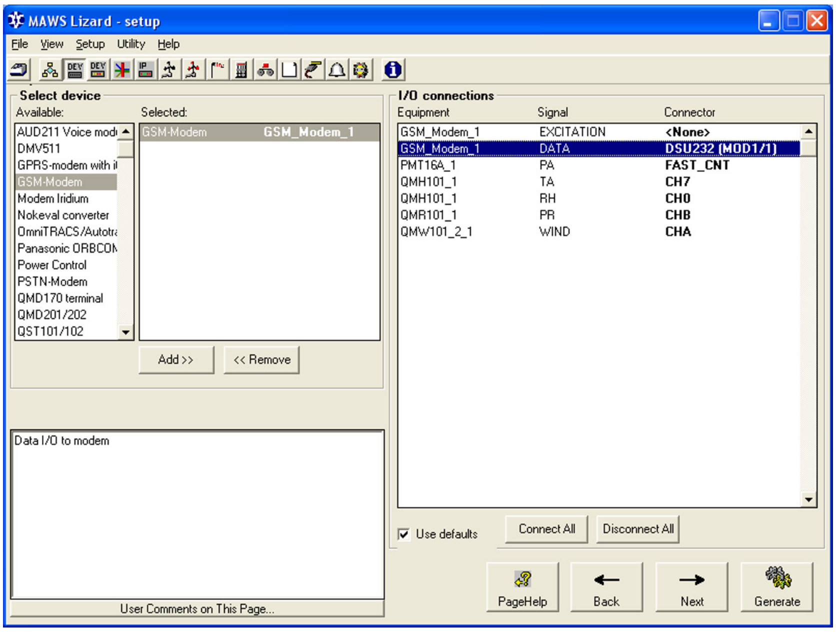

Figure 64 Modem I/O Connection

To obtain the correct default values you must first select the correct modem type from the **Available** list. Table 11 on page 133 lists the devices and the corresponding equipment selections.

Table 11 Devices with Corresponding Equipment Selections

Device	Equipment Selection in Lizard
DXM421	PSTN modem
MC35	GSM-Modem

Adding a modem to your configuration creates two I/O connections: DATA and EXCITATION.

DATA is the RS-232 connection for the modem, and EXCITATION is the optional power control output. For DATA, in the **I/O Connections** frame select DSU232 (MOD1/1) as the default, which means that the DSU232 module is installed at the module place 1 of the QML logger.

NOTE	Excitation control must be used with the **Answer incoming calls** option, as in optional power control the modem is powered only when a connection is established by the QML logger.

The QML logger modem control allows optional timed powering to gain lower total power consumption. This means that the modem is powered only when necessary, for example, when it is transmitting data or it is needed for service operation.

In optional power control, the power cable is connected to the terminal strip via the relay that is controlled using a free excitation pin from the logger channels 0, 1, 2, or 3. These pins supply only 12 V/25 mA, which is not sufficient for powering the modem without the relay.

In excitation control, the additional relay control for the power supply is set in the **Timers** view.

Modem Control Parameters

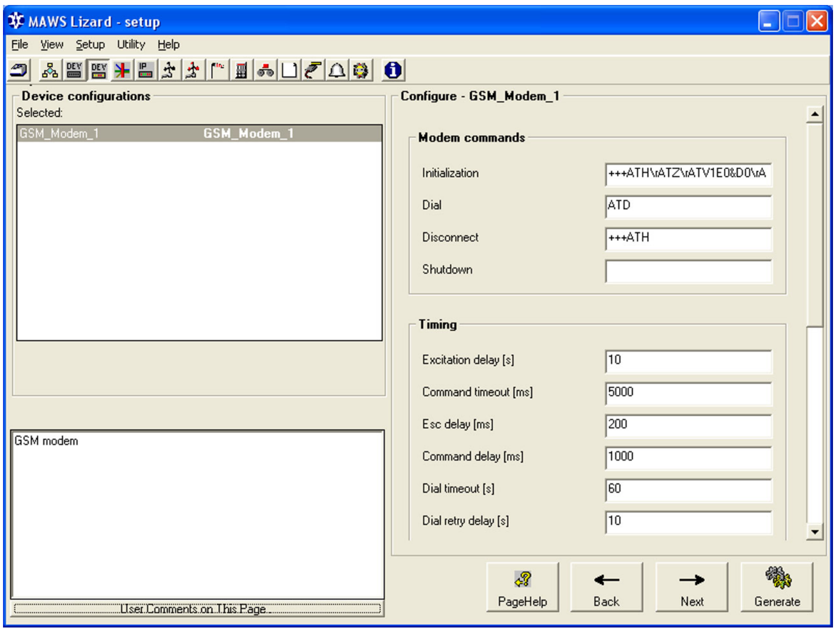

Figure 65 Device Configurations View

Regardless of the external interface used (PSTN, GSM network or other), similar interface and parameters are used for configuring the operation of the modem.

The modem control parameters are accessible in the **Device configurations** view. Reports are linked to the modem in the **Reports** view. You should link a report under **Available** reports to a communications port (**Port**) with a modem (**Device**) attached to it. Depending on the configuration, a call, SMS transmission, or Internet operation may be initiated as soon as a new report is generated. Alternatively, an inbound call and/or a poll may be required for triggering the transmission.

NOTE	The following sections describe all the parameters visible on the **Advanced** user level. Although the Advanced user level includes some additional parameters that are not accessible on the Normal user level, modifications to these parameters should be avoided. Therefore, it is recommended that you use Lizard Setup Software on the **Basic** user level.

Modem Commands

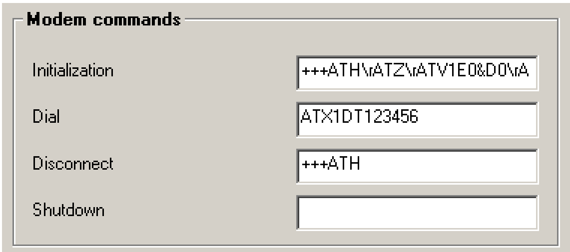

Figure 66 Modem Command Configuration Options

All modem types use the standard AT command interface with device-specific extensions. For example, you can use "," (comma) for delay after 0 for the external line. The available modem command parameters are described in the following sections. For more detailed information on the commands, refer to the applicable modem user manual.

Initialization

The **initialization** string is sent to the modem once at startup, and then at user-selectable intervals. The initialization string may contain any acceptable modem commands, but for correct operation with the QML logger, the ones listed in Table 12 on page 136 should always be present. Note that the ones marked with footnote 1 are only for GSM connection.

Table 12 Initialization Strings

String	Description
ATV1	Verbose responses, that is, the modem responds with whole words instead of numbers.
ATE0	Echo off
ATS0=0	Disables modem from automatically answering incoming calls (to enable answering use option Answer incoming calls).
AT+CMGF=1	SMS messages in the text mode [1]
AT+CNMI=2,1	Inbound SMS message buffering mode [1]

1. GSM only

With the PSTN modems, it is important that some kind of error correction is used for the link between modems. For this purpose, the initialization string for the PSTN modem contains the following command:

AT\\N2 = Forces LAPM or MNP connection. Failure to obtain a reliable connection causes the modem to disconnect.

NOTE

When a command contains the backslash "\" character, for example, **AT\N2**, the backslash must be duplicated in the initialization string. Otherwise, modem control attempts to interpret \N as a control sequence, and the command will fail. To make the previous command work correctly, it must be entered in the initialization string as **AT\\N2**.

You can combine multiple AT options in the same command, and separate the commands with <CR> (carriage return) that is entered as \r in the initialization string.

NOTE

The commands sent are not checked. Any errors are indicated by warnings, and can be viewed in the debug mode, see section Troubleshooting Modem Operation on page 150.

Dial

The **Dial** parameter is used as a prefix for the destination number when making the call. Any acceptable AT commands and options can be used.

For example, the format for the **Dial** parameter is the following:

ATX1DT123456

where

ATX1DT	=	Dial string
123456	=	Destination number, which can be saved as a static parameter *at_dial* . The number should be given separately for each report that is sent

The destination number is set with the station settings through the service interface either by entering value for the *at_dial* parameter in the **Station settings** window in the terminal software or with the **spset** command in the service connection. The format for the **spset** command is:

spset *at_dial "123456"*

NOTE	Always use quotation marks around the phone number when setting the value with the station settings.

Disconnect

The **Disconnect** parameter is sent to the modem when closing the line after a completed or failed transmission. Usually this parameter consists of the control sequence +++ followed by the **ATH** command for hanging up.

Shutdown

The **Shutdown** parameter is optional and may be required by certain modems before the modem power is switched off. This parameter is only used when the modem is powered through the excitation pin.

Timing

The modem control sequence and its timing are configured by various timing parameters. As the AT interface does not give strict restrictions for handling or timing the interface, there are quite many parameters to configure, and the settings may vary from one modem to another. Lizard Setup Software contains tested defaults for the DXM and Siemens GSM/GPRS modems. These should be changed only with extreme caution and experience.

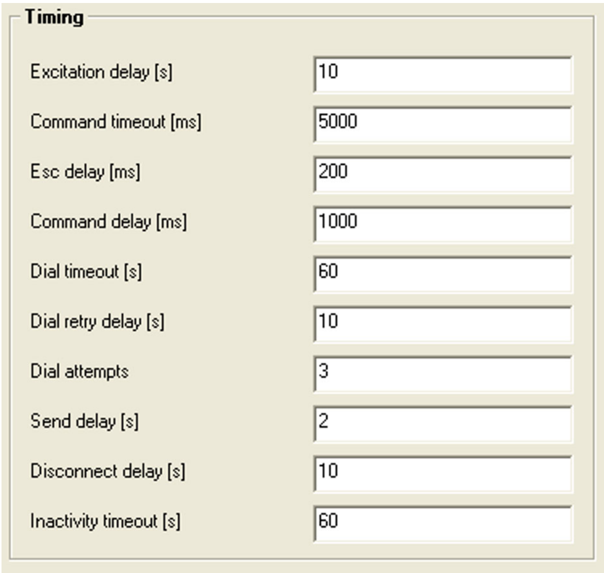

Figure 67 Timing Parameter Configuration Options

Excitation Delay

The **Excitation delay** parameter is in use only when the power control through excitation pin is in use. It defines the time in seconds from the power switch-on to the start of the initialization or the transmission command sequence. When used with a GSM modem, the time required for finding and logging into the network must also be included.

Command Timeout

The **Timeout** value in milliseconds is used for all other modem commands except for initialization (no response tests) and dial (own timeout). If the modem does not respond during this time, the sequence is aborted and a warning is issued.

NOTE	This parameter is only available on the Advanced user level.

Esc Delay

The **Esc delay** parameter defines the delay in milliseconds between the + characters when sending the control sequence.

NOTE	This parameter is only available on the Advanced user level.

Command Delay

The **Command delay** parameter defines the delay in milliseconds between each command when sending modem commands separated by <CR>.

NOTE	This parameter is only available on the Advanced user level.

Dial Timeout

The **Dial timeout** parameter defines the time in seconds to be waited after a dial command for either connect or dial failed indication from the modem. If neither is received during this period, the sequence is aborted and a warning is issued.

Dial Retry Delay

The **Dial retry delay** parameter defines the time in seconds to be waited after a failed call, caused by, for example, a busy line, before redialing.

Dial Attempts

The **Dial attempts** parameter defines the maximum number of dial attempts to be made before canceling the transmission due to an unreachable destination number, caused by, for example, a busy line.

Send Delay

The **Send delay** parameter defines the time in seconds to be waited after establishing the connection before sending data.

Disconnect Delay

The **Disconnect delay** parameter defines the time in seconds the line is kept open after the last report is sent from the QML logger to the modem. This time must be long enough to ensure that all the data has been sent by the modem.

Inactivity Timeout

In certain applications, it may be desirable to allow external, for example, service access to the QML logger through the modem. The **Inactivity timeout** parameter defines the time in seconds the line is allowed to stay idle before shutting the modem down. The setting *0* disables the operation.

NOTE	This parameter is only available on the Advanced user level.

Options

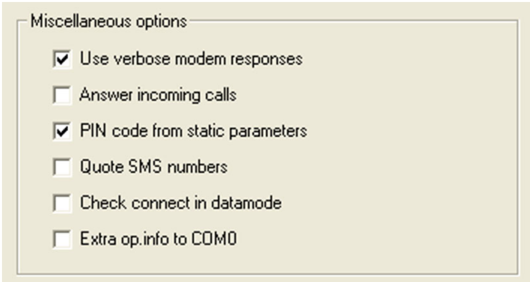

Figure 68 Miscellaneous Configuration Options

Use Verbose Modem Responses

This option defines whether or not the QML logger gets verbose responses from the modem. Do not change the default value (checked) unless specifically instructed.

Answer Incoming Calls

When this option is checked, the QML logger answers an incoming call when the first ring is indicated by the modem. SMS reception is also controlled by this setting. Check this option to enable SMS polling or command processing.

NOTE The previous modem control version used the initialization string parameter ATS0 to control inbound call answering. With this new version, do not change this parameter, always set it to **ATS0=0**.

PIN Code from Static Parameters

Cellular and satellite phones provide the PIN (personal identification number) authentication feature against unauthorized use. The PIN code is usually a four digit code, and it is stored on the removable SIM card, not in the phone.

When this option is selected, the modem control checks and issues the PIN code when needed. The PIN code is set with the *at_pin* parameter either through the terminal software **Station settings** window or using service terminal command **spset** *at_pin "code"*, where *code* is the four-digit PIN code.

NOTE	Always use quotation marks around the PIN code when setting the value with the *at_pin* parameter.

NOTE	If an erroneous PIN code is entered more than three times in a row, the SIM card will lock and a PUK code is needed for reopening it. This code can be obtained from the operator.

You can change the PIN code in two alternative ways:

1. You can use a mobile phone to change the PIN code. For detailed information, refer to the appropriate mobile phone user manual.

   or

2. You can connect the GSM data terminal to a PC and use a terminal program in the PC to change the PIN code properties.

   a. To query the PIN code status, use the following command:

      `AT+CPIN?`

      Modem responds with `SIM PIN` when it is waiting for the PIN code to be given or with `READY` when the PIN code is not in use or has already been entered.

      `+CPIN: SIM PIN`

      `+CPIN: READY`

   b. To enter the PIN code, use the following command:

      `AT+CPIN=0000`

      where *0000* is the PIN code.

   c. To enable the PIN code validation, use the following command:

      `AT+CLCK="SC",1,0000`

      where *0000* is the PIN code. Modem responds with `OK`.

d. To disable the PIN code validation, use the following command:

```
AT+CLCK="SC",0,0000
```

where *0000* is the current PIN code. Modem responds with `OK`.

e. To change the existing PIN code, use the following command:

```
AT+CPWD="SC",0000,1234
```

where *0000* is the old PIN code, and *1234* is the new PIN code.

Quote SMS Numbers

Some modems, for example, the old Siemens M20T, require SMS destination numbers to be enclosed in quotation marks in the transmission command. When this option is checked, the modem control sends the number inside the quotation marks.

Extra Op. Info to COM0

Normally the only way to monitor modem operation is to browse the generated warnings. By checking this option, more detailed information is sent to the fixed serial port COM0.

Station Name As Path Extension

This option is available only for GPRS modems when the FTP transfer is selected. When the option is selected, the modem control saves reports as files in a subdirectory named according to the parameter **sname** (station name).

Check Connect in Datamode

By default when the line is open, for example, when the modem is in the data mode, the modem control does not test responses. When this option is selected, the modem responses are tested even in the data mode. For example, the modem control detects if the connection is closed due to an error detected by the modem.

NOTE	When this option is selected, the remote system should not send any poll or acknowledge commands containing numbers, as they may be interpreted as modem responses.

Inbound Message Handling

It is possible to perform basic service operations using SMS messages. This can be enabled by selecting *Extparser* for the parameter **Inbound message handling** and enabling incoming call/SMS handling by checking the option **Answer incoming calls**. For more information, see section Remote Maintenance Commands on page 199.

Report Transmission Configuration

Report transmission is configured using a separate parameter set for each report. A new **Report transmission** frame is created each time you link a new report with the modem through the port. The existing **Report transmission** frame is deleted when you unlink the report.

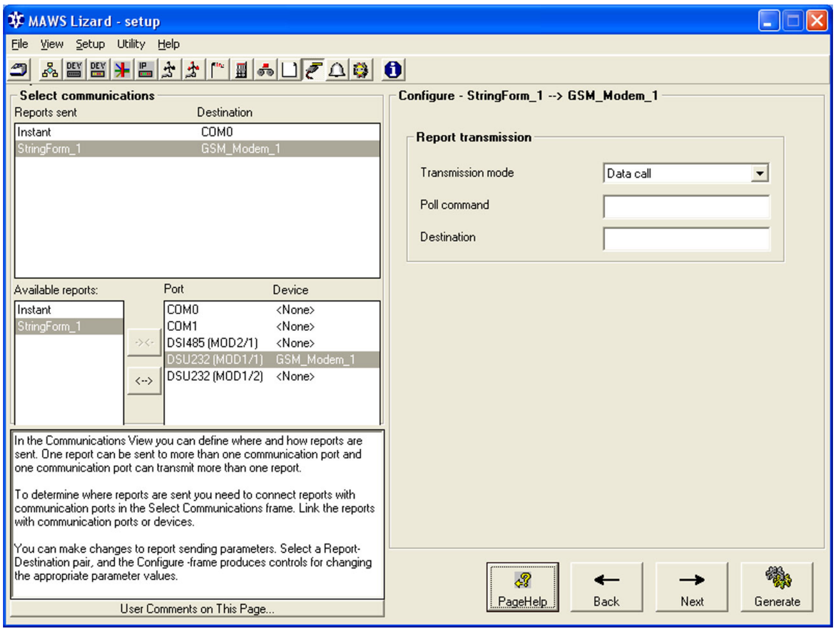

Figure 69 Report Transmission Configuration Options

Common Configurations

Report Name

The **Report name** parameter indicates which report you are configuring for the modem transfer. Because it is automatically generated by Lizard, you cannot modify it.

Poll Command

The **Poll command** parameter allows reports to be retrieved on demand with a user-definable command. When the poll request is received, the latest report is sent.

NOTE	The poll string sent by the remote system must always be terminated with carriage return (<CR>, ASCII character 13). You do not need to configure it to the **Poll command** parameter, as it is automatically expected.

NOTE	To enable polling, you must select the **Answer incoming calls** option from the **Options** dialog.

The same report can be configured for automatic sending and polling.

Destination

The **Destination** parameter specifies the destination to which the report is to be sent automatically. Depending on the setup the value can be one of the following:

- Telephone number
- E-mail address
- FTP server name
- Name of the **Station setting** parameter, which can contain one of the above.
- Empty, when the report is sent only when polled by the remote system.

It is recommended that you use the **Station setting** parameter because it provides the possibility to change the destination without changing the setup. The **Station setting** parameters can be set using either of the following:

1. With the terminal software **Station settings** window.
2. With the **spset** <*param*> <*value*> command in the service connection.

NOTE	Always use quotation marks around the parameter value for destination. For example: **spset** *sms_to "0401234567"*

PSTN-Dependent Configurations

The modem control can only use data calls for transferring the reports over the public telephone network (PSTN).

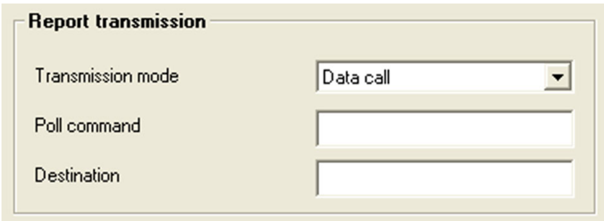

Figure 70 PSTN Transmission Configuration Options

Transmission Mode

The **Transmission mode** parameter allows selection between the two modes:

1. **Data call**: The report is automatically sent to the destination by making a call. The report can also be polled, if the **Poll command** is defined.
2. **Polled only**: The report is only sent when polled by the remote system.

Destination

The **Destination** parameter either sets the phone number or gives the name of the **Station setting** parameter that contains the phone number.

GSM-Dependent Configurations

The configuration for transmitting the reports over the GSM (or cellular) network is basically the same as for PSTN, except that the GSM option provides the possibility to transmit reports as short text messages (SMS).

Transmission Mode

The **Transmission mode** parameter allows a selection between three modes:

1. **Data call**: The report is automatically sent to the destination by making a call. The report can also be polled, if the **Poll** command is defined.
2. **SMS message**: The report is sent using the SMS message. The report can also be polled, if the **Poll** command is defined.
3. **Polled only**: The report is only sent when polled by the remote system.

NOTE

The maximum length allowed for the SMS message is 160 characters. Depending on the used modem, longer messages may be split into two or more messages. SMS messages use 7-bit coding. The use of control characters (<ASCII 32) may produce unpredictable results. For the GSM 7-bit character set, see Appendix B, GSM 7-Bit Character Set, on page 377.

Destination

The **Destination** parameter either sets the phone number or gives the name of the **Station setting** parameter that contains the phone number.

GPRS Modem -Dependent Configurations

In addition to data calls and SMS messages, the GPRS modem can be used to transfer data over the Internet.

Depending on the QML logger software version, GPRS communications can be configured in two different ways. For software version 6.00 and above, the TCP/IP functionality is provided directly by the QML logger. For instructions, see GPRS-IP Modem on page 209. For software versions earlier than 6.00, the additional iConnector module is required for TCP/IP connectivity. For instructions on GPRS configuration with the iConnector module, see section GPRS Configuration Using the iConnector Module on page 365.

Application Alternatives

Autosend and/or Polled Operation

The autosend and poll functions can be freely mixed. For example, a report that is automatically sent using e-mail transmission can also be configured to be polled with a data call and issuing appropriate poll command.

Another example of mixed use is an application where data is periodically retrieved using data calls or polling, but urgent alarm messages are sent immediately using SMS transmission.

To enable polling, enter the polling string as the **Poll command** parameter and select **Answer incoming calls** in the **Options** dialog.

NOTE	The poll string sent by the remote system must always be terminated with carriage return (<CR>, ASCII character 13). You do not need to configure it to the **Poll command** parameter, as it is automatically expected.

Answering Incoming Calls

In addition to polling, capability to answer incoming calls is needed for allowing remote service connection through the modem connection.

To enable the QML logger to answer incoming calls or to process inbound SMS messages, select the **Answer incoming calls** option.

SMS Polling

Although the QML logger can be configured to reply to incoming SMS messages, a weather station with one GSM modem is not suitable for applications where heavy inbound traffic is to be expected.

The SMS polling is meant to be used either for retrieving the data periodically or to serve as a backup for automatic SMS transmissions.

Service Operation

The service connection can be opened through the modem if the **Service connection enabled** option in the **Configure** frame of the **Optional hardware** view is checked for the port to which the modem is connected. In addition, select the **Answer incoming calls** option.

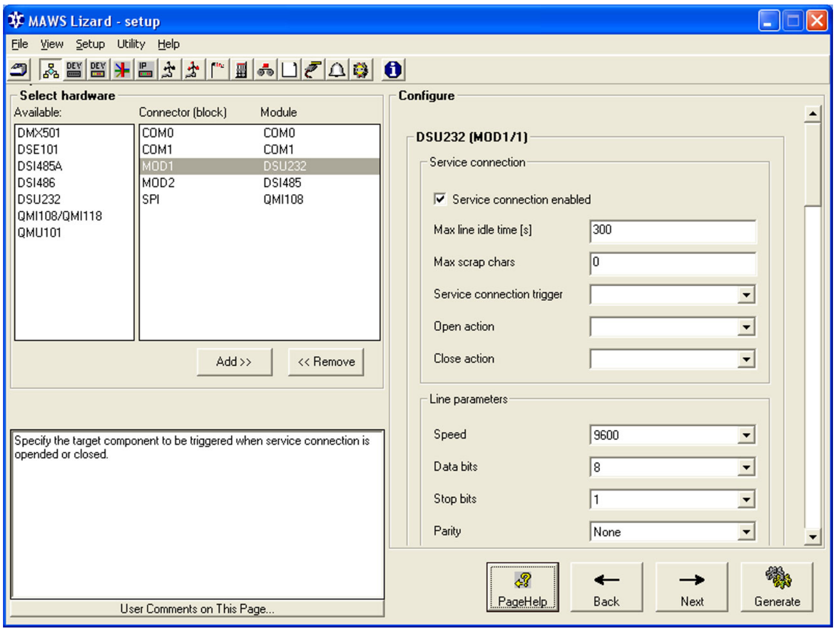

Figure 71 Service Connection Enabled Option

NOTE	Accessing the service connection while report transmission is in progress may lead to unexpected results, such as timeouts.

Troubleshooting Modem Operation

Typical possibilities for troubleshooting modem operation with the QML logger are as follows:

1. Use the **warnings** command to check for any modem related warnings.
2. To monitor different command sequences, check the **Extra op. Info to COM0** option when configuring a modem. This enables the modem control software to output various status information to the fixed RS-232 port, that is, to COM0. For example, you can monitor what is sent to the modem and how it responds. The printout includes additional internal debug data. The output is sent only when the service connection is closed.
3. To send the AT commands manually, the command interface of the modem can be directly accessed. To control the modem directly, open the service connection to the QML logger. For example, when the device is connected to the first DSU232 communication

port at the module place MOD1, type **open** *DSU232_0_0*. To terminate this operation, type **close**. While the direct connection to the modem or sensor is open, any automatic operation through the connected port is blocked. Typical parameters for the open command are presented in Table 13 on page 151. Information concerning the correct connector can be obtained in Lizard in the **I/O Connections** frame of the **Equipment** view.

Table 13 Parameters for the Open Command

Connector in Lizard	Parameter	Alias
COM0	COM0	COM0
COM1	COM1	COM1
DMX501 (MOD1/1)	DMX501_0	MOD1
DMX501 (MOD2/1)	DMX501_1	MOD2
DSI485A (MOD1/1)	DSI485_0	MOD1
DSI485A (MOD2/1)	DSI485_1	MOD2
DSI486 (MOD1/1)	DSI485_0_0	MOD1_2
DSI486 (MOD1/2)	DSI486_0_1[1]	MOD1_1
DSI486 (MOD2/1)	DSI486_1_0	MOD2_2
DSI486 (MOD2/2)	DSI486_1_1[1]	MOD2_1
DSI486SDI (MOD1/3)	DSI486SDI_0	MOD1_3
DSI486SDI (MOD2/3)	DSI486SDI_1	MOD2_3
DSU232 (MOD1/1)	DSU232_0_0[2]	MOD1_1
DSU232 (MOD1/2)	DSU232_0_1	MOD1_2
DSU232 (MOD2/1)	DSU232_1_0	MOD2_1
DSU232 (MOD2/2)	DSU232_1_1	MOD2_2
DSU232SDI (MOD1/3)	DSU232SDI_0	MOD1_3
DSU232SDI (MOD2/3)	DSU232SDI_1	MOD2_3

1. With the dual RS-485 module, the RS-232 connection is possible only to the channel B on the module, and thus the last number is 1.
2. With the RS-232 and dual RS-485 modules, the number between the underline characters stands for the module place, that is, MOD1 or MOD2, and the last number for the channel on that particular module.

NOTE	Use the **SYSINFO PORTS** command to find out which connectors to use.

4. It is impossible to have default parameters that would work everywhere with all different telephone network infrastructures. If the default parameters do not seem to work, you should try manually, for example, by using a PC terminal program, to establish a modem connection between the weather station and the data collection system, and to find out the correct parameters. Also note that the default parameters are seldom optimal, for example, in minimizing the connection time.

ORBCOMM Interface

The ORBCOMM interface uses Low-Earth-Orbiting (LEO) satellites, enabling the use of low power and small antenna in the transmitter terminals. The ORBCOMM system provides also bidirectional satellite communications through Internet e-mail.

There are two main transfer modes between the Ground Control Center (GCC) and the remote site (system).

- Message mode: Both the remote site and the GCC have simultaneous contact with the satellite.
- Globalgram mode: The satellite receives a message from the remote site and the satellite stores the message for transmission until it has contact with the GCC.

The ORBCOMM interface complies with ORBCOMM Serial Interface Specification with Document ID E89959915-Revision C. It has been tested with the Panasonic KX-G7101 ORBCOMM subscriber communicator. Panasonic KX-G7101 contains also a GPS receiver, which can be used for acquiring location and accurate time information. More information on ORBCOMM is available at http://www.orbcomm.com.

Configuring ORBCOMM Communicator

Hardware Setup

The ORBCOMM communicator is connected to the QML logger using the RS-232 interface, through the RS-232 communication module. You should add the communication module to the setup and set the parameters in the **Optional hardware** view of Lizard Setup Software. Under **Line parameters**, enter the following configuration parameters for the module:

- Speed: *4800 bps*
- Data bits: *8*
- Stop bit: *1*
- Parity: *None*

In the **Devices** view, add the communicator to the setup. Under **I/O Connections**, connect the transceiver to the previously configured communication module.

Formatting Reports

Configure the reports to be sent to ORBCOMM interface using the StringForm report template in the **Reports** view. The report must be formatted as follows:

<*address*>,<*subject*>,<*message*>

where

address	=	Destination e-mail address, or the preset speed dial number for communicator. Eight preset speed dial addresses in total can be defined when the communicator is commissioned for data transactions. Using these addresses reduces the amount of data per transmission and allows the use of the **Globalgram** mode
subject	=	E-mail message subject, for example, station name
message	=	Report contents

The fields are separated from each other with a comma (',' or ASCII character 44). The report data may contain commas as only the first two ones are interpreted to separate the fields.

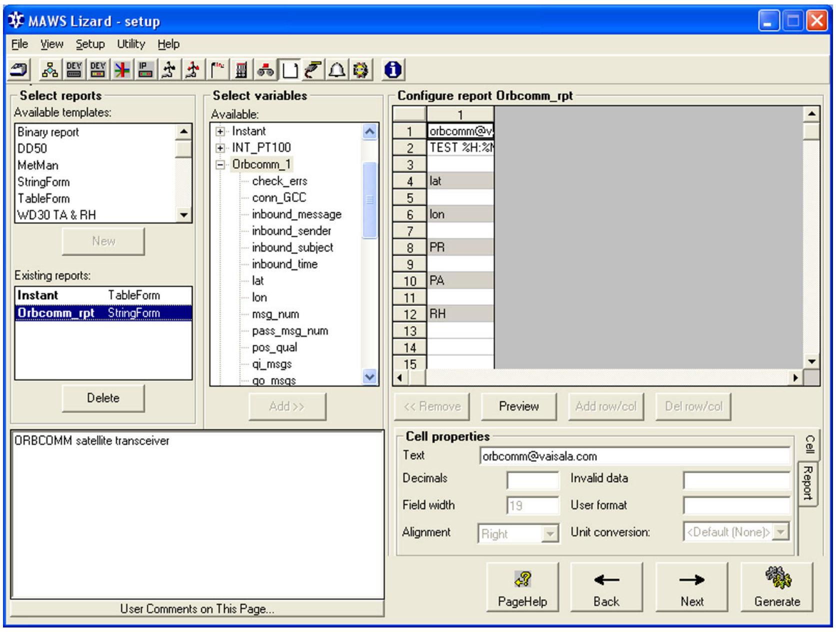

Figure 72 Reports View: ORBCOMM Report configuration Options

Because the ORBCOMM transmissions are charged by the amount of bytes transmitted, you should omit all unnecessary characters from the message as follows:

1. Insert a separator, that is, a cell containing, for example, a space or comma between the values.
2. Under **Cell properties**, enter 1 for **Field width**. This causes all cells to consume only the space needed for presenting the current value.

Configuring Communications

The ORBCOMM communications parameters are configured in the **Device configurations** view as shown in Figure 73 on page 155. For the parameter values, refer to Table 14 on page 155.

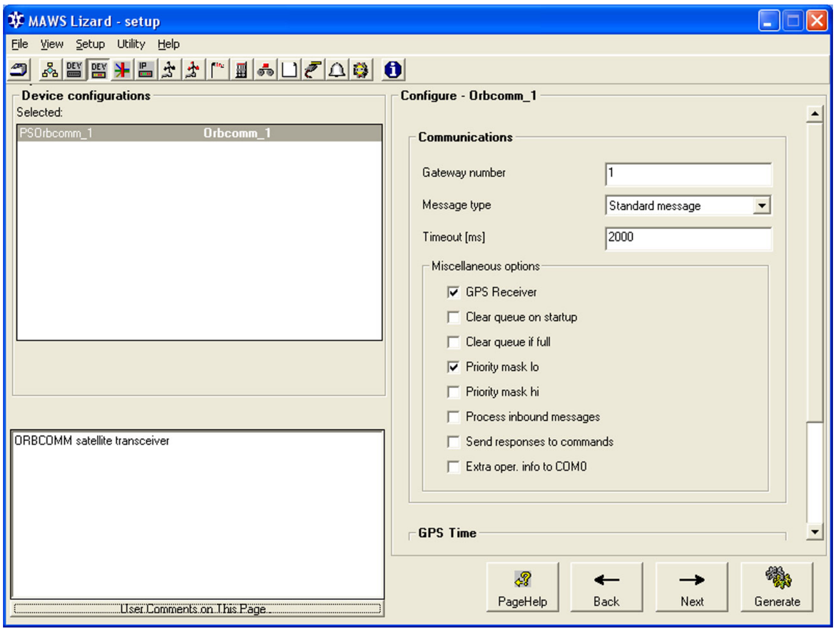

Figure 73 ORBCOMM Configuration

Table 14 ORBCOMM Communicator Parameters in Lizard

Parameter	Value	Description
Gateway number	List of active GCCs can be obtained from the ORBCOMM web site	Number of the destination GCC
Message type	Standard message	Preferred message type: Direct communication from the remote site to GCC via satellite
	Globalgram	Messages are on the queue in satellite until link can be established between satellite site and receiving GCC [1]
	Enhanced globalgram	Same as previous but you may use an e-mail address instead of the speed dial [1]
	Report	Short (max 5-byte) data report

Table 14 ORBCOMM Communicator Parameters in Lizard (Continued)

Parameter	Value	Description
Timeout	2000 ... 30000 ms	Timeout for the communication between QML logger and the communicator
Options	GPS receiver	Communicator is equipped with GPS
	Clear queue on startup	Clears transmission queue upon startup
	Clear queue if full	Clears transmission queue if it becomes full
	Priority mask lo Priority mask hi	Priority mask for communications (do not modify)
	Process inbound messages	QML logger polls and processes inbound messages as remote commands
	Send responses to commands	QML logger sends responses to the inbound commands
	Extra op. info to COM0	Sends extra operation information to COM0 for troubleshooting
Time adjustment tolerance	-1 = disable 1 ... 3600 s	When the current system time differs from the GPS time more than the tolerance set with this parameter, the system time is adjusted to match the GPS time
Parser component [2]	Empty, <*None*>, or Extparser	Parser used to process remote commands

1. The maximum length of the user data is 229 characters for the globalgram message type, while with enhanced globalgram recipient the subject and message body can have 225 characters in total.
2. Visible only in advanced Lizard operation mode.

The interface allows processing of remote maintenance commands as listed in Table 25 on page 199. To enable the use of remote commands, proceed as follows:

1. Select the **Process inbound messages** option.
2. Under **Command processing**, select *Extparser* in the **Command parser** field.
3. If you want to receive responses to commands, select also the **Send responses to commands** option.

Linking Reports

The report is linked to the ORBCOMM communicator in the **Communications** view as shown in Figure 74 on page 157.

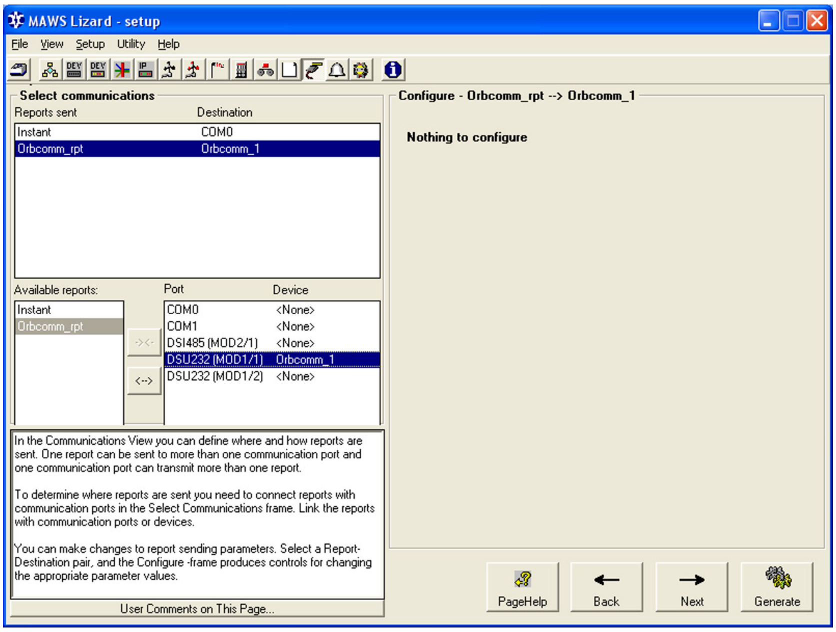

Figure 74 Communications View: Linking ORBCOMM Report

Output Variables

Table 15 on page 158 lists the available output variables that can be requested for logging and reporting.

Table 15 Available Output Variables

Variable	Data Type	Description
check_errs	Integer	Number of the failed downlink checksums
conn_GCC	Integer	Number of the connected ground stations (GCC)
inbound_message	String	Content of the latest inbound message
inbound_sender	String	Sender of the latest inbound message. If the sender is listed in the speed dial list in the GCC, only the number is available instead of the full internet address, for example, OR08
inbound_subject	String	Subject of the latest inbound message
inbound_time	String	UTC time when the latest inbound message was received
lat	Float	Latitude from GPS
lon	Float	Longitude from GPS
msg_num	Integer	Number of the next message to be sent to the communicator (SC). For example, when two messages have been sent to communicator, the value is three
pass_msg_num	Integer	Number of the message that has been acknowledged by gateway last, i.e., successfully transmitted. Value is available after first transmission. Not applicable with globalgrams.
pos_qual	Integer	Quality of the position information: 0 = GPS 1 = Good doppler ... 15= Poor doppler
qi_msgs	Integer	Number of inbound messages in SC
qo_msgs	Integer	Number of outbound messages in SC
sat_in_view	Integer	The ID number for the currently visible satellite. Zero if none is visible.
sr_status	Integer	Status code from the last system response. The code is available after first transmission. For details, refer to the Orbcomm specification.
st_diag_code	Integer	Internal diagnostic code. For values, refer to the Orbcomm SC specification.
status	Integer	General interface status: 0 = OK 1 = SC not accessed yet 2 = Reserved for future use 3 = Communication timeout 4 = Reserved for future use 5 = SC reports internal failure 6 = SC queue full
stored_sats	Integer	Number of the stored orbital elements. This indicates how many different satellites the SC has acquired.
UTC	String	Current UTC date and time from GPS. Example: Thu Nov 01 11:35:16 2007

GOES Interface

GOES Satellite Transmitters

To set the GOES satellite transmitters to operate with the QML logger, you first need to configure serial communications, data collection, data formats, and other settings with Lizard Setup Software. After this, you have to configure the transmitter-specific settings using the AWS Client software. You may also activate the GOES random and emergency transmissions on the secondary channels. The primary and secondary transmissions can be used simultaneously. The setup procedure for both the GOES primary and secondary transmissions is described in the following sections.

The QML logger has been tested with the Signal Engineering transmitter referred to as QST101/QST102 option in the device list of the Lizard setup software.

Configuring GOES Satellite Transmitter

This section describes the configuration of communication ports, log groups, reports, transmitter, and timers with Lizard Setup Software.

Hardware Setup

The GOES satellite transmitter is connected to an RS-232 communication port. It is recommended that you use a DSU232 communication module to keep the COM0 port free for service use. The port can be configured in the **Optional hardware** view of Lizard Setup Software. The configuration options are as follows:

1. Under **Service connection**, leave the **Service connection enabled** option unselected.
2. Under **Line parameters**, enter the following values:
   - Speed: *9600*
   - Data bits: *8*
   - Stop bit: *1*
   - Parity: *None*
3. Under **Handshake options**, leave all options unselected.
4. Under **Transmit control**, set the options as follows:
   - **Tx control enabled** must be selected.
   - Under **RTS polarity**, select **Normal**.
   - **RTS on before TX** can be left to its default value of *50* ms.
   - **RTS off after TX** should be set to *3000* ms.
5. Under **Buffers**, use the default value of *4096* bytes both for **Transmit** and **Receive**.

Adding Device

In the **Device** view, add QST101/102 to the setup and connect it to the first channel (MOD1/1) of the DSU232 communication module.

Device Configuration

The communications settings for the QST101/102 transmitter are configured in the **Device configurations** view. The following transmitter settings can be configured under the **Configure** frame:

- **Timeout** of the serial communication between the QML logger and the transmitter can be left to the default value of 2000 ms.
- **Transmission type**, which defines the transmission type and bit rate for the primary transmissions. The **Transmission type** selection applies both to the primary and secondary transmissions unless you define a different bit rate for secondary transmission with the **goes_sec_bps** command. Refer to Table 20 on page 177.
- Under **Miscellaneous options**, either short or long preamble can be selected for GOES 100 bps primary transmissions. For the secondary transmissions, the short preamble is always used regardless of the **Long preamble** option selection under **Misc** options. This setting has no affect either for GOES 300 bps, GOES 1200 bps, or METEOSAT transmissions because a fixed preamble is always used with these. Refer to Figure 75 on page 162.
- Under **Miscellaneous options**, you can select whether the transmission is started at the beginning of the transmission window or centered in the transmit window. Selecting the **Tx at window start** option will cause the transmission to start at the beginning of the transmit window; leaving it unchecked will center the transmission in the transmit window.
- **Time adjustment tolerance**: the system time is always synchronized according to the transmitter's clock. When the clocks differ more than this setting (in seconds), the system time is adjusted.

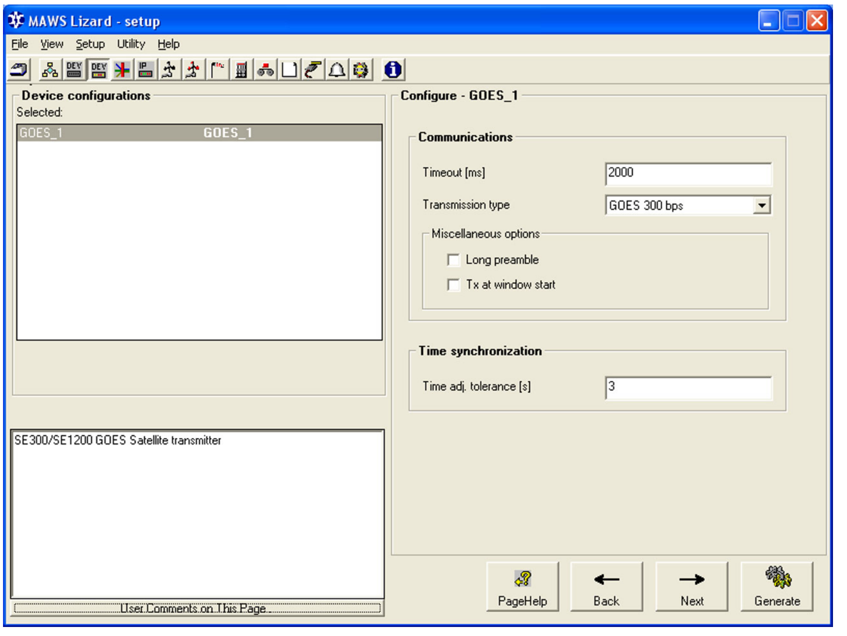

Figure 75 Device Configurations View: Configuring GOES

Report Setup for the Primary Transmissions

All data that will be sent by the satellite transmitter must first be included in a log group and then correctly formatted.

1. First create a new log group in the **Logging** view, and select all the data items that will be included in the primary transmission report. The variables will be sent in the same order as they are selected.
2. Under the **Reporting** tab in the **Type** field, select whether the data is to be sent in scan or channel ordered report format. Scan ordered format means that each data record will be sent as stored in the log group, ending with the latest record. Channel ordered format means that all measured values of the first data item will be sent first, then all values of the second data item, and so on. The latest measured value will always be the last one in the message. An example of the two formats, when temperature, humidity, and air pressure are measured, is as follows:

   Scan Ordered

   23.4 76 1023.7

   23.5 76 1023.7

   23.5 77 1023.7

   23.5 77 1023.8

   Channel Ordered

   23.4 23.5 23.5 23.5

   76 76 77 77

   1023.7 1023.7 1023.7 1023.8

3. Next, format the data. The format of the data string is defined in the **Format** field. By clicking the **Default** button, a default format string for the selected data items appears. The format string for the scan ordered example report above would be as follows:

   %[////]4.1f %[///]3.0f %[//////]6.1f\r\n

   where

%	=	Starts each data field
[ ]	=	Contains the characters for invalid data, for example, //////
4	=	Indicates the number of characters
.1	=	Indicates the number of decimals
f	=	Ends the data field (of a decimal number)

   Any characters between the previous item and the next % character will be read as a field separator before the next data value. In the example above the field separator is a space. To use a tabulator as the field separator, replace the space with \t. The characters \r\n at the end of an item indicates that a line-feed is inserted at the end of each data record.

   When specifying the number of characters shown for a data item, the data field always has fixed length. Table 16 on page 164 shows a few examples of the data value 2.345 formatted using different format strings.

Table 16 Comparing Different Format Strings

Format String	Data Item	Description
%[////]4.1f	" 2.3"	Fixed length, padded with leading spaces
%[////]04.1f	"02.3"	Fixed length, padded with leading zeroes
%[/////]+05.1f	"+02.3"	Sign of value included also for positive values
%[//].1f	"2.3"	Dynamic length, automatically changed depending on data value
%[//]+.2f	"+2.34"	Dynamic length, sign included

When the channel ordered format is used, the data string is not composed directly according to the default format string. All the values of the first data item are sent one after another according to the format string section between the first and the second % character. After this, a line-feed is automatically inserted and then all the values of the next data item are sent according to the next part of the format string. Note that if a space is required as the field separator for the last data item, the format string must end with a space. For the channel ordered example report above, the formatting would be as follows:

%[////]4.1f %[///]3.0f %[//////]6.1f

4. Set the timing and the amount of data. In each transmission, all the data in the latest records of the log group is sent. The log group saving interval, the number of records included in each transmission, and the transmission interval must all be set to match each other. For example, if data is logged at a 10-minute interval and the transmission is performed every four hours, each transmission will include the 24 latest data record, which are 10-minute data values within 4 hours. In this case, the timing parameters would be set as follows:

   - In the **Logging** view, under **Log groups** set the **Interval** to *600* seconds (=10 minutes).
   - Under the **Reporting** tab, select *24* in the **Records** field. It is the number of records included in each transmission.
   - The transmission interval is set to *4* hours with the **spset** *goes_interval* command. For information on how to set the transmission interval, see section Transmission Control for Primary Transmissions on page 175.

5. Create the primary report. In the **Reports** view, create a new report using the StringForm template and add the *orderData* variable from the *HistServer* component to it. This variable includes the scan ordered or channel ordered data string that was set up in the log group.

Report Setup for the Secondary Transmission

You do not have to set up any log group for the secondary transmissions, as only the latest data is included in the random and emergency reports. Instead, set up a report including the latest data as follows:

1. In the **Reports** view, create a new StringForm report.

NOTE	In the next step, enter also the trailing comma (,) when entering text SECONDARY, in the **Text** field.

2. In the first cell, add the text **SECONDARY**, in order to define that the report will be transmitted on the secondary channel. Add the text in the **Text** field under the **Cell properties** tab.
3. Starting from the second cell, add all desired variables to the report.
4. To include the transmission count in the report, add the variable Counter of **Report** to the report. The counter starts from zero and it will increase by one each time a new report is transmitted. The counter will not increase when repeating the same transmission.
5. All data included in the secondary transmissions must be compressed using the pseudo-binary format. Under the **Report properties** tab in the **Report formatting** field, select **Pseudo-Binary**. Configure the pseudo-binary compression of each variable as described in section Pseudo-Binary Report Format on page 167.

NOTE	The **Report formatting** field is visible only on the Advanced user level.

Pseudo-Binary Report Format

Reporting provides the pseudo-binary data compression format, which is commonly used in the GOES applications. The pseudo-binary compression uses the following algorithm:

- Each data value is compressed into a string of one or several 8-bit printable ASCII characters.
- Each 8-bit character includes 6 data bits. The two most significant bits are set to 00 if all six data bits are set; otherwise, they are set to 01. The six least significant bits include the data.
- The first character in the string includes the 6 most significant data bits and the last character includes the 6 least significant bits.
- Negative values are represented in 2's complement notation.

The user interface for configuring this function is mostly analogous to other formatting options. The **Report formatting** field is available only on the Advanced user level.

The difference to the configuration of other compression formats is that **Pre-conversion offset** is replaced with the **Signed value** option. This setting must be selected when the reported value may have negative values, and must be left unselected when the value always remains positive. Refer to Figure 76 on page 168.

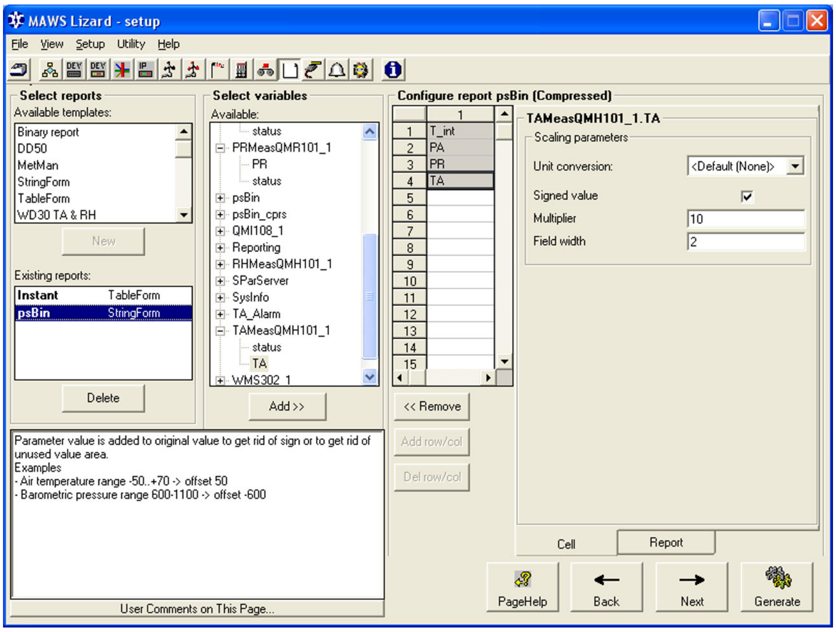

Figure 76 Configuring Pseudo-Binary Format

You have to separately define the following settings for each data item:

1. Select the **Signed value** option if the reported value may have negative values. Leave the **Signed value** option unselected if the value remains always positive.
2. To include the decimals in the data, enter a value from 1 to 1000 in the **Multiplier** field (**Pre-conversion scale** in early QML logger software versions). No decimals are included in the compressed data itself, but the data value can be multiplied by the entered value before compression. For example, if you want to have two decimals, enter 100 in the **Multiplier** field. If you do not want any decimals, enter *1*.
3. For the **Field width** option, enter the number of characters to be used for each data item. Set the field width according to the data range of the item and whether the value is signed or unsigned. The data range is calculated as the range between the possible minimum and maximum data values multiplied by the pre-conversion scale set in step 2. Select the field width so that the data range fits into the unsigned or signed data range column shown in

Table 17 on page 169. A maximum of six characters can be used for one data item.

Table 17 Field Width against Maximum Data Range

Field Width	Data Range Unsigned Data	Data Range Signed Data
1	0 ... 63	-32 ... +31
2	0 ... 4095	-2048 ... +2047
3	0 ... 262143	-131072 ... 131071
4	0 ... 16777215	-8388608 ... +8388607
5	0 ... 1073741823	-536870912 ... +536870911
6	0 ... 4294967295	-2147483648 ... +2147483647

Example 1:

Configure the report as follows to compress the measured air pressure value using the pseudo-binary format:

1. Leave the **Signed value** option unselected as the air pressure values are always positive.
2. Enter *10* as the **Multiplier** in order to include one decimal in the compressed data.
3. Use Table 17 on page 169 as reference when entering the **field width** value. In this case, enter *3* as the highest possible data value is 11000. The greatest possible air pressure value is 1100 hPa and one decimal should be included.

An air pressure value of 991.9 hPa will then be compressed as follows:

1. The value is first multiplied with the **Multiplier** value (10), resulting in 9919 which in binary format is 10011010111111.
2. The binary value is then divided into three separate 6-bit parts and compressed to the three 8-bit ASCII-coded characters as follows:

    a. The 6 least significant bits: 111111 -> 00111111 = 63 = '?'

    b. Next 6 bits: 011010 -> 01011010 = 90 = 'Z'

    c. The remaining 2 bits: 000010 -> 01000010 = 66 = 'B'

    d. The string of characters representing the compressed data value 991.9 is written as "BZ?" where the first character includes the most significant bits.

The string "B_j" representing compressed data would be decompressed as follows:

1. First the 6-bit parts of each character are extracted as follows:

    a. 'B' = 66 = 01000010 -> 000010

    b. '_' = 95 = 01011111 -> 011111

    c. 'j' = 106 = 01101010 -> 101010

2. As the last character includes the least significant bits, the resulting binary number is 000010011111101010, which corresponds 10218 in the decimal format.
3. The decimal number (10218) is divided with the pre-conversion scale (10) resulting 1021.8 hPa as the measured air pressure value.

Example 2:

Configure the report as follows to compress the measured air temperature value using the pseudo-binary format:

1. Select the **Signed value** option, as the data may be either positive or negative.
2. Enter *10* as the **Multiplier** in order to include one decimal in the compressed data.
3. Use Table 17 on page 169 as reference when entering the **field width** value. In this case, enter *2* in the **Field width** box as the air temperature value may vary between -50.0 °C and +80.0 °C, and when the value is multiplied, the value to be compressed may vary between -500 and +800.

An air temperature of -2.6 °C will then be compressed as follows:

1. The value is first multiplied with the pre-conversion scale (10), resulting in -26 to be compressed. The value is converted into the 12-bit binary 2's complement value 111111100110.
2. The binary value is then divided into two 6-bit parts and compressed into ASCII-coded characters as follows:
   a. The 6 least significant bits: 100110 -> 01100110 = 102 = 'f'
   b. The remaining bits: 111111 -> 00111111 = 63 = '?'
3. The string of characters representing the compressed data value - 2.6 is finally written as "?f".

The string 'DK' representing the air temperature would be decompressed as follows:

1. The 6-bit parts of both characters are first extracted:
   a. 'D' = 68 = 01000100 -> 000100
   b. 'K' = 75 = 01001011 -> 001011
2. The resulting binary number is 000100001011, which is +267 in the decimal format.
3. The decimal number is divided with the pre-conversion scale (10) resulting +26.7 °C as the measured air temperature value.

Linking Report to Communication Port

Proceed to the **Communications** view and link the report to the communication port to which the satellite transmitter is connected.

Timers Setup for Primary Transmissions

The timing of events must be set according to the transmission time and interval. The setup of the timers is done in the **Timers** view of Lizard Setup Software.

Data is read from a log file and the data string to be sent is composed at the occurrence of the **Create log report** event of the **HistServer_Lx** target. The transmission is scheduled at the **Create report** event of the primary transmission report. These events must be set to occur at the scheduled transmission interval and in advance of the actual transmission, for example, one minute before the start of the transmission. The **Create log report** event must be set to occur before the **Create report** event.

The transmission is not performed at the occurrence of these events, instead when the transmission is made, the data previously read by these events is sent. For information on how to configure the timing of the actual transmission, see section Transmission Control for Primary Transmissions on page 175.

Timers Setup for Secondary Transmissions

For secondary transmissions, the transmission and a given number of retransmissions occur randomly within a defined interval. The length of the transmission interval and the number of retransmissions are set with the AWS Client software as described in section Transmission Control for Secondary Transmissions on page 177. The starting time of the transmission interval is set up depending on the transmission type: emergency or random transmission.

Emergency Transmission

When the emergency transmission is selected the report is sent as the measured value passes the limit, for example, when the air temperature rises above +30 °C. Emergency transmissions are set up in the **Alarms** view of Lizard Setup Software by adding the **Create report** alarm. The transmission interval, that is, the time within which the report is randomly sent one or several times, will start 30 seconds after the report is created. Under **Condition**, the **Change delay** parameter should be set to a time that is longer than the transmission interval in order to prevent the creation of overlapping transmissions.

Random Transmission

When random transmission is selected the report is sent at a fixed interval, for example, once every six hours. The timing of random transmissions is set up in the **Timers** view of Lizard. The transmission interval, that is, the time within which the report is randomly sent once or several times, starts 30 seconds after the **Create report** event.

NOTE	The **Compress data** event must be defined to occur before the **Create report** event.

Configuring GOES Satellite Transmitter with AWS Client Software

Before any transmissions can be made, the following settings must be configured with the AWS Client software. The common settings for the primary and secondary transmissions:

- Transmitter ID
- Transmitter clock (only transmitters without embedded GPS)

For the primary transmissions you also need to enter the following settings:

- Time of the first transmission on each day
- Transmission interval
- Maximum length of the transmission window
- Transmission channel

For the secondary transmissions you also need to enter the following settings:

- Interval length of the random transmissions
- Number of repeated transmissions
- Secondary channel and bit rate
- Unitload settings

Transmitter Configuration

The configuration parameters of the transmitter are set through the service terminal with the following command:

CFG *<name> <parameter> <value>*

where

CFG	=	Configuration command
name	=	Transmitter name as given in Lizard; for example, GOES_1
parameter	=	Any of the items in the **Parameter** column in Table 18 on page 174
value		Any of the items in the **Value** column in Table 18 on page 174

Table 18 Transmitter Configuration Parameters

Parameter	Value	Description
setid	8-character hexadecimal string, for example, "010012D4"	Sets the ID of the transmitter without storing it to EEPROM
setid_eeprom	8-character hexadecimal string, for example, "010012D4"	Sets the ID of the transmitter and stores it to EEPROM
setclock	String including Year, Month, Day, Hours, minutes, and seconds in the format *YYMMDDhhmmss*	Sets the transmitter clock (only used for transmitters without embedded GPS)

A response to the command can be received by giving the command **LASTVAL** *<name> service*. The response can be either `ID set`, `ID set failed`, `Clock set`, or `Clock set failed`.

After power-up, before any transmissions can be made, the clock must be set if the transmitter does not have an embedded GPS. The transmitter ID stored to EEPROM is used after the power-up. If no ID has been stored, one must be set before any transmissions can be made.

Examples:

The transmitter ID is set to "010012D4" and stored to EEPROM as follows:

```
open

/ > CFG GOES_1 setid_eeprom "010012D4"

/ > LASTVAL GOES_1 service

Status:1  Value: ID set

/ >
```

The transmitter clock is set to 15.11.2005 13:17:00 as follows:

```
/ > CFG GOES_1 setclock "051115131700" / >

/ > LASTVAL GOES_1 service

Status:1  Value: Clock set

/ >
```

Transmission Control for Primary Transmissions

The parameters that are needed for controlling the transmissions are set with the following command:

spset *<parameter>* *<value>*

You must set the parameters listed in Table 19 on page 175.

Table 19 Transmission Control Parameters for Primary Transmissions

Parameter	Value	Description
goes_txtime	0...86400 seconds	Time of the first transmission on each day, given in seconds since midnight
goes_window	0...n seconds	Length of the transmission window in seconds
goes_interval	0...n seconds	Interval between transmissions in seconds
goes_channel	0...266	Transmission channel number

The current value of a parameter can be read with the following command:

spset *<parameter>*

The transmission can either start at the beginning of the transmit window or be centered within the configured transmission window. This is controlled by the **Tx at window start** option in the **Device configurations** view; see Device Configuration on page 161. For transmissions centered in the transmit window, if the transmission takes, for example, 40 seconds and the window length is set to 60 seconds, the transmission will start 10 seconds after the scheduled transmission time. For transmissions starting at the beginning of the transmit window, sending the message will start at the scheduled transmission time. If a message is too long to fit into the configured window, it will be cut and a # character will be added to its end.

As an example, the following command sequence will set first transmission to 02:10 (=7800 seconds after midnight), transmit window to 60 seconds, transmission interval to 4 hours (=14400 seconds), and channel number to 151.

```
/ > spset goes_txtime 7800

/ > spset goes_window 60

/ > spset goes_interval 14400

/ > spset goes_channel 151
```

Transmission Control for Secondary Transmissions

Set up the transmission control for secondary transmissions similarly to primary transmission using the **spset** command. However, different control parameters are required for secondary transmissions. The transmission control parameters are listed in Table 20 on page 177.

Table 20 Transmission Control Parameters for Secondary Transmissions

Parameter	Value	Description
goes_sec_window	0 ... n seconds	Length of the interval within which the report is randomly transmitted (Required)
goes_sec_repeat	0 ... 2 times	Number of randomly repeated transmissions (Required)
goes_sec_channel	0 ... 266	Channel number of secondary transmissions (Required)
goes_sec_unitload	1 ... n minutes	Unitload limit in minutes per day (Required)
goes_sec_load_div	1 ... n	Divisor of the unitload interval (Optional; the default value (1) is used if not defined)
goes_sec_bps	100, 300, or 1200	Bit rate of secondary transmissions (Optional; if not set, the bit rate of primary transmissions set in Lizard is used)

The unitload setting (*goes_sec_unitload*) defines the maximum allowed transmission time per day on the secondary channel. For example, when unitload is set to 8, then a maximum transmission time of 8 minutes is allowed within every 24-hour interval.

The divisor of unitload interval setting (*goes_sec_load_div*) divides this interval of 24 hours into shorter periods. For example, when **unitload** is set to 8, and the unitload divisor (*goes_sec_load_div*) is set to 4, then a maximum transmission time of 2 minutes is allowed within every 6-hour interval.

The total transmission time within an interval will never exceed the defined unitload. When the unitload limit has been reached, all further secondary transmissions within the interval will be stopped.

Troubleshooting GOES Transmission

If a transmission fails, a `Transmission failed` warning will be generated. An error code showing the reason for the failure can be read by giving the following command:

LASTVAL *<name> service*

Example:

```
LASTVAL GOES_1 service
```

Below is a list of the possible error codes:

- 0x04: Invalid transmission start time
- 0x05: Transmission overlaps another transmission
- 0x06: Invalid transmission channel
- 0x0A: Transmitter clock not set

If any of required static parameters (parameters set with the **spset** command) are invalid or missing, no transmission will be made. A warning will be generated indicating which parameter is missing or invalid.

Reading Settings and Diagnostics Information

The settings and diagnostics information of the transmitter can be read by giving the following command in the service terminal:

DO *<name><command>*

where

name	=	Name of the transmitter as given in Lizard, for example, GOES_1
command	=	Any of the commands presented in Table 21 on page 179

Table 21 Diagnostics Commands

Command	Description
reset	Resets the transmitter.
run_diag	Executes the transmitter self-test.
show_diag	Reads the results of the previous self-test.
version	Reads the version information of the transmitter.
getid	Reads the transmitter ID currently in use.
getid_eeprom	Reads the ID stored in the transmitter EEPROM.
getclock	Reads the transmitter clock.

A response to each command can be read by giving the following command:

LASTVAL *<name> service*

As an example, the transmitter ID ("010012D4") can be read as follows:

```
/ > DO GOES_1 getid

/ > LASTVAL GOES_1 service

Status:1  Value:010012D4

/ >
```

Forced Transmission

To test the connection, a forced transmission of a user-defined message can be made as follows:

1. Store the message string to be sent into the static parameter *goes_usertx_msg* with the **spset** command.
2. Store the time at which the message is to be sent into the static parameter *goes_usertx_time*, in seconds since midnight. If this parameter is not set, the transmission will be made 30 seconds after issuing the **DO** command.

   To unset the parameter, use the command: **spclear** *goes_usertx_time*
3. Activate the message with the command **DO** <*name*> *usertx*. The message will not be sent until this command has been given.

As an example, the following command sequence will send the string "Test transmission" at 08:00 (28800 seconds since midnight).

```
/ > spset goes_usertx_msg "Test transmission"

/ > spset goes_usertx_time 28800

/ > DO GOES_1 usertx
```

Meteosat Interface

The Meteosat transmissions with the GOES satellite transmitters are supported. For setting up the transmitter, refer to section GOES Interface on page 159. The main difference to GOES transmission is that in Lizard Setup Software the transmission type is set to *Meteosat 100 bps* in the **Device configurations** view, as shown in Figure 77 on page 181. Note also that the secondary transmissions are not supported with the Meteosat interface.

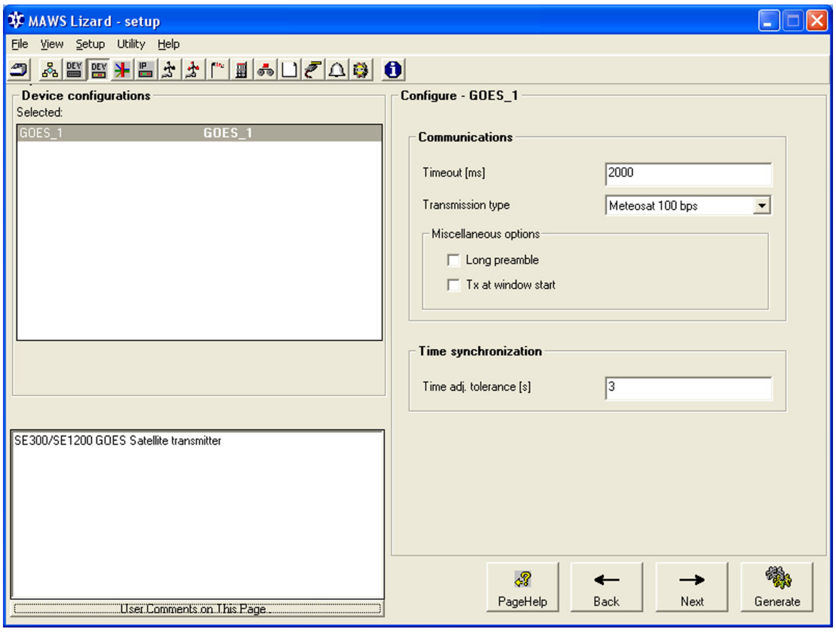

Figure 77 Device Configurations View: Meteosat Configuration Options

Autotrac Interface

The Autotrac transceiver provides bidirectional satellite communications between the weather station and the Autotrac software package of the end user. In addition, the transceiver contains a GPS receiver for acquiring location and accurate time information. The QML logger has been tested with the Qualcomm transceiver, referred to as the **Omnitracs/Autotrac** option in the **Device** list of Lizard Setup Software.

Configuring Autotrac Transceiver

Hardware Setup

The Autotrac transceiver is connected to the QML logger using RS-232 interface, typically through the DSU232 communication module. Add the communication module to the setup and set the parameters in the **Optional hardware** view. Under **Line parameters**, enter the following configuration parameters for the module:

- Data speed: *1200 bps*
- Data bits: *8*
- Stop bit: *1*
- Parity: *None*

In the **Devices** view, add the transceiver to the setup. Under **I/O Connections**, connect the transceiver to the previously configured communication module.

Device Configuration

The Autotrac transceiver is configured in the **Device configurations** view as shown in Figure 78 on page 183. For the parameter values, refer to Table 22 on page 183.

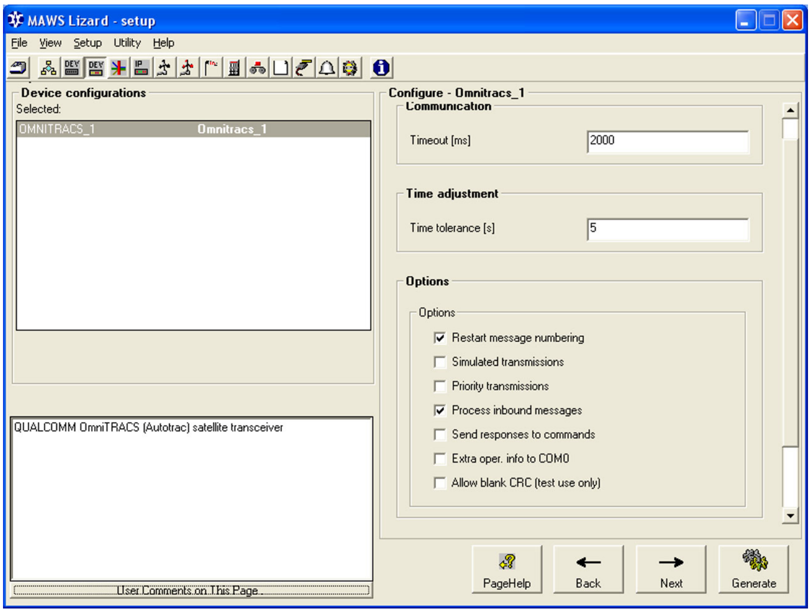

Figure 78 Autotrac Device Configuration

Table 22 Lizard Parameters for the Autotrac Transceiver

Parameter	Value	Description
Timeout	2000 ... 30000 ms	Timeout for QML logger to transceiver communications.
Time tolerance	-1 = Disable 1 ... 3600 s	When the current system time differs from the GPS time more than the tolerance set with this parameter, the system time is adjusted to match the GPS time.
Options (available only on **Advanced** user level)	Restart message numbering	Resets message sequence numbering on startup.
	Simulated transmissions	Makes simulated transmissions (tests RS-232 interface).
	Priority transmissions	Uses high priority transmission.
	Process inbound messages	Processes inbound messages as remote commands.
	Send responses to commands	Sends responses to the received commands.
	Extra op. info to COM0	Sends additional printouts to COM0 for troubleshooting.
	Allow blank CRC	Test function; not for application use.
Command parser (available only on **Advanced** user level)	Empty, <None> or Extparser	Parser used to process remote commands.

The interface allows processing of remote commands as listed in Table 25 on page 199. To enable the use of the remote commands, proceed as follows:

1. Select the **Process inbound messages** option.
2. Under **Command processing**, select *Extparser* for the **Command parser** field.
3. If you want to receive responses to commands, select also the **Send responses to commands** option.

Formatting Reports

Configure the reports to be sent to Autotrac interface using the StringForm report template in the **Reports** view. You must always use the **StringForm** report template for creating reports to the Autotrac system.

The report must be formatted with the following restrictions:

- The maximum message length is 1900 bytes.
- The ASCII messages use 6-bit coding for the transmission. This means that only the character codes from 32 to 95 (from <space> to "_") are allowed in the reports. The interface checks the presence of illegal characters, and conflicting messages are rejected with a warning.

When needed, extra spaces can be removed from the report as follows:

1. Insert a separator, that is, a cell containing, for example, a space or comma between the values.
2. In the **Cell properties** tab, enter *1* for **Field width**. This causes all cells to consume only the space needed for presenting the current value.

Linking Report to Communication Port

Report is linked to the Autotrac transceiver in the **Communications** view; see Figure 79 on page 185.

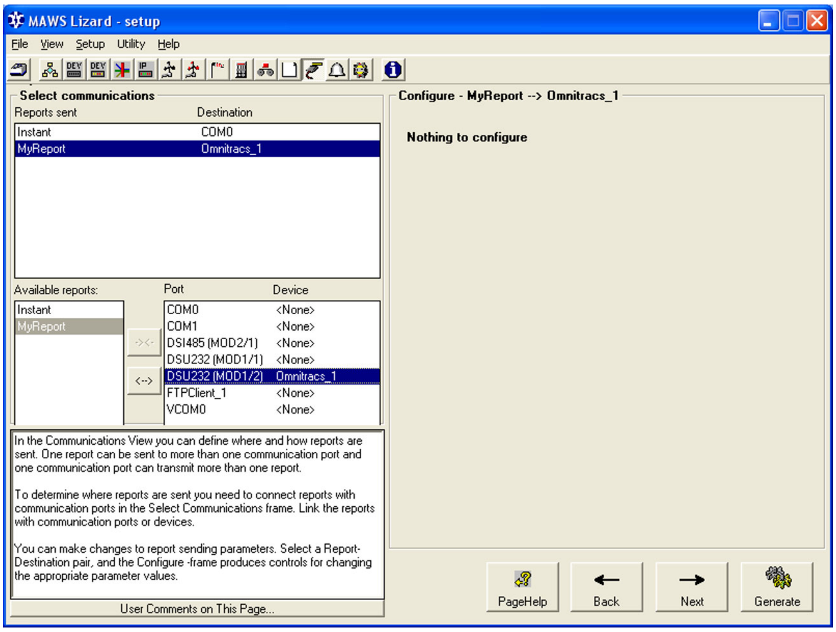

Figure 79 Linking Report to Autotrac Transceiver

Inmarsat-C Interface

The Inmarsat satellite network is the world wide satellite network of four geostationary satellites providing excellent network coverage between 70° North and 70° South. Inmarsat-C, a global satellite communication service, enables the users to send and receive data from/ to a mobile terminal via Internet e-mail and PSTN fax.

The QML logger supports Inmarsat-C transceiver model TT-3026L/M manufactured by Thrane & Thrane. The transceiver is connected to the logger using the RS-232 connection. With the transceiver you can send StringForm reports up to 10 Kilobytes from the QML logger via Internet e-mail or PSTN fax. You can also send terminal commands via Internet e-mail to the logger. The logger processes the commands and sends command response back as an e-mail reply.

The QML logger also has support for the GPS receiver included in TT-3026L/M. The GPS receiver provides accurate time, position, and also speed. These can be utilized in calculations and reports, and to automatically adjust the system time.

Hardware Setup

In the **Optional hardware** view of Lizard Setup Software, configure the serial port as follows:

- Speed: *4800 bps*
- Data bits: *8*
- Stop bit: *1*
- Parity: *None*

In the **Devices** view, add the Inmarsat transceiver to the setup and connect it to a serial port.

Configuring Communications

In the **Device configurations** view, you can configure the communications settings for the Inmarsat transceiver.

For the fax transmission, under **Configure** in the **Device configurations** view, the **Transmit control string** must be changed to contain the following parameters:

- t 1 = Destination PSTN
- e T30 = Destination extension FAX

NOTE	Do not use control characters, that is, ASCII characters below ASCII 32 in the message body as they may have special use in the RS-232 protocol used.

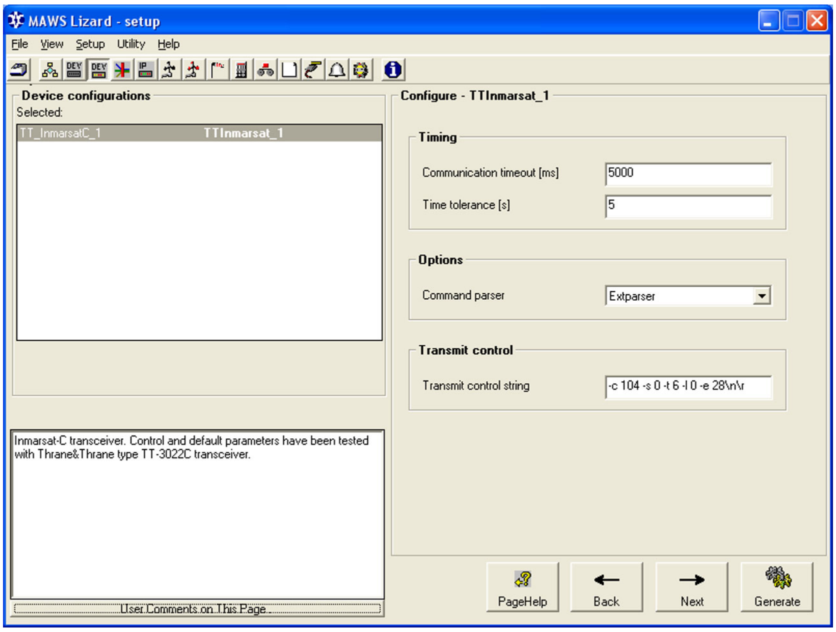

Figure 80 Configuring Inmarsat-C Options

1. Under **Timing**, set the following parameters:
   a. Set the **Communication timeout** (in milliseconds) for the RS-232 link between the QML logger and the transceiver.
   b. Set the **Time tolerance** (in seconds) for adjusting the logger clock according to the GPS time from the transceiver.

2. Under **Options**, select **Command parser** for handling remote commands. When you select **Extparser**, the remote commands are processed as described in section Remote Maintenance Commands on page 199.

3. Under **Transmit control**, you need to enter a string that depends on the service provider and destination network. The default parameters are listed in Table 23 on page 188.

Table 23 Default Transmit Control Strings for Inmarsat-C

String	Description
c 104	Destination ground LES/provider EIK/Sonera
s 0	Delivery service, store-and-forward
t 6	Destination Internet
e 28	Destination extension e-mail
l 0	Defines standard 8 bit alphabet

Formatting Report

In the **Reports** view, the reports that are sent to Inmarsat must be formulated according to the following examples. The report structure for the Internet e-mail is as follows:

```
TO:some.one@somewhere.net<CR><LF>
CC:someone.else@somewhere.else.net<CR><LF>
SUBJECT:Your mail subject<CR><LF>
<CR><LF>
message body
```

The report structure for fax (phone number 3589123456) is as follows:

```
3589123456+<CR><LF>
<CR><LF>
message body
```

Linking Report to Communication Port

In the **Reports** view, link the report to the communication port used by the Inmarsat transceiver.

Iridium Interface

This section provides information on the Iridium satellite transmitter communications option, how to interface the Iridium satellite transmitter to the QML logger, and more specifically, how to use the Short Burst Mode in binary transmission. Iridium transmission using data calls can be used like any other modem.

Iridium is the only data solution with a complete coverage of the Earth including oceans, airways, and Polar Regions. Iridium delivers the essential communications services to and from remote areas where no other form of communication is available. The Iridium constellation of 66 low-earth orbiting (LEO), cross-linked satellites operates as a fully meshed network and is the largest commercial satellite constellation in the world.

Short Burst mode (SBD) is a mechanism to deliver short data messages over the Iridium satellite network to the Internet. Each SBD message can be up to 1960 bytes in length.

For more information about Iridium communication, refer to www.iridium.com.

Configuring Iridium Transmission

Iridium transmission is configured using Lizard Setup Software. Proceed as follows:

1. Configure the serial transmission hardware in the **Optional hardware** view as shown in Figure 81 on page 190.

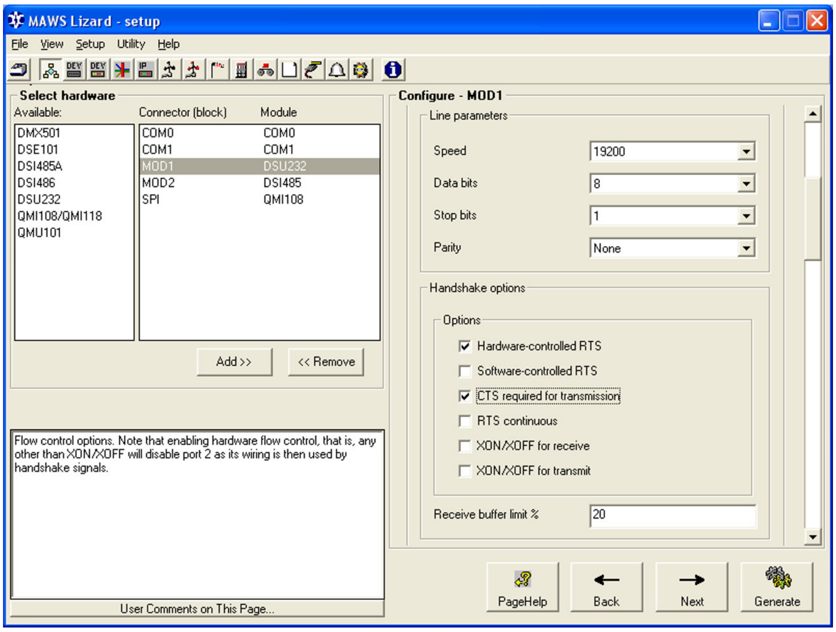

Figure 81 Optional Hardware View: Selecting Configuration Options for Iridium Transmission

In this example, MOD1/1 on the DSU232 communication module is selected for use with Iridium.

It is recommended that you keep the default settings in the **Configure** frame:

Option	Default Value
Service connection enabled	selected
Speed	*19200*
Data bits	*8*
Stop bits	*1*
Parity	None
Hardware-controlled RTS	selected
CTS required for transmission	selected

2. Proceed to the **Devices** view by clicking **Next**.
   a. Select Modem Iridium in the **Available** list.
   b. Click the **Add** button.
   c. Double-click the **Connector** list for Iridium_modem_1 in the **I/O Connections** frame in order to select the COM port.
   d. Leave the EXCITATION selection to <None>.

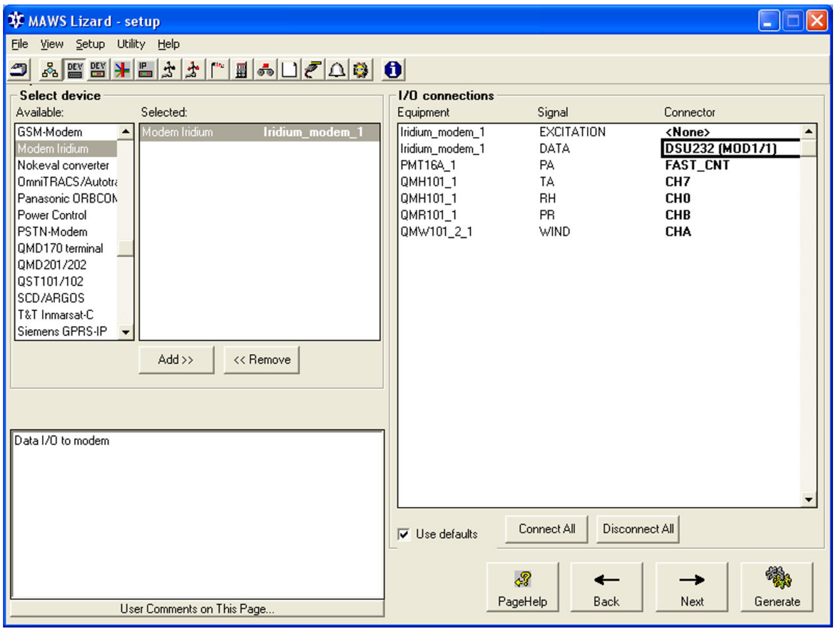

Figure 82 Equipment View: Selection I/O Connection for Iridium Modem

3. Proceed to the **Device configurations** view to configure the Iridium communication settings.

   a. For the device **Iridium_1**, you can select whether extra information on the connection is sent to COM0. If you select the **Read clock time from Iridium** option, the Iridium clock counter is read when the SBD transmission has been completed, and the QML logger clock is set according to this counter.

   b. For the device **Iridium_modem_1**, you can configure the modem control options as for any other modem type. See Figure 83 on page 192.

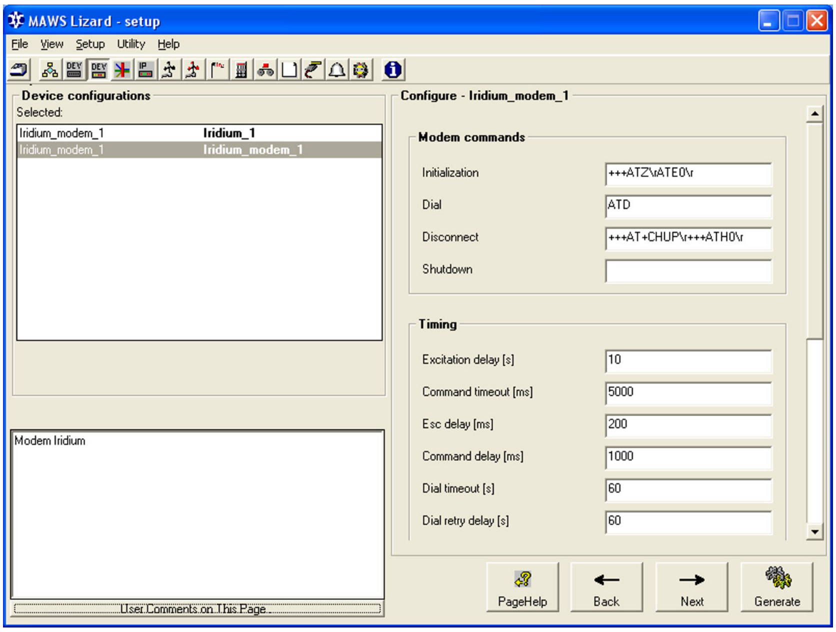

Figure 83 Device Configurations View: Configuring Iridium Transmission Parameters

The **Powering device** option lists all the power control components included in the setup; see Figure 84 on page 193. For more information on using this option, see section Power Control Option in Vaisala HydroMet™ Data Collection Platform User's Guide, Volume 2.

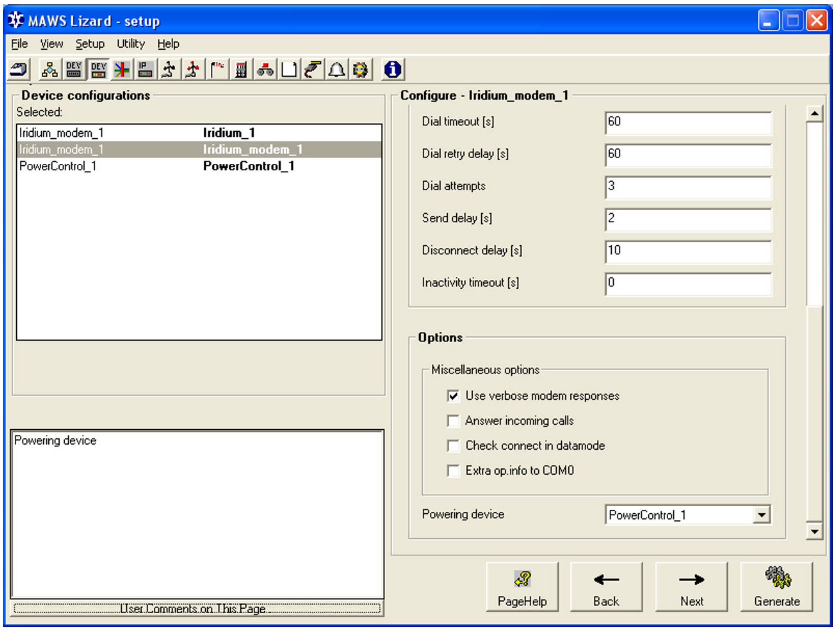

Figure 84 Power Control Option for Iridium

4. Proceed to the **Communications** view.

5. In the **Select Communications** frame, link a report to the Iridium mode. See Figure 85 on page 194.

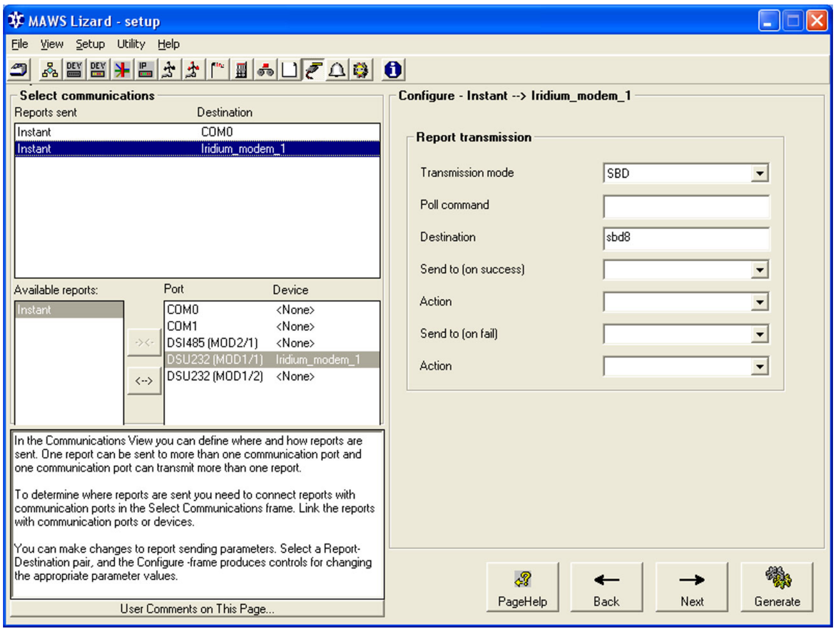

Figure 85 Communications View: Selecting Report Transmission Options for Iridium

a. Select the report form from the **Available reports** list.

b. Select a port having *Iridium_modem_1* as the device.

c. Click the linking button to make a connection between the report and the Iridium modem.

6. Select the Short Burst mode in the **Report transmission** frame by choosing SBD from the **Transmission mode** list; see Figure 85 on page 194.

The fields **Send to** (**on success** and **on fail**) are currently used only when transmitting via SBD. You should leave the fields empty unless otherwise informed.

SCD/ARGOS Interface

The HAL2 satellite transmitter can be used with the QML logger to send data using either the SCD-1 or ARGOS satellite systems. This section includes instructions on how to configure the transmissions using Lizard Setup Software and HyperTerminal or other such program.

HAL2 in Lizard Setup Software

Before the HAL2 satellite transmitter can be successfully used with the QML logger, it must first be configured for use in the Lizard Setup Software, after which the setup thus created is exported to the QML logger.

In order to make HAL2 part of the setup, you need to run an external script. This script is included in the setup provided with the MAWS system.

HAL2 Transmitter Configuration

To configure the HAL2 SCD/ARGOS transmitter for use, proceed as follows:

1. On the QML logger, connect the HAL2 transmitter to port DSU232_0_0.
2. On the QML logger, connect your PC to port COM0.
3. Using HyperTerminal or other such program on your PC, open a maintenance connection to the QMS logger (AWS Client may not show escape sequences correctly; therefore, it is recommended that you use HyperTerminal or an equivalent program).
4. Give the following command:

   ```
   HAL2 enter
   ```
5. Redirect the maintenance connection to HAL2 by giving the following command:

   ```
   open DSU232_0_0
   ```
6. Display the HAL2 menu, press ENTER. The following menu is displayed:

   ```
   MODE REGLAGE USINE
   Software:4.03 [00000000]

   A REGLAGE DE LA FREQUENCE
   B REGLAGE DE LA PUISSANCE RF
   C REGLAGE DE LA MODULATION
   D DEBUG CARTE
   E TCXO

   F PARAMETRES EMISSIONS
   G SEQUENCE EMISSIONS

   H DOWNLOAD
   I UPLOAD

   J AUTORISER BOOTLOADER
   K RAFRAICHIR INFO
   L NUMERO DE SERIE

    Tapez [X] pour sortir
   ```

7. Modify the HAL2 configuration by pressing F. The following menu is displayed:

```
A) EMIS. COURANTE : 1
B) TR FIXE        : 0
C) ADRESSE 28     : 0
D) MESSAGE INTERNE: 0
E) HEADER PRESENT : 0
F) NBR OCTETS EXT : 32
G) ID             : 00000001
H) FREQUENCE : 401647000
) PUISSANCE : 1W
J) HEADER : 25
K) RESOL PUISSANCE: 0
L) RESOL ALIM : 10
M) RESOL TEMP. : 0
N) RESOL COMPTEUR : 0
O) TOR1 : 0
P) TOR2 : 0
Q) RESOL AN1 : 0
R) RESOL AN2 : 0
S) RESOL AN3 : 0
T) RESOL AN4 : 0
U) MSG:
FFFFFFFFFFFFFFFFFF001FFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFC00
V) MEMORISE
```

8. To change a value for a menu item, first press the corresponding letter key and then press ENTER to change the values. Refer to HAL2 documentation for the correct values. At minimum, the following menu items need to be modified:

Table 24 Minimum Changes to HAL2 Parameters

Parameter	Value
C) ADDRESS	Use a 25-bit or 28-bit address
D) MEDSSAGE INTERNE	To enable external messages to be sent from MAWS, set to 0.
G) ID	Set to an appropriate value.
H) FREQUENCE	Set to an appropriate value.
I) PUISSANCE	Set to an appropriate value (transmission power)

9. Save your configuration by pressing V.

10. Exit the HAL2 menu by pressing X.
11. Close the maintenance connection by typing the following command:

    ```
    close
    ```

12. Reopen the maintenance connection by typing the following command:

    ```
    open
    ```

13. Enable reports to be sent to the HAL2 transmitter by typing the following command:

    ```
    HAL2 exit
    ```

NOTE	The last step is important! To enable reports to be sent to HAL2, run the HAL2 EXIT command.

Remote Maintenance Commands

The remote maintenance commands provide parallel functionality to the service interface. The commands included are a subset of the service interface commands, and responses are more suitable to the non-interactive use.

NOTE	Remote command interface operates only with satellite transceivers having bi-directional communications capability, such as Orbcomm, Inmarsat C, and Autotrac.

The processing of the remote maintenance commands is enabled for the satellite interfaces by selecting *Extparser* as the **Command Parser** parameter. In addition, with Autotrac and Orbcomm, the **Process inbound messages** option needs to be selected.

If no data is expected in return from the QML logger, the string `OK` is returned after a successful command. A failed commands return the string `Fail:` followed by a more detailed explanation.

The available remote maintenance commands are listed in Table 25 on page 199.

Multiple commands can be sent in a single message separated by commas, for example:

```
spset par1 55.2, spset par2 66, spset par3 abacab
```

NOTE	The commands themselves are case insensitive, but for parameters, the upper/lowercase selection is in effect.

Table 25 Remote Maintenance Commands

Command	Use	Parameters	Output
spset <*param*> [*value*] [1] [2]	Setting or reading the static parameters	Parameter name Parameter value	When sent without [*value*], command returns the current setting for parameter. String '<*param*> not set' is returned if no value is available.
spclear <*param*>	Clearing of the static parameters	Parameter name or ALL for clearing all parameters	n/a

Table 25 Remote Maintenance Commands (Continued)

Command	Use	Parameters	Output
reset	Resetting the device		n/a, not even OK.
ctev <*tevID*><*interval*> [P]	Changing the timer interval	Timer id from Lizard information view. Interval in [s]. P to preserve the parameters over reset	n/a
errors [*clear*] **warnings** [*clear*]	Reading or clearing a warning or error logs	No parameters to read errors or warnings. Clear to clear errors or warnings	Example return value for the log readout Cnt: 1 Src:idle.cpp(73) - Cnt: number of occurrences - Src: source file - (nn): source file line number. Indications are separated with commas.
lastval <*comp*> <*param*> <*status*>	Reading of the application variable	Component name Variable name The status value included to output, 0=no, 1=yes	Value with the optional status as listed in the sensor status list.
do <*comp*> <*action*>	Executing an action, for example, a measurement	Component name Action identifier	n/a
usermsg	User message without any action		n/a
log <*group*> <*starttime*> <*records* | *endtime*> [3]	Reading set of logged data	Loggroup Ln Starttime in YYMMDDhhmmss Number of records or end time	Logged data from single group split into one row per record. Values only, no header data.

1. <> = Parameter is compulsory
2. [ ] = Parameter is optional
3. | = Parameters are alternatives

Command example:

```
spset my_setting "This is a string with spaces"
spclear my_setting
ctev 15 60
warnings clear
lastval Instant rep 0
do TAMeasQMH101_1 TA
usermsg this has no meaning
log L0 030703120000 5
```

CHAPTER 5

CONFIGURING TCP/IP-BASED TELEMETRY

This chapter provides instructions for configuring TCP/IP-based telemetry options.

NOTE	Configuring TCP/IP-based telemetry requires a working knowledge of TCP/IP networking. This User's Guide only describes configuration carried out at the QML logger, and does not attempt to provide general guidelines for designing TCP/IP-based networks.

Configuration Buildup

Hardware and protocol layers of the TCP/IP implementation form a network setup. The configuration for the network setup can come from the following sources:

- Setup file for the QML logger
- Static parameters groups (values can be set via AWS Client at runtime)
- Network configuration protocols (TCP/IP behavior configured via the network itself)
- Configurations of devices other than the QML logger

Physical and Logical Interfaces

The same physical communication device can be accessed via multiple logical communication interfaces. This way, the same device can be used to access different networks or services. These logical interfaces can, in turn, be accessed through different application services, for example, to send reports and log files.

The relationship between physical and logical interfaces and the services using them are illustrated in Figure 86 on page 202.

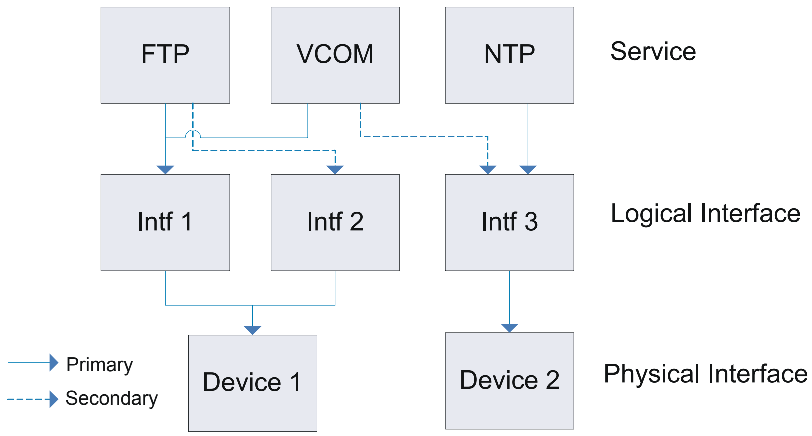

Figure 86 Relationship between Logical and Physical Interfaces

In the figure, the FTP service uses interface 1 as its primary connection and interface 2 as a backup connection. Both these interfaces use device 1 as their physical interface. Even though both interfaces use the same physical device, they can be connected to different networks, thereby enabling the FTP application to, for example, access a backup FTP server in case the primary FTP server is down. The secondary logical interface for a service can also use another physical interface, creating a backup mechanism if one communication device breaks down. The different services can also use the same logical or physical interface as another service.

TCP/IP communications channels (such as TCP sockets) are accessed via virtual COM ports in the QML logger software. The virtual ports map to the actual logical and physical interfaces, depending on the configuration. The report or measurement configurations are created as with earlier Lizard versions, with the difference being that the source or destination of data can be a virtual COM port in place of a physical serial port, a FTP client for sending a report, and so on.

General Order of Configuration

For communications devices, interfaces, and TCP/IP-based services, the general order of configuration is as outlined below. The configuration order is the same as that of MAWS Lizard configuration views.

1. Add the communication hardware in the **Optional hardware** view. This hardware can either be the Ethernet Communication Module DSE101 for Ethernet connections or, for modems, the hardware module to which the modem is connected.
2. For modems, add the communication device and connect it to a port in the **Devices** view.

NOTE	For Ethernet Communication Module DSE101, no additional device needs to be added in the **Devices** view.

3. For modems, configure the required parameters for your device in the **Device configurations** view.
4. Add and configure the interfaces using the communication device in the **Communication interfaces** view.
5. Add and configure the TCP/IP-based services using the interface in the **IP Services** view.

Network Warnings and Errors

As errors, often without significant functional impact, yet providing some information about the status of the network connection, occur quite frequently in IP communications, IP-related runtime warnings are not shown with the standard **warnings** command nor indicated by the command prompt. Use the **net warnings** command to list the network-related warnings.

Communication Devices

This section presents the TCP/IP-based communication devices and their configuration in MAWS Lizard.

Ethernet Communication Module DSE101

Ethernet Communication Module DSE101 is added to the setup in Lizard's **Optional hardware** view. The module uses an automatic configuration mode, and there are no user adjustable parameters.

External Communications Devices

Hardware configuration for external communication devices is done in the **Devices** view, shown in Figure 87 on page 204.

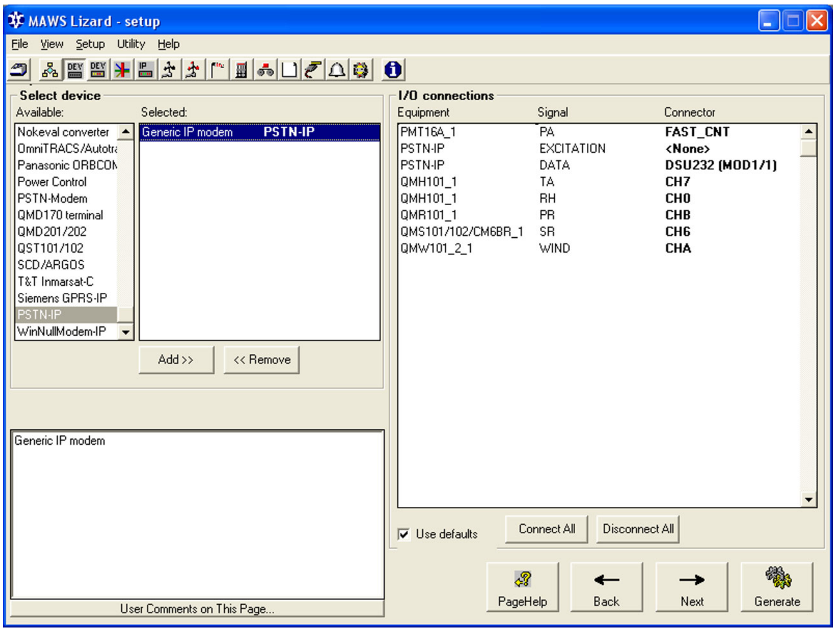

Figure 87 Communication Device Selection

NOTE	Adding a modem into the setup automatically introduces a power control component. It is strongly recommended that automatic modem power control, using, for example, a relay, is included whenever an external modem is used. The relay is used in order to prevent the modem from stalling.

The TCP/IP-capable communication devices included in the list of available equipment in the **Devices** view are listed in Table 26 on page 205.

Table 26 TCP/IP Communication Devices in Lizard

Device name	Description	Use
NullModem-IP	Used to connect QML to another computer using RS232 cable. Note: this null modem does not work with a Windows PC.	Connecting the QML logger locally to a router, Linux PC, or similar devices
WinNullModem-IP	Windows-specific version of the null modem device	Connecting the QML logger locally to a Windows PC
Siemens GPRS-IP	TCP/IP connection over GSM/GPRS network using Siemens modem. Tested to work with MC35i.	Connecting the QML logger to remote systems using the GSM/GPRS network as the ISP connection
PSTN-IP	TCP/IP connection by dialing remote modem(bank) using a PSTN modem. Tested to work with DXM421.	Connecting the QML logger to remote systems using the PSTN network as the ISP connection

NOTE	All devices listed above are for IP use only. They provide a traditional, readable ASCII format modem operation only for inbound connection with the communication interface parameter **Incoming** set to **Yes, without PPP**. Non-IP modem controls must be used if ASCII operating mode is required. For configuration instructions on non-IP modems, refer to Chapter 4, Configuring Telemetry Options, on page 131.

Common Modem Parameters

All different IP-modem types share the same configuration view: **Device configurations**, shown in Figure 88 on page 206. This section describes the parameters for basic use. For advanced modem configuration, refer to section Generic IP Modem Configuration on page 275.

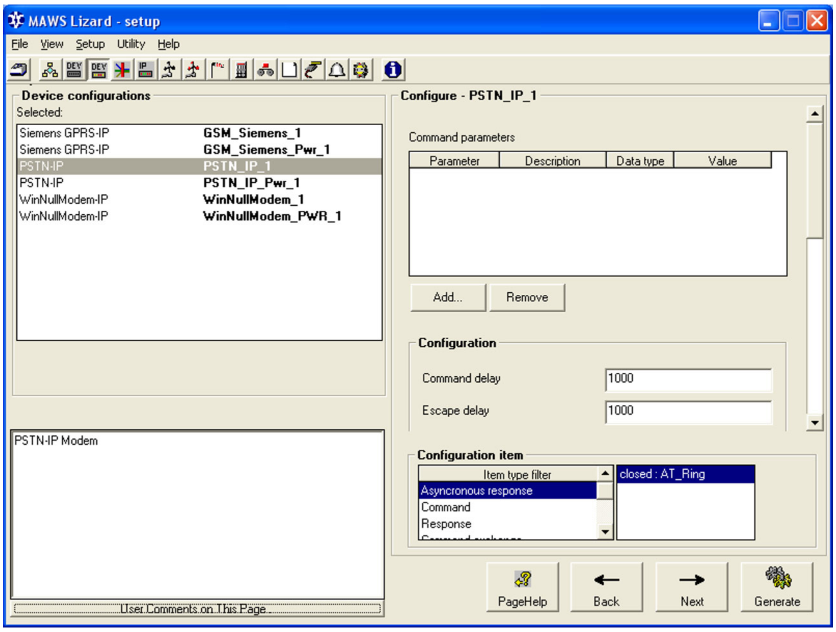

Figure 88 Modem Configuration View

The top of the view shows the parameter set values associated with the device. The parameter set and the configured component have the same name. Parameters include station-specific settings and commissioning time settings, such as PIN codes. For certain modem types, this list may be empty. Usually there is no need to add or remove parameters, but just to change the value to an appropriate default value.

Common configuration parameters for modem controls are listed in Table 27 on page 207. All parameters may not be applicable for a certain modem type.

Table 27 Modem Control Parameters

Parameter	Use
Command delay	Delay in milliseconds between successive commands.
Escape delay	Delay in milliseconds between escape characters (+) used when closing the connection.
Dial retry count	Number of retries if opening the connection fails.
Dial retry delay	Delay in milliseconds after a failed connection attempt before attempting a new connection.
Options/Reset after failure	Use power control output to reset the device after failure.
Options/Preserve power	Keep modem powered only with active connection.
Options/Answer incoming	Answer incoming calls.
Options/Debug output	Provide additional operation information to COM0.
Power control	Power control component
Powering delay	Delay in milliseconds between modem power-up and the first command.
Carrier detect input	Optional input component used to monitor the carrier detect signal from the modem.
Carrier detect variable	Variable to monitor in the carrier detect component.
Mask value for CD status	Mask logically ANDed with value of the carrier detect variable. If not set, value 0 = CD off, else CD ON.

The lowest part of the view provides an interface for managing modem commands and responses. For configuration instructions, refer to section Generic IP Modem Configuration on page 275.

Null Modem

Null modem is not a physical communication device, but a commonly used name for a cross-connected RS-232 cable for interconnecting two computers.

To include this component to setup, add the device **NullModem-IP** or **WinNullModem-IP** as shown in Figure 89 on page 208. Note that for a direct cable connection to a Windows PC, the device to use is **WinNullModem-IP**.

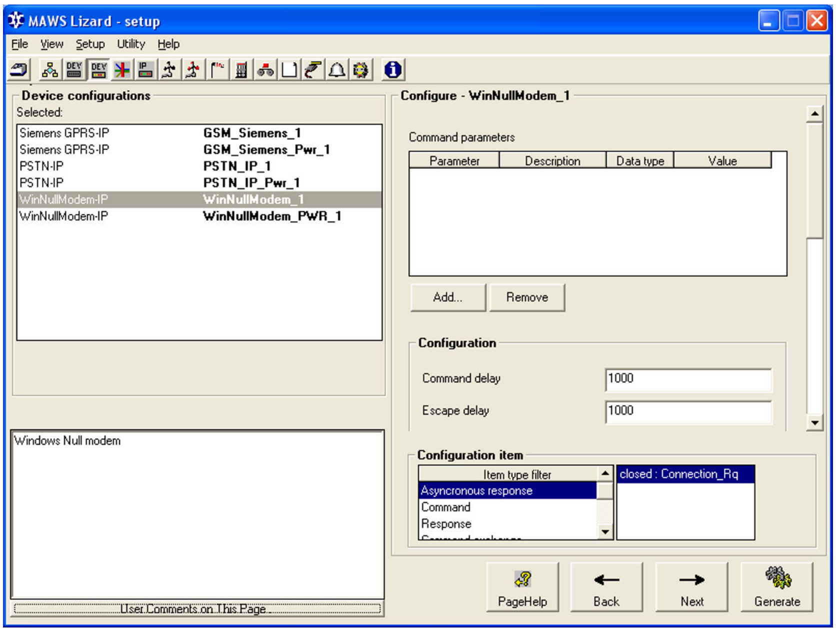

Figure 89 Null Modem Configuration

NOTE	There are no station-specific parameters associated with the null modems. A power control component is included, but usually there is no real need for it.

GPRS-IP Modem

A GSM modem with GPRS capability can be used to connect the QML logger to the Internet, or directly to a private network through a dedicated access point (APN). Also data call (CSD) can be used as an IP carrier.

This modem control has been tested with Siemens MC35i GSM/GPRS modem.

To include this component to setup, add the device **Siemens GPRS-IP** and configure it as shown in Figure 90 on page 209. The only station-specific parameter to configure is **pin**, the PIN code for the GPRS modem.

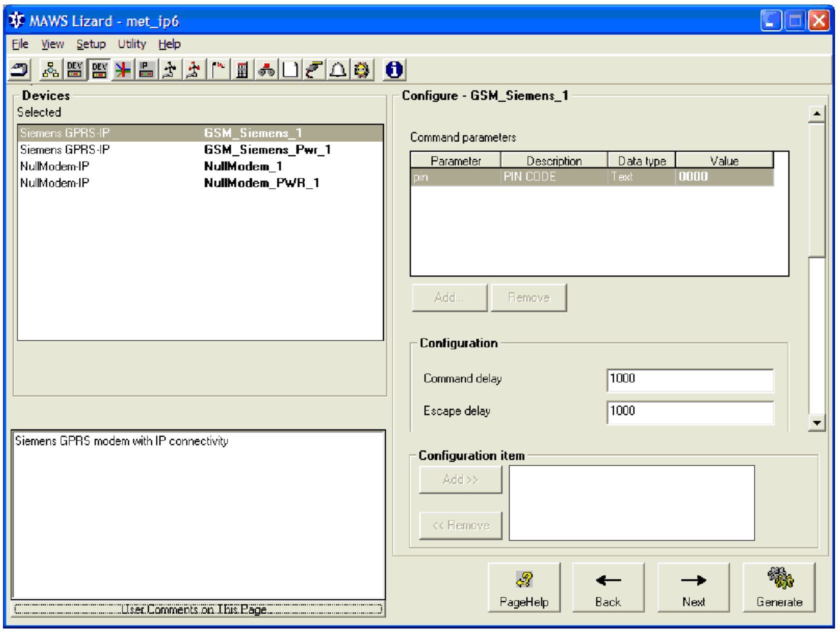

Figure 90 GPRS-IP Modem Configuration

Using Extended Characters with GPRS-IP Modem

In GSM-based communications, the GSM 7-bit character set is used instead of the ASCII character set. Due to this, the character values for extended characters are different from ASCII-based communications. With the QML logger, extended characters based on the GSM character set may be used in access point names, user names, passwords, and reports to be sent over a GSM-based link. The GSM 7-bit character set is presented in Appendix B, GSM 7-Bit Character Set, on page 377.

For example, if your report contains the @ character, its character value in the GSM character set is 0 instead of the 64 in ASCII. Note also that the GSM character set does not include certain control characters, such as <SOH> (ASCII character number 1) or <STX> (ASCII 2) often used in ASCII-based reports.

You can enter extended characters using the GSM character set values by entering the characters with the format **\xHH**, where **HH** is the GSM character number in hexadecimal. This format can be used in the **Communication interfaces** view for configuring parameters for interfaces using a GSM-based communication device and the **Reports** view for using extended characters in reports.

PSTN-IP Modem

A modem for public switched telephone network can be used to provide an IP connection carrier for the QML logger. This modem control has been tested with Vaisala PSTN modem DXM421.

To include this component in your setup, add the device **PSTN-IP** and configure it as shown in Figure 91 on page 210.

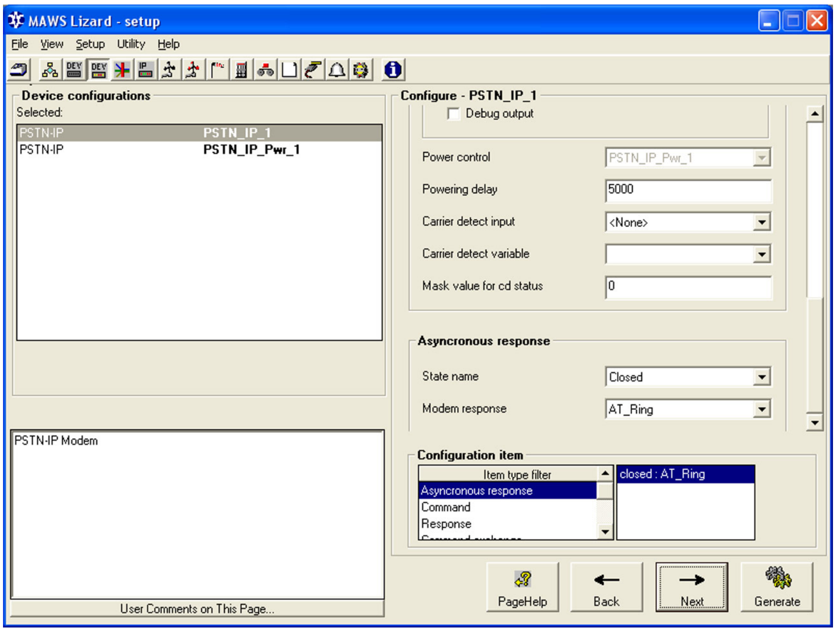

Figure 91 PSTN-IP Modem Configuration

There are no station-specific parameters associated with the PSTN modem.

Communication Interfaces

As the same physical device can provide connection to different network gateways and/or services, configurable logical communication interfaces are included in Lizard Setup Software. For example, the same GSM modem can provide different connections and have different connection-dependent parameters for GPRS as the primary connection and data call (CSD) as the backup connection.

Interface Configuration

The configuration view **Communication interfaces**, shown in Figure 92 on page 211, contains functions for adding, deleting, and editing interfaces.

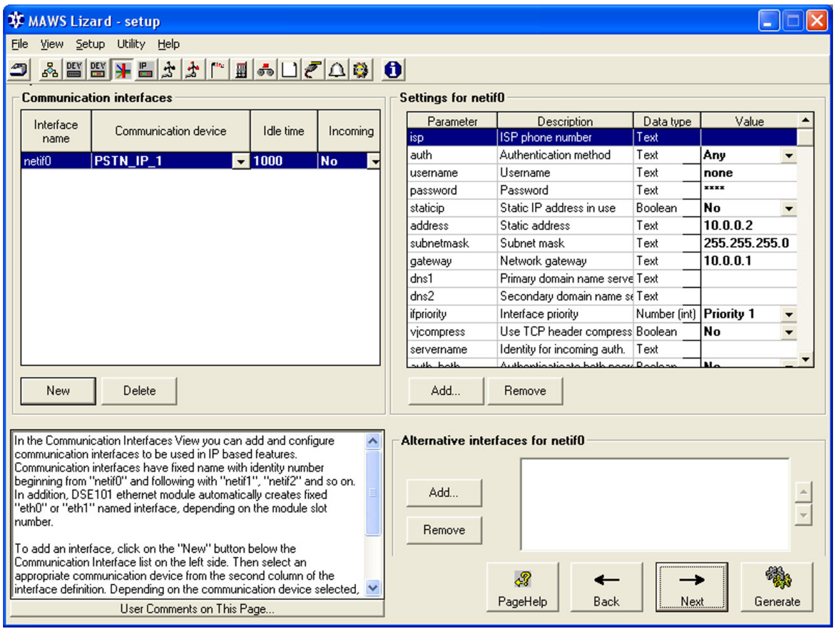

Figure 92 Interface Configuration

NOTE	Adding Ethernet Communication Module DSE101 to the setup automatically creates a communication interface with a name in the format **ethX**, where X is 0 or 1, depending on the module place where DSE101 is installed.

To add a communication interface, proceed as follows:

1. In the **Communications interfaces** view, click **New**. The naming for logical interfaces is fixed so that each new interface will get a name in the format **netifX**, where X is an automatically increasing interface index.
2. Select the physical communication device used by the interface.
3. Set the idle time for the device as needed by entering the value in the **Idle time** field. Idle time defines the time, in milliseconds, for which the interface is kept open when it is not being used. For example, if opening the interface is a time consuming or costly operation, it is advisable to keep it open for some time, so that all transmitted information can pass during one session.
4. Select if you want the QML logger to accept incoming connections from this interface. For interfaces using the devices **Siemens GPRS-IP** and **PSTN-IP**, you can select whether the incoming connection uses a TCP/IP-based connection or not: for TCP/IP, select the option **Yes** from the list; for connections without TCP/IP (such as data calls), select the option **Yes, without PPP**. If you select the latter option, you can use data calls, for example, for the service connection to a GPRS interface, even though the data transmission on the interface uses TCP/IP.

NOTE	In order to accept an incoming connection through a dial-up device, such as a modem, you need to enable incoming call answering also for the modem. See section Common Modem Parameters on page 206.

Station-specific settings for communication interfaces are stored in parameter sets with the same names as the communication interfaces. Setup defaults are set from the top right configuration frame named **Settings for netifX / ethX**.

To edit the parameter defaults, first select the interface from the **Communication interfaces** list. This brings up parameters for the selected interface. To delete a communication interface, select the interface to be deleted and then click **Delete**.

Table 28 on page 213 lists all station-specific parameters for the communication interfaces. The required settings depend on both the selected communication device and the network configuration. Lizard Setup Software contains an appropriate default set corresponding to the selected interface.

Table 28 Station-Specific Parameters for Communication Interfaces

Parameter	Use	Example or Allowed Values
apn	Access point name	internet, vaisala.fi
auth	Authentication type	**none** = no authentication; **any** = PAP or CHAP; **pap** = PAP; **chap** = CHAP
auth_both	Authenticate both peers. In addition to the client (AWS) authenticating to the server, the server must authenticate to the client. If the parameter value is set to **yes**, authentication credentials must be added to the parameter group **pppclients**. For further information, see Authentication on page 217.f	Yes = true
username	Outgoing username	registereduser
password	Outgoing password	mysecret
servername	Identity when acting as server. The default value is the station name.	MAWS, AWS
isp	ISP number to dial	GPRS connection: *99***1# Dial up: 01234567
staticip	Static IP address in use	Yes = true
address	IP address if static	10.0.0.1[1]
subnetmask	Network mask	255.255.255.0[1]
gateway	Gateway if static	10.0.0.2[1]
dns1	Domain name server 1. It is recommended that you leave this setting blank unless you use a static IP address or do not obtain a DNS server from your network in DHCP or PPP negotiation.	10.10.10.10
dns2	Domain name server 2. It is recommended that you leave this setting blank unless you use a static IP address or do not obtain a DNS server from your network in DHCP or PPP negotiation.	10.10.10.11
ifpriority	Usage priority for interface. Defines the routing preference for packets: the open interface with the highest priority will be used.	1 ... 9; 1 = lowest priority, 9 = highest priority
vjcompress	Use IP header compression	Yes = true
autoinit	Automatic DSE101 startup	Yes = true
autoinitinterval	Interval (in milliseconds) after which automatic startup of the interface is retried if it initially fails.	5000

1. Used only if the staticip option is selected.

For more information on configuring interfaces, see sections IP Settings on page 216 and Authentication on page 217.

Station-specific values for these parameters can be set using the **paramset** commands described in section Commands for Parameter Sets on page 293.

Alternative Interfaces

Alternative interfaces are interfaces that provide similar networking functionality, but use different routes and/or media. For example, if **netif1** is marked as alternate to **netif0** it means that:

- If a service is trying open connection through **netif0** and **netif1** is already open, **netif1** will be used instead.
- If **netif1** is requested for use, **netif0** will not be used as its alternate.

Using alternative interfaces is optional, and should not be mixed with primary/secondary interfaces associated with TCP/IP-based services, which are used to provide backup mechanisms for services.

To add an alternative for an interface, proceed as follows:

1. From **Communication interfaces**, select the interface you want to add alternatives for.
2. Click **Add** in the **Alternative Interfaces** frame. The **Add alternative interfaces** dialog shown in Figure 93 on page 215 is displayed.

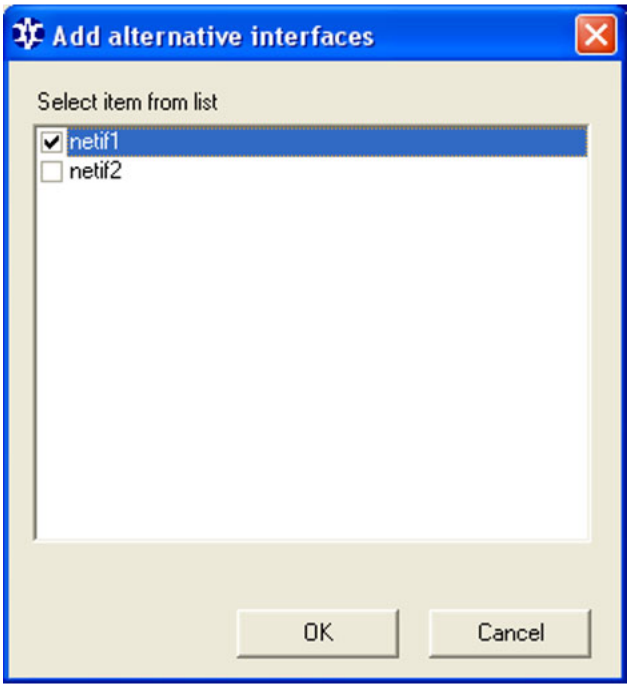

Figure 93 Adding Alternative Interfaces

3. Select the interface(s) you want to use as alternate.
4. Click **OK** to close the dialog.

To remove an alternate interface, select the item you want to remove from the **Alternative Interfaces** list and click **Remove**. You can also change the order of the alternative interfaces using the arrows on the right of the **Alternative interfaces** frame.

IP Settings

As IP settings depend greatly on the network environment, the aim of this document is not to provide comprehensive instructions on network design. The following describes a few basic rules and practices.

Static and Dynamic IP Addresses

- LAN environments typically provide private dynamic addressing using DHCP, which is supported by the QML logger when using Ethernet Communication Module DSE101. It is possible to use static addressing in LAN as well, but then the addresses have to be provided by the organization operating the LAN.
- Without special operator agreement, public networks such as telephone, cellular, or broadband do not usually provide static IP addresses to connected client devices. This means that clients, such as the QML logger, cannot be directly reached using IP. Dial-up IP networking can be used when applicable.
- Broadband links, such as ADSL or satellite broadband, are mostly LAN connections from the QML logger point of view, as in this case it is usually connected to the communication device using Ethernet. Depending on communication device configuration and operator requirements, the operation mode can be:
    - Bridged, that is, the communication device performs media conversion between Ethernet and broadband. IP addressing, domain name, and other services come directly from operator's network.
    - Routed, that is, the communication device acts as a router between Ethernet and broadband connection. Router usually provides IP addressing to connected clients. Often NAT (Network Address Translation) is also used, making the client devices connected to Ethernet inaccessible from broadband, unless configured to the router.

When connecting the QML logger directly to another computer using, for example, null modem, it is possible to use dynamic or static addresses. In each case, the settings on both ends have to match.

If a static IP address is used, all other address settings need to be set manually as well. With dynamic addressing, only the domain name server addresses can be set manually, if not provided by DHCP or PPP negotiation.

IP header compression is an algorithm which reduces the amount of redundant header data. Using it is recommended whenever operating over a RS-232 line, that is, with modem and null modem connections.

Authentication

QML supports two authentication methods: Password Authentication Protocol (PAP) and Challenge Authentication Protocol (CHAP). These can be used with modem and null modem connections.

PAP is very simple protocol where user credentials are transmitted in plain text, making it insecure. CHAP is more secure as it does not transmit user credentials over the link. It is the preferred authentication option to use with the QML logger.

NOTE	Microsoft has its own version of CHAP called MS-CHAP, which is not supported by QML. PAP is currently the only possibility for authenticating the QML logger when connecting to a Windows PC. When connecting from Windows PC to QML, also CHAP can be used.

Outgoing user credentials are stored in parameter set of the communication interface used, for example, parameter set **netifX**:

S:00000001: netif0:username = pppuser
S:00000001: netif0:password = userppp

For incoming authentication, there is a separate parameter set reserved, named **pppclients**. For example, below are listed the user credentials for users router1 and router2:

S:00000001: pppclients:router1:password = userppp
S:00000001: pppclients:router2:password = r2pass

Authentication can also be configured to be required both ways. For example, when the QML logger is authenticating to remote computer, it can simultaneously request the remote computer to authenticate itself to QML. This option is set on by the communication interface parameter **auth_both**. If bidirectional authentication is required, you need to set outgoing credentials for the communication interface used and incoming credentials in the **pppclients** parameter set.

Authentication for Incoming Connections

Authentication credentials for incoming users are configured using the parameter set **pppclients**. The parameter set contains the user name and password combinations to be supplied to the QML logger when users connect to it. To configure incoming authentication, proceed as follows:

1. On the **Setup** menu, select **Parameter Sets**.
2. Select **Current parameter set owners** from the **Category** menu.
3. Select **pppclients** as the parameter to configure and click **Add** to add a new parameter.
4. Enter the user ID and password to be used for the incoming connection in the **Parameter** field. The format to use is <user ID:password>, for example, *user1:password*. See Figure 94 on page 218.

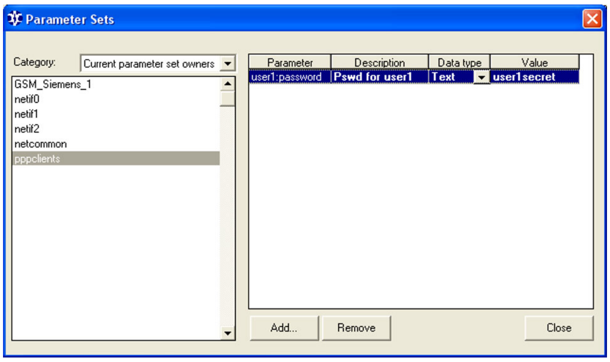

Figure 94 User Credentials for Incoming Connections

5. If required, enter a description for the parameter in the **Description** field.
6. In the **Type** field, select **Text** as the parameter type.
7. Enter the user's password in the **Value** field.
8. Close the **Parameter Sets** view by clicking **Close**.

The users connecting to the QML logger can use the credentials configured in the pppclients parameters: their user ID is the first part of the parameter name, and their password is the value of the **username:password** parameter. You can also add more user–password combinations in the same manner.

NOTE	Users can connect to the QML logger using any interface configured to accept incoming connections. The users need to supply the credentials configured in the **pppclients** parameter set. It is not necessary to configure the incoming credentials separately for each interface.

Parameter Set hosts

Managing hosts

Parameter set *hosts* groups general information needed for accessing remote servers into single location in the logger. The information contained in parameter set *hosts* includes:

- User credentials
- Proxy settings
- Backup servers and interfaces

NOTE	Storage does not provide encryption for the stored user credentials. Externally secured remote access and possibly logger userlevels are recommended to protect the information from unauthorized access.

Parameter set is managed using Lizard Setup Software for creating the parameters and providing default settings and AWS Client for making station-specific adjustments.

Storage paths for parameters in the set are defined as:

<hostname>:<protocol>:<parametername> <value>

or

<hostname>:<protocol>:<interface><parametername> <value>

To create and update the default settings for *hosts*, in Lizard, open the **Setup** menu and choose **Parameter Sets**. The dialog shown in Figure 95 on page 220 opens.

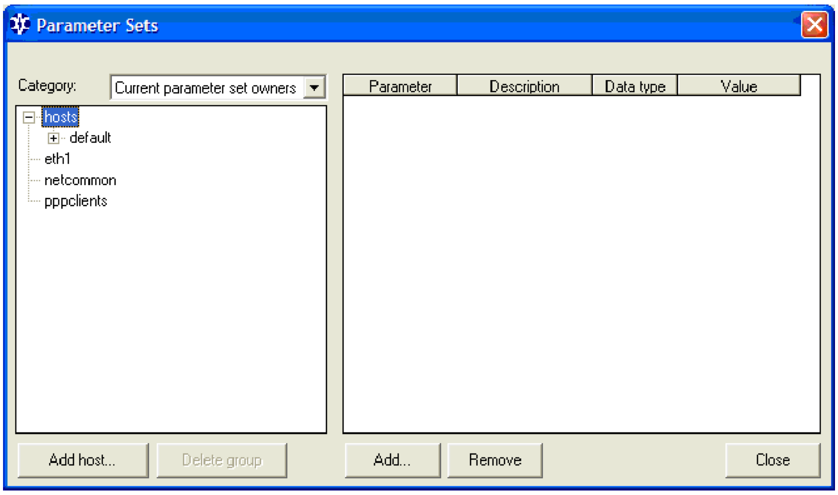

Figure 95 Configuring Parameter Set hosts in Lizard

All remote hosts that require having settings stored in the logger are listed under setname *hosts*.

Settings from hostname *default* are used if parameters for the given host are not found. For example, as typically all HTTP traffic passing through particular interface uses the same proxy server, this server can be defined under *default* instead of defining it for each accessed host.

To add a new host to the parameter set, proceed as follows:

1. In the **Category** window, select *hosts*, and click **Add host**. The dialog shown in Figure 96 on page 221 appears.

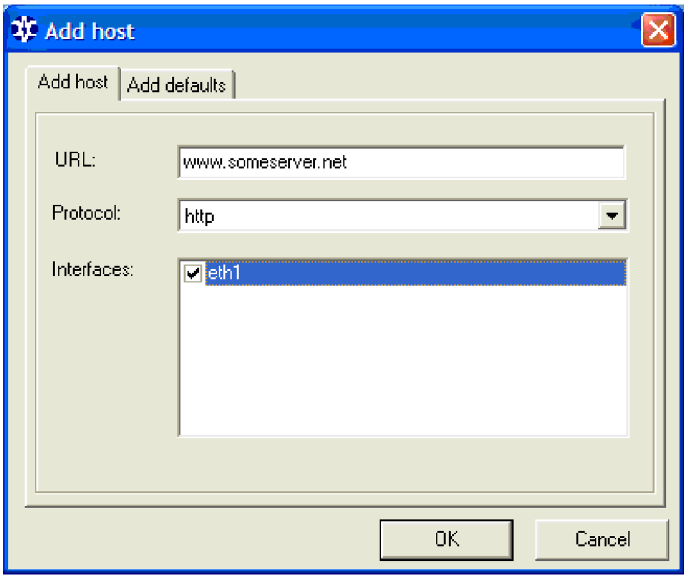

Figure 96 Adding New Host to hosts

2. Enter hostname, select protocol, and select interfaces, if applicable. Interface selection is needed only for HTPP proxy settings. Click **OK** when done. Now the settings dialog looks as shown Figure 97 on page 221.

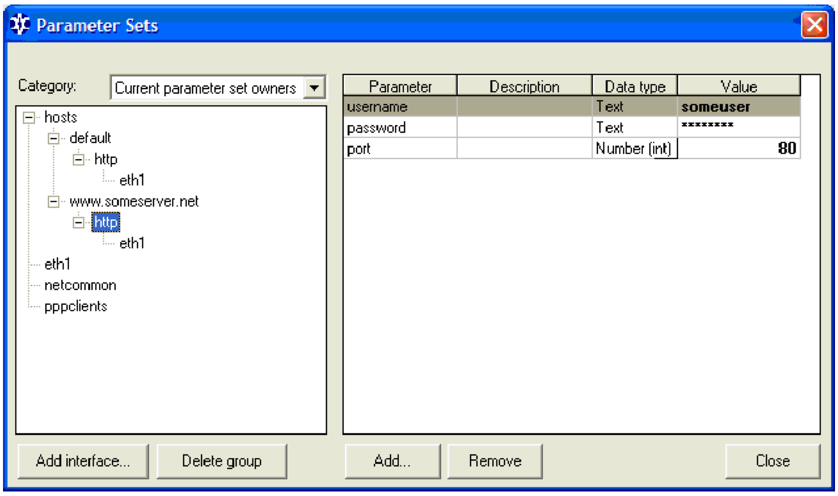

Figure 97 Configuring Settings for a Host

3. Enter username, password, and service port, as required.

4. If proxy configuration is required (http only), open interface(s) under the protocol. Enter proxy name/address and its TCP port. If *default* proxy configuration for interface is applicable, there is no need to define the proxy here, and it is recommended that you use the *default* settings instead; see Figure 98 on page 222.

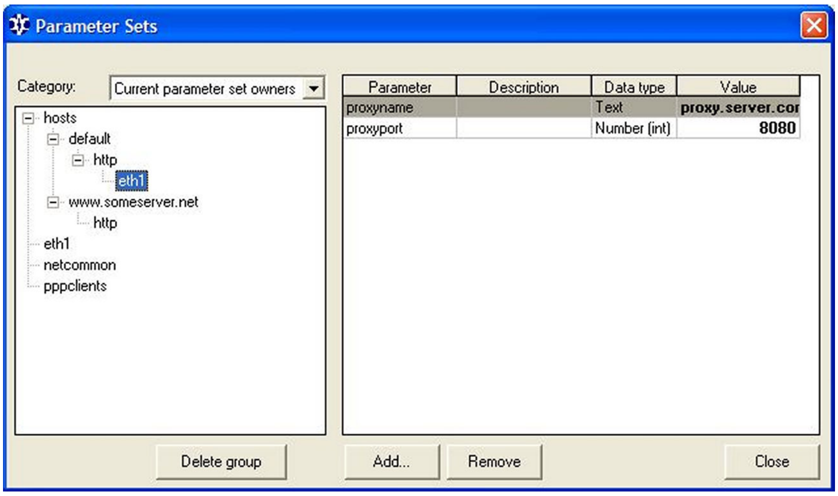

Figure 98 Configuring Default HTTP Proxy for Interface

Default settings are available only for defining proxy settings for network interfaces.

All station-specific configuration regarding settings in parameter set *hosts* is managed using AWS Client.

To access parameters in *hosts* using AWS Client, select **Settings - Parameters - Static** and open the hosts set. Refer to AWS Client documentation for instructions on how to update the values.

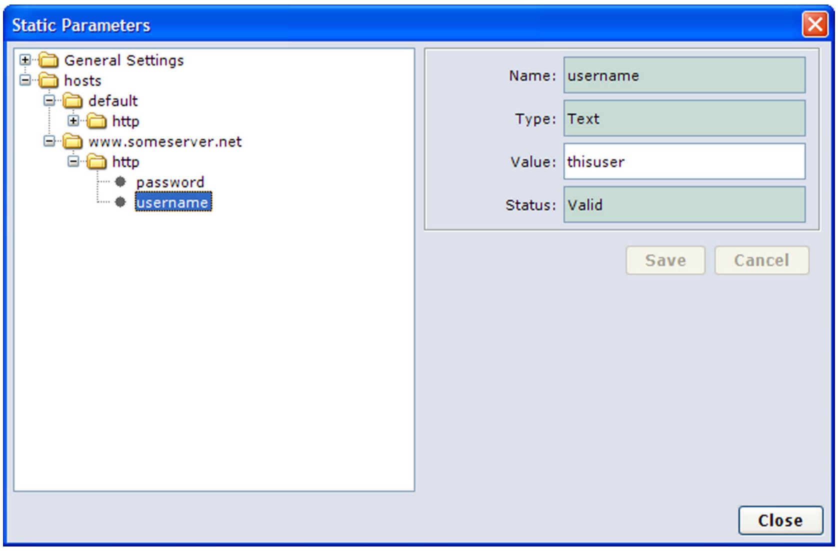

Figure 99 Updating Parameter Set hosts with AWS Client

Backup Servers

Parameter set *hosts* also provides backup definitions for application level IP protocol clients implemented in the logger.

To define a backup for a host in Lizard, proceed as follows:

1. In the **Setup** menu, open **Parameter Sets**.
2. Select the protocol under the server for which the backup is added.
3. Add the following parameters using the **Add** button:

Table 29 Parameters for Backup Host

Parameter Name	Data Type	Use
backuphostNo[1]	Text	Name or IP address of the backup server.
backupportNo	Number (int)	Port for protocol in the backup server.
backupifNo	Text	Interface to use for accessing the backup server.

1. No = Running number. 1 ... 5 indicate the backup sequence number. There may be up to five backups defined.

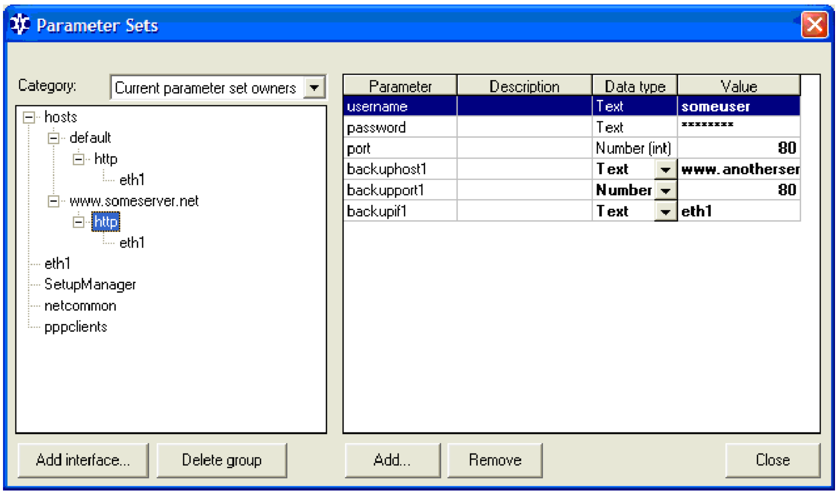

Figure 100 HTTP Host with One Backup Defined

Command hostset

Logger provides shell command **hostset** to manage parameter set *hosts*. The syntax of the command is:

hostset *{[<hostname>]{:[<protocol>]:[<param> | <netif>:<proxyset>] } {<value>}}*

where

<hostname>	=	Name of the host or *default*
<protocol>	=	*http*, *ftp*, *ntp*, or *smtp*
<param>	=	username password port backuphostNo backupportNo backupifNo
<netif>:<proxyset>	=	Network interface name followed by *proxyname* or *proxyport*

Examples of **hostset** command:

/ > hostset

```
ftp.vaisala.com
ntp.vaisala.com
smtp.dna044.com
smtp.vaisala.com
```

/ > hostset ftp.vaisala.com

```
S:00000001: ftp.vaisala.com:ftp:backuphost1 =
ftp.nosuch.com
S:00000011: ftp.vaisala.com:ftp:backupif1 = netif0
I:00000011: ftp.vaisala.com:ftp:backupport1 = 21
S:00000001: ftp.vaisala.com:ftp:password = wontpass
I:00000011: ftp.vaisala.com:ftp:port = 21
S:00000001: ftp.vaisala.com:ftp:username = thisuser
```

/ > hostset ftp.vaisala.com:ftp:username newuser

/ > hostset www.vaisala.com:http:eth1:proxyname myproxy

/ > hostset www.vaisala.com:http:eth1:proxyport 8080

NOTE	Host *default* is reserved name, and its values are used when no host-specific settings are present.

TCP/IP-Based Services

This section describes the TCP/IP-based services used in Lizard Setup Software.

FTP Client

The FTP client functionality enables the QML logger to send reports and log files to an FTP server. File retrieval is currently implemented only for the **ftp** command available in the QML logger command shell accessible via a terminal connection.

A buffering mechanism is used for outgoing data, so in case of network failure, data will be temporarily stored by the QML logger, and sent when the communications link is restored.

General Configuration

General FTP client parameters are configured in the **IP Services** view as shown in Figure 101 on page 227. Report and log file -specific parameters are configured in their respective views as described in sections Sending Reports on page 230 and Sending Log Files on page 233.

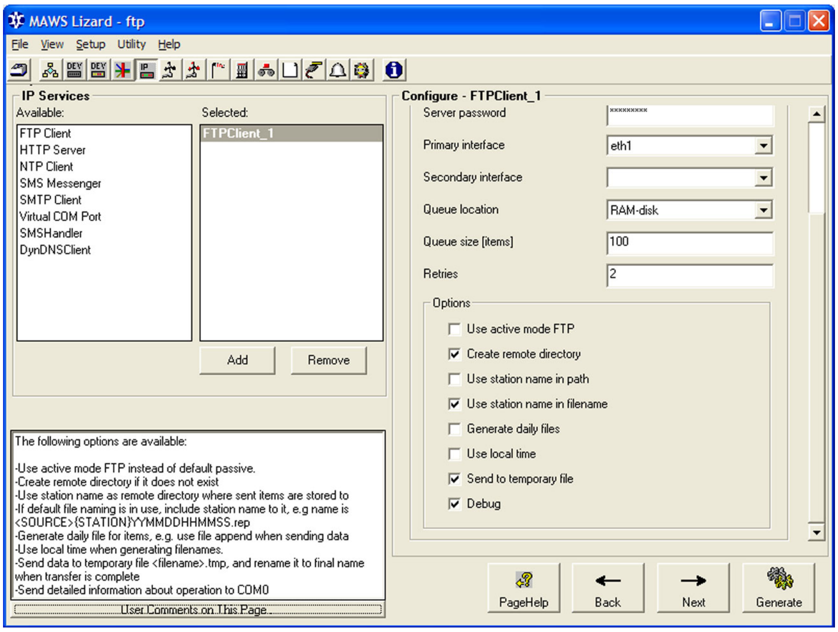

Figure 101 FTP Client Configuration

The FTP client configurable parameters are presented in Table 30 on page 227.

Table 30 FTP Client Parameters

Parameter	Use	Note
Destination server	FTP server to which data will be sent	
Server username	Username for the FTP server	
Server password	Password for the FTP server	The password is displayed as asterisks in Lizard Setup Software.
Primary interface	Primary communication interface used for connecting to the server	
Secondary interface	Secondary communication interface used for connecting to the server. This will be used in case of connection failure in the primary one.	
Queue location	File system location where outgoing reports will be queued for transmission. The available locations are: **RAM-disk**, **External CFS**,[1] [2] and **Internal logfile system**.[2]	Specify the same location for all FTP clients in the setup.

Table 30 FTP Client Parameters (Continued)

Parameter	Use	Note
Queue size	Number of items, for example, reports stored in the queue. If this limit is exceeded due to, for example, network congestion, the oldest queued item will be deleted for each new one.	The QML logger can buffer data within the limitations of the selected storage location and certain file system constraints. An absolute maximum of 500 items is in effect. Buffering data in **RAM-disk** will consume dynamic memory from rest of the application; similarly, the other storage alternatives will consume logging space.
Options	- **Use active mode FTP** = see RFC959 - **Create remote directory** = create specified remote directory if it does not exist - **Use station name in path** = use station name for path - **Use station name in filename** = report is formatted as follows: <Report_Name><Station_Name>YYMMDDhhmmss.rep -or- <Report_Name><Station_Name>YYMMDD.rep See section Sending Reports on page 230. - **Generate daily files** = append reports to daily files - **Debug** = print debug output to COM0 - **Use local time** = use local time instead of UTC time when creating daily files	For the file naming conventions, see section Sending Reports on page 230.

1. Requires a CompactFlash card.
2. Non-volatile.

If needed, it is possible to provide station-specific settings for the FTP client parameters, that is, settings that override the setup values. This is done by creating a new parameter set with the same name as the FTP client in the setup.

NOTE	If a station-specific setting for an FTP parameter is created, it must have a proper default value, or the value must be entered during commissioning. Any value, including empty text, will override the value that the parameter has received from the setup.

To create station-specific settings for an FTP parameter, proceed as follows:

1. In Lizard, select **Parameter sets** from the **Setup** menu.
2. From the **Category** list, select **IP Services**.
3. Select the FTP client to which you want to add parameter(s).
4. Click **Add**, then enter a parameter name from the table below.
5. Select the data type and enter the default value.

Table 31 Station-Specific Parameters for FTP Client

Parameter name	Use[1]	Data Type	Example Value
server	FTP server to use	Text	ftp.vaisala.com
user	Username for the FTP server	Text	user
pass	Password for the FTP server	Text	wontpass
primif	Primary interface	Text	eth1
secif	Secondary interface	Text	netif0
pipe	Queue location	Text	/Ram
pipesize	Queue size (items)	Number (int)	50
retries	Retries	Number (int)	3

1. For more information, see table FTP Client Parameters.

Station-specific values for these parameters are set using AWS Client.

To create station specific-parameters for an FTP client, see the following sections.

Sending Reports

As with all other communication configuration for report transmission, transmission using the FTP client is configured, for each report, in the **Communications** view as shown in Figure 102 on page 230.

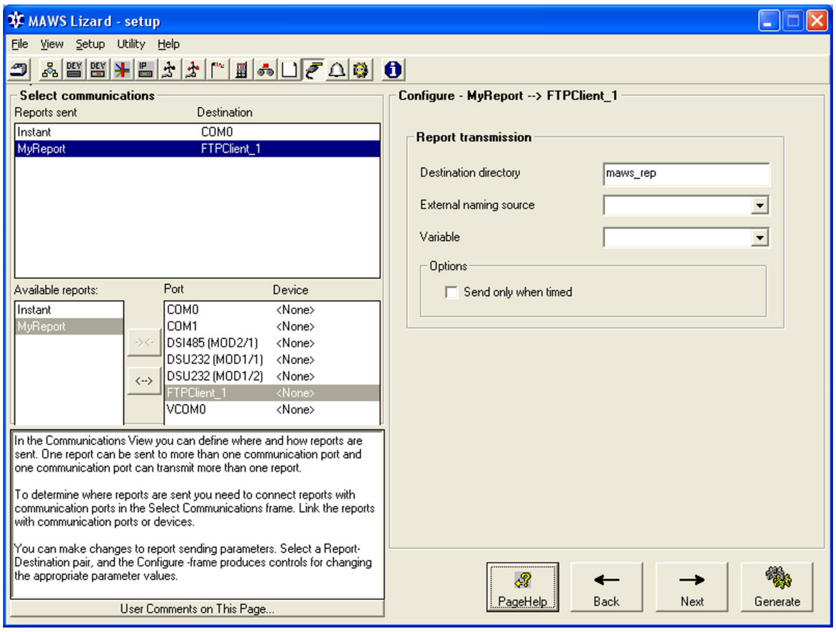

Figure 102 Report Configuration for FTP Transmission

The report transmission parameters are described in Table 32 on page 230.

Table 32 Parameters for Sending Reports via FTP

Parameter	Use	Note
Destination directory	Fixed FTP server path under the given user's home directory to which reports will be placed.	
External naming source	Optional external source for the remote file name. This can be used if the built-in naming convention is not suitable	Reporting can be used to generate file names.
Variable	Variable in the external source.	Usually **rep**.
Options/send only when timed	Do not send the report immediately when created, but when triggered by the **File transfer** timer. This timer needs to be manually enabled in the **Timers** view.	

The destination file naming rules for transmitted reports are as follows:

1. If an external naming source is selected from the **External naming source** list, the path and file name provided by it are used as is. If it is not given, the rules below apply.
2. The destination directory is the first part of the path. It may be empty, that is, no fixed path exists.
3. FTP client uses the built-in file naming rules to create rest of the path and the file name:

   a. If the option **Use Station Name in Path** is selected in the FTP client configuration; the station name will be added as the next path element.
   b. If the option **Generate daily files** is selected in the FTP client configuration in the **IP Services** view, reports for one day are appended into a single file. Its name is created from the report name and current timestamp. UTC or local time is used as specified by the option **Use local time**. The file name follows the format:
   <Report_Name>YYMMDD.rep
   where YY = year, MM = month, and DD = date.
   c. If the option **Generate daily files** is not selected, each report will be placed in its own file named in a similar manner as the daily file, but with hours, minutes, and seconds appended to the file name:
   <Report_Name>YYMMDDhhmmss.rep,
   where hh = Hours, mm = Minutes, ss = Seconds.

d. If the option **Use station name in filename** is selected, the file name is formatted as follows:

<Report_Name><Station_Name>YYMMDD.rep
<Report_Name><Station_Name>YYMMDDhhmmss.rep

where

<Report_Name>	=	Name of the report
<Station_Name>	=	Set by station parameter sname
YY	=	Year
MM	=	Month
DD	=	Day
Hh	=	Hour
Mm	=	Minute
Ss	=	Second

Example:

When the name of the sent report is *obs* and the station name is *maws*, target file name is *obsmaws081210160000.rep*.

The following special content convention allows any file from the QML logger to be sent to the FTP server.

If an item, for example, a report, linked to the FTP client for transmission starts with the schema **file://**, the name following the schema will be extracted and the file referenced by it will be sent to the FTP server.

For example, a report with the content **file:///Log/L1070523.dat** will trigger transmission of the referenced log file each time the report is generated.

Sending Log Files

FTP settings for log file transmission are configured in the **Logging** view as shown in Figure 103 on page 233. For each log group, there is an FTP tab where transmission parameters can be set.

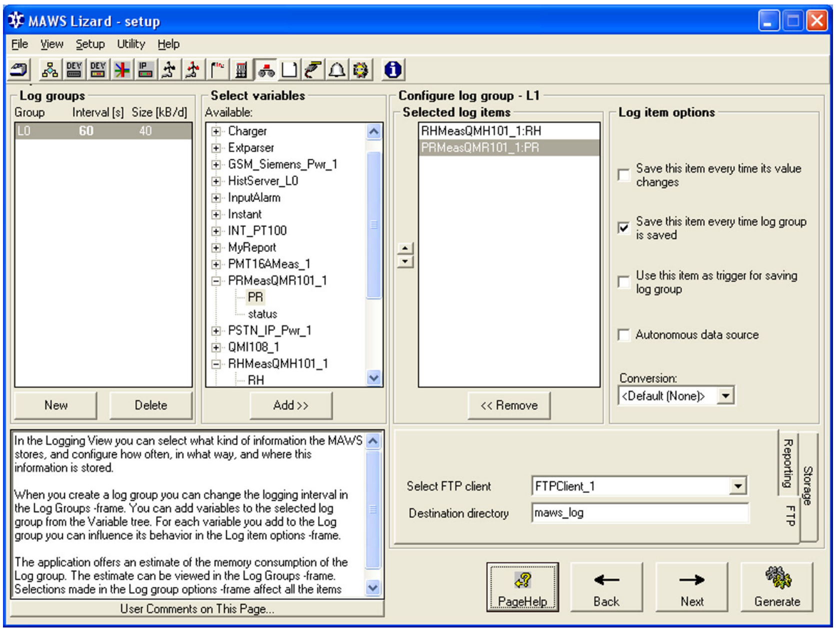

Figure 103 Log File Configuration for FTP Transmission

Log file transmission parameters are listed in Table 33 on page 233.

Table 33 FTP Transmission Parameters for Log Files

Parameter	Use
FTP client	FTP client used for sending the log files. Usually, it is sufficient to have one FTP client in the setup for transmitting log files.
Destination directory	Fixed path to which files are stored. This will be automatically followed by the station name.

Log files' naming is preserved when sent to the FTP server. By default, the log files are sent two minutes after midnight.

Automatic and Timed Transmission

By default, reports are sent immediately when created, and multiple reports per one FTP session will be sent only when they are created at the same time.

If it is needed to buffer reports to, for example, reduce communication costs or to preserve power, sending multiple reports during one FTP session can be done with a timer. A file transfer timer for FTP client is available, but it is disabled by default. To enable this mode of operation, proceed as follows:

1. In the **Communications** view, select the option **Send only when timed** for the report(s) .
2. In the **Timers** view, set the required transmission interval for FTP client's **File transfer** timer.

NTP Client

The QML logger NTP client functionality provides a means for synchronizing the QML real-time clock with an external time server. There are public timeserver pools for this purpose, or, alternatively, the observation network can provide an internal NTP service. NTP clients are configured in the **IP Services** view as shown in Figure 104 on page 235.

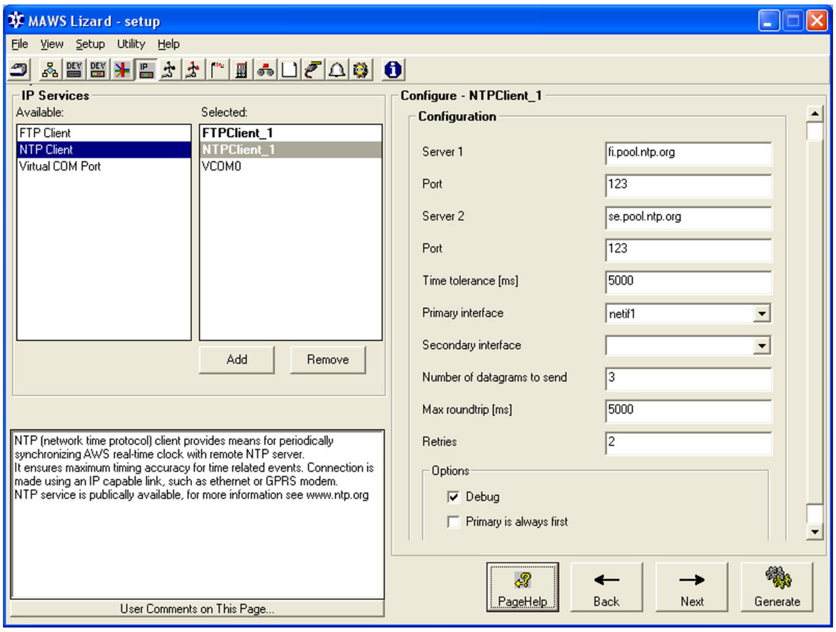

Figure 104 NTP Client Configuration

Parameters for NTP client are listed in Table 34 on page 236.

Table 34 NTP Client Parameters

Parameter	Use
Server 1	First NTP server to acquire time from.
Server 2	Optional backup NTP server to acquire time from.
Time tolerance	Maximum allowed time error in milliseconds. If the difference between server time and logger clock exceeds this value, the logger clock will be adjusted to match.
Primary interface	Primary network interface for contacting the server.
Secondary interface	Optional secondary network interface for contacting the server.
Number of datagrams to send	Number of datagrams to send for averaging the time error.
Max roundtrip	Maximum allowed roundtrip in milliseconds, that is, the time taken from request transmission to response reception. Datagrams with longer roundtrip will be ignored.
Retries	Number of retries in case of failure when contacting the server.
Options	**Debug** = List debug output to COM0; **Primary always first** = Always request time first from the primary server. If this option is not set, the QML logger will keep using the same server as for the previous successful time request.

If needed, it is possible to provide station-specific settings for the NTP client parameters, that is, settings that override the setup values. This is done by creating a new parameter set with the same name as the NTP client in the setup.

NOTE	If a station-specific setting for an NTP parameter is created, it must have a proper default value, or the value must be entered during commissioning. Any value, including empty text, will override the value that the parameter has received from the setup.

To create station-specific parameters for an NTP client:

1. In Lizard, select **Parameter sets** from the **Setup** menu.
2. From the **Category** list, select **IP Services**.
3. Select the NTP client to which you want to add parameter(s).
4. Click **Add**, then enter a parameter name from the table below.
5. Select the data type and enter the default value.

Table 35 Station-Specific Parameters for NTP Client

Parameter name	Use[1]	Data Type	Example Value
server1	Primary NTP server	Text	ntp.vaisala.com
server2	Secondary NTP server	Text	fi.pool.ntp.org
timetol	Time error tolerance [ms]	Number (int)	5000
primif	Primary interface	Text	eth1
secif	Secondary interface	Text	netif0
datagrams	Number of datagrams	Number (int)	3
retries	Retries	Number (int)	1
timeout	Timeout/max roundtrip [ms]	Number (int)	5000

1. For more information, see table NTP Client Parameters.

Station-specific values for these parameters are set using AWS Client.

E-Mail (SMTP) Client

The QML logger e-mail client functionality provides a means for sending QML logger reports via e-mail using the SMTP protocol. Using the SMTP client functionality requires a TCP/IP based connection to an SMTP server. Typical use is sending alarm messages triggered by, for example, observation value limits, door switch, or battery charge levels. A buffering mechanism is used for outgoing data, so in case of a network failure, data will be temporarily stored by the QML logger, and sent when communications are restored. SMTP clients are configured in the **IP Services** view as shown in Figure 105 on page 238.

NOTE	The SMTP client does not support authentication extensions. It can typically be used only on closed networks, or networks where user identification is externally provided by means such as a SIM card or device identification on cellular networks.

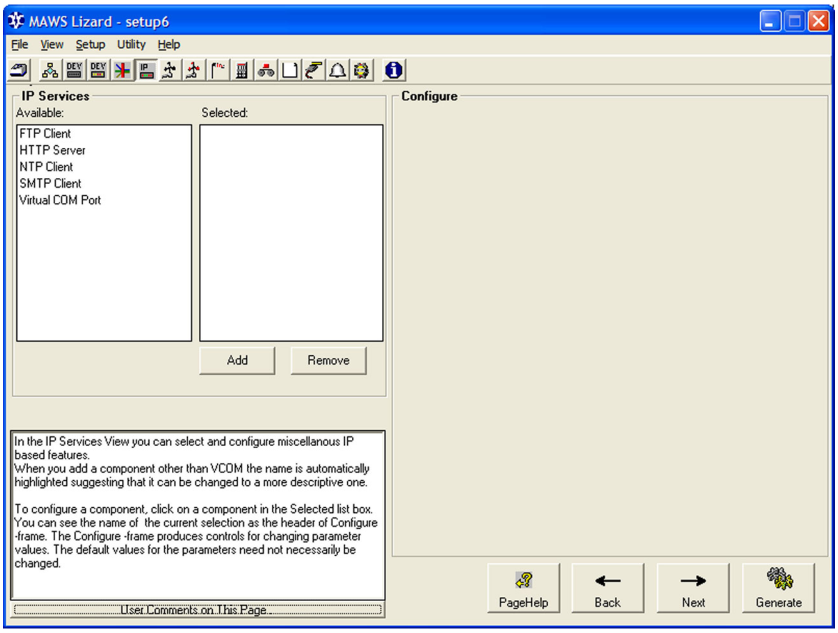

Figure 105 IP Services

General parameters for SMTP clients are configured in the **IP Services** view as shown in Figure 106 on page 239. Message-specific parameters are configured in the **Communications** view.

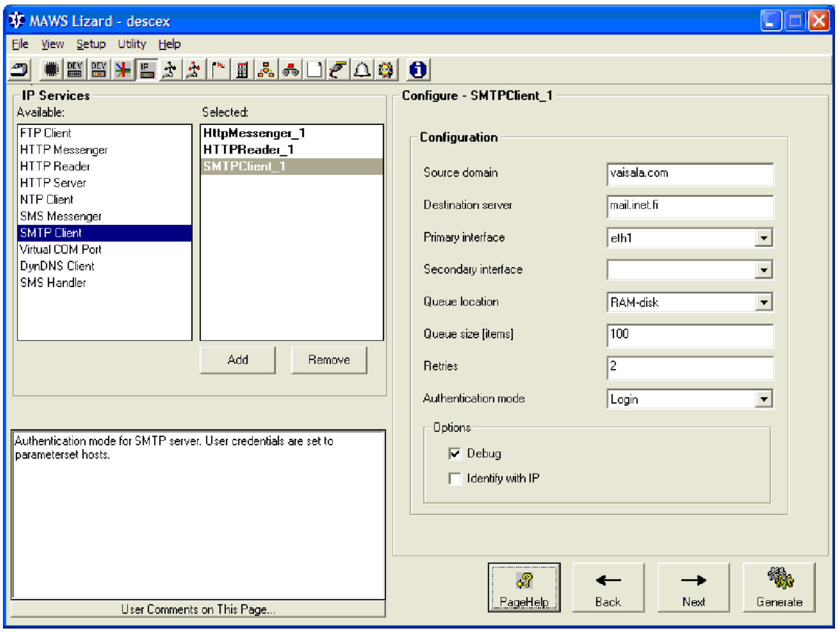

Figure 106 SMTP Client Configuration

The parameters to be configured for the SMTP client are presented in Table 36 on page 240.

Table 36 Parameters for SMTP Client

Parameter	Use	Note
Source domain	Source domain where messages are sent from. If the field is left empty, the QML logger interface IP address will be used as default. This can also be used as part of message sender identity; see Table 37 on page 243.	Operators usually require the source domain to be an existing one.
Destination server	SMTP server to use for sending messages.	
Primary interface	Primary network interface for contacting the server.	
Secondary interface	Optional secondary network interface for contacting the server.	
Queue location	File system location where outgoing messages will be queued for transmission. The available locations are: **RAM-disk**, **External CFS**,[1] [2] and **Internal logfile system**.[2]	Specify the same location for all SMTP clients in the setup.
Queue size (items)	Number of items, for example, reports stored in the queue. If this limit is exceeded due to, for example, network congestion, the oldest queued item will be deleted for each new one.	QML can buffer data within limitations of the selected storage location and certain file system constraints. An absolute maximum of 500 items is in effect. Buffering data in **RAM-disk** will consume dynamic memory from rest of the application; similarly, the other storage alternatives will consume logging space.
Retries	Number of retries if the SMTP server cannot initially be contacted	
Authentication mode	Selection for SMTP authentication.	For further details, see section User Credentials for Authentication on page 241.
Options	**Debug** = print debug output to COM0	

1. Requires a CompactFlash card.
2. Non-volatile.

User Credentials for Authentication

Parameter set *hosts* is used to manage user credentials for SMTP authentication. Credentials are stored as follows:

<servername>:smtp:username <username>

<servername>:smtp:password <password>

where

<servername>	=	Name of the server
<username>	=	Assigned username
<password>	=	Assigned password

For more information, see section Parameter Set hosts on page 219.

If using authentication mode PLAIN, an additional credential **Authorization identity**, may be required. By default, this is the same as username, and no separate setting is necessary. If needed, authorization identity is created in the configuration dialog by clicking the **Add** button and entering the following attributes for the parameter:

- Name: authid
- Data type: text
- Value: authorization identity

NOTE	Even if these authentication modes use encoding for sending the user credentials, this encoding can be easily translated to plain text. No encrypting for the mail transaction is provided by QML201.

NOTE	Any station-specific settings to the authentication selection or user credentials become effective only after logger restart.

Sending E-Mail Messages

The e-mail messages to be sent are constructed as reports in the **Reports** view like any other report. The transmission options for the SMTP client are configured, on a per-report basis, in the **Communications** view as shown in Figure 107 on page 242.

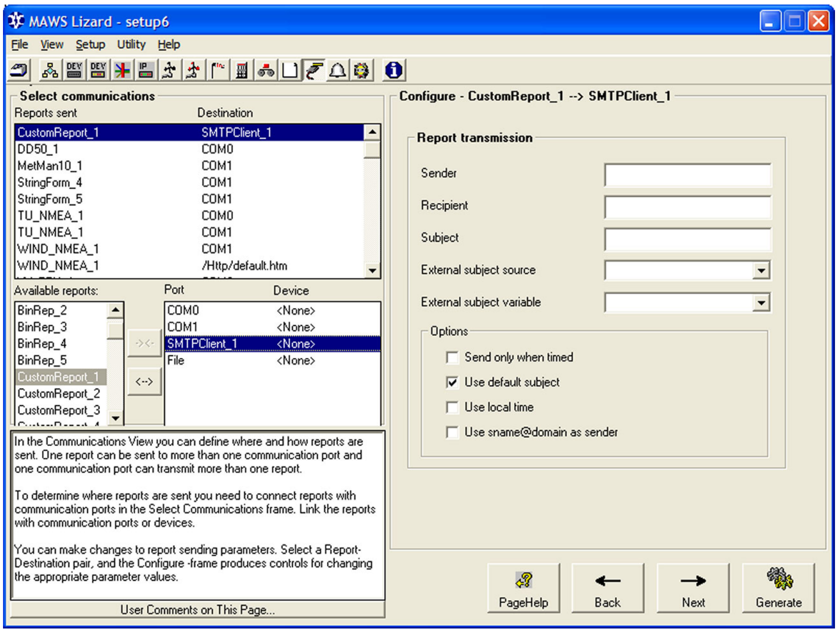

Figure 107 Report-Specific Transmission Options for SMTP Client

The parameters for messages transmitted using the SMTP client are listed in Table 37 on page 243.

Table 37 Parameters for Sending Reports via E-Mail

Parameter	Use
Sender	Fixed message sender identity. Alternatively, option **Use sname@domain as sender** can be used. See later in the chapter for naming formats
Recipient	E-mail address for message recipient
Subject	Message subject. Alternatively, an external source for subject or option Use default subject can be used. See immediately below this table for naming formats
External subject source/ Variable	Optionally message subject can be generated by external source, typically by a report. Enter here the source and variable in the source.
Options: Send only when timed	Do not send the message immediately when created, but when triggered by 'E-mail transmit' timer. This timer needs to be manually enabled from the timers view. See section Automatic and Timed Transmission of E-Mail Messages on page 244.
Options/Use default subject	Use built-in default for message subject. See naming formats after this table.
Options/Use local time	Use local time when generating timestamp for the default subject.
Options/Use sname@domain as sender	If the source domain is set in the generic SMTP configuration, it can be used with station name to construct sender identity. See naming formats after this table.

Built-in defaults for the field formats are composed as follows:

1. Use default subject:
   - The message subject follows the format <*source name*> <*station name*> YYMMDDhhmmss where
     - *Source name* is the name of the message content source, typically a report
     - *Station name* is the name of the weather station
     - YYMMDDhhmmss is a time stamp
2. Use sname@domain as sender:
   - The first part of the sender identity is the weather station name, followed by the @ sign and the source domain as set in the general SMTP client configuration. For example: maws1@vaisala.com
   - When this option is selected, the other parameters must be configured as follows:
     - The **Source domain** parameter must be configured in the general SMTP client configuration in the **IP Services** view
     - The **Sender** parameter can be left empty in the e-mail message configuration in the **Communications** view

Automatic and Timed Transmission of E-Mail Messages

By default, messages are sent immediately when created, and multiple reports per one SMTP session will be sent only if they are created at the same time.

If buffering messages is required to, for example, reduce communication costs or preserve power, multiple messages can be sent during one SMTP session using a timer. The **E-mail Transmit** timer for the SMTP client component is configurable in the setup but disabled by default. To enable this mode of operation, proceed as follows:

1. In the **Communications** view, select the option **Send only when timed** for the report(s) you want to send based on the timer.
2. In the **Timers** view, set the desired transmission interval for the SMTP client component's **E-mail Transmit** timer.

HTTP Client

HTTP client provides access to resources on HTTP servers. In AWS applications, its typical uses are:

- Post observation messages and other information to data collection system
- Retrieving setups and other configuration information from network management system
- Retrieving images from a network camera

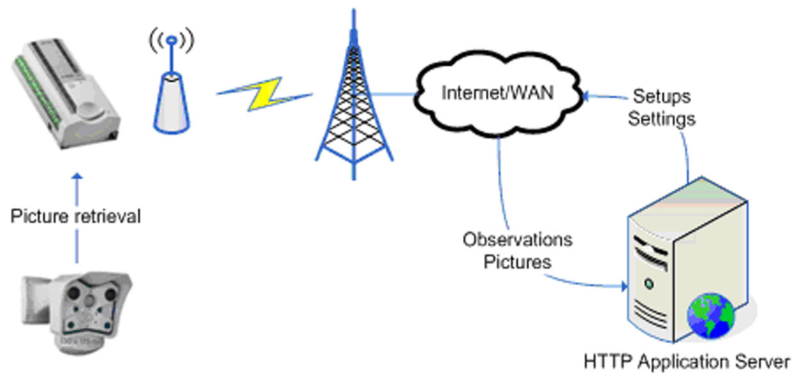

Figure 108 Using HTTP in AWS Application

In Lizard, HTTP client is introduced to the setup from the **IP Services** view. It has two different basic setups:

- HTTP Messenger for sending information
- HTTP Reader for reading information

Sending Messages with HTTP

To send messages from data logger to HTTP application server, add and configure HTTP Messenger to the setup. One messenger can be used to send multiple messages, even to different servers.

1. In the **IP Services** view, add HTTP Messenger to the setup.

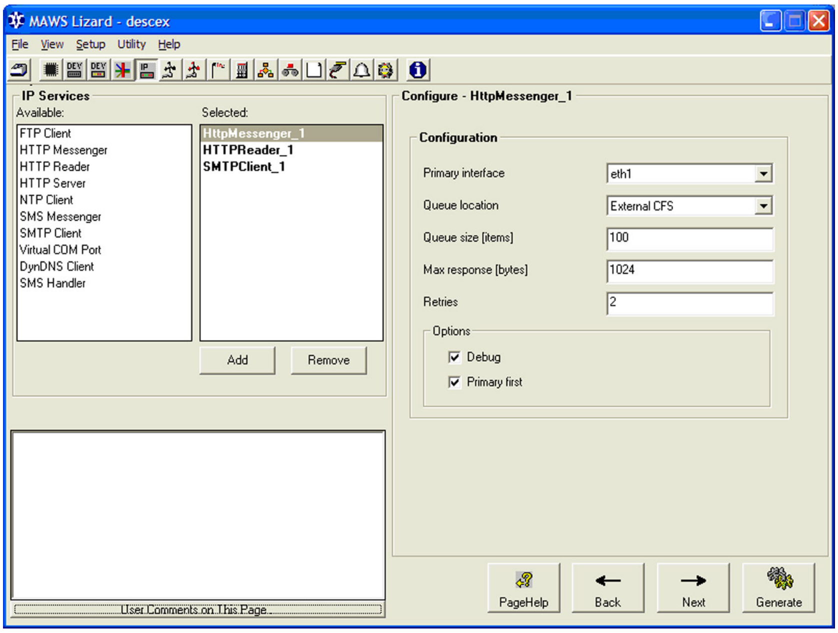

Figure 109 Configuring HTTP Messenger Common Settings

2. Make the following common configurations:

Table 38 Configurations for HTTP Messenger

Parameter	Use	Note
Primary interface	Primary network interface for contacting the server.	Backup(s) may be specified in parameter set *hosts*.
Queue location	File system location where outgoing messages will be queued for transmission. The available options are: - RAM disk - External CFS (requires CF card) (non-volatile) - Internal logfile system (non-volatile)	Specify the same location for all HTTP clients in the system.
Queue size	Number of items, for example, reports, stored in the queue. If this limit is exceeded due to, for example, network congestion, the oldest queue item will be deleted for each new one.	QML can buffer data within limitations of the selected storage location and certain file system constraints. Absolute maximum of 500 items is in effect. Buffering data in RAM disk will consume dynamic memory from the rest of the application. Likewise, the other storage alternatives will consume logging space.
Max response	Maximum size of the expected HTTP response (bytes).	Depending on server configuration, HTTP responses can sometimes be quite verbose and even contain formatting.
Retries	Number of retries in case of failure.	
Options	- Debug information to COM0 - In case a backup interface is used, the first attempt is always made to primary target.	Backups for HTTP are configured using parameter set *hosts*.

3. Create message(s) for transmission using the **Reporting** view. HTTP Messenger does not require any specific format to be used; instead, the format depends on the receiving server application.

4. In the **Communications** view, link the message(s) for transmission.

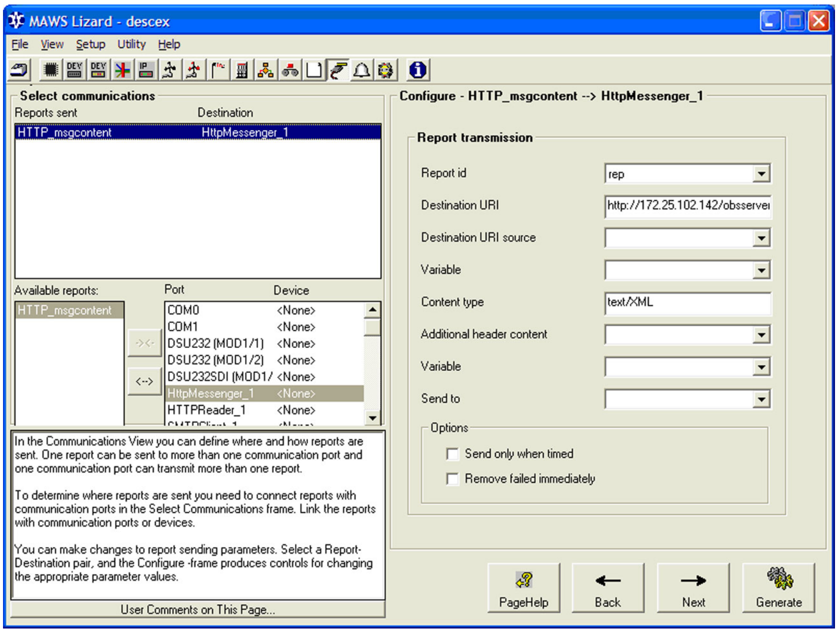

Figure 110 Linking Message for Transmission with HTTP

Parameters configured for each transmitted message are as follows:

Table 39 Parameters Configured for Transmitted Message

Parameter	Use	Note
Report id	Source identifier *rep* when used to send content of a report.	
Destination URI	HTTP URI of the receiving web application.	Alternatively, dynamic URI can be used. See below. Backup(s) may be specified in parameter set *hosts*.
Destination URI source	Source for dynamic or station-specific URI, for example, report or static parameter.	
Variable	Variable in the above source containing the URI, *rep* when used with a report.	
Content type	Content type of the message; see http://www.iana.org/assignments/media-types.	Default *text/plain*.

Table 39 Parameters Configured for Transmitted Message (Continued)

Parameter	Use	Note
Additional header content	Source for additional HTTP header fields, that is, others than those inserted by HTTP Client by default.	Extra content is appended to the HTTP header, and needs to be terminated by <CR><LF><CR><LF>.
Variable	Variable in source containing the additional HTTP header fields.	
Send to	Component where information about HTTP result is sent to.	Vaisala internal use.
Options/Send only when timed	Allows queuing several messages for transmission during the same HTTP session. Reports are not sent immediately when created, but when triggered by 'HTTP transfer' timer. This timer needs to be manually enabled from the timer's view.	
Options/Remove failed immediately	When retries for transmitting a message are exhausted by failure, remove the message instead of leaving it to the queue for later transmittal.	

The following special content convention allows any file from the logger to be sent to HTTP server. If an item, for example, a report, linked to HTTP Messenger for transmission starts with schema file://, the name following the schema will be extracted and the file referenced by it will be sent to server.

For example, a report with content **file:///Ext/images/current.jpg** causes the file */Ext/images/current.jpg* to be sent.

Reading Information

To read information from HTTP server, add and configure an HTTP Reader to the setup. One reader can be used to serve multiple HTTP requests, even from different sources.

All configurations for HTTP Reader except timing are managed from the **IP Services** view.

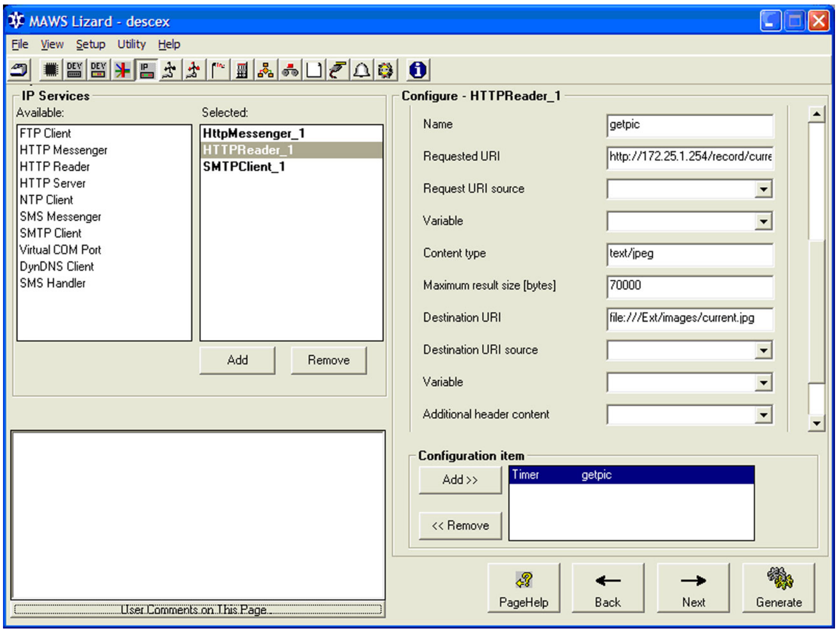

Figure 111 Configuring HTTP Reader

The following parameters are common for all requests served by a particular HTTP reader.

Table 40 HTTP Reader Common Parameters

Parameter	Use	Note
Primary Interface	Primary network interface to use for requests.	Backup(s) may be specified in parameter set *hosts*.
Retries	Number of retries for a request.	
Options	Debug information to COM0. In case backup is used, always attempt the primary interface first.	

New requests to the HTTP Reader are added by adding Configuration Items using the **Add** button. For each request, the following parameters are available.

Table 41 HTTP Reader Request Parameters

Parameter	Use	Note
Name	Unique name for the request.	
Requested URI	HTTP URI to read, for example, URI of the latest camera image.	Alternatively, dynamic request URI may be used; see below.
Request URI source	Source for dynamic or station-specific URI, for example, report or static parameter.	
Variable	Variable in the above source containing the URI, *rep* when used with report.	
Content type	Expected content type; see http://www.iana.org/ assignments/media-types. HTTP Reader will reject return content with different type.	Leave empty if ignored or unknown.
Maximum result size	Maximum size of the returned content, for example, maximum size of a photo [bytes].	Take into consideration limited resources in the logger.
Destination URI	File URI for storing the returned content, for example, file://<path. Example URI: file:///Ext/images/current.jpg. Remember to respect 8.3 file naming rules in the logger.	Alternatively dynamic destination URI may be used; see below.
Destination URI source	Source for dynamic or station-specific URI, for example, report or static parameter.	
Variable	Variable in the above source containing the URI, *rep* when used with report.	
Additional header content	Source for additional HTTP header fields, that is, others than those inserted by HTTP client by default.	Extra content is appended to the HTTP header, and needs to be terminated by <CR><LF><CR><LF>.
Variable	Variable in source containing the additional HTTP header fields.	
Requesting component	Component where information about the result will be sent to.	Vaisala internal use.

HTTP Authentication

HTTP Client supports basic authentication to server. Credentials are set to parameter set *hosts* as follows:

<servername>:http:username <username>

<servername>:http:password <password>

where

<servername>	=	Name of the server
<username>	=	Assigned username
<password>	=	Assigned password

For more information, see section Parameter Set hosts on page 219.

NOTE	Credentials are not encrypted in the logger, not even when transmitted over the network.

Proxy Servers

Accessing HTTP service may involve using a proxy server. Proxy parameters are set to parameter set hosts as follows.

<servername>:http:<ifname>:proxyname <proxy>

<servername>:http:<ifname>:proxyport <port>

where

<servername>	=	Name of the server
<ifname>	=	Name of the interface. Different interfaces may have different proxy settings
<proxyname>	=	Name or IP address of the proxy
<proxyport>	=	Port used for proxy connection

For more information, see section Parameter Set hosts on page 219.

HTTP Server

The QML logger contains a very basic server implementation for HyperText Transfer Protocol connections. Upon an HTTP request from, for example, a browser, it can return the content of a file from the QML file system. Documents to be viewed over HTTP connections are formatted similarly to other QML logger reports, and they are written to the QML logger file system using the file writer component.

NOTE	The QML logger can support a limited number of concurrent HTTP connections. The recommended use is as a single user system, the maximum allowed by Lizard Setup Software being 10 users.

HTTP connections are configured in the **IP Services** view as shown in Figure 112 on page 253.

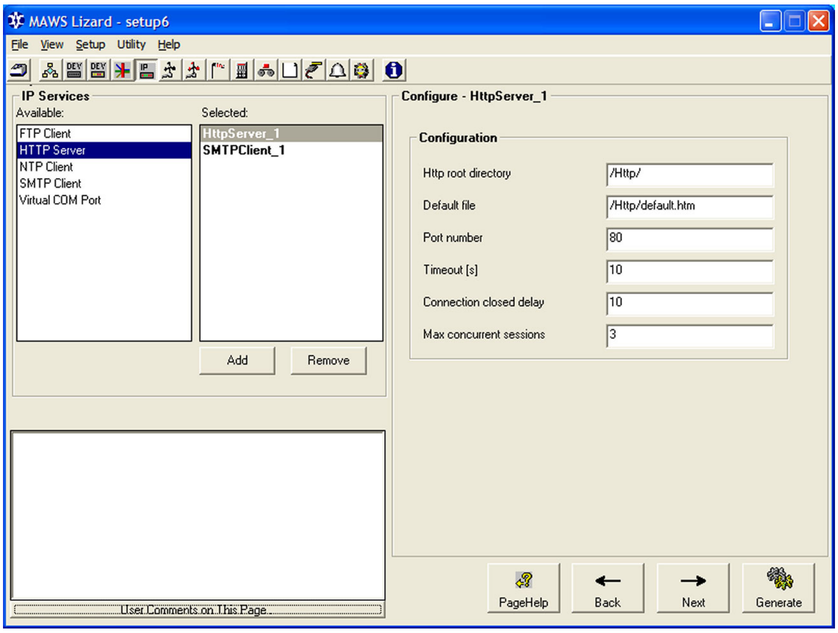

Figure 112 HTTP Server Configuration

The parameters to be configured for the HTTP server are presented in Table 42 on page 254.

Table 42 Parameters for HTTP Server

Parameter	Use
HTTP root directory	Location in the QML file system in which files accessible using HTTP are stored.
Default File	File to be returned as default, that is, when the HTTP request does not specify a file name.
Port Number	Port number where the server will be listening.
Timeout [s]	Timeout in seconds for sending the data.
Connection closed delay [s]	Timeout for waiting the remote end to close the connection. After the timeout, the QML logger will close the connection.
Max concurrent sessions	Maximum number of concurrent sessions.

Creating HTML Reports

Documents in HTTP servers are mostly formatted according to the HyperText Markup Language (HTML) specification. Such documents can be created in the **Reports** view using the **Custom report** template. The report is then linked to the **File** port in the **Communications** view.

To create HTML reports, proceed as follows:

1. In the **Reports** view, select the **Custom report** template from the **Available templates** list.
2. There are two options for starting to create the report:
   a. Writing the report from scratch by selecting **New**. If you select this option, a screen shown in Figure 113 on page 255 is displayed.

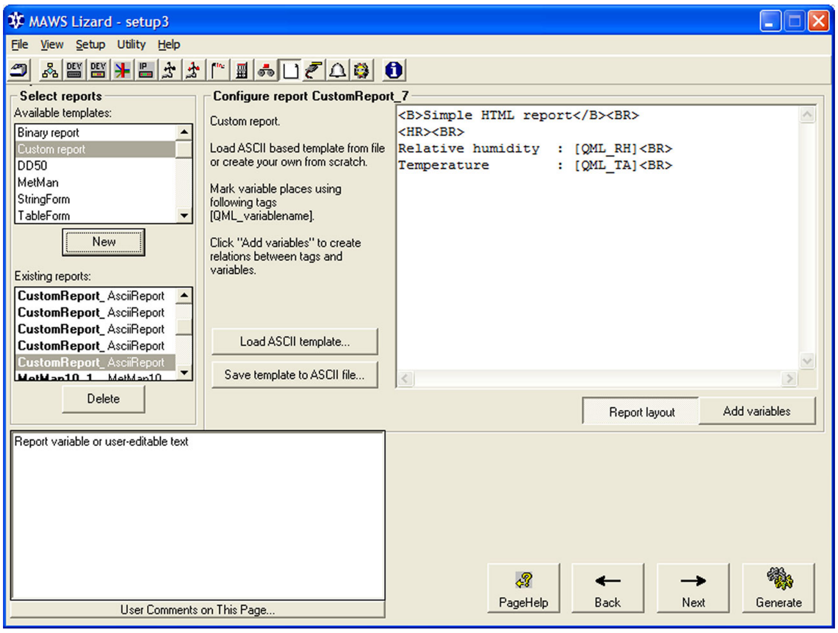

Figure 113 Creating Custom HTML Report from Scratch in Lizard Setup Software

b. Importing an HTML or XML report template from a file by selecting **Load ASCII template** and then browsing for the template file. If you select this option, a screen shown in Figure 114 on page 256 is displayed:.

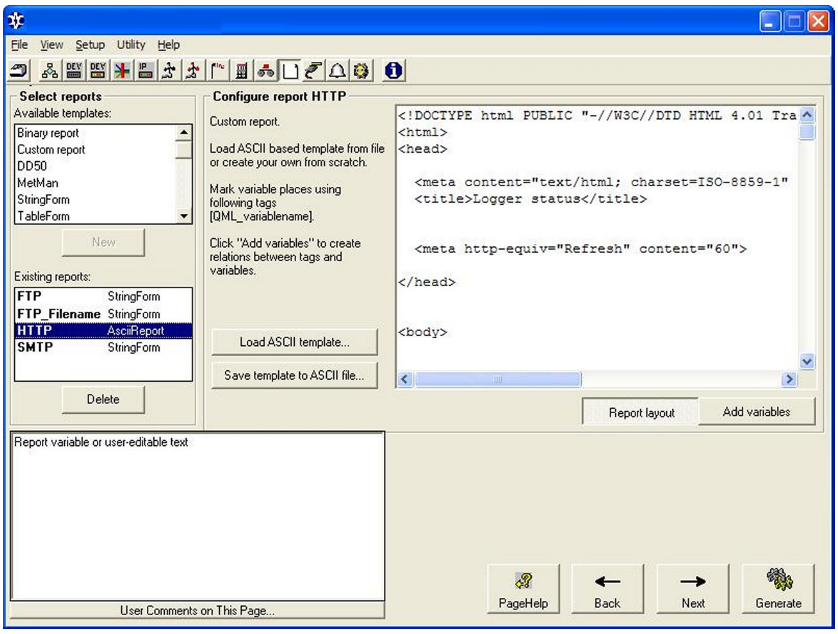

Figure 114 Loading Custom ASCII Template into Lizard Setup Software

Both options are fairly similar, the main difference being that if you load a custom ASCII template into Lizard, you probably load more HTML and more Lizard syntax into Lizard than you do if you select the **New** option.

If you loaded a custom ASCII template, skip steps 3 and 4 below. If you are creating a custom template from scratch, perform steps 3 and 4 before proceeding.

3. Write the HTML code for the report in the **Configure report** frame. You can use HTML formatting tags as you would when writing any HTML page.
4. The Lizard variable placeholder tags are written using the syntax **[QML_variablename]**, where QML_variablename is a placeholder for the variable to be inserted at that point in the report.
5. Once you have written the layout of your report and inserted the variable placeholders, start adding the Lizard variables to your report by selecting **Add variables**.

6. Clicking the **Add variables** button opens a list of the variable placeholders that is used for linking the QML logger variables to their placeholders. The variable placeholders displayed are based on the [QML <variable name>] tags in the report layout.

7. Add your variables by selecting them from the **Select variables** list and dragging them to the corresponding cells in the **MAWS variables** list. Alternatively, you can select the variable and the destination cell in the list and then click **Add**. After you have linked variables to your custom template, it should look similar to the one shown in Figure 115 on page 257.

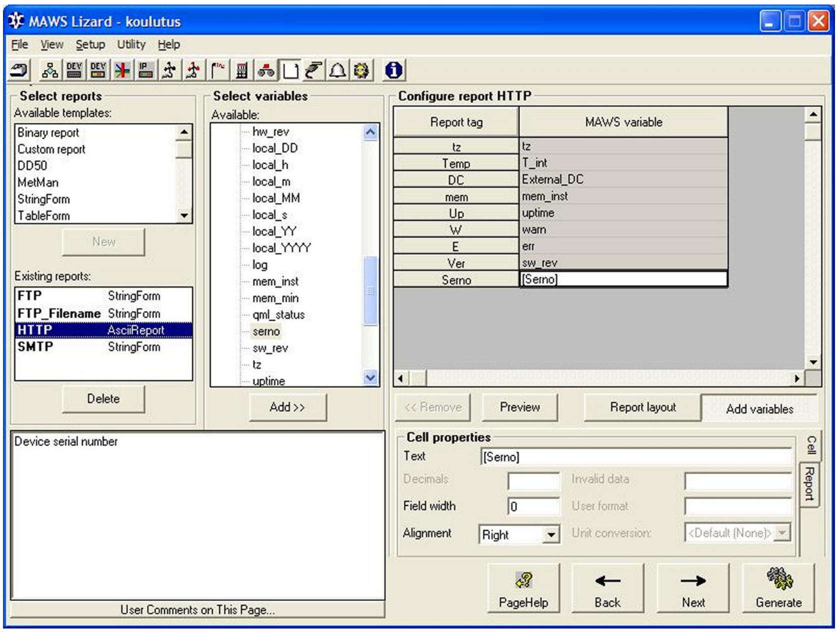

Figure 115 Linking Lizard variables to Custom Report

8. You can preview your report by selecting **Preview**. The **Preview report** window, shown in Figure 116 on page 258, provides two views of the report: one with example values inserted for the variables, and one with the QML logger variable names for the variables. With the variable names, you can ensure that you have added the correct variable to each point in the report.

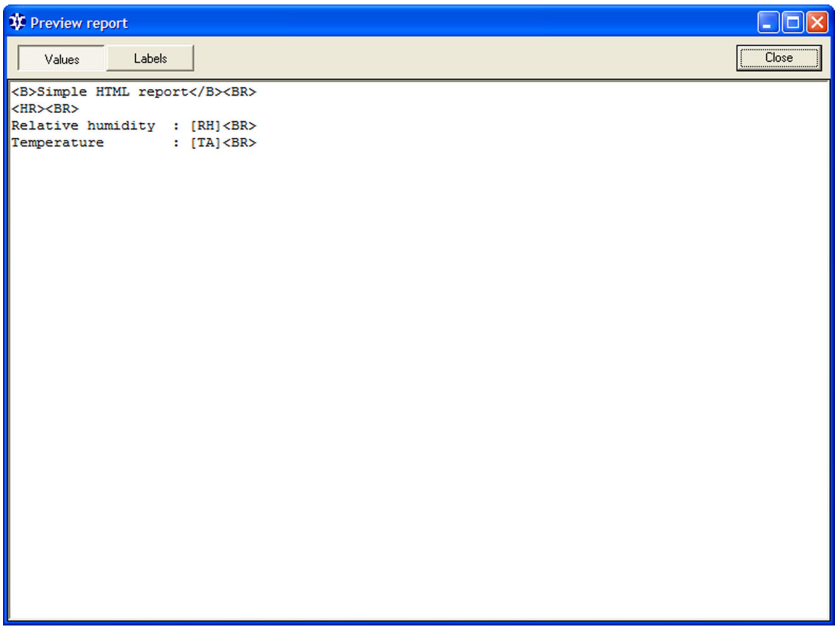

Figure 116 Custom HTML Report Preview

9. Once you have created your report, link it to the **File** port in the **Communications** view as shown in Figure 117 on page 259.

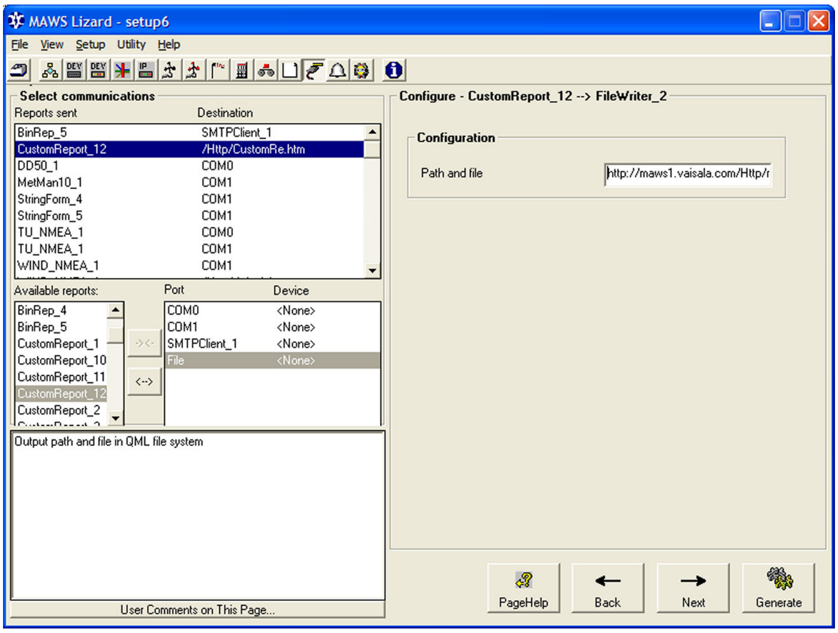

Figure 117 Linking Custom HTML Report to File Port

10. Specify the path and file name for the report in the QML logger in the **Path and file** field. This path and file name is used in the browser to specify the file to be viewed: for example, if the DNS name of the QML logger is **maws1.vaisala.com** and the path of the report is **/Http/report1.html**, the report would be accessed at **http://maws1.vaisala.com/Http/report1.html** in the browser.

Dynamic DNS Client

Dynamic DNS is a service where devices with dynamic IP addresses can be made accessible via Domain Name Service. Whenever the client device's IP address is changed, the client device sends an update to the Dynamic DNS service. The services that request a connection to the device request the device's address from Dynamic DNS using normal DNS query mechanisms.

Dynamic DNS Client is added to setup from Lizard's **IP Services** view. It includes two application components: the client and its connection.

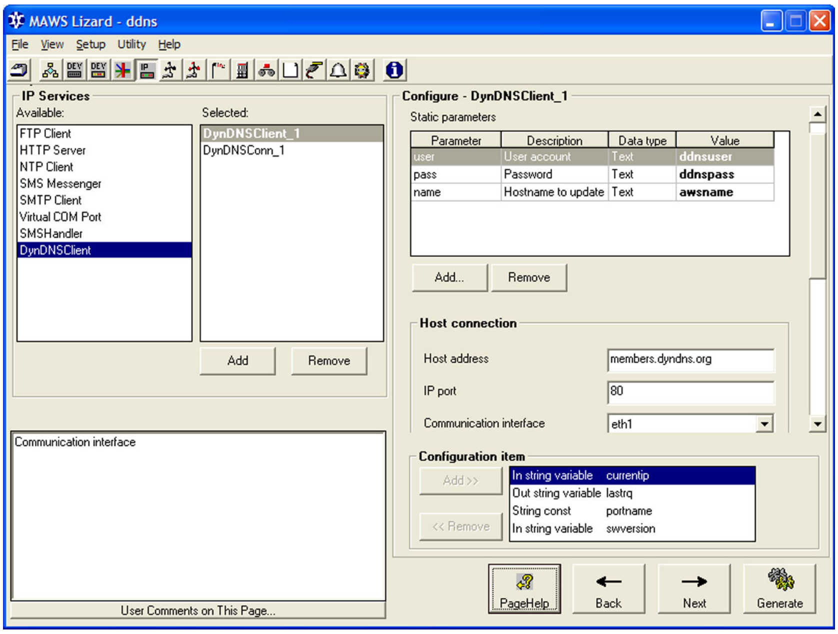

Figure 118 Dynamic DNS Client

Station-specific settings with setup defaults from Lizard and the setup-defined configuration parameters are listed in Table 43 on page 261 and Table 44 on page 261.

NOTE	You must establish a user account at Dynamic DNS service before using the configuration parameters presented in Table 43 on page 261 and Table 44 on page 261. Otherwise, the client device may end up blacklisted. A common service to use is, for example, Dynamic DNS Client (see www.dyndns.org).

Table 43 Station-Specific Settings for Dynamic DNS Client

Parameter	Use	Note
user	User name to use when updating IP to service.	User account must be established prior to using. Otherwise, the client device may end up blacklisted.
pass	Password for the user.	
name	Hostname to update. For example, *testaws.dyndns.org*.	

Table 44 Configuration Parameters for Dynamic DNS Client

Parameter	Use	Note
Host address	Name or address of the Dynamic DNS host running the service.	
IP port	IP port in the host system. By default, Dynamic DNS uses HTTP port 80.	
Communication interface	Interface to use for the connection.	
Timeout	Timeout [ms] for connecting the host.	
Retries	In case of failure, the number of retries for attempting host connection.	
Debug	Produce additional informative output to COM0.	

DynDNSConn, an additional virtual connection, is a connection that the client uses for contacting the defined Dynamic DNS service. Normally, there is no need to make any configuration for it.

Setup Management

Automatic Setup Update

Automatic setup update enables centralized management of the logger setup files (.adc/.dtg) and efficient distribution of the setups to the observation sites.

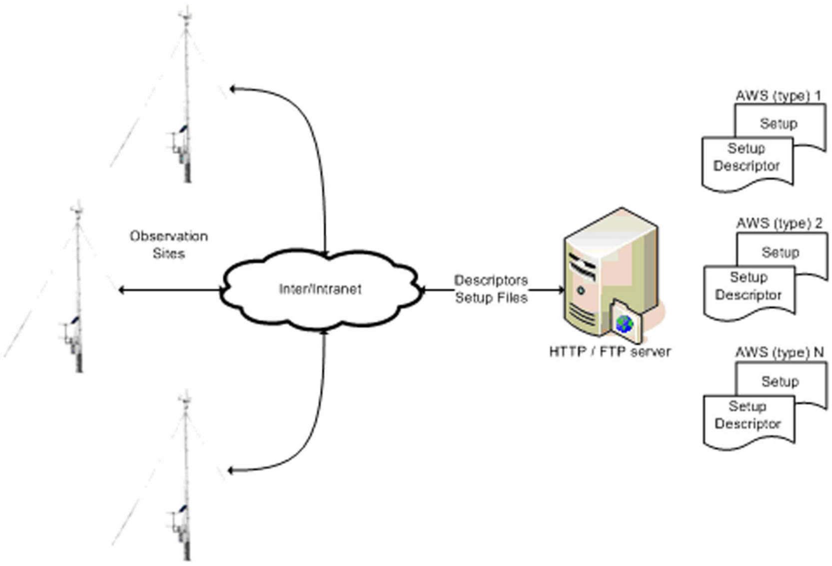

Figure 119 Automatic Setup Management Principle

The operating principle is as follows:

- When a setup (*.adc* file) is generated, MAWS Lizard also automatically generates a small setup descriptor file with extension *.asd*.
- Both setups and descriptors are made available to the network by placing them in HTTP or FTP server.
- According to the given configuration, data loggers in the observation sites periodically read their setup descriptors from the server.
- If information contained by descriptor on the server differs from the currently running setup, data logger reads its setup file from the server. Setup file integrity is validated after the download.
- Depending on the information in the descriptor, the setup is taken into use immediately or at a given point in time.

When generating the setup with Lizard, the following setup descriptor information dialog can be opened using the **Advanced** button.

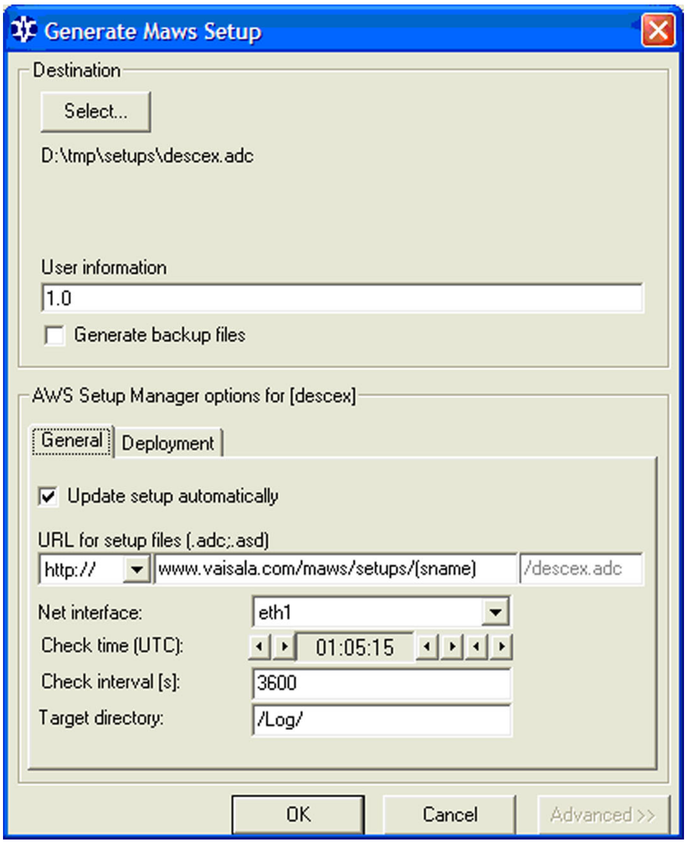

Figure 120 Setup Information Dialog

Table 45 Elements in Setup Information Dialog

Configuration	Use
User information	User-specific information to be included, for example, setup version number.
Generate backup files	When the **Generate backup files** option is selected, Lizard also creates a zip file named <setupnameYYMMDDhhmmss.zip>
Update setup automatically	Selection if the automatic setup update is used.
URL for the setup files	HTTP, FTP, or file URL to check for updated setup from. Accepts the following placeholders, that is, values that will be replaced by logger-/site-specific value: - (parameter name) = static parameter - (serno) = serial number
Net interface	Network interface used to access the URL.

Table 45 Elements in Setup Information Dialog (Continued)

Configuration	Use
Check time	Synchronization time for the check. Avoid making checks too frequently or at times when the logger is executing other time-consuming operations, for example, at midnight.
Check interval	Check interval in seconds.
Target directory	Directory on the logger where the files will be loaded to.

The **Deployment** tab contains information for synchronized setup update.

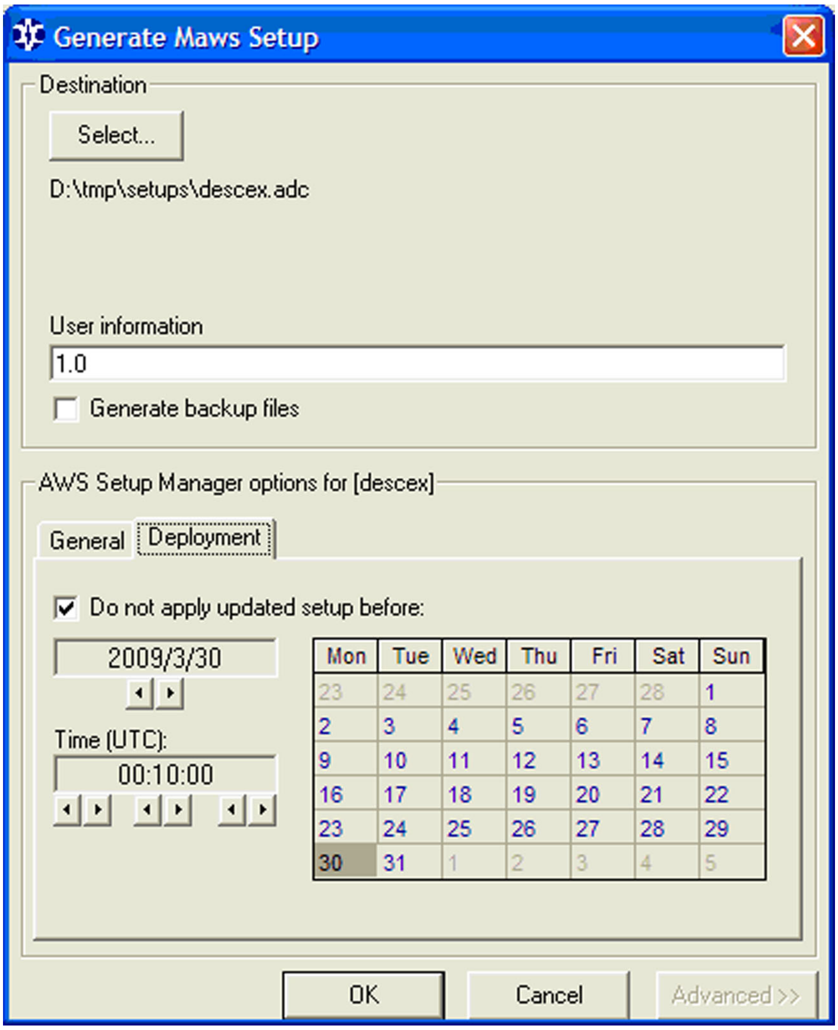

Figure 121 Setup Deployment Timing

Deployment timing can be later updated by updating the generated setup descriptor.

Table 46 Elements in Setup Deployment Timing Dialog

Configuration	Use
Do not apply updated setup before	Timed setup update in use. Setup will be loaded to the QML logger as soon as it is available, but it will be taken into use at a given time.
Date/calendar	Use the arrow buttons below the date to select the month, and the calendar to select the day.
Time	Use the arrow buttons below the time to adjust the hours, minutes, and seconds.

Setup descriptor is an ASCII file with extension *.asd*. It uses XML syntax and contains the following information.

```
<?xml version="1.0" encoding="us-ascii"?>
      <SetupInfo>
      <Name>descex.asd</Name>
      <SwVersion>7.00</SwVersion>
      <Size>3863</Size>
      <Creator>Vaisala</Creator>
      <Created>2009-03-30T05:37:56+00:00</Created>
      <CRC>377163335</CRC>
      <Info>1.0</Info>
      <URL>http://www.vaisala.com/maws/setups/
      descex.adc</URL>
      <ValidFrom>2009-03-30T00:10:00+00:00</ValidFrom>
</SetupInfo>
```

Table 47 User-Editable Fields in Setup Descriptor

Field	Use
Info	User-specific information to be included, for example, setup version number.
Creator	Setup created by.
URL	HTTP, FTP, or file URL to check for updated setup from.
ValidFrom	Time when the setup will be taken into use. ISO8601 format YYYY-MM-DDThh:mm:ss±hh:mm

NOTE	Do not modify other fields than the listed ones. Preferably, validate your changes using an XML editor.

Command setupupdate

The data logger setup can be manually loaded from HTTP or FTP server. The shell command syntax is as follows:

setupupdate *{-uri <uri>} {-dst <filename>} {-if <netif>}*

Table 48 Parameters for Command setupupdate

Parameter	Use
-uri <uri>	Specifies HTTP, FTP, or file URI where setup descriptor will be loaded from. For example, -uri http://172.25.112.180/setups/descex.asd - If URI specifies a descriptor, the descriptor has to be different from the currently running setup before the setup it defines can be taken into use. - If URI specifies a setup file, *descex.adc* in this example, it is always taken into use. - If not given, the command uses setup manager settings from parameter set *SetupManager*.
-dst <filename>	File on the logger where file will be loaded to for temporary use. Preferably, use /Ext/<filename>. - Use external CF card if present. - Otherwise, use /Log/<filename>. - Alternatively, target can be set to parameter set *SetupManager*, parameter *target*.
-if <netif>	Network interface to use for loading. Needed only if more than one interface is available.

Examples of command **setupupdate**:

/ > setupupdate -uri http://172.25.112.180/setups/demo1.asd

```
Received 306 bytes of 306
Received 113414 bytes of 113414
Setup is ok
Resetting device now
```

/ > setupupdate -uri http://setupmgr/descex.adc -dst /Ext/descex.adc -if eth1

```
Received 3863 bytes of 3863
Setup is ok
Resetting device now
```

/> setupupdate

```
Received 306 bytes of 306
this setup is already in use
```

Virtual COM Ports

Virtual COM ports, in Lizard Setup Software referred to as **VCOM**, provide similar functionality as physical COM ports. In addition to configuration, from a setup point of view there is no practical difference between the two.

Instead of using a physical serial port, virtual COM ports tunnel the traffic through a TCP socket connection, that is, it performs similar functionality as the COM Server module.

Virtual COM ports can be configured to function as server (passive connection), when remote computer initiates the connection to QML, or as client (active connection), when QML makes the connection to the remote computer. Typical uses for virtual COM ports are:

- Sending reports to a server socket on a remote server
- Accepting an incoming connection for service
- Connecting a sensor with a TCP/IP interface to the QML logger

Virtual COM ports are configured in the **IP Services** view as shown in Figure 122 on page 267.

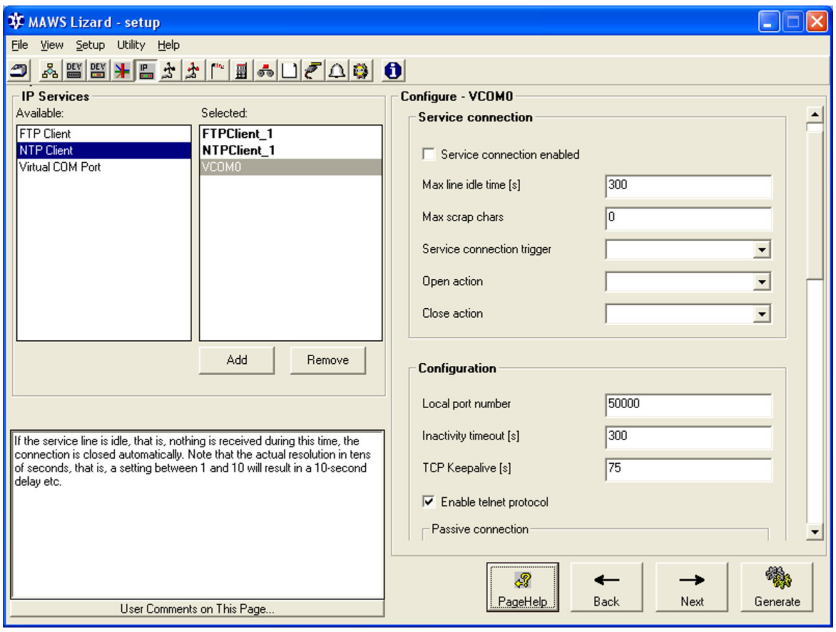

Figure 122 Virtual COM Port Configuration

Parameters for virtual COM port configuration are presented in Table 49 on page 268.

Table 49 Virtual COM Port Parameters

Parameter	Use	Note
Service connection enabled	Accept service connection from this port.	If service connection is required, there should be at least one communications interface (both logical and physical) with incoming connections enabled. The option **Accept incoming** must also be selected for the virtual COM port.
Max idle time [s]	If no commands are issued through the open service port during this time, the port will be closed automatically.	The idle time is configured in seconds.
Max scrap chars	Maximum number of "scrap" characters without <CR> to accept before automatically closing the open service connection.	Used to reject interference caused by, for example, line disturbance or malfunctioning devices.
Service connection trigger, open action, close action	Target component and action identifiers to trigger when service connection is opened or closed.	
Local port number	Number of the local TCP port.	Used only for incoming (passive) connections. The local port number must be unique for each virtual COM port accepting incoming connections.
Inactivity timeout	Time in seconds for which connection is kept open with no traffic.	
TCP keepalive	TCP keepalive transmission interval in seconds.	Ensures connection status. If three attempts fail, the connection will be closed. Generates "dummy" traffic through an open connection, thus preventing idle timeouts that may exist, for example, in GSM modems. The connection is closed if four consecutive transmission attempts fail.
Enable Telnet protocol	Enable support for Telnet escape sequence decoding and negotiation.	Virtual COM port does not provide full Telnet functionality, but it provides handling for certain escape sequences to enable binary file transfer.

Table 49 Virtual COM Port Parameters (Continued)

Parameter	Use	Note
Passive connection/Accept incoming	Allow incoming connection requests.	If selected, the QML logger opens a server socket in the port specified by the **Local port number** parameter.
Active connection	**None** = no active connection; **Auto start** = connect to remote server at startup and try to keep connection open all the time; **With any character** = connect to remote server when data is being sent to port.	
Flush input buffer	Defines the connection event which causes any data present in the input buffer to be discarded.	
Flush output buffer	Defines the connection event which causes any data present in the output buffer to be discarded.	
First remote access point	Always use the first defined remote server when opening an active connection.	
Debug/Output component	Port to which debug output will be written. No selection disables debug output.	
Debug/Telnet protocol debug	Output debug information for Telnet protocol. Used only if the debug output component is selected.	
Buffers/Transmit [bytes]	Size of the transmit buffer in bytes.	
Configure hosts	Opens the remote host configuration dialog. For instructions, see section Remote Host Configuration on page 270.	

Remote Host Configuration

Remote hosts to which active connections are made are configured using a separate dialog. To add a remote host, proceed as follows:

1. Click the **Configure hosts** button at the bottom of the virtual COM port parameter view. The **Virtual port hosts** dialog shown in Figure 123 on page 270 is displayed.

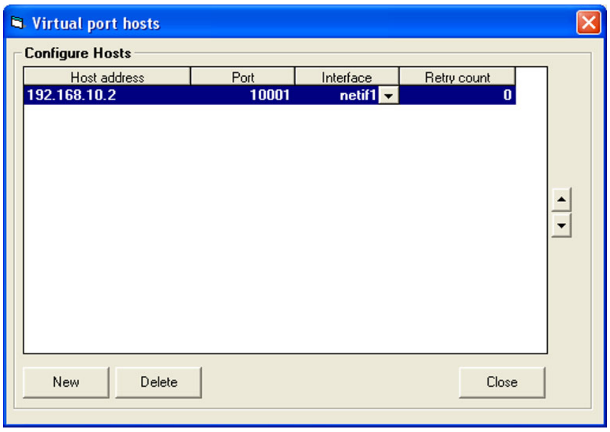

Figure 123 Remote Host Configuration for Virtual COM Ports

2. Click **New** to add a host.
3. Enter the host name or IP address in the **Host address** field.
4. In the **Port** field, enter the remote port number where the host has a server socket listening.
5. Select the interface you want to use for the connection from the **Interface** list.
6. In the **Retry count** field, enter the retry count, that is, how many connection attempts will be made before switching to next host, if any.
7. You can enter multiple hosts to act as backups for each other. If connection to one of the hosts fails, the next one will be used. Depending on the selection **First remote access point**, connection will always be initially attempted to the first one or to the last known functioning host.
8. Close the **Virtual port hosts** dialog by clicking **Close**.

Configuration for remote host(s) is stored into a parameter set with the same name as the virtual port, for example, **VCOM1**. Station-specific values for these parameters can be set using the **paramset** commands described in section Commands for Parameter Sets on page 293.

To delete a remote host, select the host from the list and click **Delete**.

NOTE	In addition to selecting the option **Accept Incoming**, it is not necessary to define an interface for incoming connections. Virtual COM ports can accept a connection from any open interface.

Virtual COM ports buffer outgoing data within the limitations of the configured transmit buffer. However, according to the buffer flushing configuration, the output buffer may be cleared when certain communication events occur.

Service Connection for Virtual COM Ports

The service connection to a virtual COM port is not a pure ASCII terminal session, but the information is transferred using a TCP/IP connection between the terminal software (for example, HyperTerminal) and the QML logger.

The primary requirement for incoming service connection is that the QML logger must have a known fixed IP address. If the address is private, as usually is the case in LAN environments, service access is possible only from inside the local network to which the QML logger is connected, not the Internet. If the address is public and there are firewall configurations allowing the connection to be made, it is possible to access the QML logger from the Internet.

NOTE	As the QML logger provides only username/password security, it is recommended to keep it behind external, strictly restrictive firewall(s).

Point-to-point dial up will establish a network with two peers, where the requirement of a known fixed IP address is also fulfilled. The known fixed IP address is required for establishing a service connection.

Enabling Service Connection for Virtual COM Ports

Two parameters must be set in order for a virtual COM port to allow service access:

1. **Service connection enabled**
2. **Passive connection**: **Accept incoming**.

Enabling Physical Connection

The physical connection used for an incoming service connection can be either an Ethernet or a dial-up link.

Ethernet

If the setup is using the Ethernet Communication Module DSE101, you can establish a physical link for external access as described below:

1. In the **Communication interfaces** view, select your interface. Ethernet interfaces are named **ethX**, for example, **eth1**.
2. Set the value of the parameter **autoinit** to **Yes**. This setting automatically starts the Ethernet connection when the QML logger is started.

Dial-Up

Setting up a dial-up link, that is, using a modem or null-modem connection, requires setting up a dial-up network connection.

NOTE	These instructions apply when using PSTN or GSM data call as the carrier. Using GPRS or a similar packet switched connection for remote service purposes is not discussed here, and is most likely a topic to be solved with the cellular network operator. The service connection requires obtaining a known fixed IP address for the weather station.

As parameters for outgoing and incoming dial-up connection are usually different, it is necessary to create a separate communication interface for the incoming connection. In Lizard's **Communication interfaces** view, create a new interface with the settings presented in Figure 124 on page 273:

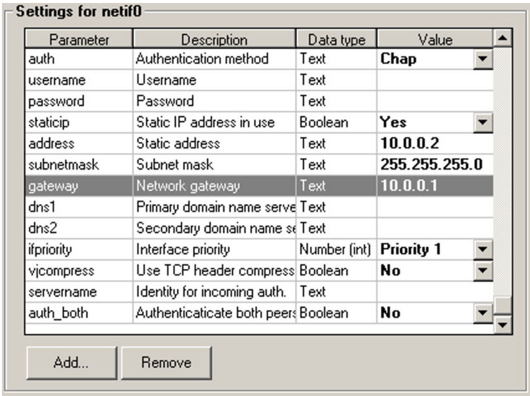

Figure 124 Settings for Incoming Dial-Up Connection

On the client computer (Windows) side, instructions for installing a modem and setting up a dial-up network connection can be found in the Windows online help.

Tips for configuring the dial-up connection on the client-computer side:

- Before configuring the dial-up network connection, there must be a dial-up modem installed.
- Use the New Connection Wizard from network connections.
- Connection type is Internet connection.
- Connection must be set up manually.
- Device type is a dial-up modem.
- Provided username and password must have a match in the QML parameter set **pppclients**. For further information, see section Authentication on page 217.
- Do not make the dial-up connection your default Internet connection.

Terminal Program Configuration

For a terminal program, for example, HyperTerminal in Windows, the connection is configured as a network connection. The default connection parameters for HyperTerminal when using dial-up are presented in Figure 125 on page 274. With Ethernet, the host address is usually different, based on the network configuration used.

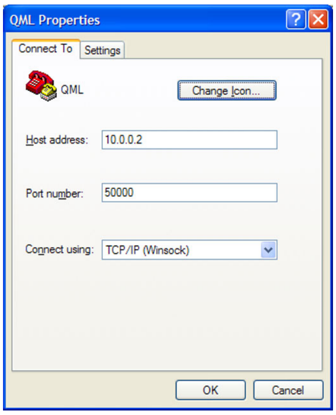

Figure 125 Connection Settings for Terminal Program

Generic IP Modem Configuration

The device **Generic IP modem** can be used to create a modem control for an unspecified modem respecting states and the command/response scheme of an AT modem. Generic IP modem is configured in the **Devices** view as shown in Figure 126 on page 275.

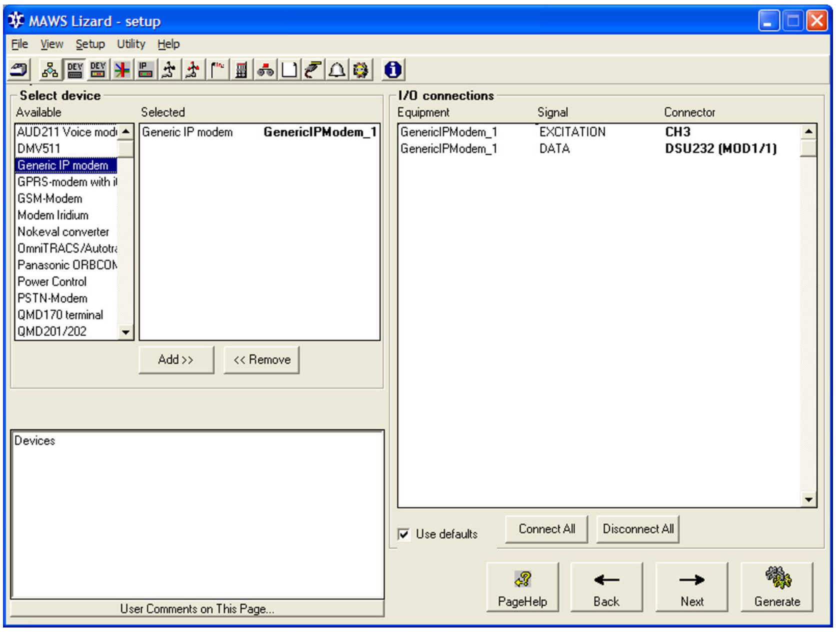

Figure 126 Generic IP Modem Configuration

NOTE	Generic IP modem is available only with the advanced tools password. The instructions presented here also apply to advanced configuration of any of the preconfigured modems, that is, the devices **(Win)NullModem IP**, **Siemens GPRS-IP,** and **PSTN-IP**.

Operating Principle

The diagram shown in Figure 127 on page 276 illustrates the normal operational states of the modem control. The state "failed" is not described, as it is a transitory state with entry from most of the other states.

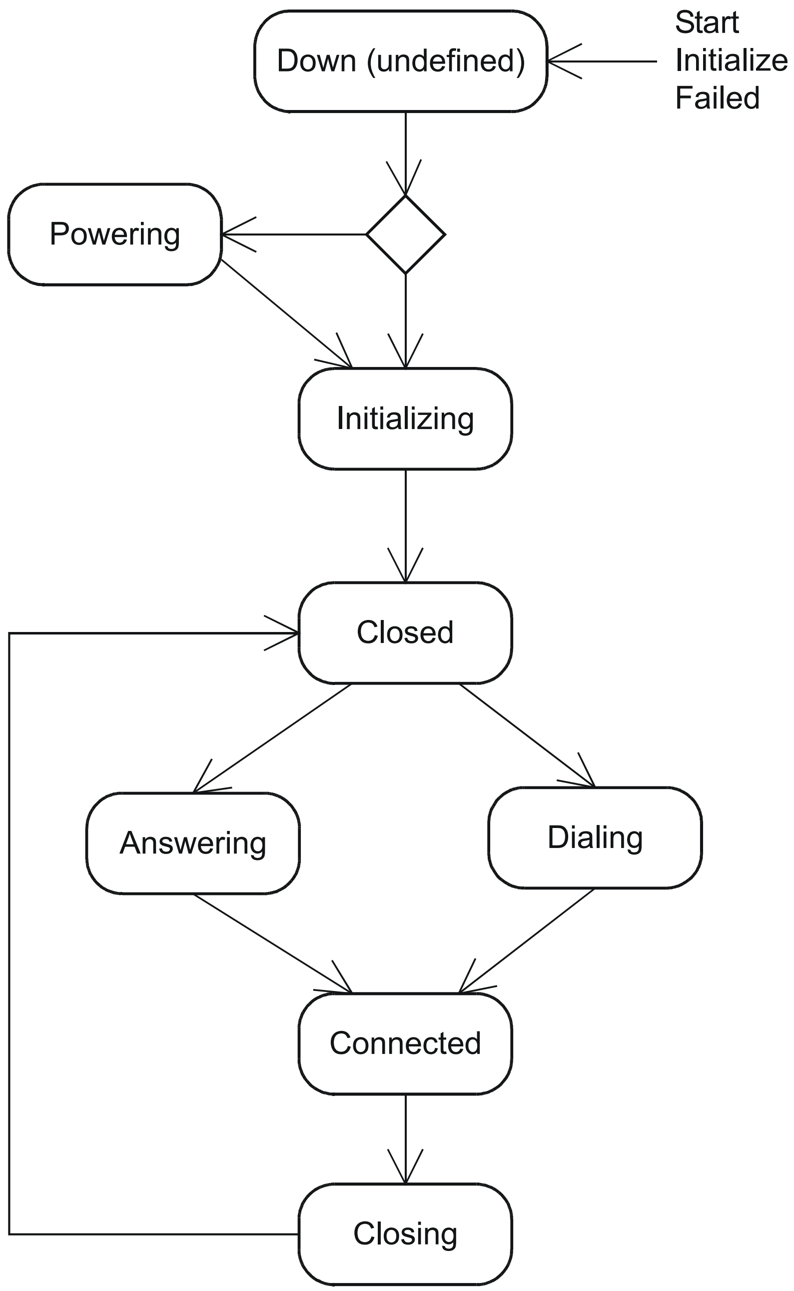

Figure 127 Generic IP Modem States

A state can have from zero to multiple command exchanges associated with it. These command exchanges can be executed when entering the state or when leaving the state. There are various options, including timeouts, repetitions, conditional, and ignoring errors associated with each command exchange.

If any of the compulsory command exchanges fails when entering a state, modem control will switch to failed state.

Depending on the power control option used for the modem, the idle state can be down, that is, conserving power, or closed.

Configuring Generic IP Modem

All basic modem parameters as described in section Common Modem Parameters on page 206 are also available with the generic IP modem. Advanced modem configuration is done in the lower part of the **Device configurations** view as shown in Figure 128 on page 277.

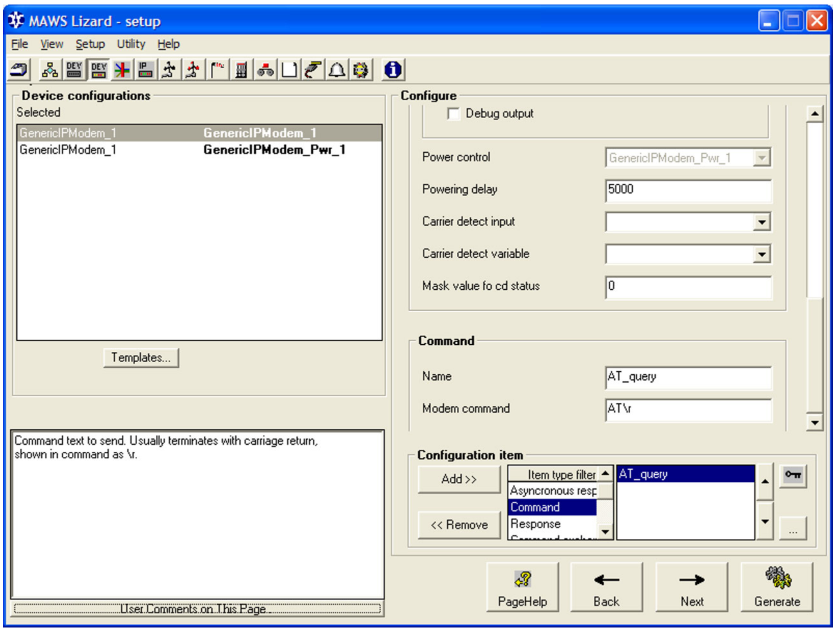

Figure 128 Generic IP Modem Configuration

The **Item type filter** list shows the currently selected configuration item type. It is used to control the content of the list of existing items, and to select the type for a new item. The available configuration item types are listed in Table 50 on page 278.

Table 50 Configuration Item Types for Generic IP Modem

Type	Use
Command	Modem commands.
Modem response	Modem responses.
Command exchange	Sequence where a command is sent to the modem and one or multiple responses can be received in return.
Asynchronous response	Similar to command exchange but triggered by the spontaneous modem response which should trigger an action. For example, an incoming call.

To add a new item, proceed as follows:

1. Select the type of item to add in the **Item type filter** list.
2. Click **Add**. The new item and its configuration parameters appear above the **Configuration item** frame.

To remove an existing item, proceed as follows:

1. Select the type of the item in the list of available items.
2. Select the item to remove.
3. Click **Remove**.

NOTE The order of items is significant for command exchanges: items belonging to same state will be executed in order defined here. You can use the up and down arrows to change the order of execution.

To configure the modem operation in different states, proceed as follows:

1. Define the commands to be sent to the modem. The parameters for a command are presented in Table 51 on page 278.

Table 51 Parameters for Modem Commands

Parameter	Use
Name	Unique name for the command
Modem command	Command string

It is recommended that you use command names that are easy to remember later during the configuration. At this point, the commands are not yet associated with any state.

Parameters can be inserted in commands by enclosing them in curly brackets, **{ }**. Parameters named inside the brackets are attempted to be read from the following sources:

- Parameter set for the associated logical interface, that is, **netifX**.
- The modem control's own parameter set, for example, **GSM_Siemens_1**.

If the parameter cannot be found, the command will be sent to the modem as it is written.

A built-in delay command can be used to pause the sequence for a specified period of time (in milliseconds). The delay command syntax is:

DELAY(TIME_MS)

Non-printing characters can be inserted using the C programming language syntax, for example:

- \r = carriage return
- \x1B = ASCII character 1BH, that is, <ESC>
- If the command contains quotation marks ("), they must be entered as \".

NOTE	The Null (\x00) character is not allowed in the command.

Below are examples of valid commands:

- AT\r
- ATV1E1&D0\r
- ATD{isp}\r
- \x7E\xFF\x7D\x23\xC0\x21\x7D\x25\x7D\x23\x7D\x20\x7D\0User request\x79\x7B\x7E
- DELAY(5000)

NOTE	For a comprehensive set of AT commands supported by your modem, please refer to the modem documentation.

An example of the generic modem command configuration is presented in Figure 129 on page 280.

Figure 129 Example Modem Command Configuration

2. Define the responses received from the modem. Parameters for a response are presented in Table 52 on page 280.

Table 52 Parameters for Modem Response

Parameter	Use
Name	Unique name for the response.
Type	Response type: - **Ok**: indicates successful action. - **Failed**: indicates failed action.
Modem response	Response string
Options	**Message reception** = not used; **Request to connect** = indicates that the modem control should open the connection, if enabled, when this response is received.

It is recommended that you use response names which are easy to remember later during the configuration. At this point, the responses are not yet associated with any state.

Note that with the standard AT interface the same response may indicate successful or failed operation depending on the state of the modem. For example, **NO CARRIER** indicates a failure when dialing, but is an **Ok** response when closing the connection. In this case, it is necessary to define two response instances with the same modem response, but with a different name and type.

It is not necessary to define the full response, but any part of the response distinguishing it from other responses can be used. For example, it is not necessary to define all connection responses with different link indicators, such as **CONNECT 9600**, but it is enough to just define the modem response as **CONNECT**. Additional information received from the modem will be ignored.

The C programming language syntax can be used to insert non-printing characters into the response (see the previous step). The response cannot contain any parameters.

Examples of valid responses are listed below:

- OK
- ERROR
- RING
- +CPIN: SIM PIN\r\n\r\nOK

An example of generic modem response configuration is presented in Figure 130 on page 281.

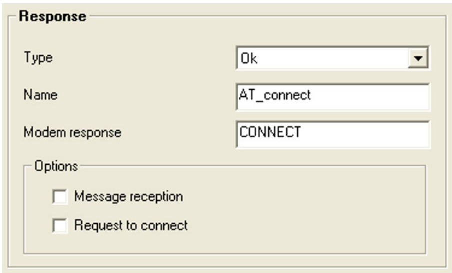

Figure 130 Example Modem Response Configuration

3. Define the command exchanges associated with the different states. The parameters for a command exchange are presented in Table 53 on page 282.

Table 53 Parameters for Modem Command Exchange

Parameter	Use
Type	Selection if executed when entering or exiting the state.
State name	Name of the state in which the command exchange is executed.
Command to send	One of the defined commands to send.
Expected response	Name(s) of expected response(s). The list may be empty, that is, no response, or it may contain multiple responses separated by commas. Do not enter the modem response string(s) here, but use the response names defined in the previous step. Make sure that you type the name(s) correctly, including upper/lower case. List also the error responses; otherwise, they will exit only through a timeout.
Conditional response	Name of the previous response for conditional execution, that is, this command exchange will be executed only if the previously received response matches the given response. For example, the PIN code setting for GSM modem will be executed only if the previous response indicates that it is needed. For an example, see the command exchange **AT_pinset** for device **Siemens GPRS-IP** in Lizard Setup Software.
Timeout	Timeout for the command in milliseconds.
Retries	Number of retries in case of failure.
Options	**Negate conditional** = Execute only if the previous response was not the given one; **Ignore error** = ignore error responses.

4. As needed, adjust the order of execution as described in section Generic IP Modem Configuration on page 275.

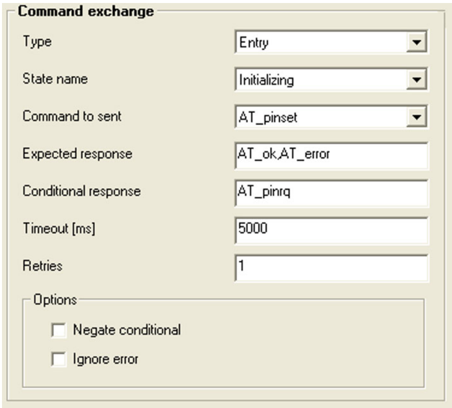

Figure 131 Example Command Exchange Configuration

5. Define asynchronous responses. Parameters for an asynchronous response are presented in Table 54 on page 283.

Table 54 Parameters for Asynchronous Response

Parameter	Use
State name	State where processed
Modem response	Name of the modem response

Typically, the only processed asynchronous response is the indicator for an incoming call, that is, it is associated with the **RING** response.

An example of generic modem asynchronous response configuration is presented in Figure 132 on page 283.

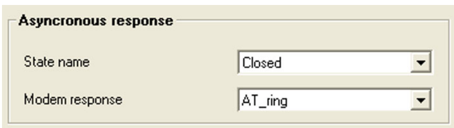

Figure 132 Example Asynchronous Response Configuration

6. Once the configuration is completed, you can test the operation:

   a. Ensure that the option **Debug output** is selected and observe the debug output from the QML logger service port COM0.

   b. Lizard Setup Software makes only basic checking of the modem parameters. Many configuration problems are reported as warnings or errors during and after the QML logger startup. Use the QML logger shell commands **warnings** and **errors** to check for problems.

SMS Handling

NOTE	Configuration presented in this chapter requires that MAWS Lizard is used on the Advanced user level.

SMS processing capability enables the QML logger to do the following:

- Send short reports or alarms as SMS messages
- Receive SMS as commands to be processed to, for example, activate a network connection for maintenance

Modem Configuration

When SMS capability is required with IP functionality, the modem to select in MAWS Lizard is **Siemens GPRS-IP SMS**; see Figure 133 on page 284. However, using SMS features is optional, that is, this IP-modem control can be used even if SMS functions are not in use.

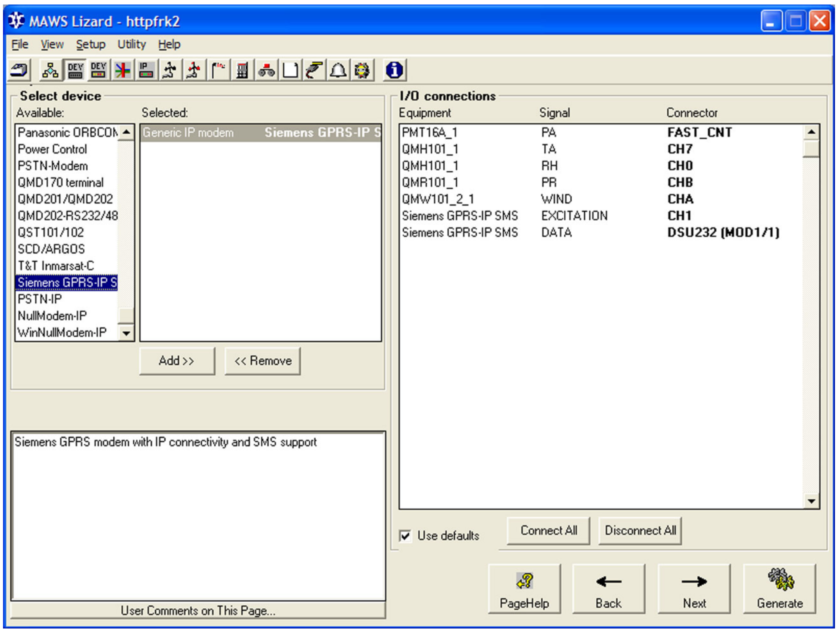

Figure 133 Siemens GPRS-IP SMS Modem

Enabling SMS handling does not necessarily require additional configuration to the modem control. The following settings affect SMS handling and may be necessary, depending on the planned use.

Table 55 Modem Parameters for SMS Processing

Parameter	Use	Note
SMS centre	Telephone number for the operator's SMS center. **This is usually available from the network** without a fixed setting. When needed, the number has to be set using the international number format. For example, +3584012345.	To enable this setting, command exchange AT_smscentre needs to be enabled by removing the **disabled** selection. For further instructions, see section Command Exchange Option Disabled on page 286.
Answer incoming	If incoming SMS processing is required, that is, SMS handler component is in use, the **Answer incoming** option needs to be enabled in the modem.	

Command Exchange Option Disabled

Modem command exchange option **Disabled** is selected by default for command exchange **AT_smscentre**. See Figure 134 on page 286 below.

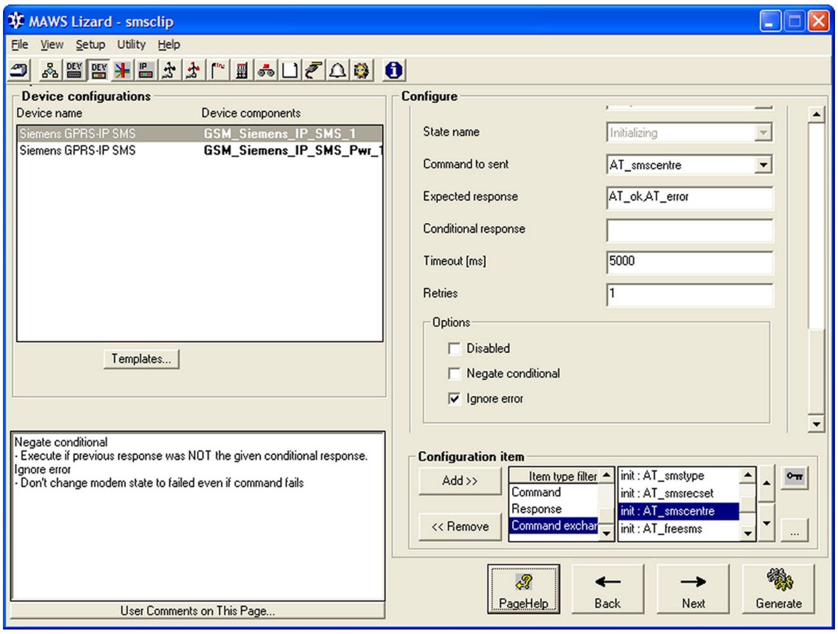

Figure 134 Modem Command Exchange Enabled

To enable execution of the SMS centre setting, proceed as follows:

1. In Lizard, open the **Device configurations** view for telemetry devices and select the modem to configure; see Figure 134 on page 286.
2. In the left-side list in the **Configuration item** box, select **Command exchange**.
3. In the right-side list in the **Configuration item** box, select **AT_smscentre**.
4. Clear the **Disabled** check box.

Application Services

There are two application components available in Lizard's **IP Services** view for SMS processing:

- SMS Messenger for sending messages
- SMS Handler for processing incoming messages

It should be noted that even though SMS services are not IP services, they are configured from the **IP Services** view, as they are related to GPRS modem, the device that provides IP services.

SMS Messenger

General parameters for SMS Messenger, shown in Figure 135 on page 287, are configured in the **IP Services** view. Messages are linked for transmission in the **Communications** view.

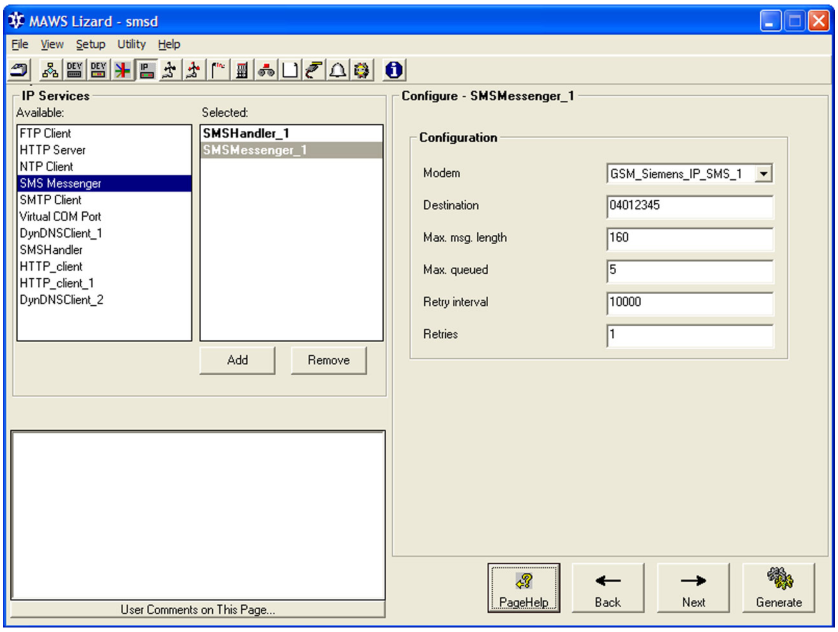

Figure 135 SMS Messenger Advanced User View

The general configuration parameters for SMS Messenger are presented in Table 56 on page 288.

Table 56 Configuration Parameters for SMS Messenger

Parameter	Use	Note
Modem	Modem control to use; see section Modem Configuration on page 284.	
Destination	The number where the SMS is sent to.	1
Max. msg. length	Maximum message length. Longer messages will be cut to fit.	Message length is limited by GSM standard. Thus, do not exceed 160 characters. Support for longer SMS depends on the modem used.
Max. queued	Maximum number of SMS queued for transmission. If this count is exceeded, messages will be discarded starting from the oldest one.	
Retry interval	Message transmission retry interval [ms].	
Retries	Number of entries.	

1. Destination number can be set as station parameter to a parameter set with the same name as SMS Messenger, for example, as set SMSMessenger_1. Parameter name within this set is sendto.

When linking messages to SMS Messenger, there are no parameters to configure, as shown in Figure 136 on page 289.

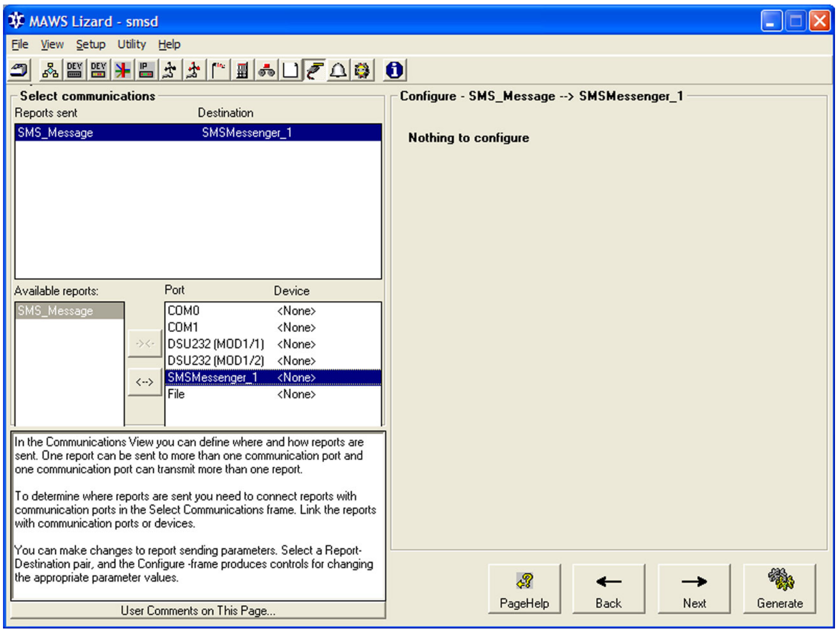

Figure 136 Report to SMS Messenger

SMS Handler

Typical uses for SMS Handler are applications where network connection -related actions need to be triggered from outside the AWS, for example, from network management system. This feature provides SMS wakeup mechanism in cellular network environment, where IP-connections typically can be opened from AWS to Internet but not from Internet to AWS.

Requestable actions are:

- Opening a network interface, that is, connecting to GSM network and acquiring an IP address through GPRS connection
- Opening a preconfigured connection to remote server using virtual COM port

If the received SMS is not recognized as a request message, its content and basic information are stored into output variables and can be used by other application components.

Parameters for SMS Handler are configured in the **IP Services** view, as shown in Figure 137 on page 290.

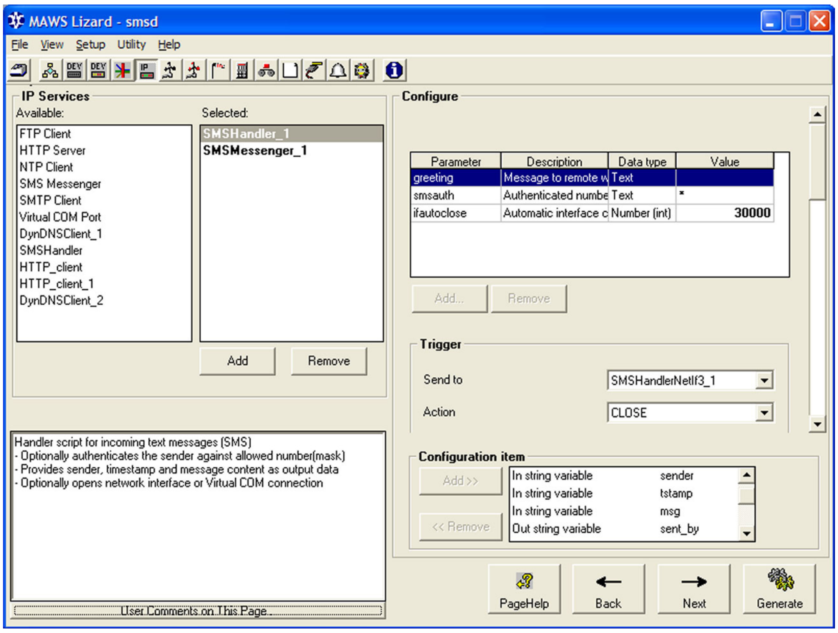

Figure 137 SMS Handler

Station-specific settings with setup defaults from Lizard and setup-defined configuration parameters are listed in Table 57 on page 290 and Table 58 on page 291.

Table 57 Station Settings for SMS Handler

Parameter	User	Note
greeting	Message to be sent when VCOM connection to remote host is opened.	
smsauth	Allowed SMS sender number or number area. Examples: +3584012345 = allow exactly +35840123* = allow all numbers starting with +35840123	Wildchar (*) works only as a postfix.
ifautoclose	Time to automatic interface close [ms] after opening. Setting 0 disables automatic close.	

Table 58 Configuration Parameters for SMS Handler

Parameter	User	Note
Controlled network interfaces	Lists names of the controlled network interfaces, if any. At maximum, there can be three network interfaces controlled by the same SMS Handler.	
Communication device	Modem to listen for incoming SMS.	
Info printouts to COM0	Produces additional informative output to COM0.	

When using the SMS Handler for interface state management without automatic connection to remote server:

1. In Lizard, select the name(s) of the managed interface(s) in the lists available in the **Controlled network interfaces** window.

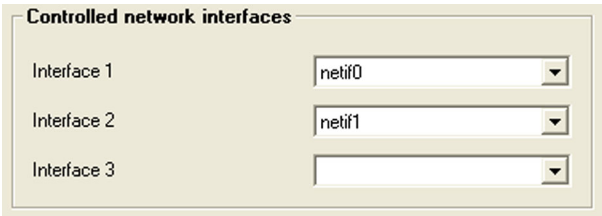

Figure 138 SMS Handler Interfaces

2. When you need to connect an interface to network, send an SMS with the following content:

   connect intfX

   where X is the number of the interface in the **Controlled network interfaces** window. For example, to connect to interface number 2, send:

   connect intf2

When using SMS Handler to open a connection to a predefined remote server:

1. Add a Virtual Com port to the setup with the configuration option **Active connection with any character** and set the target server information for it from the **Virtual port hosts** dialog.
2. Open SMS Handler configuration. In the **Configuration items** list, select one of the following variables: *conn1*, *conn2*, or *conn3*. Select the newly created VCOM as **Send to target**.

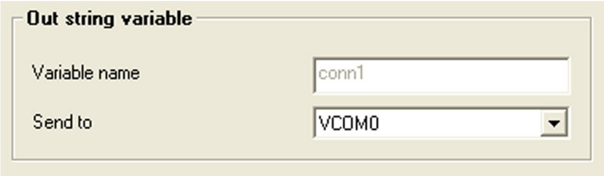

Figure 139 SMS Handler VCOM

3. When SMS-triggered connection needs to be opened, send an SMS with the following content:

   open connX

   where connX is the name of the output variable associated to VCOM in step 2. For example, to open a connection using VCOM associated to variable *conn2*, send:

   open conn2

SMS Handler produces the output variables shown in Table 59 on page 292. There are also additional variables visible, but they are not useful for normal application purposes.

Table 59 Outputs Produced by SMS Handler

Variable	Content	Note
message	Content of the latest received message.	
sent_at	Timestamp of the latest received message. The format is according to GSM specification: *YY/MM/DD,hh:mm:ss<tz>*, where <tz> is the time zone in quarters of an hour, preceded by a plus sign. For example, *08/10/20,16:46:46+12*.	
sent_by	Originating phone number of the latest received message.	

Commands for Terminal Connection

This section presents the QML logger shell commands for configuring parameter sets and TCP/IP-based connections.

Commands for Parameter Sets

The commands available for manipulating parameter sets are presented in Table 60 on page 293.

Table 60 Commands for Parameter Sets

Command	Use
paramset	Lists all parameter sets
paramset <set>	Shows values of all parameters in the given set
paramset <*set*> <*parameter*>	Shows value of single parameter is the given set
paramset <*set*> <*parameter*> <*value*>	Creates or sets value for parameter in the given set.
paramsetclear <set>	Removes a complete parameter set
paramsetclear <*set*> <*parameter*>	Removes a single parameter from the given set

NOTE When setting station-specific IP addresses, phone numbers, or PIN codes using the QML service terminal, double quotation marks must be used around the value. For example: **paramset eth1 address "10.0.0.1"**

Examples of **paramset** command output:

```
/ > paramset
GSM_Siemens_1
pppclients
netif0
netif1
VCOM0
```

```
/ > paramset GSM_Siemens_1
S:00000011: GSM_Siemens_1:pin2 =
S:00000011: GSM_Siemens_1:pin = 1234

/ > paramset VCOM0 host1 "10.0.0.1"

/ > paramset pppclients:myuser password wontpass

/> paramsetclear GSM_Siemens pin2
```

Command netif

The **netif** command is used to control and check the status of network interfaces. The command syntax is:

netif [**open** <*ifname*> | **close** <*ifname*>]

Parameters for the **netif** command are described in Table 61 on page 294.

Table 61 Parameters for Command netif

Parameter	Use
No parameters	Lists current status of all configured network interfaces
open	Opens the given interface
close	Closes the given interface
ifname	Name of interface to be opened or closed, for example, *netif0*

NOTE

In spite of an issued close command, the interface may still remain open if a service is using it. Closing will always be delayed by the specified idle time (set in configuration).

Depending on the connection used, changing the state of a network interface may take a while. For example, with the PSTN modem, the **netif open** command will take time for establishing the telephone connection plus the time for network connection negotiation. This can easily total over one minute.

Examples of **netif** command output are presented below:

```
/> netif
Network interface states:
interface 0 (netif0) : Closed
interface 1 (eth0)   : Opened
Physical network interface states:
DSE101_0             : Opened   owned by 1
PhyNullModemWin_1    : Closed

/>netif open netif0
Trying to open network interface 0
interface 0 state: Opening

/> netif close netif1
Trying to close network interface 1
interface 1 state: Idle
```

Command ipconfig

The command **ipconfig** shows the IP configuration for all open network interfaces. The command has no parameters.

An example of **ipconfig** command output is presented below:

```
/> ipconfig
pp1: (default network interface)
ÜP POINTOPOINT NOARP MTU:1500
DHCP Enabled. . . : No
IP Address. . . . : 62.78.127.31
Subnet Mask . . . : 255.255.255.0
Default Gateway . : 62.72.127.1
DNS Servers . . . : 217.78.192.22
                    217.78.192.78
```

Command ping

Ping is a commonly used command for testing access to an IP address. The command syntax is:

ping <*destination*>

The parameter *destination* is the destination IP address or host name.

Response to the command shows:

- IP address of the destination
- Roundtrip time taken to reach the destination
- TTL, remaining time to live (= maximum number of hops)

NOTE	The **ping** command requires an open network interface to work. Use the **netif** command to open the appropriate interface.

An example of **ping** command output is presented below:

```
/ > ping www.vaisala.com
Pinging 62.61.78.68
Reply from 62.61.78.68: time=168ms TTL=241
```

Command net

The **net** command has multiple command options for showing various network statistics. The command syntax is:

net <*warnings* | *statistics* [*clear*]>

Parameters for the **net** command are described in Table 62 on page 296.

Table 62 Parameters for Command net

Parameter	Use
warnings	Displays warning level events occurred in communication
statistics	Displays statistical information about protocols and active network interfaces
clear	Clears network warnings or statistics

NOTE	All information is shown as the number of packets transmitted, except for the sent and received data counts per network interface, which are in bytes. Interface related data is available only for currently open interfaces, and it is cleared when the interface is closed.

Examples of **net** command output are presented below:

```
/> net warnings
Warning: NTP server changeover (se.pool.ntp.org)
occurred 1 times first in ntpclient.cpp[827]
during thread: 00063938 [Worker_1]
object pointer: 6AEF0 [component: NTPClient_1]
Warning: Unable to resolve NTP server IP (fi.pool.ntp.org)
occurred 1 times first in ntpclient.cpp[738]
during thread: 00063938 [Worker_1]
object pointer: 6AEF0 [component: NTPClient_1]

/ > net statistics
Total packets per protocol
TCP
Sent: 22
Recd: 22
Drop: 0
UDP
Sent: 0
Recd: 0
Drop: 0
ICMP
Sent: 0
Recd: 0
Drop: 0
IP Sent: 42
Recd: 22
Drop: 0
Sent/recd bytes and dropped packets per interface
pp0
Sent: 339
Recd: 292
Drop: 0

/ > net statistics clear
```

Command ftp

FTP client functionality can be used from the QML logger command shell to transfer files and to test the connection to an FTP server. The **ftp** command syntax is

ftp <*get* | *put* | *test*> <*user:pass*> <*source*> [*destination*] [*interface*] [*options*]

Parameters for the **ftp** command are described in Table 63 on page 298.

Table 63 Parameters for Command ftp

Parameter	Use
get	Gets file from server
put	Puts file to server
test	Tests connection to server
user:pass	Username and password to FTP server, separated by a colon (**:**)
source	Name of the source, depending on command; for example, file on local or remote system
destination	Name of the destination, that is, depending on command, file on local or remote system
interface	Interface to use; the default value is netif0
options	**f** = Force destination directory creation on remote serve **a** = Append to file **d** = Write debug output

Examples of **ftp** command output are presented below:

```
/ > ftp test user:pass ftp.vaisala.com "" netif1
Opening interface...ok
Connecting ftp.vaisala.com...ok

/Log > ftp put user:pass L3070515.dat ftp.vaisala.com/MAWS
netif0 f
Opening interface...ok
Sent successfully 658 bytes

/Cfg > ftp get user:pass ftp.vaisala.com/win_out.adc
Opening interface...ok
Received successfully 4482 bytes
```

Command ntp

The command **ntp** can be used to manually set the QML realtime clock and to test the connection to an NTP server. The command syntax is:

ntp <*set* | *show* | *test* > <*server*[:*port*]> [*maxroundtrip*] [*interface*]

Parameters for the **ntp** command are presented in Table 64 on page 299.

Table 64 Parameters for Command ntp

Parameter	Use
set	Sets QML realtime clock to server time
show	Shows server time
test	Tests server connection and shows debug output
server[:port]	NTP server to contact. Port is optional and the default value is 123.
maxroundtrip	Maximum allowed roundtrip in milliseconds, for example, time between request transmit and data reception. The default value is 5000 ms.
interface	Interface to use. The default value is **netif0**.

Examples of **ntp** command output are presented below:

```
/> ntp test ntp.vaisala.com
Opening interface...
NTP: Sending datagram to ntp.vaisala.com
1B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 C9 FF E0 96 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00
NTP: data received, time error [s] -8
Status: 00 -> Clock operating correctly
Ref updated:    Thu May 24 09:45:17 2007
Req txd :       Thu May 24 09:59:50 2007
Req rxd :       Thu May 24 09:59:58 2007
Resp txd:       Thu May 24 09:59:58 2007
Resp rxd:       Thu May 24 09:59:50 2007
roundtrip:      0

/> ntp set fi.pool.ntp.org:123 2000 netif1
Opening interface...
Time set to server UTC:  Thu May 24 09:15:46 2007

/> ntp show ntp.vaisala.com
Opening interface...
Received server UTC:  Thu May 24 10:15:18 2007
```

Application Examples

This section contains application examples illustrating the uses of the QML logger TCP/IP-based functionality in different network environments. The example setups contain configuration for using TCP/IP-based services, but do not represent any kind of meteorological or hydrological application.

Example setups containing the example configurations presented in this section are also included on the MAWS software CD, starting from version 6.01. The example setups are:

- gprsdemo.dtg for GPRS configuration
- ethdemo.dtg for Ethernet configuration
- pppdemo.dtg for PPP configuration

GPRS Connection

This example demonstrates a setup for using the GRPS-IP modem for:

- Sending reports as files to FTP-server
- Synchronizing the QML logger time from a network time server
- Accepting incoming CSD calls for a TCP/IP-based service connection

The network setup to be configured is illustrated in Figure 140 on page 301.

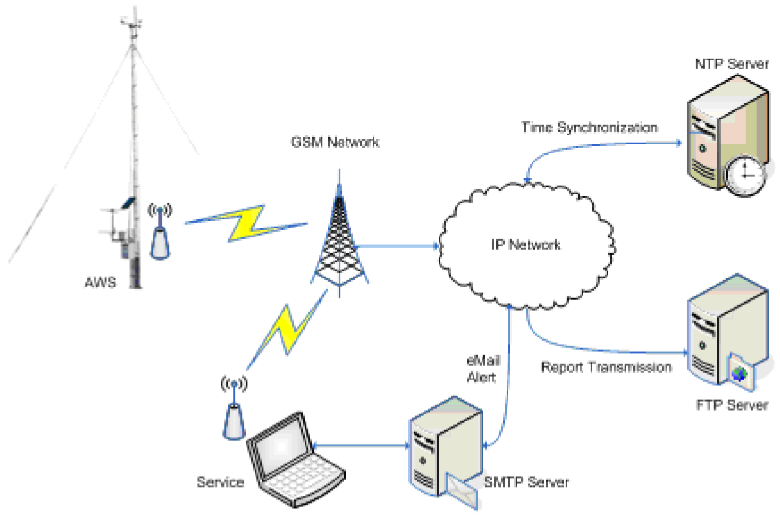

Figure 140 Example GPRS and GSM Network Configuration

To create the network configuration using Lizard, proceed as follows:

1. Add and configure the physical communication devices in the setup.

   a. In the **Optional hardware** view, add the module **DSU232** to either module place.

NOTE	It is not recommended that you use serial link speeds over 9600 bps.

   b. In the **Communication devices** view, add the modem **Siemens GPRS-IP** as shown in Figure 141 on page 302.

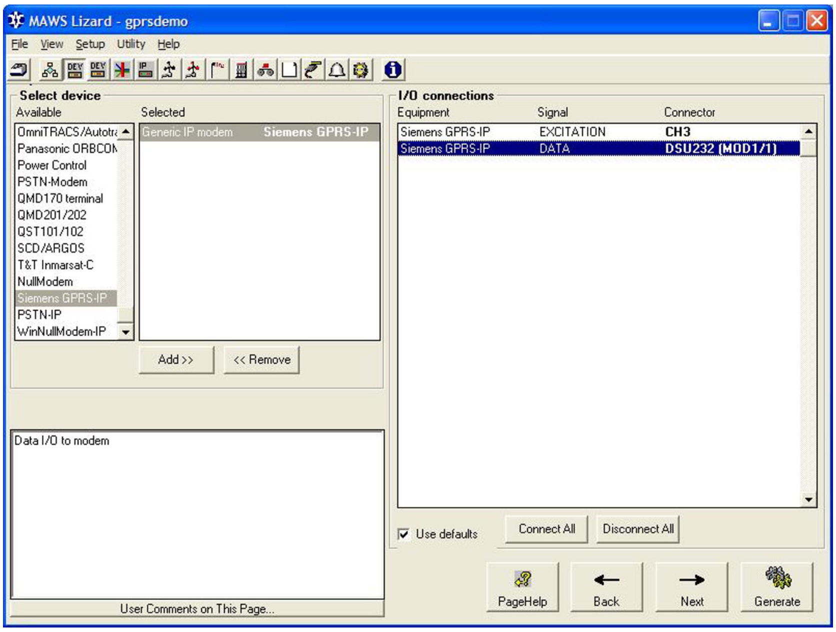

Figure 141 Connecting GSM/GPRS Modem

- Connect the **DATA** signal to a channel on DSU232.
- Connect the **EXCITATION** signal to powering channel.

c. In the **Device configurations** view, select the device **GSM_Siemens_1**.

- Set the PIN code default value, if any, in the **Command parameters** frame, parameter **pin**. See Figure 142 on page 302.
- If PIN-code authentication is not used, you do not need to delete the **pin** parameter. No PIN code will be sent to the modem unless the modem requests it after power-up.

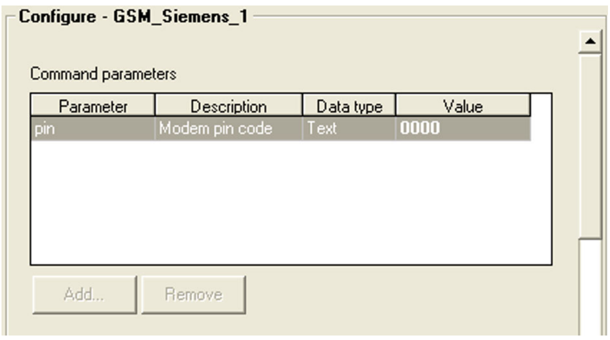

Figure 142 Entering Default PIN Code

NOTE	The PIN code value presented here is only an example. Ensure that a proper PIN code is used; otherwise, the SIM card in the modem will be locked with the PUK code after three failed PIN entries.

d. Enable answering incoming calls by selecting the **Accept incoming** option.

2. In the **Communication interfaces** view, add and configure a communication interface for the modem as shown in Figure 143 on page 303.

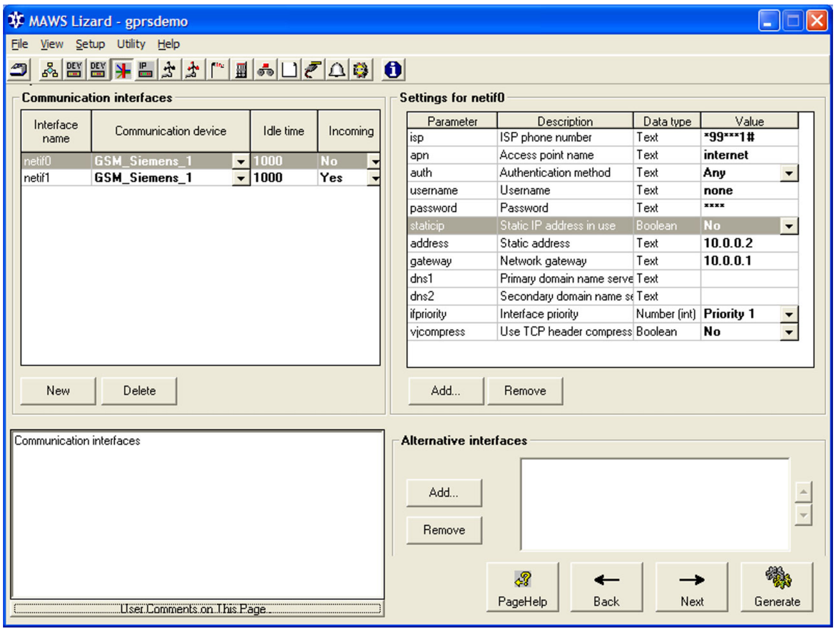

Figure 143 Configuring Network Interface for GSM/GPRS Modem

a. Click **New** to add a new interface.

b. Change the settings for the new interface (**netif0**) as required. The most common operator-dependent settings for a GPRS connection are presented in Table 65 on page 303.

Table 65 Operator-Dependent Parameters for GPRS Connections

Parameter	Use
apn	Network access point name
username	User name for network access
password	Password for network access

NOTE	The user name and/or password may be left blank if they are not required by the network operator.

c. Create another communication interface dedicated for the incoming service connection. Change the following settings:

- Allow incoming connections by changing the interface setting **Incoming** to the value **Yes**.
- Leave the **isp** parameter blank. This interface must not be used for outgoing connections.
- Select the value **Yes** for the **staticip** parameter.

3. In the **IP Services** view, add and configure an FTP client for sending reports.

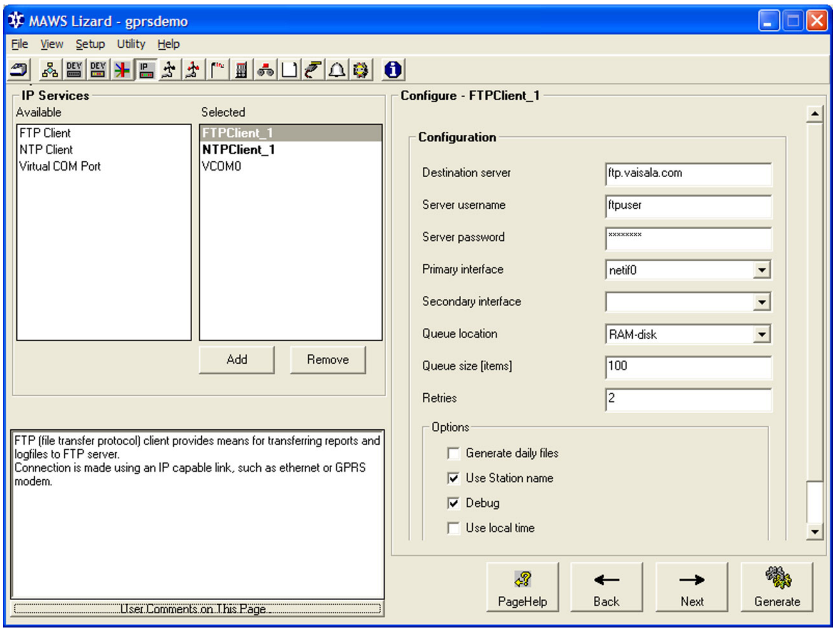

Figure 144 Configuring FTP Client

a. Add an FTP client by selecting **FTP Client** from the **Available** list and clicking **Add**.

b. The settings to configure for the FTP client are outlined in Table 66 on page 305

Table 66 Settings for FTP Client

Setting	Description
Server username	User name for the FTP-server
Destination server	Name of the destination FTP server
Server password	Password for the FTP server
Primary interface	**netif0**, that is, the GPRS/GSM modem you have previously added in the **Devices** view

NOTE The destination server ftp.vaisala.com used in Figure 144 on page 304 is used as an example only. It is not a public server, and the user credentials in the setup are not valid for it.

4. In the **IP Services** view, add and configure an NTP client for QML logger time synchronization as shown in Figure 145 on page 305.

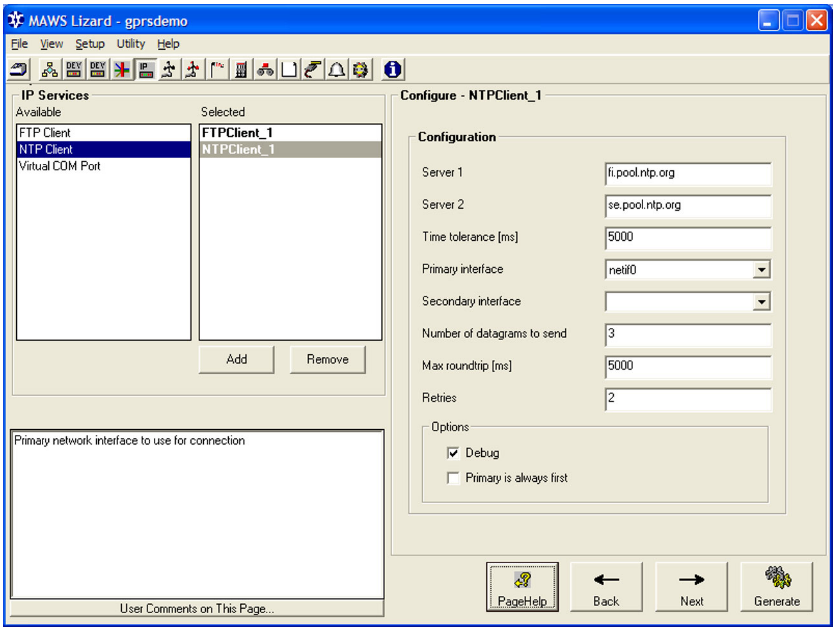

Figure 145 Configuring NTP Client

a. Add an NTP client by selecting **NTP Client** from the **Available** list and clicking **Add**.

b. The settings to configure for the NTP client are outlined in Table 67 on page 306.

Table 67 Settings for NTP Client

Setting	Description
Server 1	Primary NTP server to use
Server 2	Secondary NTP server to use
Primary interface	netif0, that is, the GPRS/GSM modem you have previously added in the **Devices** view

NOTE	If you using a public NTP server, it is recommended that you select a server close to the weather station location, for instance, a server located in the same country.

5. In the **IP Services** view, add and configure a virtual COM port for service access as shown in Figure 146 on page 306.

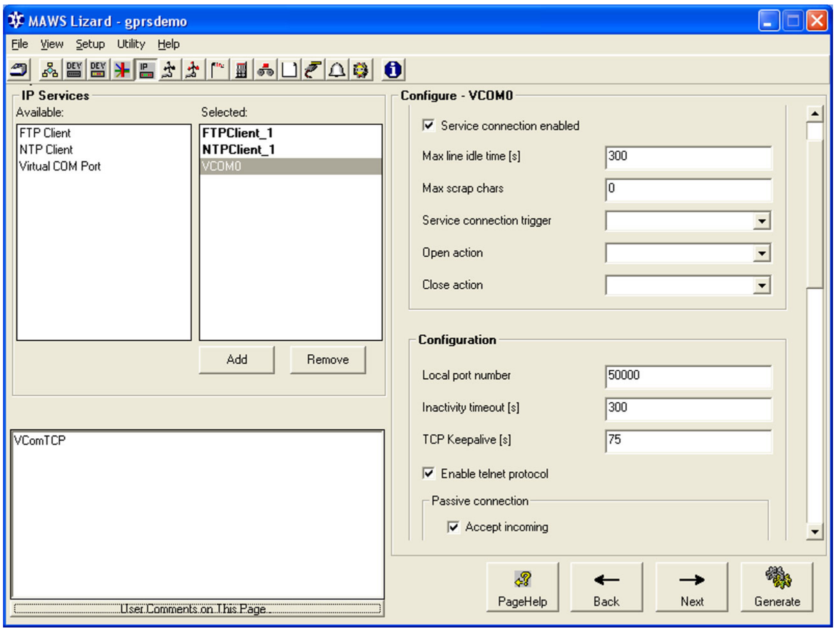

Figure 146 Configuring Virtual COM Port for Service Access

a. Add a virtual COM port by selecting **Virtual COM Port** in the **Available** list and clicking **Add**.

b. Configure the following items for the virtual COM port:

- Select the **Service connection enabled** option.
- Select the **Passive connection**, **Accept incoming** option.

6. In the **IP Services** view, add and configure an SMTP client for alarm message transmittal.
   - Add an SMTP client.
   - Configure the following items for the SMTP client:
     - **Source domain**: Use an existing one, for example, a company domain. SMTP servers often reject messages from unknown domains.
     - **Destination server**: This server must be available on the GSM operator's network. For more information, see connection settings often provided on an operator's web pages.
     - **Primary interface**: **netif0**, for instance, using the GSM modem.
7. In the **Reports** view, create a report to be sent to the FTP server.
8. Create an alarm report to be sent as an e-mail alert.
9. In the **Communications** view, shown in Figure 147 on page 307, link the report to the previously created FTP client for transmission, and configure the transmission parameters.

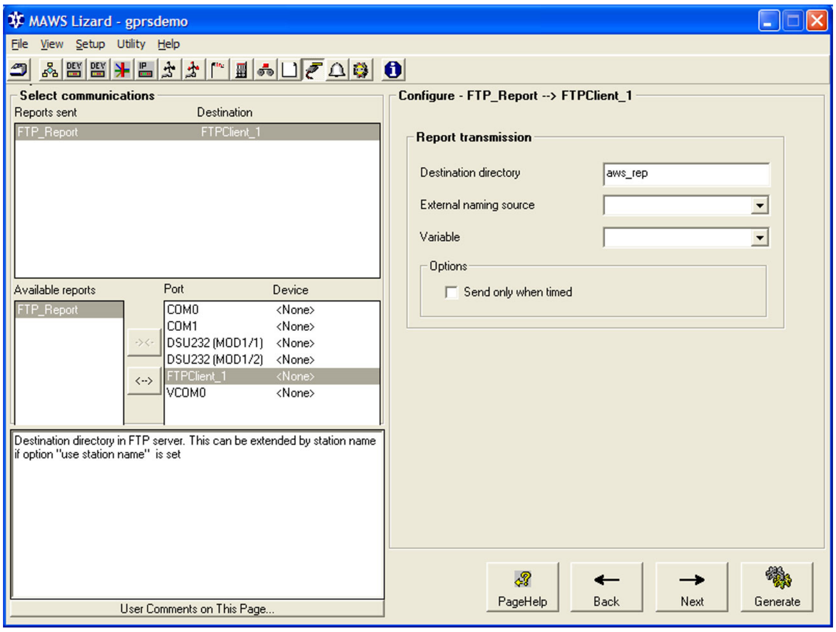

Figure 147 Configuring Report Transmission for FTP Client

a. Link the report to the FTP client by selecting the report in the **Available reports** list, the FTP client in the **Port/Device** list and then clicking the linking button.

b. Enter the directory to which reports will be sent.

- Usually there is a home directory specified for each FTP user. The directory defined here will be located under that directory. If the directory does not exist in the server, it will be created by the QML logger during the first transmission.
- This example has the option **Use station name** set for the FTP client. The final directory for the reports will be **aws_rep/<station name>**, where **<station name>** equals the name of the weather station as set in the MAWS common parameters or by the **sname** command.
- Link the **DC_Lost_Alert** to the SMTP-Client.
    - Enter the e-mail address where message will be sent to.
    - Select the **Use sname@domain as sender** option.

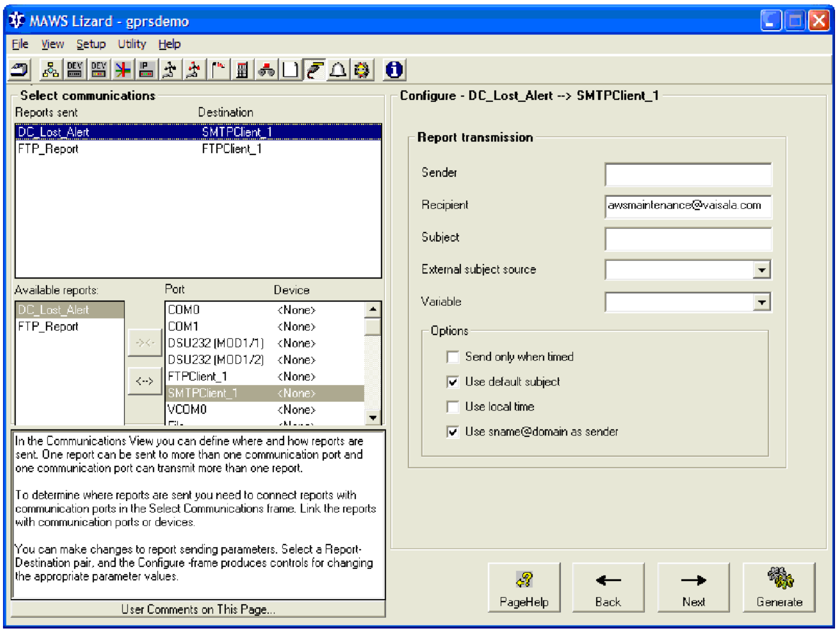

Figure 148 Configuring E-Mail Message Transmission

10. Add a **Create Report** alarm to monitor external DC supply, and connect the report **DC_Lost_Alert** as **Report on activation**.

11. In the **Timers** view, you can adjust the timers as needed. By default, the setup timer options are as follows:

    - Reports are generated and sent to the FTP server once a minute.

NOTE	The default interval may be a bit too fast for a GPRS connection, especially if the connection is dropped between transmissions. (Whether the connection is dropped depends on the modem, and it is often controllable by AT commands.)

    - The QML logger time is synchronized from the NTP server once an hour.
    - Check **External DC status** once per minute, and trigger an e-mail message if the voltage is under 6.0V

12. Your setup is now ready for testing. You can test the setup as follows:

    a. Observe the QML logger output to COM0. It will show the progress of and possible errors in the connections, FTP transfer, and other communications.

    b. Check that your reports appear in the FTP server directory at the scheduled intervals.

    c. You can test the NTP connection from the QML logger command line using the command **ntp test** <*servername*>, where <*servername*> is the NTP server name used in the NTP client configuration.

    d. You can set the QML logger time using the command **ntp set** <*servername*>.

    e. Establish a service connection to the QML logger. For instructions, see section Service Connection for Virtual COM Ports on page 271.

    f. For the external DC alarm test, the logger must have a charged battery installed. To test, remove the logger power supply from the POWER connector. Check that an e-mail is sent no later than 60 seconds after the disconnection.

Virtual COM Ports over Ethernet Connection

This example illustrates a setup for using virtual COM ports over an Ethernet connection provided by Ethernet Communication Module DSE101. The configuration includes the following items:

- Sending reports to a data acquisition system with a raw socket interface
- Reading data from a sensor via COM Server Module DXE421
- Accepting an incoming IP service connection

The network setup to be configured is illustrated in Figure 149 on page 310.

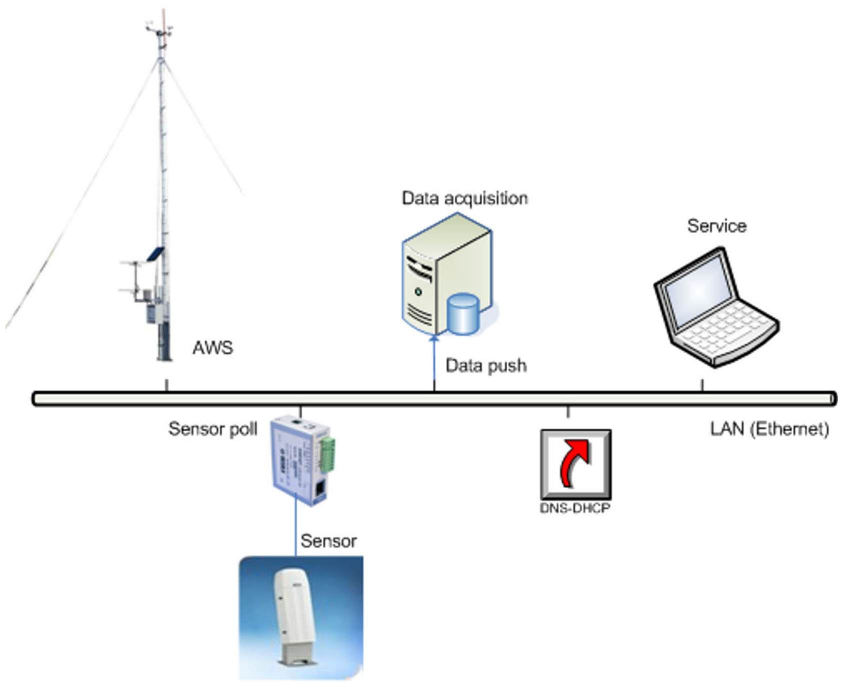

Figure 149 Weather Station with Ethernet Networking

1. Add the physical communication device to the setup.

   In the **Optional hardware** view, add Ethernet Communication Module DSE101 to an available module place.

   Note that DSE101 has no parameters to be set by the user.

2. Adding Ethernet module DSE101 to the setup automatically creates the corresponding communication interface. The interface is named **eth0** or **eth1**, depending on the module place used. See Figure 150 on page 311.

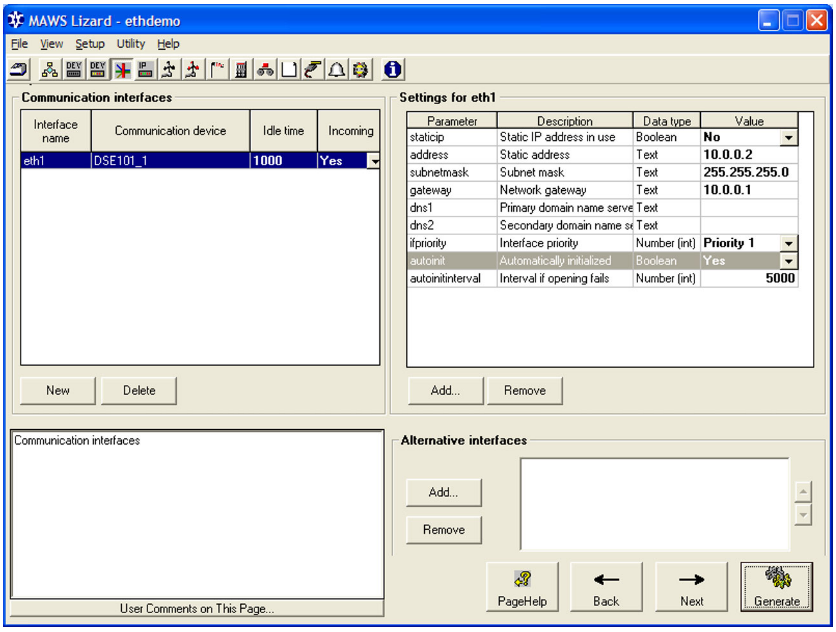

Figure 150 Network Interface Configuration for Ethernet Module

   a. Set the value of the **autoinit** parameter to **Yes**. The default network parameters are for DHCP operation, that is, with dynamic network provided IP address and other network parameters. Normally, there is no need to override these parameters in a LAN environment.

NOTE	Even if network parameters are dynamically assigned by DCHP, the lease time for an address can be long enough to allow at least testing. If this is not the case, that is, if the IP address is changed each time the QML logger is restarted, fixed IP addresses are required. To determine the IP address QML receives from the DHCP server, you need to open a service interface using a serial connection, and issue the command **ipconfig** to list the IP networking parameters.

3. In the **IP Services** view, add and configure a virtual COM port for report transmission to the data acquisition system.

   a. Remove the selection from the **Enable Telnet protocol** option.

   b. Set the value of the **Inactivity timeout** parameter to 10 seconds.

   c. Enable the active connection by selecting the **Active Connection**, **With any character** option. When this option is selected for the virtual COM port with default parameters, a dialog for configuring remote hosts is displayed automatically; see Figure 151 on page 312.

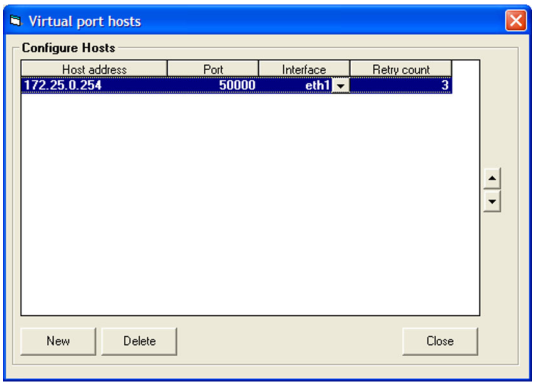

Figure 151 Configuring Remote Hosts for Virtual COM Ports

NOTE	The values shown do not represent any existing server connection. In order to have a functional system, the parameter values must be changed either in the setup or by changing the values in the parameter set **VCOM0**. The values of the parameter must represent the IP address and port of the destination server.

With this configuration, the virtual COM port will connect to the configured remote system as soon as it receives application data to be transmitted, that is, it begins to transmit a report. The connection will be automatically closed 10 seconds after the data has been sent.

NOTE	The connection to the remote system can be made automatically when a report is sent (active connection), or it is also possible to have the remote system connecting the QML logger for polling the reports (passive connection). This example setup uses an active connection.

4. Add and configure a virtual COM port for the sensor interface using DXE421. See Figure 152 on page 314.

NOTE	In this example configuration, the COM Server Unit DXE421 is only used for providing an Ethernet connection to the sensor. Factory settings can be used for the DXE421 module; its configuration is not changed in this example.

5. Add and configure a virtual COM port for the service connection.

NOTE	It is possible to use the same virtual COM port, for example, for transmitting reports and allowing an incoming service connection. This is, however, not recommended, since opening a service connection would prevent the reports from being transmitted. Also if a connection is already established for sending a report, it cannot be simultaneously established from somewhere else.

a. Select the **Service connection enabled** option.

b. Select the **Passive connection**, **Accept incoming** option.

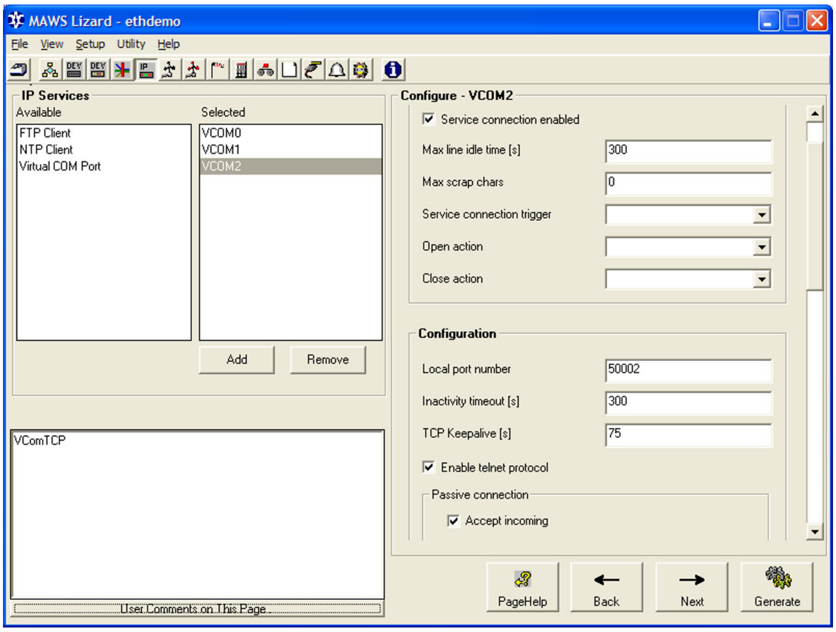

Figure 152 Configuring Virtual COM Port

6. In the **IP Services** view, add an HTTP server.

   - Leave the default HTTP server parameter values as they were.

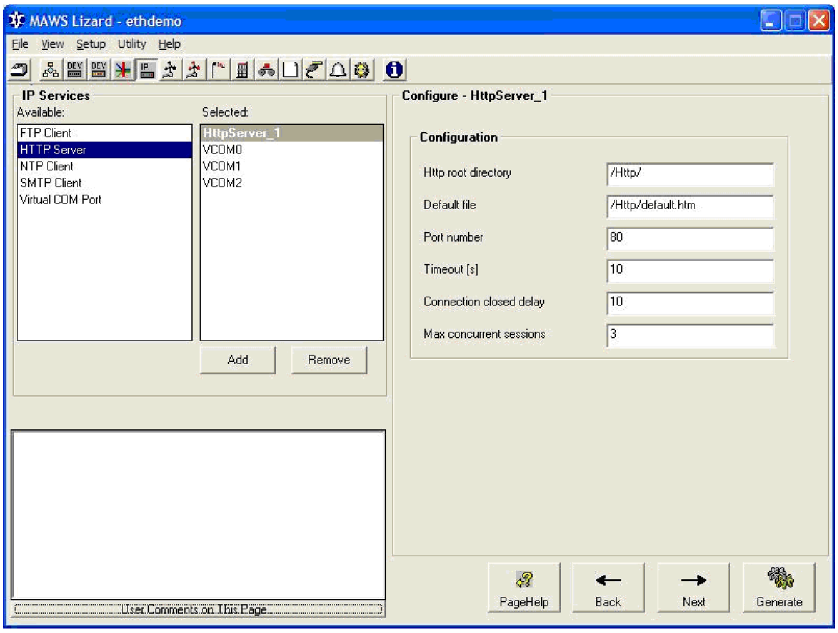

Figure 153 HTTP Server Configured

7. In the **Equipment** view, add a sensor and connect it to the virtual COM port created for the sensor interface. This example uses a ceilometer connected to DXE421 to provide the Ethernet connection. See Figure 154 on page 315.

   Sensors with an Ethernet interface are connected to virtual COM ports. From the QML application point of view, such a sensor is the same as it would be if connected to a serial port, except that instead of a physical serial port, the connection is made to a virtual COM port.

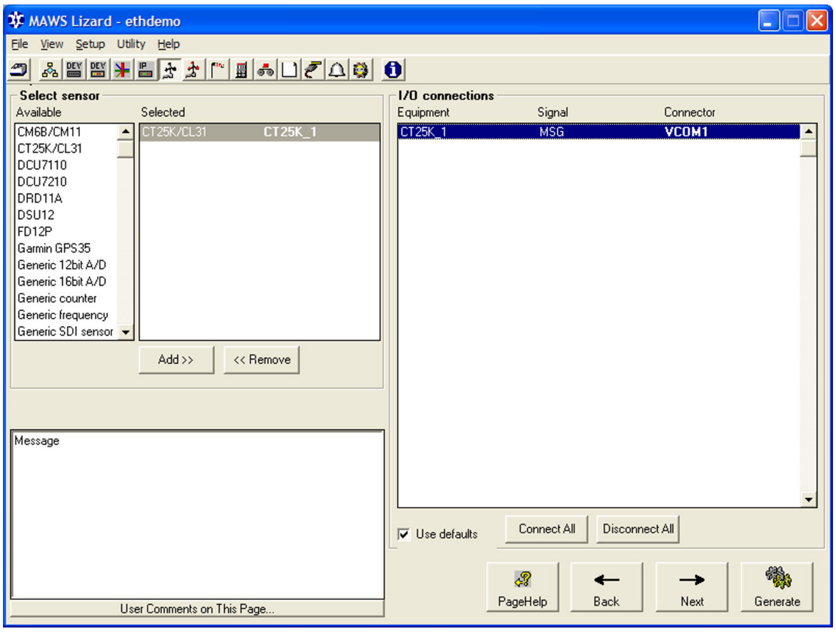

Figure 154 Ceilometer Connected to Virtual COM Port

NOTE	
	In this example, the QML logger is responsible for establishing the IP connection to the sensor. In such a case, it is essential that the virtual COM port is configured to open automatically (autostart) or that the sensor is operated in polled mode. If autostart is not active and no data is sent to the sensor port, no connection will be made. If the virtual COM port for the sensor is not constantly kept open, the poll timeout for the sensor may have to be increased to allow an additional delay caused by opening the connection. Even more time will be needed if the Ethernet module DSE101 has to be started and parameterized.

8. In the **IP Services** view, add and configure a virtual COM port for service access as shown in Figure 155 on page 316.

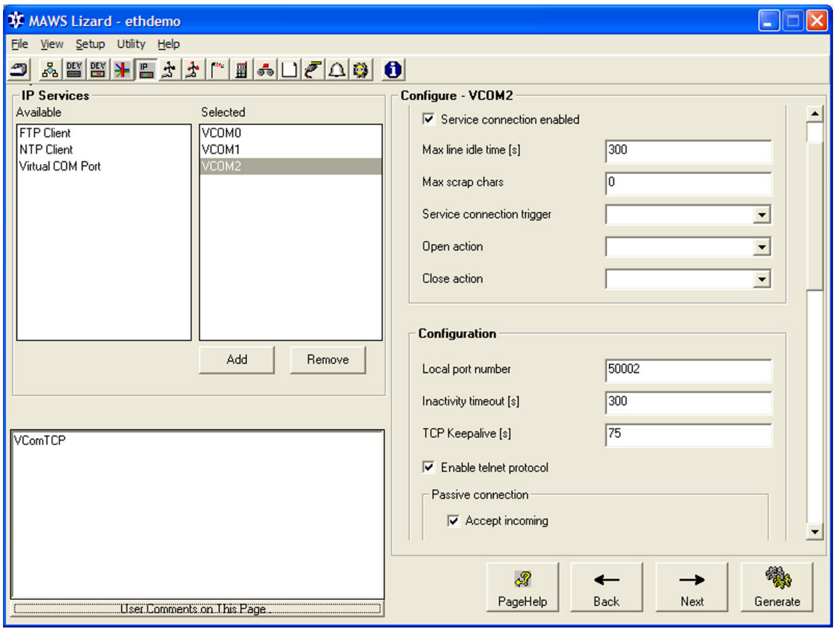

Figure 155 Virtual COM Port Configured for Service Connection

a. Add a virtual COM port by selecting **Virtual COM Port** in the **Available** list and clicking **Add**.

b. Configure the following items for the virtual COM port:

- Select the **Service connection enabled** option.
- Select the **Passive connection**, **Accept incoming** option.

9. In the **Reports** view, create a report and link it to the virtual COM port for sending data to the data acquisition system. See Figure 156 on page 317.

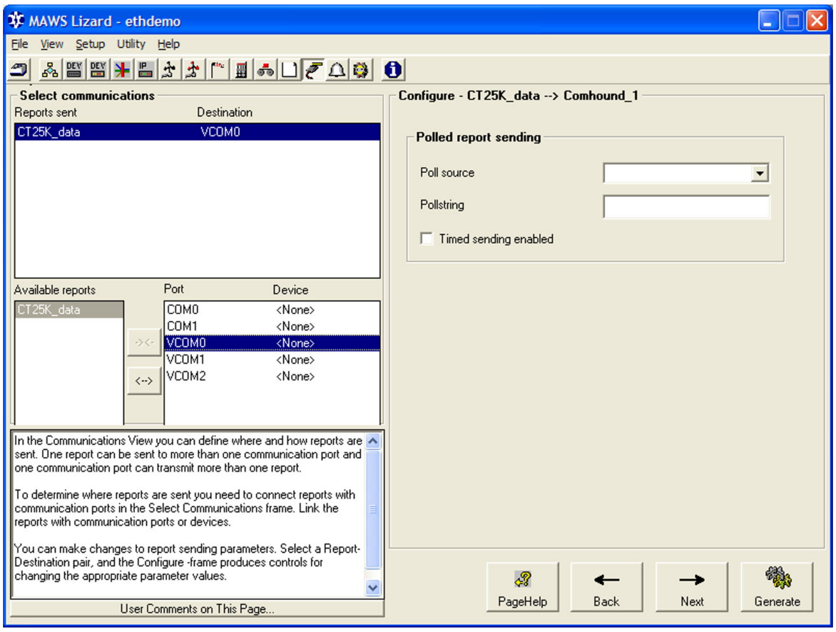

Figure 156 Linking Report to Virtual COM Port

From the QML application point of view, sending reports to virtual COM ports is the same as sending reports to a physical serial port.

10. Using **Custom Template**, create an HTML-formatted report:

   - Click the **Load ASCII Template** button and import template **QML201.html**; templates can be found in the same directory as the setup.
   - Link setup variables to the template as follows:

Table 68 Setup Variables Linked to HTTP-Formatted Report

Tag	Source	Variable
sname	SParServer	sname
base1	CT25K_1	base1
base2	CT25K_1	base2
base3	CT25K_1	base3
vvis	CT25K_1	v_vis
extdc	Charger	External_DC
warn	Sysinfo	warning
err	Sysinfo	errors

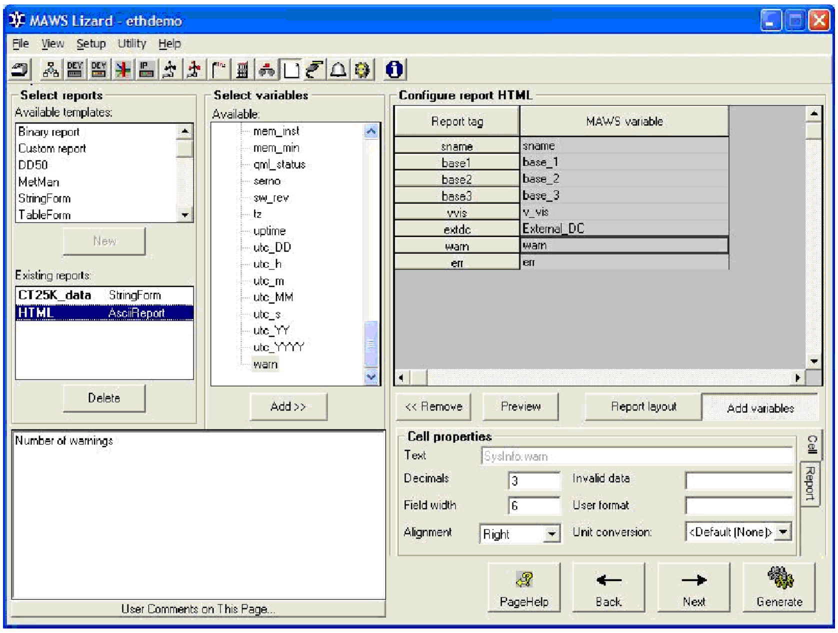

Figure 157 Linking Variables to HTTP Reports

11. Link the HTTP report to a file:

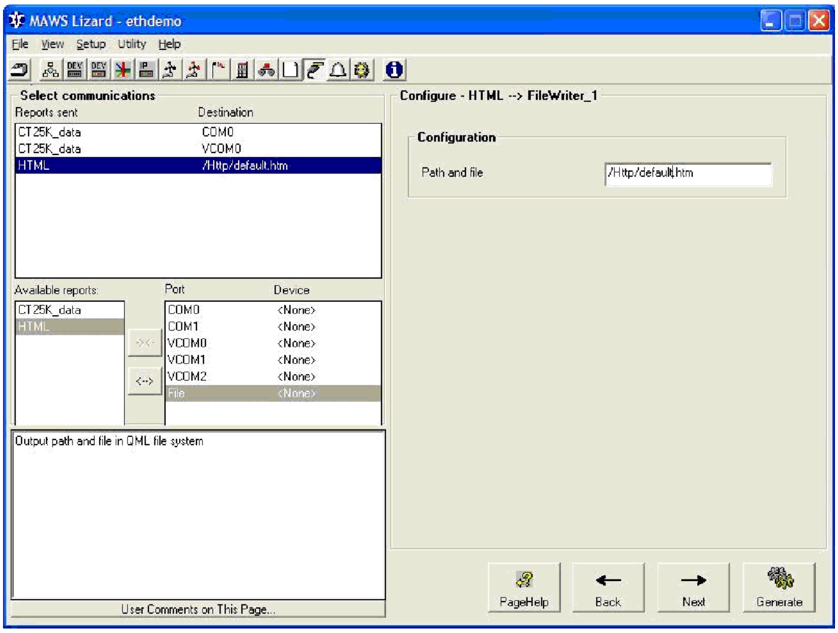

Figure 158 Linking HTTP Report to a File

12. Your configuration is now ready for testing. You can test the setup as follows:

    a. If you are not using a fixed IP addresses, determine the QML logger IP address by opening a service connection using COM0 and then issuing the command **ipconfig**.

    b. Verify that sensor data is being received using COM0.

    c. Check from the data acquisition system that reports are coming through.

    d. From your PC, open a service connection to the QML logger IP address, port 50002. For further instructions, see section Service Connection for Virtual COM Ports on page 271.

PPP over RS-232

This advanced example illustrates a setup for connecting the QML logger to a Windows PC using PPP over a serial (RS-232) line. This interface is used for the following:

- Transmitting reports with user configurable naming to FTP server
- Transmitting log files to FTP server
- Providing a service connection using a virtual COM port

The network configuration is shown in Figure 159 on page 319.

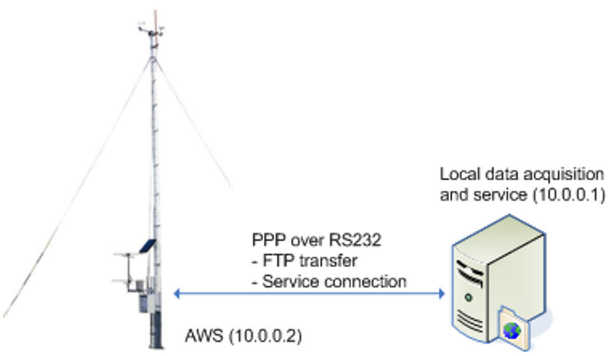

Figure 159 Weather Station with PPP Link over RS-232 Connection

The IP services, FTP server, and service terminal, reside in the local data acquisition system. The benefits of using a serial line IP connection in this kind of configuration are:

- Standard file transfer method
- Possibility to establish multiple simultaneous links over one serial line

NOTE	Half-duplex links, such as two-wire RS-485, are not suitable as PPP carriers with the QML logger. Whenever possible, using Ethernet should be preferred.

1. Configure the PPP connection in the local data acquisition system with the parameters presented in Table 69 on page 320. Further information on creating dial-up/in interfaces in Windows PCs and configuring their parameters can be found in Windows online help.

Table 69 PPP Connection Parameters in Remote Data Acquisition System

Parameter	Description
Link parameters	9600 bps, 8 data bits, 1 stop bit, no parity, no flow control
Local IP address	10.0.0.1
Remote IP address	10.0.0.2
Authentication method	PAP
PPP dial-in user name	pppuser
PPP dial-in password	userppp
PPP dial-out user name	qmluser
PPP dial-out password	userqml

2. Add and configure the physical communication devices in the setup.
   a. In the **Optional hardware** view, add module DSU232 to an available module place.

NOTE	Using link speeds over 9600 bps is not recommended.

   b. In the **Devices** view, add the device **WinNullModem-IP**.

NOTE	If the local data acquisition system is running an operating system other than Windows, use the device **NullModem-IP** instead.

c. Connect the **WinNullModem-IP** device to a channel on DSU232 as shown in Figure 160 on page 321.

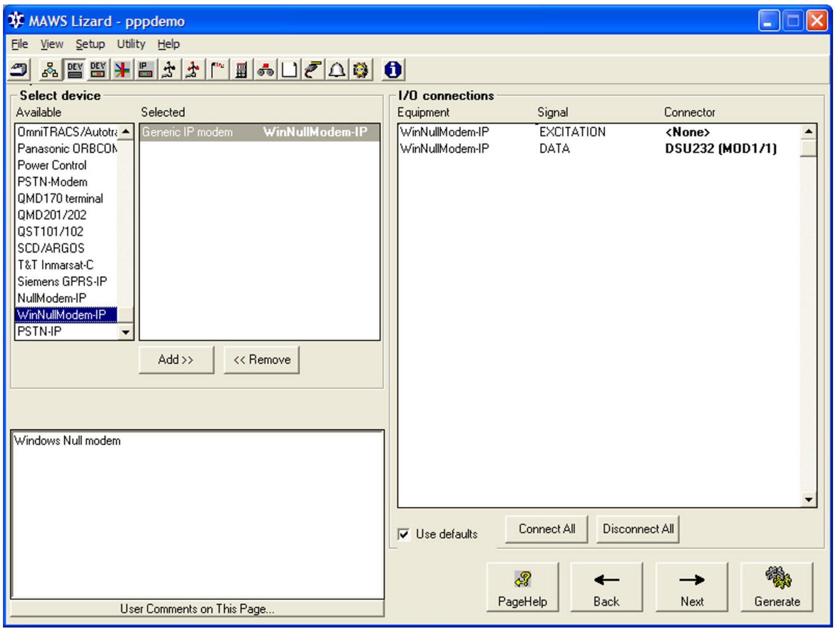

Figure 160 Connecting WinNullModem-IP Device

d. In the **Device configurations** view, enable incoming calls to **WinNullModem-IP** by selecting the **Answer incoming** option.

3. In the **Communication interfaces** view, create and configure a logical interface for the Windows Null modem. The parameters to configure are outlined in Table 70 on page 322.

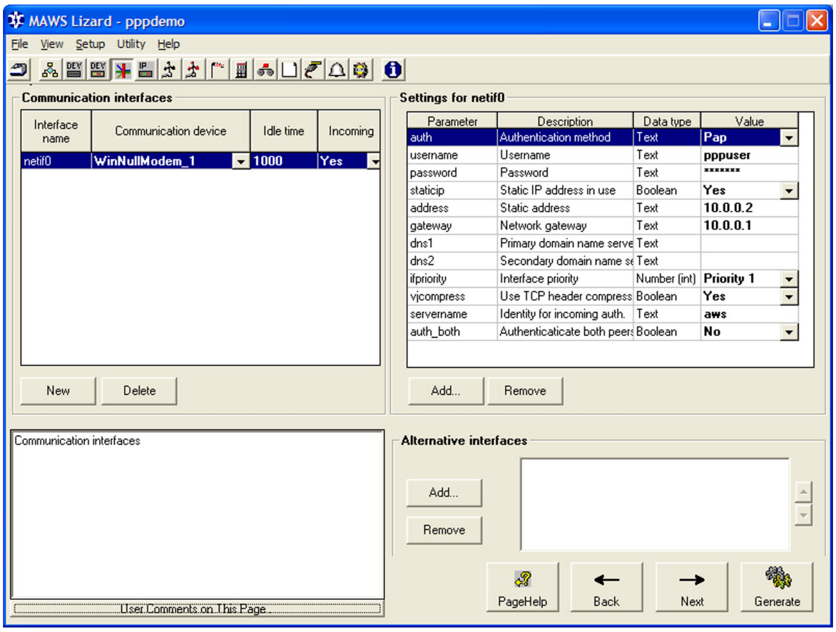

Figure 161 Configuring Null Modem

Table 70 Null Modem Parameters

Parameter	Value
Incoming	Yes
auth	PAP
username	User name available for logging in to the data acquisition system using PPP dial-in. Use **pppuser** as value.
password	Password for the user name. Use **userppp** as value.
staticip	Yes
vjompress	Yes
servername	Name seen by external devices when dialing in to the QML logger

NOTE	In this configuration, the QML logger has no domain name service configured, so it is not capable of accessing IP services by name, but only by using IP addresses. If it is necessary to access services by name, at least one DNS server (**dns1**) must be available and configured in the QML logger. DNS settings can be obtained from a Windows PC by opening a command prompt and issuing the command **ipconfig /all**.

4. In the **IP Services** view, create and configure FTP clients for transferring reports and log files. See Figure 162 on page 323.

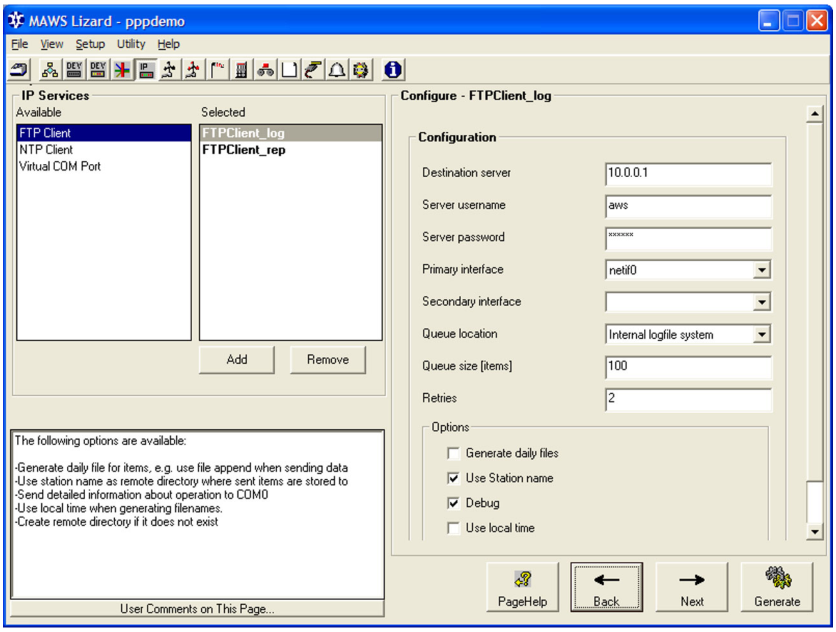

Figure 162 Configuring FTP Clients

a. Add an FTP client by selecting **FTP Client** in the **Available** list and clicking **New**.

b. Rename the FTP client as **FTPClient_rep**.

c. The parameters to be configured for **FTPClient_rep** are outlined in Table 71 on page 323:

Table 71 FTP Client Parameters for Sending Reports

Parameter	Value
Destination server	10.0.0.1
Server username	User name in the FTP server
Server password	Password for the configured user name
Primary interface	netif0
Queue location	Internal logfile system
Options, Generate daily files	Selected

d. Add another FTP client and rename it as **FTPClient_log**.

e. Make the same parameter settings for **FTPClient_log** as for **FTPClient_rep**. The option **Generate daily files** has no effect when used with log files, so it can be left not selected.

5. In the **IP Services** view, create and configure a virtual COM port for service access.

   a. Select the option **Service connection enabled**.

   b. Select option **Passive connection**, **Accept incoming**.

6. In the **Logging** view, configure the logs to be transmitted to the FTP server. See Figure 163 on page 324.

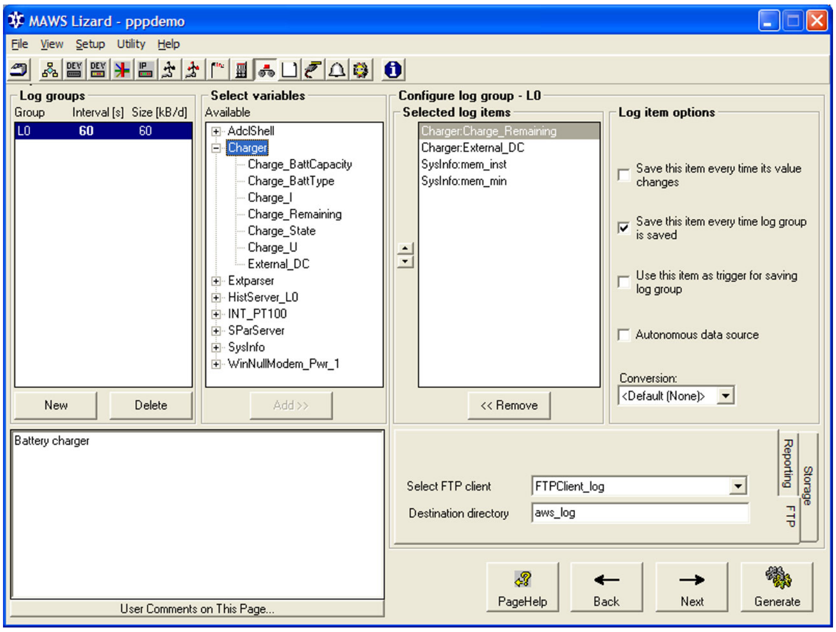

Figure 163 Configuring Log File Transfer

In the **FTP** tab, enter the following parameters:

- **FTP client**: FTPClient_log
- **Destination directory**: FTP server directory to which the log files are to be stored. This directory name will be followed by a directory named according to the station name.

Note that the log files are transferred once a day, two minutes after midnight.

7. In the **Reports** view, create a report to construct user defined file names for storing observation data on the FTP server.

   a. Create a **StringForm** report named, for example, **FTP_Name**.

      The example report illustrated in Figure 164 on page 325 generates files that are organized in subdirectories by year and month. The path for a daily file is of the format

      maws_rep/stationname/2007/10/obs_20071011.txt.

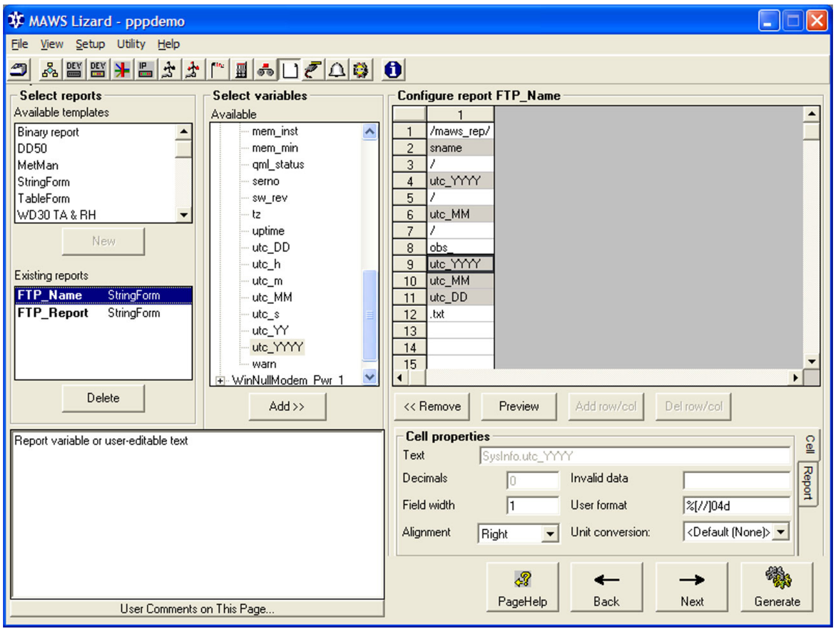

Figure 164 Report for Constructing FTP File Name

   b. This report must be triggered at least as often as the file name on the FTP server needs to be changed. It is recommended that you set the report to be triggered as often as the transferred report, so that any changes in time or station name will be reflected immediately when the next report is being sent. You can set the interval for triggering the report by changing **Interval** parameter for the report in the **Timers** view.

8. In the **Communications** view, configure the report-specific FTP transmission options. See Figure 165 on page 326.

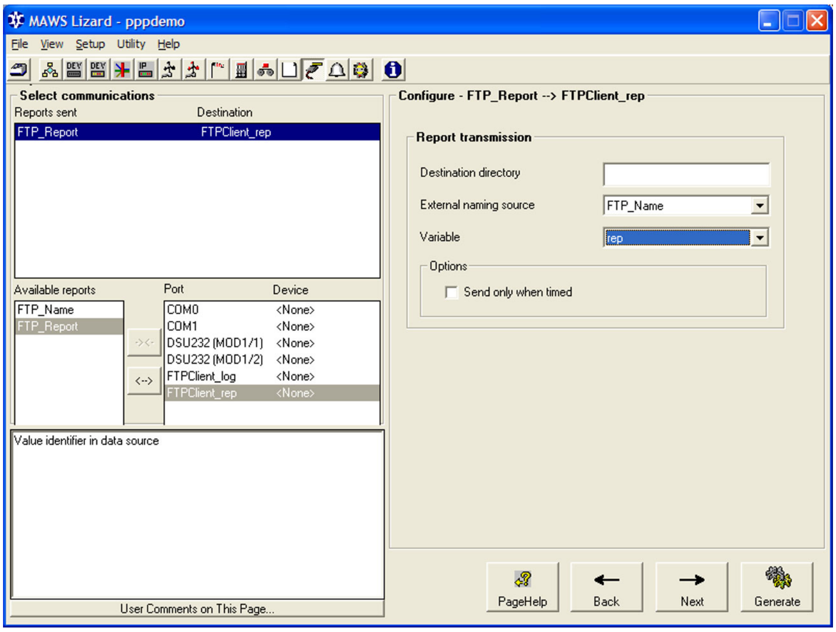

Figure 165 Using External Naming Source for Report Transfer

a. Link the report to be transmitted to the port **FTPClient_Rep**, that is, in this case the report **FTP_Report**.

b. Do not specify any destination directory, but instead select an **External naming source** and **Variable** from which the FTP file name will be read. Select the report **FTP_Name** as the naming source and **rep** as the variable.

9. Add user credentials for dialing in to the QML logger by selecting **Parameter Sets** from the **Setup** menu. See Figure 166 on page 327.

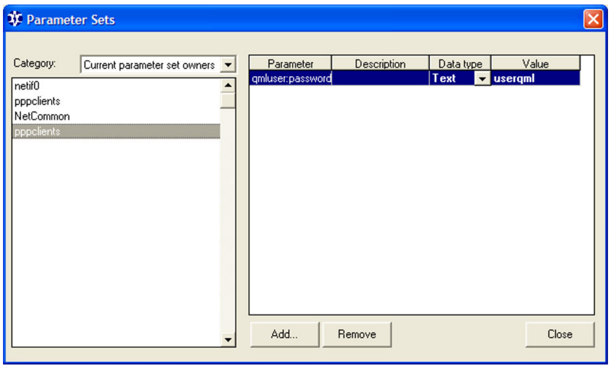

Figure 166 Parameter Set for Dial-In User Credentials

a. Add a new parameter to the pre-defined group **pppclients**.

b. Name the parameter set using the format **<username>:password**, where **<username>** is the user name to be used and **password** the fixed key for the user's password not to be changed.

c. Select **Text** as the parameter type in the **Data type** menu.

d. Enter the user's password as the value of the parameter in the **Value** field.

10. Your setup is now ready for testing. You can test the setup as follows:

a. The daily file containing report data is updated every five minutes.

b. You can access the QML logger service interface from the data acquisition PC by connecting to IP address **10.0.0.2**, port **50000**.

IP Service Connectivity with GSM

In GSM networks, IP connections from the Internet to the cellular terminals are often blocked by operator firewall, and terminals may even be provided with private IP addresses. These restrictions make the terminal devices inaccessible from the Internet when, for example, establishing an IP service connection to AWS site is impossible.

NOTE	Restrictions are operator-dependent. Operators provide additional services, such as private APNs, with different access and addressing schemes.

A "wakeup" SMS can be used to trigger the logger to establish an IP service connection via GSM terminal to a PC running AWS Client software.

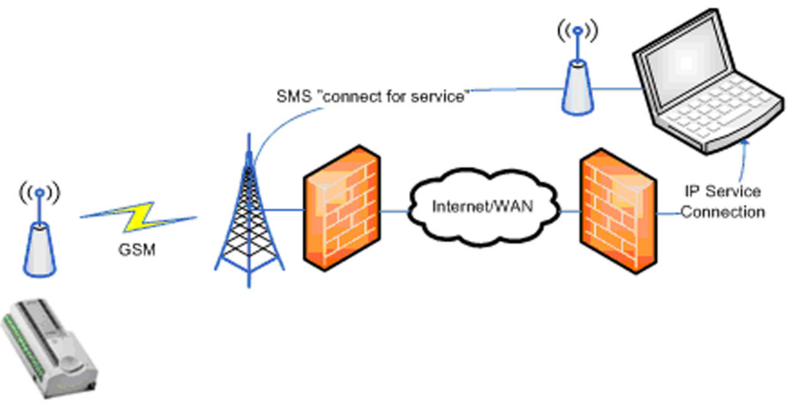

Figure 167 Using SMS to Establish IP Service Connection

NOTE	To use this feature, the PC running the AWS Client software needs a GSM modem for sending the SMS and a public IP address, either static or dynamic DNS -hosted, for accepting the incoming connection. Also, it is required that a port in the user organization firewall be opened for the incoming connection to pass to the destination PC.

By default, the logger does not process incoming SMS. For instructions on how to enable SMS handling, see section SMS Handler on page 289.

An example configuration for SMS Handler and Virtual COM port used for service connection is as follows:

1. Configure the *SMS handler* to open a connection using VCOM when SMS with content *open conn1* is received.

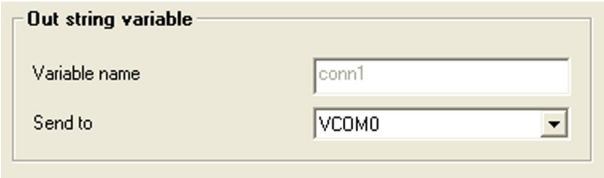

Figure 168 SMS Handler VCOM

2. Define the remote connection point, that is, IP address and port for connecting the AWS Client.

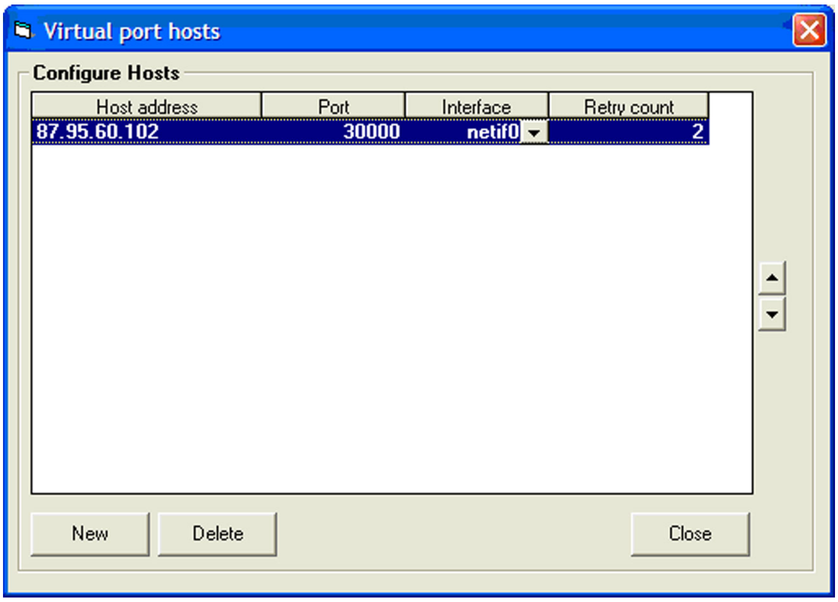

Figure 169 Configuring Remote Connection Using Virtual COM

3. To enable SMS-triggered service connection using Virtual COM, the following additional settings must be used:

   - Service connection enabled: Selected
   - Active connection: With any character

The following procedure describes how to configure AWS Client to use SMS when connecting to the logger:

1. Open menu **Settings - Address book**.
2. Click **New** to create a new Address Book entry.

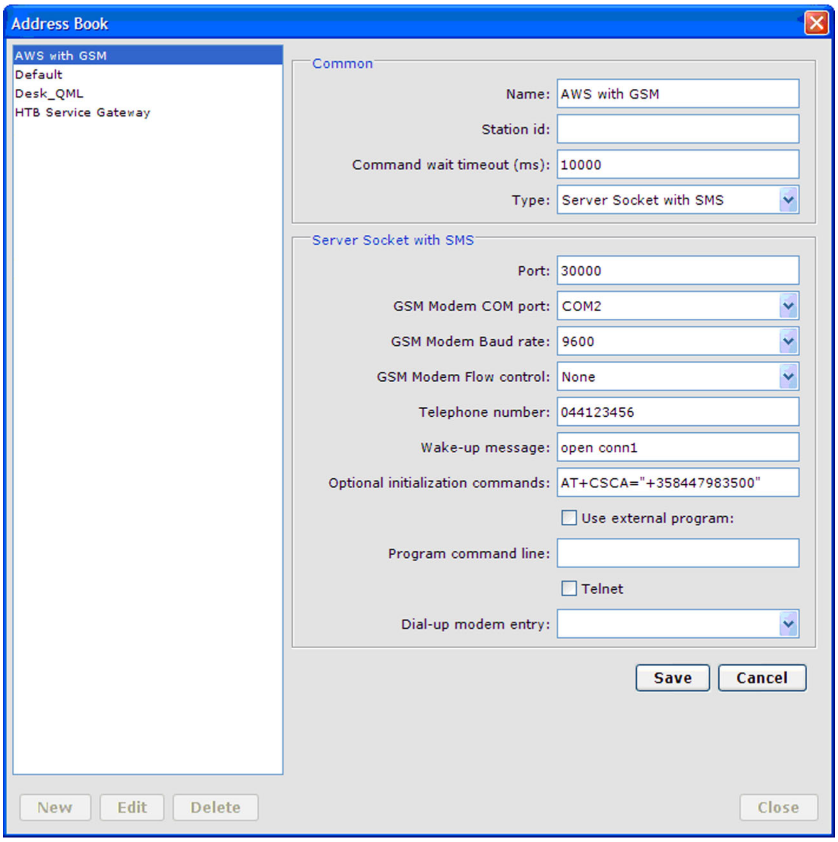

Figure 170 Entering Parameters for SMS Triggered IP Connection

3. Enter the following parameter values (only the parameters relevant for SMS configuration are listed):

Table 72 SMS Configuration Parameters

Parameter	Value	Note
Type	Server socket with SMS.	
Port	Port where the logger is configured to connect to.	For more information, see section SMS Handler on page 289.
GSM Modem COM port	PC COM port where the GSM modem is connected to.	
GSM Modem baud rate	Baud rate for the GSM modem.	
GSM Modem flow control	Flow control mode for the GSM modem.	
Telephone number	Telephone number of the logger's GSM modem.	
Wake-up message	*open <connX>*, where <connX> refers to the logger SMS handler configuration	For more information, see section SMS Handler on page 289.
Optional initialization commands	Additional initialization commands for the modem.	For example, AT+CPIN="XXXX" AT+CSCA="<sms central number>"

4. Click **Save** to close the settings.

It is also possible to use an external program to execute the sending. In this case, select **Use external command** and enter startup command and parameters for launching the external program to **Program command line**. In this case, neither the modem settings nor **Wake-up message** are used.

NOTE	Connecting to logger with this method can involve significant delays, depending on how fast the SMS gets through and the logger manages to establish a GSM connection for IP.

Troubleshooting TCP/IP-Based Telemetry

Typical possibilities for troubleshooting TCP/IP-based telemetry operation with the QML logger are as follows:

NOTE	This manual will not attempt to give general guidelines on network design or troubleshooting issues, such as IP addressing. Troubleshooting IP networking usually also involves issues not related to the QML logger or its configuration; this section includes only the troubleshooting possibilities at the QML logger end.

1. Use the **warnings** command to check for any modem-related warnings.
2. Use the **net warnings** command to check for any IP-specific warnings.
3. Use the **ipconfig** command to see the current IP configuration in the QML logger. The command output gives you information on all open network interfaces and their configuration, such as IP addresses, gateways, and DNS servers.
4. Use the **netif** command to see whether the network interface you are trying to use is open. If required, the interface can be opened with the command **netif** *open <interface name>*.
5. Check that the IP addresses used for the QML logger interfaces are correct with respect to your overall network configuration.
6. Use the **ping** command to test the connection to the destination IP address or host. If the destination cannot be reached, you can test whether the connection to the gateway works; if it does, the problem might be in the connection between the gateway and the destination host.
7. For the FTP and NTP services, use the commands **ftp** *test <server>* and **ntp** *test <server>* to check the connection to the server. You can also use the **ping** command to check whether the server is reachable.
8. As applicable in your network, ensure that there are no firewalls blocking the traffic to/from the QML logger.

9. To monitor different command sequences, check the **Extra op. Info to COM0** or **Debug output** option when configuring a device. This enables the device control software to output various status information to the fixed RS-232 port, that is, to COM0. For example, you can monitor what is sent to the modem and how it responds. The printout includes additional internal debug data. The output is sent only when the service connection is closed.

10. To send the AT commands manually, the command interface of the modem can be directly accessed. To control the modem directly, open the service connection to the QML logger. For example, when the device is connected to the first DSU232 communication port at the module place MOD1, type **open** *DSU232_0_0*. To terminate this operation, type **close**. While the direct connection to the modem or sensor is open, any automatic operation through the connected port is blocked. Typical parameters for the open command are presented in Table 73 on page 333. Information concerning the correct connector can be obtained in Lizard in the **I/O Connections** frame of the **Equipment** view.

Table 73 Parameters for the Open Command

Connector in Lizard	Parameter	Alias
COM0	COM0	COM0
COM1	COM1	COM1
DMX501 (MOD1/1)	DMX501_0	MOD1
DMX501 (MOD2/1)	DMX501_1	MOD2
DSI485A (MOD1/1)	DSI485_0	MOD1
DSI485A (MOD2/1)	DSI485_1	MOD2
DSI486 (MOD1/1)	DSI485_0_0	MOD1_2
DSI486 (MOD1/2)	DSI486_0_1[1]	MOD1_1
DSI486 (MOD2/1)	DSI486_1_0	MOD2_2
DSI486 (MOD2/2)	DSI486_1_1[1]	MOD2_1
DSI486SDI (MOD1/3)	DSI486SDI_0	MOD1_3
DSI486SDI (MOD2/3)	DSI486SDI_1	MOD2_3
DSU232 (MOD1/1)	DSU232_0_0[2]	MOD1_1
DSU232 (MOD1/2)	DSU232_0_1	MOD1_2
DSU232 (MOD2/1)	DSU232_1_0	MOD2_1
DSU232 (MOD2/2)	DSU232_1_1	MOD2_2
DSU232SDI (MOD1/3)	DSU232SDI_0	MOD1_3
DSU232SDI (MOD2/3)	DSU232SDI_1	MOD2_3

1. With the dual RS-485 module, the RS-232 connection is possible only to the channel B on the module, and thus the last number is 1.
2. With the RS-232 and dual RS-485 modules, the number between the underline characters stands for the module place, that is, MOD1 or MOD2, and the last number for the channel on that particular module.

NOTE	Use the **SYSINFO PORTS** command to find out which connectors to use.

11. It is impossible to have default parameters that would work everywhere with all different network infrastructures. If the default parameters do not seem to work, you should try manually, for example, by using a PC terminal program, to establish a connection between the weather station and the data collection system, and to find out the correct parameters. Also note that the default parameters are seldom optimal, for example, in minimizing the connection time.

CHAPTER 6
TECHNICAL SUPPORT

This chapter informs you how to contact Vaisala HelpDesk.

When contacting Vaisala technical support, please send the following information with your technical enquiry or description of a fault:

- Serial number of the QML logger.
- The captured text of the **SYSINFO** command.
- If you have modified the setup file and the setup is possibly defected, please send also the captured yyMMddHHmmSS.log file and the setup file (.dtg). Please refer to Vaisala HydroMet™ Data Collection Platform User's Guide, Volume 2, for instructions on how to export a setup file.

For technical questions, contact the Vaisala technical support:

E-mail	helpdesk@vaisala.com
Fax	+358 9 8949 2790

If the product needs repair, please follow the instructions below to speed up the process and to avoid extra costs to you.

1. Read the warranty information.
2. Contact Vaisala technical support via e-mail or fax and request for RMA (Return Material Authorization) and shipping instructions.
3. Proceed as instructed by Vaisala technical support.

NOTE	RMA must always be requested from Vaisala technical support before returning any faulty material.

CHAPTER 7
TECHNICAL DATA

This chapter provides technical data for the modules included in this manual.

Wiring Diagrams

CAUTION	All wiring diagrams in this chapter are examples only. Refer to the appropriate delivery documents for the actual wiring.

Do not change the wiring between the connectors and the logger pins. For special deliveries or with some sensors, a separate wiring diagram is supplied in order to help you connect the sensor wires to the correct connectors.

Communication Modules

RS-232 Module DSU232

The unisolated RS-232 module provides either a double serial channel without handshaking or a single RS-232 with handshaking. In addition, an optional SDI-12 interface is available.

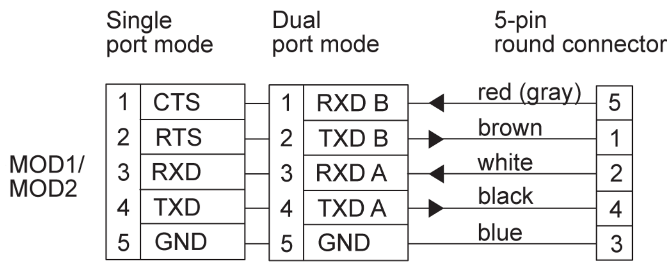

Figure 171 RS-232 Wiring Diagram

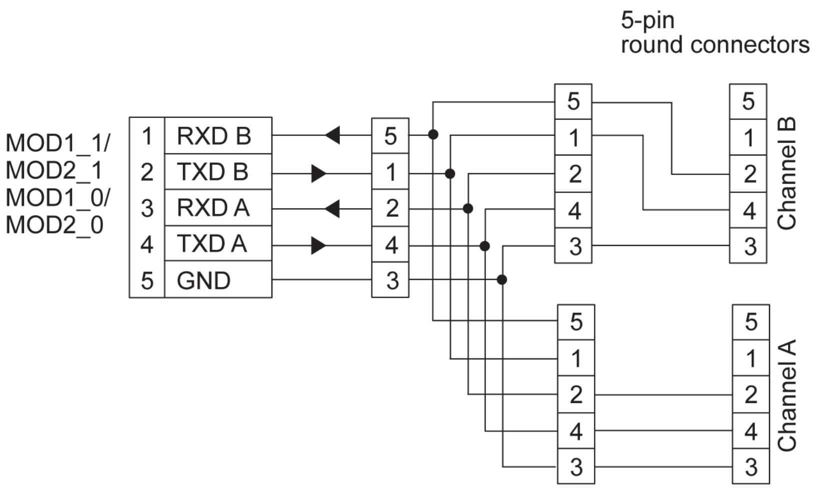

Figure 172 Suggested T-Connection in Dual Port Mode

For the SDI-12 wiring with the later version, see Figure 178 on page 343. The SDI-12 data pin is the same as in the dual RS-485 module, and the same wiring method can be used as in Figure 178 on page 343.

See Figure 178 on page 343 and Table 74 on page 339 for the SDI-12 jumper of DSU232-C.

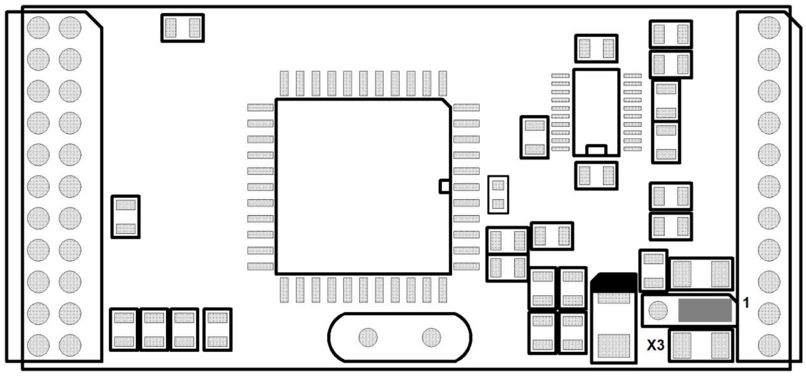

Figure 173 SDI-12 Jumper of DSU232-C

Table 74 DSU232-C Jumper Settings

Jumper	Connected Pins	Function
X3	1-2	Pin 5 is GND (default, compatible with the old DSU232)
	2-3	SDI-12 connected to pin 5

Isolated RS-485 Module DSI485

The isolated RS-485 communication module can be configured either for a 2-wire line or for a 4-wire line when the receive and transmit lines are separated. If the module is configured for a 2-wire line, the transmitter is enabled only during the transmission. Normally, the 2-wire connection is used to connect several devices to the same communication line. The 4-wire mode is the default mode.

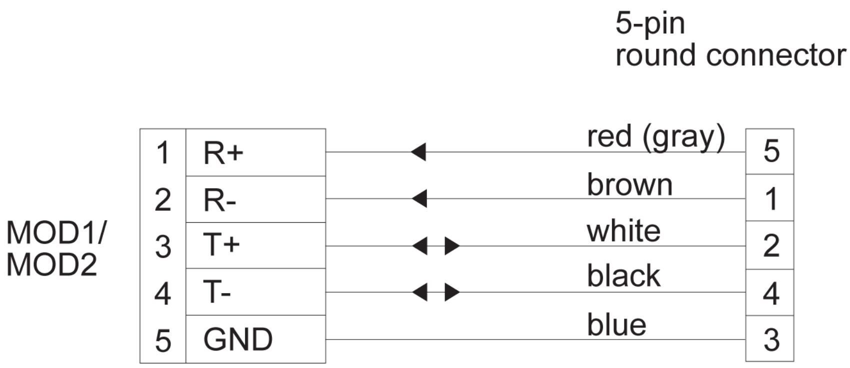

Figure 174 Isolated RS-485 Module Wiring Diagram

NOTE	In 2-wire mode, only T+ and T- pins are used.

Dual RS-485 Module DSI486, Version A

Channel A is always used in the RS-485 mode. In 2-wire RS-485, both transmitted and received data is sent via this channel. In 4-wire RS-485 this channel can either transmit or receive depending on the configuration. Jumper X4 defines the line terminating resistor for the data channel A. Remove the jumper X4 if you do not need the terminating resistor of the dual RS-485 module. Figure 175 on page 341 provides a schematic wiring diagram for dual RS-485.

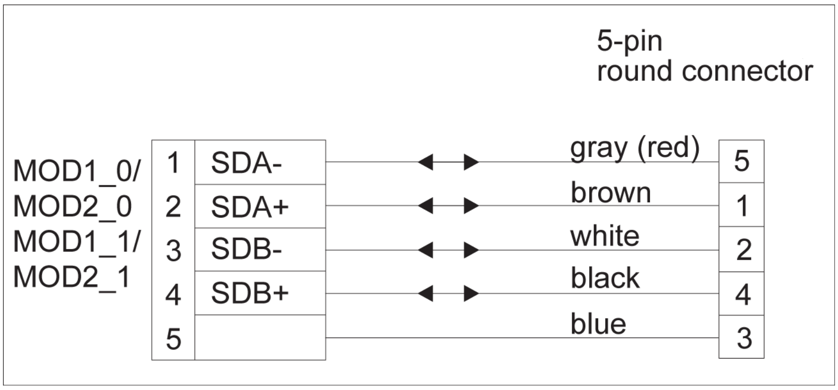

Figure 175 Dual RS-485 Module Wiring Diagram

Channel B can be used either in the RS-485 mode or in the RS-232 mode. In 2-wire RS-485, both transmitted and received data is sent via this channel. In 4-wire RS-485, this channel can either transmit or receive depending on the configuration.

Figure 175 on page 341 provides a schematic wiring diagram for the dual RS-485 connection, the dual 2-wire connection utilizing both channels. The correct jumper settings for the channel B are listed in Table 75 on page 341. The jumpers are located on the module as illustrated in Figure 176 on page 342.

Table 75 Jumper Settings for Channel B in the RS-485 Mode

Jumper	Connected Pins	Function
X3	1-2	Sets the RS-485 mode active for the channel B.
	3-4	
X6	1-2	
X5	1-2	The line terminating resistor is in use with RS-485.

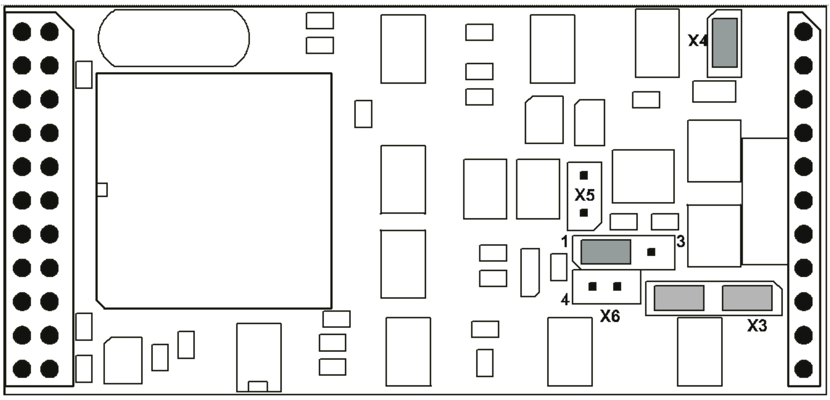

Figure 176 Dual RS-485 Module Default Jumper Locations

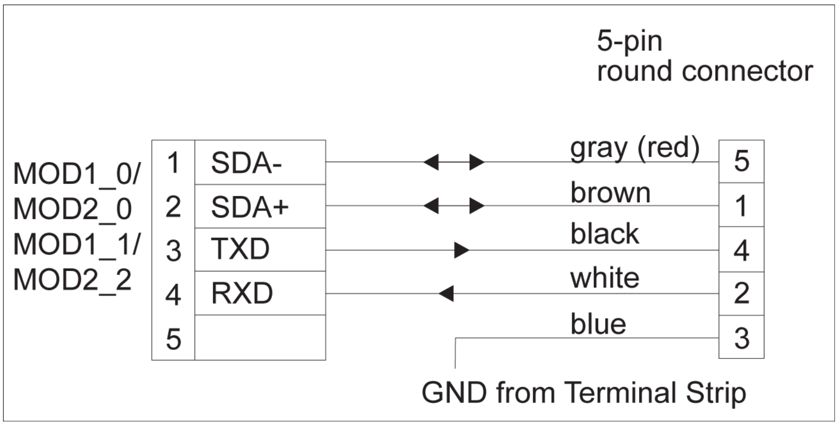

Figure 177 Dual RS-485 Wiring Diagram for RS-485 and RS-232

Figure 177 on page 342 provides a schematic wiring diagram for the combination of the RS-485 and RS-232 connection. The correct jumper settings for the channel B are listed in Table 76 on page 342.

Table 76 Jumper Settings for Channel B in the RS-232 Mode

Jumper	Connected Pins	Function
X3	2-3	Sets the RS-232 mode active for the channel B.
X6	1-4	
	2-5	
X5	None	The line terminating resistor is not in use at all.

The dual RS-485 module also provides an SDI-12 connection. The SDI-12 line uses one wire for data and is limited to a maximum length of 60 meters. Figure 178 on page 343 provides a schematic wiring diagram for the SDI-12 connection and the 12 VDC power supply for a sensor. The jumper settings should be as shown in Figure 176 on page 342.

Simultaneously with the SDI-12, you can connect channels A and B in the 2-wire RS-485 mode. If you take all three channels in use, you either need three free connectors in the flange or an optional junction box.

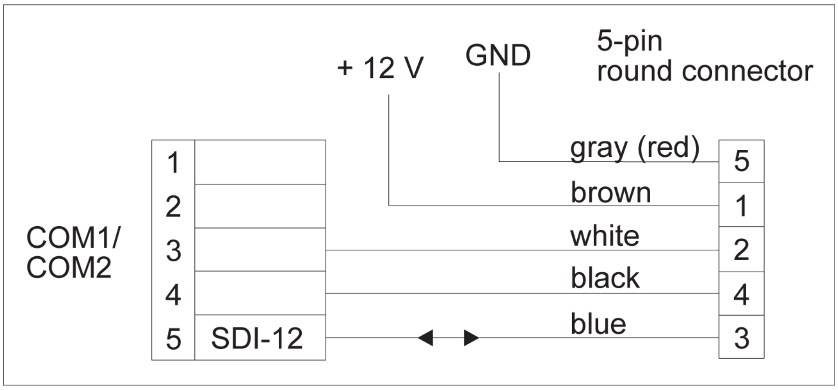

Figure 178 Dual RS-485 Wiring Diagram for SDI-12 and 12 VDC Power Supply

Dual RS-485 Module DSI486, Version B

The later module version, DSI486-B, has otherwise the same features as module DSI486-A, but its jumper settings are slightly different. The settings have been simplified to make the module easier to use. Notably, jumpers X4 and X5 still select the termination resistors for channels A and B, respectively, only their locations have been moved a bit (see Figure 179 on page 344).

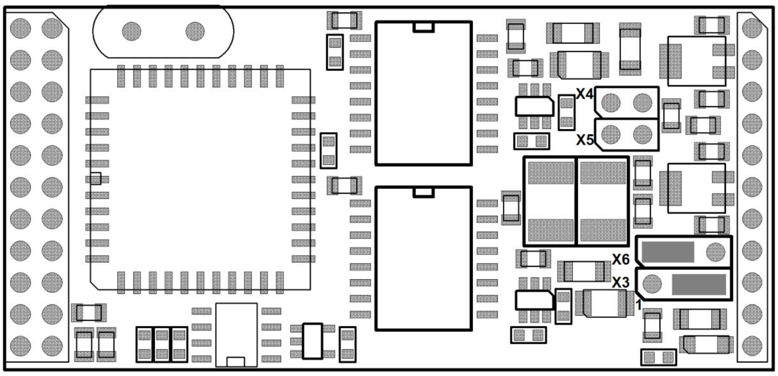

Figure 179 RS-232 Jumper Settings

Jumpers X3 and X6 are used to select between the RS-485 and RS-232 modes for channel B, but their settings have been simplified. The jumper positions are described in Table 77 on page 344.

Table 77 Jumper Settings for DSI486-B

Jumper	Connected Pins	Function
X3	1-2	Channel B RS-232 mode
	2-3	Channel B RS-485 mode (default)
X6	1-2	Channel B RS-485 mode (default)
	2-3	Channel B RS-232 mode
X4	1-2	Channel A RS-485 line terminating resistor active
X5	1-2	Channel B RS-485 line terming resistor active. Do not use in Rs-232 mode

Digital I/O Module QMI118

The QML logger provides the possibility to extend the I/O capacity with the optional digital I/O module (QMI108 or QMI118) equipped with eight inputs and eight outputs. The digital I/O module interfaces to the logger via the SPI connector. The module is located inside the enclosure beside the logger, and it conforms to the same environmental immunity and emission standards as the logger.

The module has eight open collector outputs with 30 VDC / 1 A continuous drive capability and LED indicators. The eight inputs of the module tolerate voltages from 0 to 25 VDC (the absolute maximum 30 VDC), and they have 40 ms (typical) contact debouncing circuitry. The module allows connection of slow pulse inputs, for example, with tipping bucket rain gauges.

The later version, QMI118, is otherwise the same as QMI108 except there are four open collector outputs and four outputs with high side switching capability. This means the module can switch, for example, +VB directly to the peripheral that needs it without additional relays. The rating of the high side outputs is the same as open collector outputs, 30 VDC / 1 A. They have a common positive terminal labeled +VB. The voltage does not have to be system battery voltage but can vary from 3.3 V to 30 V.

Note that the GND pin associated with the output block of O4 ... O7 (both blocks in QMI108) should always be wired to the system GND if any of the outputs are used to switch currents that exceed 10 mA.

Table 78 Technical Data QMI108/118

Property	Description/Value
Max. voltage all outputs, +VB	30 VDC
Max. current all outputs	1 A
Max. voltage all inputs	±25 V
Input default state	ON (max. 100 k weak pull up to 3.3 V built in)
Input debounce duration	40 ms typ., 60 ms max.
Input threshold	2 V
Input hysteresis	300 mV

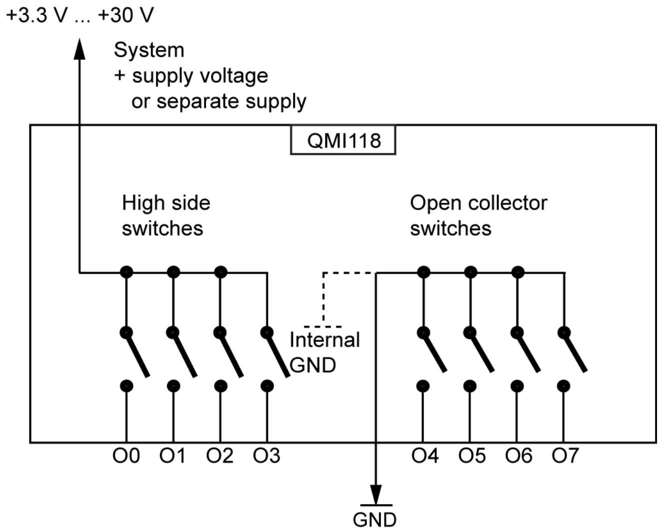

Figure 180 Digital I/O Module Wiring Diagram (Digital Outputs)

NOTE	The digital I/O module can only be connected to the QML201 logger or newer. The type of the logger can be checked with the **VER** command.

Fixed Line Modem DMX501

The fixed line modem module can be configured for a point-to-point line or for a multidrop modem network. If a modem is configured for multidrop use, the outgoing carrier is valid only during transmission. If the QML logger is the master in the multidrop network, the fixed line modem can be normally configured for point-to-point use.

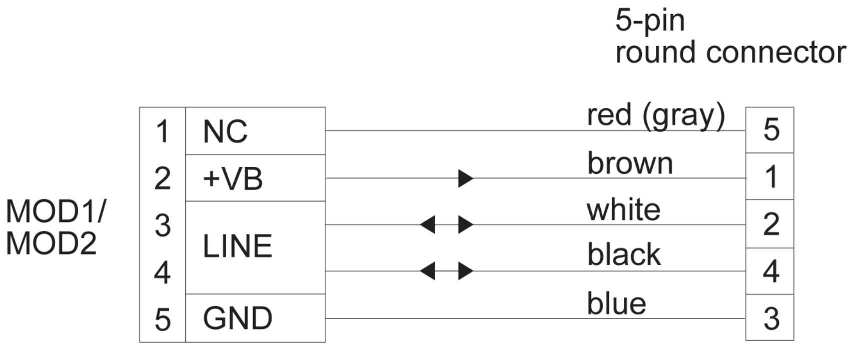

Figure 181 Fixed Line Modem Wiring Diagram

Ethernet Communication Module DSE101

The Ethernet module DSE101 provides a 10base-T connection, which is compatible with 100 Mbps and 1 G switches; DSE101, however, operates at 10 Mbps. Supported by software, the module allows virtual serial port connections and socket connections to be established to the logger.

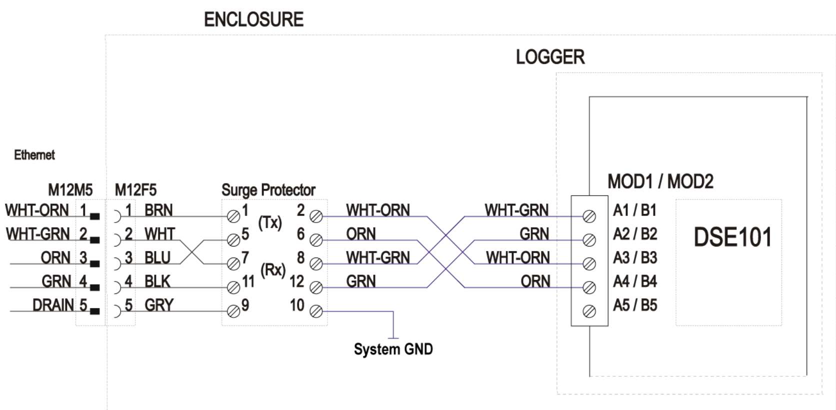

Figure 182 DSE101 Ethernet Module Wiring Diagram

Telemetry Options

PSTN Modem

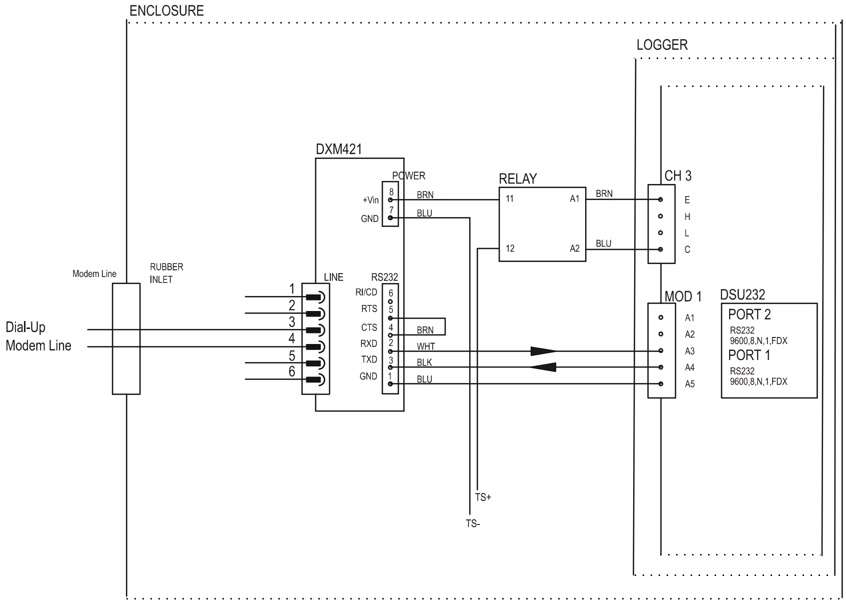

Figure 183 PSTN Modem Wiring Diagram

Leased-Line Modem

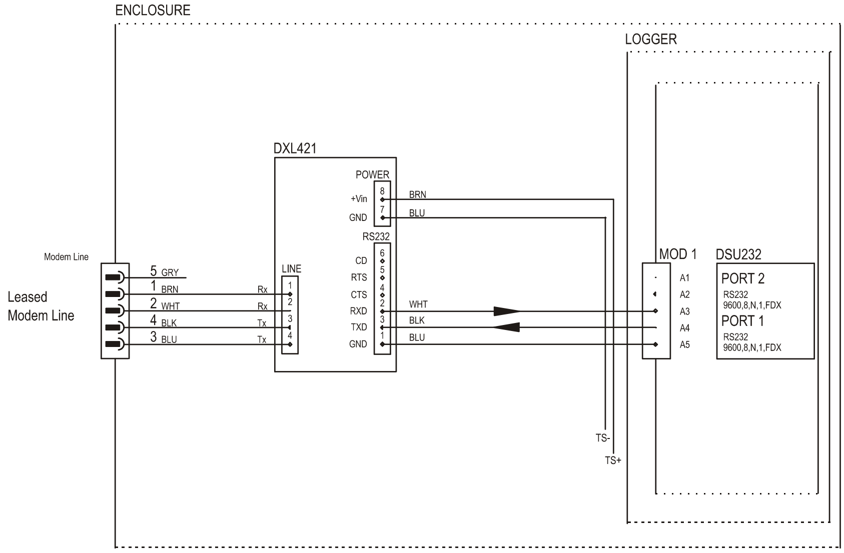

Figure 184 Leased Line Modem Wiring Diagram

GSM/GPRS Modem

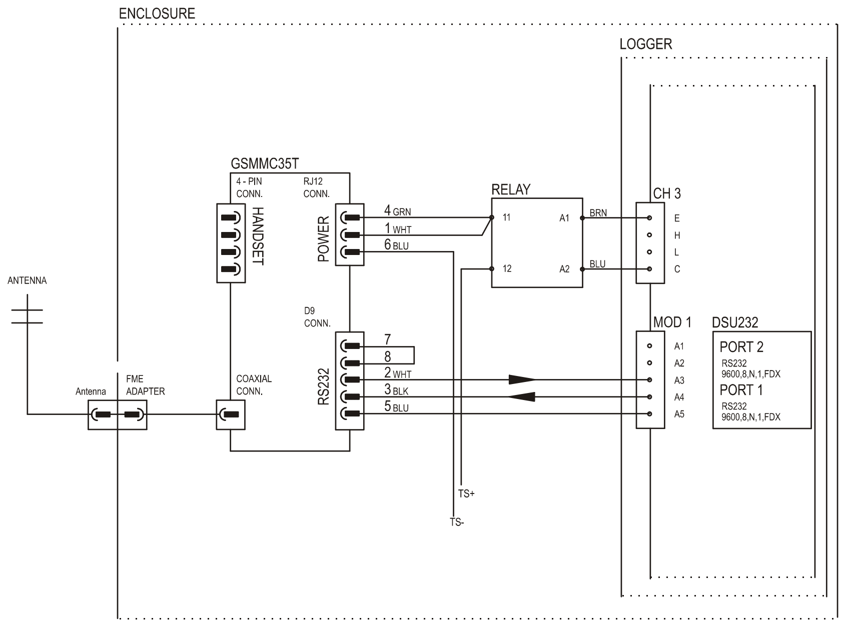

Figure 185 GSM/GPRS Modem Wiring Diagram

ORBCOMM Satellite Transceiver

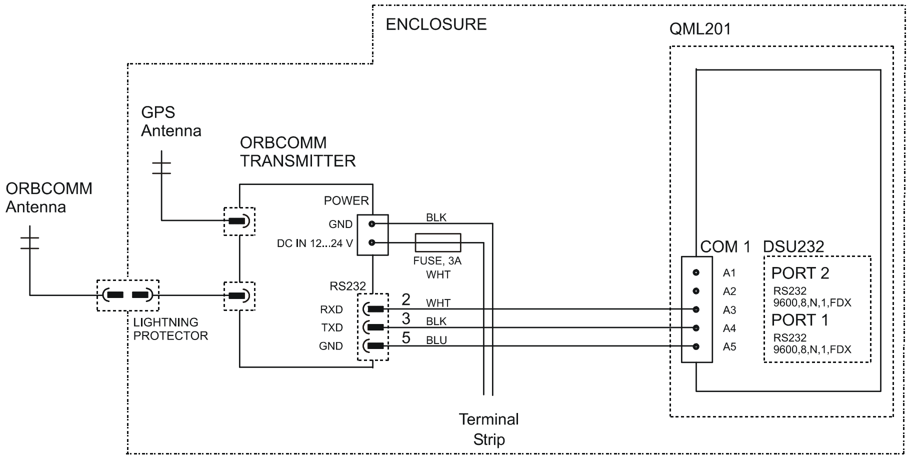

Figure 186 ORBCOMM Satellite Transceiver Wiring Diagram

GOES Satellite Transmitter

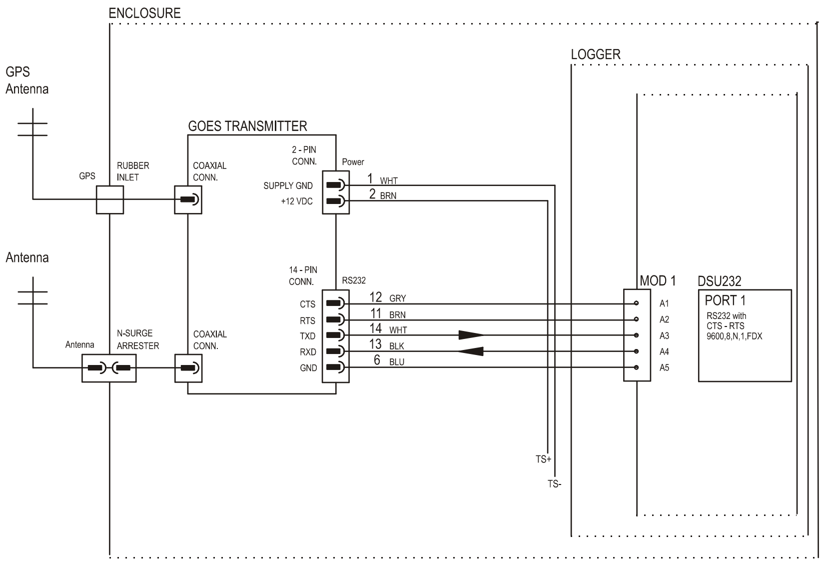

Figure 187 GOES Transmitter Interface Wiring Diagram

Sensors

Vaisala WA15 Set Sensors

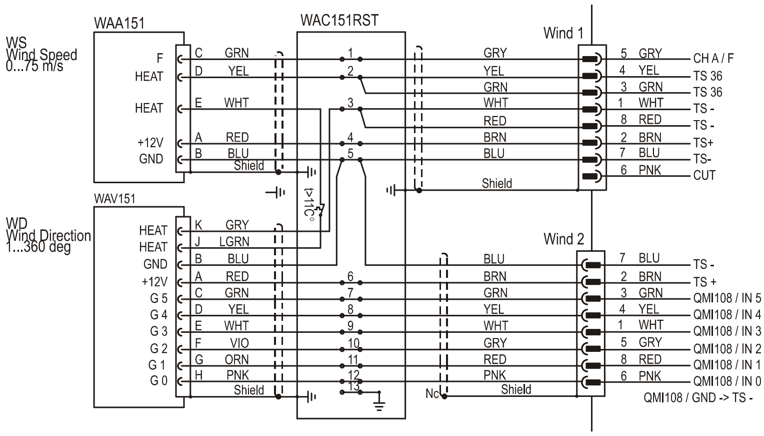

Figure 188 Example Wiring Diagram for Digital I/O Module with Anemometer and Wind Vane - Sensors Powered Continuously

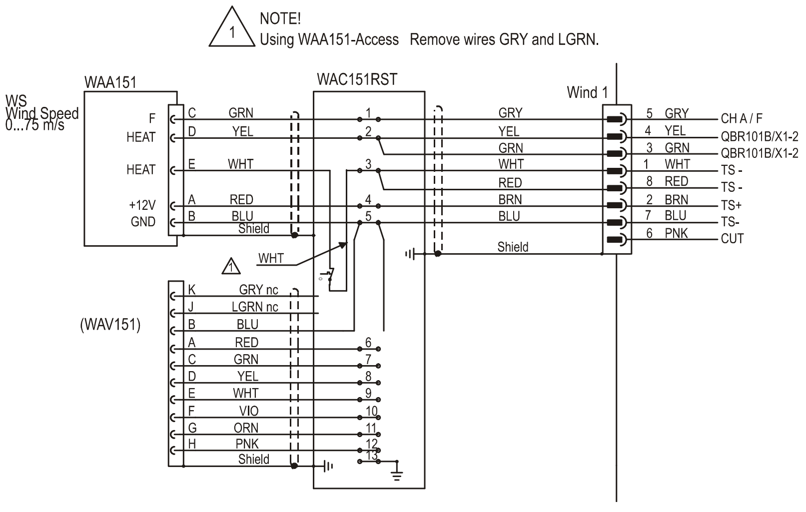

Figure 189 Example Wiring Diagram for Digital I/O Module with Anemometer Only

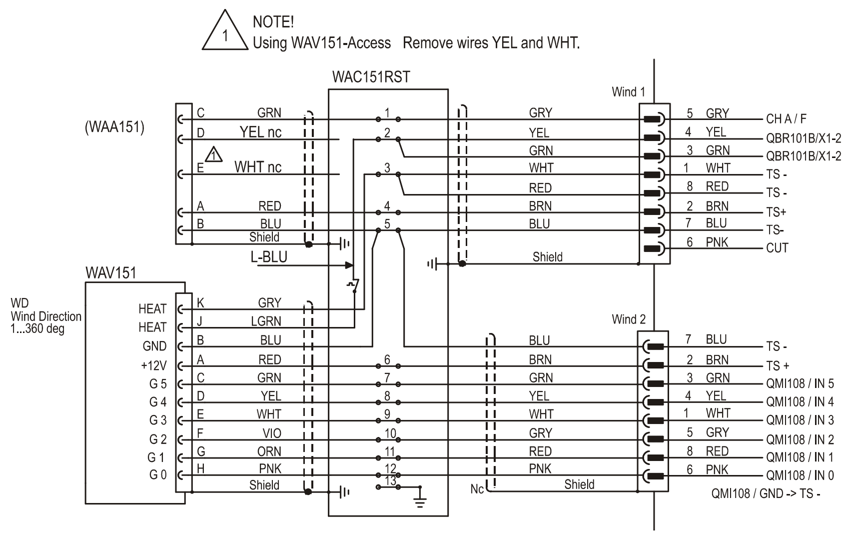

Figure 190 Example Wiring Diagram for Digital I/O Module with Wind Vane Only

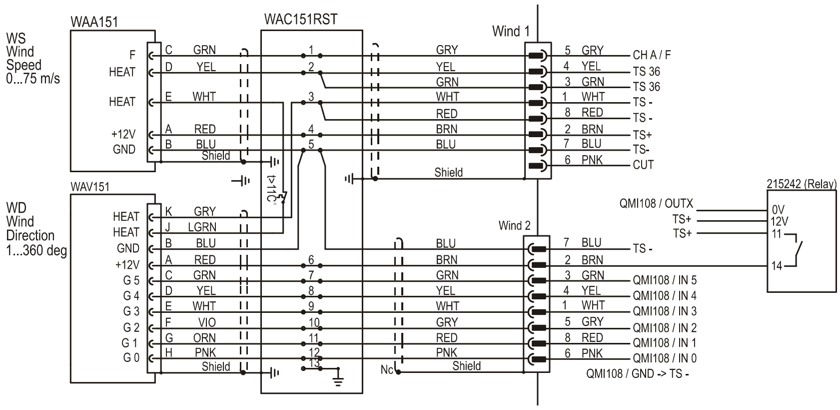

Figure 191 Example Wiring Diagram for Digital I/O Module with Anemometer and Power Reduction for Wind Vane - Wind Vane Powered Only When Measured

Temperature Measurement PT1000

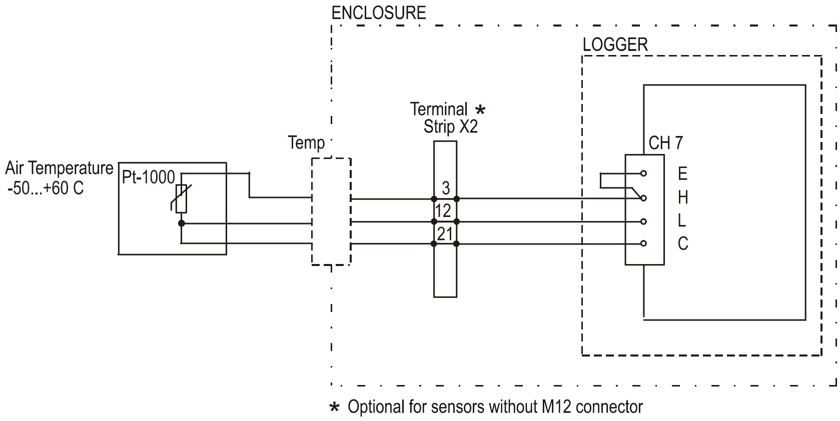

Figure 192 Example Wiring Diagram for PT1000 3-Wire Connection

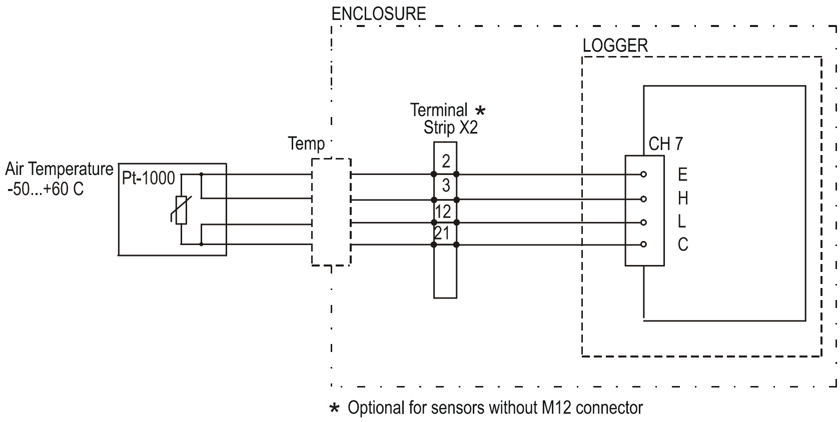

Figure 193 Example Wiring Diagram for PT1000 4-Wire Connection

Specifications

PSTN Modem DXM421

Table 79 PSTN Modem Power Input

Property	Description/Value
Supply voltage	+5 ... +30 VDC
Power consumption	0.8 W, standby 0.45 W
Fuse	0.5A PTC
Transient protection	Transil diode
Reverse connect protection	Yes
Connector	2 pin screw terminal
Indicator	One green LED: Power on

Table 80 PSTN Modem Serial RS-232 Interface

Property	Description/Value
Topology	RS-232C interface
Serial port speed	300 ... 57600 bps
Connector	6 pin screw terminal
Indicators	Three orange LEDs: RI/CD, TxD, and RxD

Table 81 PSTN Interface

Property	Description/Value
Modem compatibility	ITU V.90, K56flex; ITU-T V.34 enhanced, V.34, V.32bis, V.32, V.22bis, V.22Bell 212A and 103/113; ITU-T V.29, V.42, V.42bis; ITU-T V.21 & V.23
Error correction	ITU-T V.42 (LAP-M or MNP 3-4)
Data compression	ITU-T V.42bis and MNP 5
Transient protection	350 V surge arrestor, PTC-fuses, sidactor
Isolation	1500 V
Connector	RJ-11, 2-wire

Table 82 PSTN Modem Mechanical Specifications

Property	Description/Value
Housing material	EN AW-1050A aluminum, gray anodized
Housing classification	IP30 (NEMA 1)
Dimensions w × d × h	25 × 80 × 76 mm (0.98 × 3.15 × 2.99 in.)
Weight	94 g (3.3 oz.)

Table 83 PSTN Modem Environmental Specifications

Property	Description/Value
Operating temperature	-40 °C ... +55 °C (-40 ... +131 °F)
Storage temperature	-50 °C ... +75 °C (-58 ... +167 °F)
Humidity	20 ... 90 % RH, non-condensing
EMC	EN 55022
Immunity	
ESD	EN 61000-4-2, contact dis. 4 kV, air dis. 8 kV
RF field	EN 61000-4-3
Fast transient bursts (EFT)	EN 61000-4-4
Transient surges	EN 61000-4-5
Conducted RF	EN 61000-4-6
Vibration	IEC 68-2-64 Fh, 10 ... 500 Hz, up to 2.0 G
Free fall	IEC 68-2-32 Ed, fall height 1000 mm
Cold	IEC 68-2-1, -40 °C 16 hours
Heat	IEC 68-2-2, +55 °C 16 hours
Slow change of temperature, 1°C/min	IEC 68-2-14, -40 °C to +55 °C, two cycles
CE marked	Yes

Leased Line Modem DXL421

Table 84 Leased Line Modem Power Input

Property	Description/Value
Supply voltage	+5 ... +30 VDC
Power consumption	0.09 W
Fuse	0.5 A PTC
Transient protection	Transil diode
Reverse connect protection	Yes
Connector	2-pos. screw terminal
Indicator	One green LED for power on

Table 85 Leased Line Modem Serial RS-232 Interface

Property	Description/Value
Topology	RS-232C interface
Serial port speed	300 ... 57600 baud
Connector	6-pos screw terminal
Indicators	Three orange LEDs for CD, TxD, and RxD

Table 86 Leased Line Modem Interface

Property	Description/Value
Modem compatibility	V.21 300/300 bps FSK, V.22 1200/1200 DPSK and V.23 1200/1200 bps FSK.
	2- or 4-wire, point-to-point or multi-point
Transient protection	350 V surge arrestor, PTC-fuses, sidactor
Isolation	1500 V
Connector	4-pos screw terminal

Table 87 Leased Line Modem Mechanical Specifications

Property	Description/Value
Housing material	EN AW-1050A aluminum, gray anodized
Housing classification	IP30 (NEMA 1)
Dimensions w × d × h	25 × 80 × 76 mm
	(1.0 × 3.1 × 3.0 in.)
Weight	94 g (3.3 oz.)

Table 88 Leased Line Modem Environmental Specifications

Property	Description/Value
Operating temperature	-40 °C ... +55 °C (-40 ... +131 °F)
Storage temperature	-60 °C ... +75 °C (-76 ... +167 °F)
Humidity	5 to 95 % RH, non-condensing
EMC	EN 55022
Immunity ESD RF field Fast transient bursts (EFT) Transient surges Conducted RF	 EN 61000-4-2, contact dis. 4kV, air dis. 8kV EN 61000-4-3 EN 61000-4-4 EN 61000-4-5 EN 61000-4-6
Vibration	IEC 68-2-64 Fh, 10 ... 500 Hz, up to 2.0 G
Free fall	IEC 68-2-32 Ed, fall height 1000 mm
Cold	IEC 68-2-1, -40 °C 16 hours
Heat	IEC 68-2-2, +55 °C 16 hours
Slow change of temperature, 1°C/min	IEC 68-2-14, -40 °C ... +55 °C, two cycles

Cellular Modems

Table 89 GSM Terminal MC35 Specifications

Property	Description/Value
Sensitivity	-104 dBm
Dual Band	EGSM900/GSM1800
Data transmission	GPRS class 8 (up to 85.6 kbps)
	Full PBCCH support
	GPRS mobile station, class B
	CSD up to 14.4.kbps
	USSD
	Non transparent mode
	V.110
	7E1-character framing
SMS	Included
Tx power	2 W (class 4)
	1 W (class 1)
Supply voltage	8 ... 30 V
Power consumption Transmit mode Idle mode	 300 mA (average) 10 mA
Sleep	3 mA
Power down	0.1 mA
Temperature range	-20 ... +55 °C
Weight	130 g (4.6 oz.)

Table 90 GSM Antenna Specifications

Property	Description/Value
Frequency range	Dual band 900/1800 MHz
No. of elements	16
Gain	7.5 dBd
Polarization	Vertical
Weight	420 g (14.8 oz.)
Cable	2.5 or 10 m

Radio Modems

Table 91 Radio Modem SATELLINE 3AS Specifications

Property	Description/Value
Frequency range	380 ... 470 MHz
Channel spacing	12.5/25 kHz
Number of channels	160 / 80
Frequency stability	<±1.5 kHz
Type of emission	F1D
Communication mode	Half-duplex
Carrier power	10 mW ... 1 W / 50 Ω
Carrier power stability	+2 dB/-3 dB
Adjacent channel power	acc. to EN 300 220-1 / ETS 300 113
Spurious radiations	acc. to EN 300 220-1 / ETS 300 113
Sensitivity	-116 ... -110 dBm (BER <10 E-3)
Co-channel rejection	>-12 dB
Adjacent channel selectivity	>60 dB/>70 dB
Intermodulation attenuation	>65 dB
Spurious radiations	<2 nW
Interface	RS-232 or RS-422, RS-485
Interface connector	D 15, female
Data speed of RS interface	300 ... 38 400 bps
Data speed of radio interface	19 200 bps (25 kHz channel) 9600 bps (12,5 kHz channel)
Data formats	Asynchronous data
Operating voltage	+9 ... + 30 VDC
Power consumption	1.8 VA typical (receive) 6.0 VA typical (transmit) 0.05 VA typical (when DTR is "0")
Temperature range	-25 ... +55 °C (-13 ... +131 °F)
Antenna connector	TNC, 50 Ω, female
Construction	Aluminum enclosure
Dimensions h x w x d	137 × 67 × 29 mm (5.4 × 2.6 × 1.1 in.)
Installation plate	130 × 63 × 1 mm (5.1 × 2.5 × 5/128 in.)
Weight	250 g (0.55 lb.)

Table 92 Radio Modem SATELLINE 3AS Epic Specifications

Property	Description/Value
TRANSCEIVER	
Frequency range	400 ... 470 MHz
Channel spacing	12.5/25 kHz
Number of channels	160/80
Frequency stability	<±1.5 kHz
Type of emission	F1D
Communication mode	Half-duplex
TRANSMITTER	
Carrier power	1 ... 10 W/50 Ω
Carrier power stability	+2 dB/-3 dB
Adjacent channel power	acc. to ETS 300 113
Spurious radiations	acc. to ETS 300 113
RECEIVER	
Sensitivity	-116 ... -110 dBm (BER <10 E-3)
Co-channel rejection	>-12 dB
Adjacent channel selectivity	>60 dB/>70 dB
Intermodulation attenuation	>65 dB
Spurious radiations	<2 nW
DATA MODEM	
Interface	RS-232 or RS-422. RS-485
Interface connector	D 15, female
Data speed of RS interface	300 ... 38400 bps
Data speed of radio interface	19200 bps (25 kHz channel)
	9600 bps (12,5 kHz channel)
Data formats	Asynchronous data
GENERAL	
Operating voltage	+11.8 ... + 30 VDC
Power consumption	3 VA typical (receive)
	25 VA typical (transmit)
	0.05 VA typical (when DTR is "0")
Temperature range	-25 ... +55 °C
Antenna connector	TNC, 50 Ω, female
Construction	Aluminum enclosure
Dimensions h × w × d	151 × 123 × 31 mm (5.9 × 4.8 × 1.2 in.)
Weight	550 g (1.2 lb.)

ORBCOMM Satellite Transceiver Set

Table 93 ORBCOMM Satellite Transmitter Specifications

Property	Description/Value
Power source (current) Sleep mode Power save Receive Transmit	 0.5 mA (@ 12 V), 1.2 mA (@ 24 V) 210 mA (@ 12 V), 115 mA (@ 24 V) 220 mA (@ 12 V), 120 mA (@ 24 V) 1.4 A (@ 12 V), 0.7 A (@ 24 V)
Interface Serial Input port Output port Analog input Power control Status monitor	 RC232C 2 ch (TTL) 2 ch (TTL) 2 ch, 8 bit (0 ... 3.3 V) 1 ch (SW) 2 ch (TTL)
Up-link (TX) RF TX power Frequency Modulation	 5 W 148 ... 150.05 MHz, 819 ch SDPSK 2400 bps
Down-link (RX) Sensitivity Frequency Modulation	 -118 dBm @ BER 1E-5 137 ... 138 MHz, 399 ch SDPSK 4800 bps
Position (determination) Method Accuracy TTFF	 8 ch parallel, C/A code <100 m 2 Drms <1 min
UTC time Pulse period Pulse (VER C)	 1 ms pulse every sec 2/4/8/16/32 sec
Timing accuracy	1 µs
Temperature Operating Storage	 -30 ... 75 °C -40 ... 85 °C
Mechanical	SAEJ1455
Vibration	4.9 g
Dimensions h × w × d	221 × 89 × 33 mm (8.7 × 3.5 × 1.3 in.)
Weight	660 g (1.46 lb.)

GOES Satellite Transmitter

Table 94 GOES Satellite Transmitter Specifications

Property	Description/Value
Frequencies	GOES satellite channels 1 to 200. International satellite channels 202 to 266 (even numbers).
	Meteosat international channels 1 to 33 and regional channels 1 to 33.
	Argos and SCD channels 0 to 9.
Frequency stability	Over temperature ±0.5 ppm.
	Over time (long term) ±1 ppm
Frequency setting accuracy	±1 Hz
Output	50 Ω impedance. Short and open circuit protected
RF power output	GOES: 10 W standard, 20 W optional Meteosat: 10 W Argos: 1.8 W SCD: 2 W nominal, 1 ... 4 W factory adjustable.
Clock drift	<15 sec per year
Input voltage	12.5 VDC nominal (11 ... 15 VDC)
Current requirements quiescent transmit (GOES, 10 W) transmit (Argos/SCD 2 W)	 <6.0 mA typical 3.5 A 1.2 A
Temperature ranges Operating (GOES) (Argos/SCD) Storage	 -40 ... +50 °C (-40 ... +122 °F) max. +65 °C (+149 °F) -55 ... +75 °C (-67 ... +167 °F)
Dimensions l × h × w	16.5 × 12.7 × 5.1 cm (6.5 × 5 × 2 in.)
Weight	0.9 kg (1.25 lb.)
Certification	NOAA/NESDIS certified for self-timed and random GOES transmissions under Signal Engineering, Inc. model numbers SEH-100(10) and SEH-100(20).
	EUMETSAT agency for the Meteosat system under Signal Engineering, Inc. model number SEHM-100.
	Argos/SCD transmitter certified for the Argos system by the CNES Certification Laboratory (France) under Signal Engineering, Inc. model number SEH-400.

Fixed Line Modem Module DXM501

Table 95 Fixed Line Modem Module Specifications

Property	Description/Value
Modem chip	73K324L
Modem chip crystal frequency	11.0592 MHz
USART clock frequency	11.0592 MHz/2
Register access	Operated through an 8-bit bus interface
Connection	2-wire Point-to-point line or Multidrop modem network
Modem protocols	V.21, 300 bps FSK V.23, 1200/75 bps FSK V.22, 1200 bps DPSK
Line interface	Matched to 600 Ω
TxControl signal	Configurable
Supply voltage	5 V (+4.75 ... +5.50 V)
Current consumption Reset/power-down Operation	 9 mA 26 mA
Transmit level	-10 dBm
Distance between modules	19 km (~12 mi.) with 26 AWG standard cable
Operating and storage temperature	-50 ... +70 °C (-58 ... 158 °F)
Humidity	0 ... 100 % RH, non condensing

Ethernet Communication Module DSE101

Table 96 Ethernet Module DSE101 Specifications

Property	Description/Value
Operating mode	10Base-T
Power consumption (transmitting)	50 mA at 12 V
Temperature	Normal: -40 ... +70 °C (-76 ... 158 °F) Extended: -60 ... +70 °C (-76 ... 158 °F)
Humidity	0 ... 100 % RH

APPENDIX A
CONFIGURATION INFORMATION FOR LEGACY ITEMS

This appendix contains configuration information for devices not necessarily needed in MAWS versions 6.00 or later.

COM Server Unit

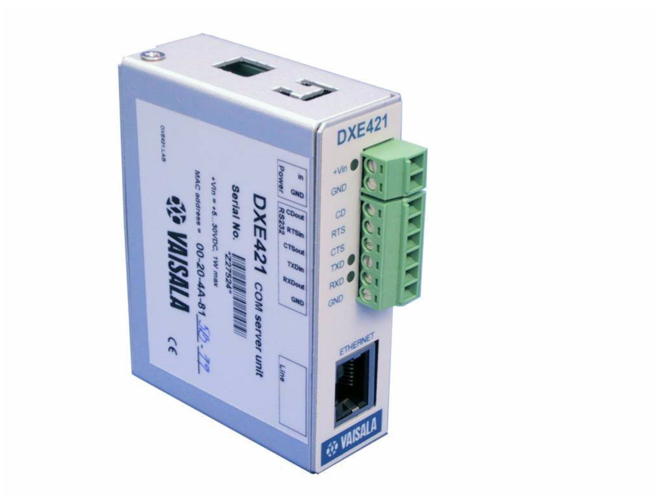

Figure 194 COM Server Unit

With MAWS versions 6.00 or later, the COM Server unit for TCP/IP connections is no longer needed. Instead, the QML logger provides the necessary TCP/IP functionality directly.

The COM Server Unit provides a direct LAN network connection for weather stations and other systems. The module converts a standard RS-232 port to a 10Base-T/100Base-T Ethernet connection making connected systems Internet-Enabled devices. Application software, for example, display software, can receive the data:

- by reading the data directly from TCP/IP
- from another COM server unit installed at the receiving site to convert the TCP/IP data stream back to RS-232 for the COM port connection
- by installing ComServer software, which establishes a virtual COM port to the IP address making a transparent RS-232 connection to a HydroMet™ system.

The module has a small enclosure with DIN-rail mounting support that can be connected into various enclosures. It has face connectors where the power supply, RS-232, and Ethernet lines can be connected.

When connecting the COM server unit to the QML logger, you need to use the terminal strip to provide power for the unit. For the example wiring diagram, see Figure 195 on page 364.

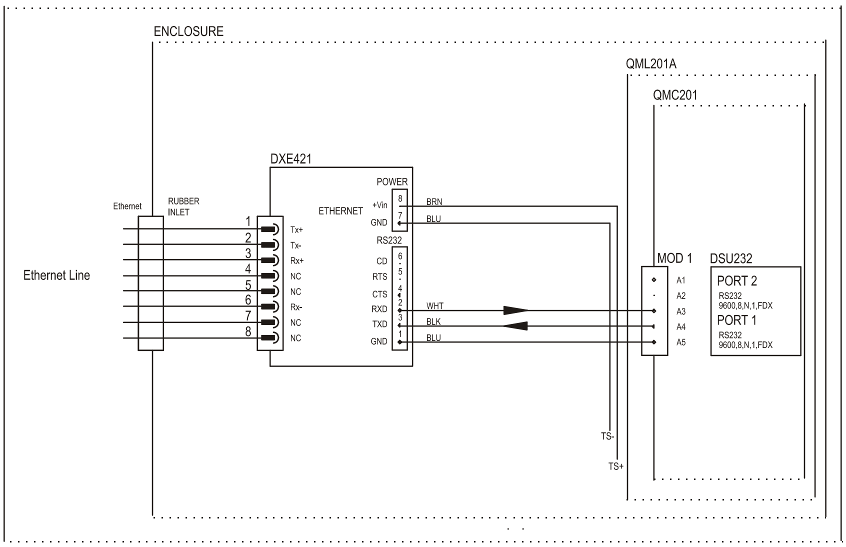

Figure 195 COM Server Unit Wiring Diagram

GPRS Configuration Using the iConnector Module

With MAWS versions 6.00 or later, the iConnector module for wireless TCP/IP connections is no longer needed. Instead, the QML logger provides the necessary TCP/IP connection to the GPRS modem directly; in Lizard Setup Software, the communication device Siemens GPRS-IP can be used.

iConnector iC101 is a small adapter that enables installed devices to use the Internet for messaging via wireless modems and data-enabled phones that operate over AMPS, CDMA, CDPD, GPRS, GSM, iDEN, and TDMA wireless networks. iConnector provides "Instant Internet™" connectivity by eliminating the need for any hardware modification to a host device when connecting it to an Internet Service Provider (ISP). iConnector supports, for example, FTP client basic features and enables the user to communicate with the server using the FTP protocol.

Wiring GSM Modem with iConnector

When connecting GSM Modem with iConnector to the QML logger, you need to use an extra relay. For the wiring diagram, see Figure 196 on page 366.

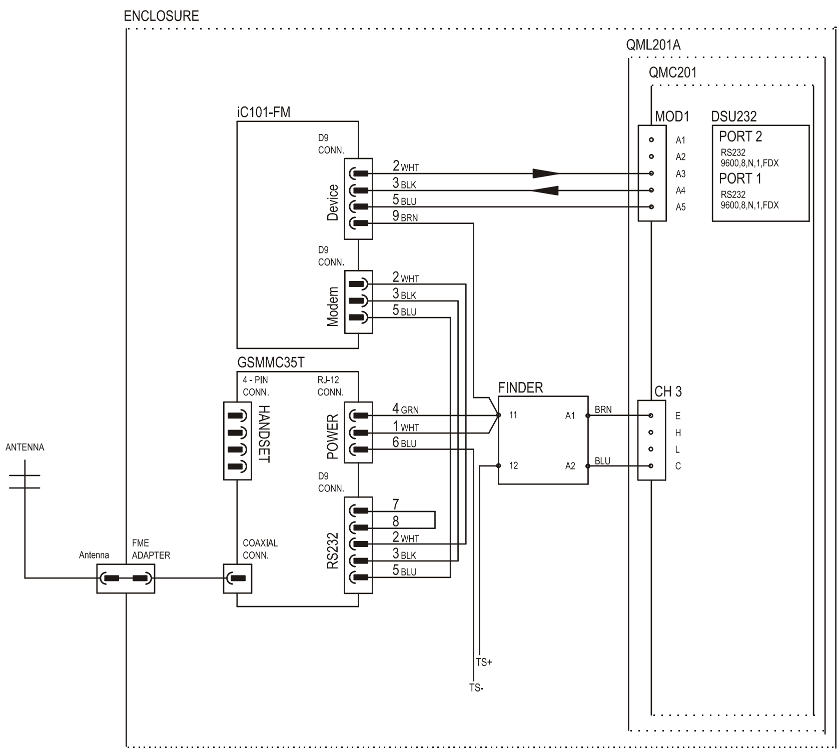

Figure 196 GSM Modem with iConnector Wiring Diagram

GPRS Modem Configuration

In addition to data calls and SMS messages, the GPRS modem can be used to transfer data over the Internet.

The device GPRS-modem with iChip is added to the setup and connected to a serial port in the **Devices** view.

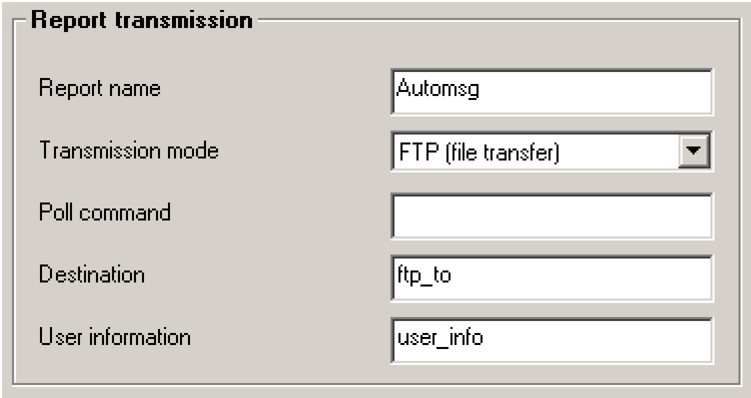

Figure 197 GPRS Transmission Configuration Options

Transmission Mode

The **Transmission mode** parameter allows selection between five modes:

1. **Data call**: The report is automatically sent to the destination by making a call.
2. **SMS message**: The report is sent using SMS message.
3. **E-mail**: The report is sent to an e-mail address.
4. **FTP**: The report is sent as a file to the FTP server using the file transfer protocol.
5. **Polled only**: The report is only sent when polled by the remote system using a data call or SMS message.

For all of the above listed modes, polling using data call or SMS is possible. You simply need to define the **Poll** command and enable the **Answer incoming calls** option.

Destination

The value for the **Destination** parameter depends on the selected transmission mode as follows:

- Phone number for the data call and SMS transmission.
- E-mail address for the E-mail transmission.
- FTP server name followed by the optional path for the data. For example, when entering *ftp.vaisala.com/data* the reports are sent to directory data under user's home directory in the *ftp.vaisala.com* server. In addition, the **Station name as path extension** option can be used to place files in subdirectories with station names. For example, the full storage path for station MAWS1 would be *ftp.vaisala.com/data/MAWS1*.

NOTE	The iConnector firmware does not support the creation of directories; thus the specified directory has to exist in the destination FTP server.

The **Destination** parameter either sets the phone number or gives the name of the **Station setting** parameter that contains the phone number.

User Information

The **User information** parameter is required only for FTP transmission. The format for **User information** is:

<*username*>,<*password*>

where

username = Valid username for the destination FTP server

password = Valid user password for the FTP server

Example:

```
maws1,weatherpass1
```

You may either set the appropriate value directly or specify the name of the Station setting parameter containing the value.

iConnector Configuration

The iConnector can be configured by selecting the device **iConnector_x** in the **Device configurations** view.

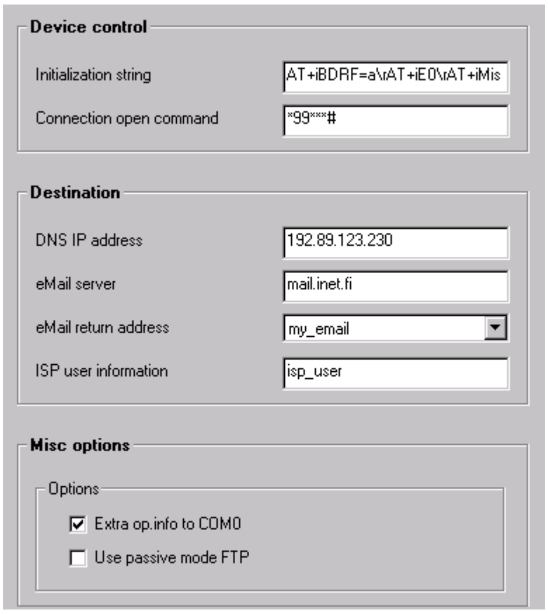

Figure 198 iConnector Configuration Options

Initialization String

The **Initialization string** contains the set of parameters that is sent to iConnector each time the modem is initialized. The default parameters are listed in Table 97 on page 369.

Table 97 Initialization Strings for iConnector

String	Description
AT+iBDRF=a	Autobaud on
AT+iE0	Echo off
AT+iMis=\"at+CGDCONT=1, \\\"IP\\\",\\\"internet\\\"\"	Initialization string for the GPRS interface (defines a PDP context)
AT+iATH=2	CHAP authentication
AT+iMTYP=102	Modem type is GSM

Connection Open Command

The **Connection open command** parameter defines the dial string for opening the GPRS connection.

DNS IP Address

The **DNS IP address** parameter defines the IP address for Domain Name Server. The server in that particular address provides services that are needed for translating domain names into IP addresses. The correct value can be obtained from the Internet service provider.

eMail Server

The **eMail server** parameter defines the name of the server for handling outgoing mail. The correct name can be obtained from the Internet service provider.

eMail Return Address

The **eMail return address** parameter specifies the name of the **Station setting** parameter containing the e-mail address that is shown as sender for output messages. New **Station setting** parameters can be created using the **Static variables** dialog in the **Setup configuration** view. To set the value for the parameter, use either the Terminal software **Station settings** window, or the **spset** command in service connection. For example, the **eMail return address** parameter *my_email* in Figure 198 on page 369 is a static parameter that can be defined in the **Station settings** window.

This setting has to be formatted as *user@domain*, for example, *station1@maws.net*. Note that you can freely set this value, but to maintain compatibility with future features, prefer selecting names that can be real e-mail addresses. The domain has to be an existing one.

NOTE	Always use quotation marks around the **eMail return address** parameter when setting the value with the **Station settings** window.

NOTE	The ISP eMail system sends all delivery error messages and problem reports to the given **eMail Return address**. You may utilize this feature for troubleshooting purposes: set, for example, your own eMail address as the **eMail return address** and you will receive information about the failed messages. However, with this feature you cannot solve problems that occur before the data reaches the eMail system, such as GPRS connection problems.

ISP User Information

The **ISP User information** parameter (ISP = Internet Service Provider) is required only when accessing Internet using PSTN or GSM data. The format for **ISP User information** is:

<*username*>,<*password*>

where

username	=	Valid username for the destination ISP account
password	=	Valid user's password for the destination ISP account

Example

```
maws1,weatherpass1
```

You may either set the appropriate value directly or specify the name of the **Station setting** parameter containing the value.

Use Passive Mode FTP

When this option is selected, iConnector uses passive mode FTP sessions. PASV is then used instead of PORT when opening an FTP session. This should generally not be selected, but it may be required in some firewalled environments. For a detailed description of this feature, refer to the FTP standard (RFC 959).

Log File Sending with FTP

Log files can be sent to an FTP server using the iConnector and a modem. When using this feature, the log files of the previous day are automatically sent every day after midnight.

To use this feature, proceed as follows:

1. Configure the iConnector and modem as described in section Report Transmission Configuration on page 144.
2. To send the Log files to an FTP server, create a log group in the **Logging** view, and add the items to be logged.
3. Select the **FTP** tab in the **Configure Log Group** frame, as shown in Figure 199 on page 372.

Figure 199 FTP Settings Tab

4. Select **Upload logs via FTP** and give the following parameters:

   a. **Destination**: This is the FTP server name followed by an optional path to which the files will be sent. For example, when entering *ftp.vaisala.com/data* the reports are sent to directory data under user's home directory in the *ftp.vaisala.com* server.

NOTE	The iConnector firmware does not support the creation of directories; thus the specified directory has to exist in the destination FTP server.

   b. **Use station name in ftp server path**: When this option is selected, the log files will be placed in subdirectories with station names. For example, the full storage path for station MAWS1 would be: *ftp.vaisala.com/data/MAWS1*.

c. **User information**: The user name and password for the FTP server, given in the format: <*username*>,<*password*>

where

username	=	Valid username for the destination FTP server
password	=	Valid user password for the FTP server

d. **Retries**: The maximum number of upload retries in case of communication errors.

e. **Retry delay**: The delay between these retries in seconds.

Both parameters **Destination** and **User information** can be set either directly or by specifying the name of a static parameter that contains the value.

Specifications

COM Server Unit

Table 98 COM Server Unit Power Input

Property	Description/Value
Supply voltage	+5 ... +30 VDC
Power consumption	0.8 W
Fuse	0.5 A PTC
Transient protection	Transil diode
Reverse connect protection	Yes
Connector	2-pos screw terminal
Indicator	One green LED power on

Table 99 COM Server Unit Serial RS-232 Interface

Property	Description/Value
Topology	RS-232C interface
Serial port speed	300 ... 57600 baud
Connector	6-pos screw terminal
Indicators	Two orange LEDs for TxD and RxD

Table 100 COM Server Unit Ethernet Interface

Property	Description/Value
Protocols	TCP/IP, UDP/IP, ARP, ICMP, SNMP, TFTP, Telnet, DHCP, BOOTP, HTTP, and AutoIP
Interface	Ethernet 10/100Base-T (Auto-Sensing)
Password protection	Version 2.0/IEEE 802.3
Internet Web Server	Serves web pages and Java applets
	Storage capacity 384 Kbytes
Connector	RJ-45 Shielded
Indicators	10Base-T and 100Base-T connection

Table 101 COM Server Unit Mechanical Specifications

Property	Description/Value
Housing material	EN AW-1050A aluminum, gray anodized
Housing classification	IP30 (NEMA 1)
Dimensions w × d × h	25 × 80 × 76 mm
	(1.0 × 3.1 × 3.0 in.)
Weight	94 g (3.3 oz.)

Table 102 COM Server Unit Environmental Specifications

Property	Description/Value
Operating temperature	-40 °C ... +55 °C (-40 ... +131 °F)
Storage temperature	-60 °C ... +75 °C (-76 ... +167 °F)
Humidity	5 to 95 % RH, non-condensing
EMC	EN 55022
Immunity ESD RF field Fast transient bursts (EFT) Transient surges Conducted RF	 EN 61000-4-2, contact dis. 4kV, air dis. 8kV EN 61000-4-3 EN 61000-4-4 EN 61000-4-5 EN 61000-4-6
Vibration	IEC 68-2-64 Fh, 10 ... 500 Hz, up to 2.0 G
Free fall	IEC 68-2-32 Ed, fall height 1000 mm
Cold	IEC 68-2-1, -40 °C 16 hours
Heat	IEC 68-2-2, +55 °C 16 hours
Slow change of temperature, 1°C/min	IEC 68-2-14, -40 °C ... +55 °C, two cycles

Cellular Modems

Table 103 GSM Terminal TC35i Specifications

Property	Description/Value
Sensitivity	-104 dBm
Dual Band	EGSM900/GSM1800
Data transmission	CSD up to 14.4.kbps
	USSD
	Non transparent mode
	V.110
	7E1-character framing
SMS	Included
Tx power	2 W (class 4)
	1 W (class 1)
Supply voltage	8 ... 30 V
Power consumption Transmit mode Idle mode Sleep Power down	 300 mA (average) 10 mA 3 mA 0.1 mA
Temperature range	-20 ... +55 °C
Weight	130 g (4.6 oz.)

Table 104 iConnector Specifications

Property	Description/Value
Supported cellular protocols	AMPS, CDMA, CDPD, GPRS, GSM, IDEN, and TDMA
Supported Internet access protocols	email, HTTP, WAP, FTP, Telnet, and TCP/UDP
Connector	DB-9
Host interface	RS-232
Power supply	Any 5 ... 24 VDC external, or via pin 9 of the host DB-9 connector
Power Consumption	0.4 W
Weight	37 g (1.3 oz.)
Measures	42.67 x 69.19 x 20.17 mm (1.68 x 2.724 x 0.794 in.)
Temperature range	-40 ... +85 °C (-40 ... 185 °F)

APPENDIX B
GSM 7-BIT CHARACTER SET

This appendix contains the 7-bit character set for GSM communications. The character set is provided as an aid for configuring and troubleshooting GSM-based connections.

Hex.	Dec.	Character name	Character	ISO-8859-1 DEC
\x00	0	COMMERCIAL AT	@	64
\x01	1	POUND SIGN	£	163
\x02	2	DOLLAR SIGN	$	36
\x03	3	YEN SIGN	¥	165
\x04	4	LATIN SMALL LETTER E WITH GRAVE	è	232
\x05	5	LATIN SMALL LETTER E WITH ACUTE	é	233
\x06	6	LATIN SMALL LETTER U WITH GRAVE	ù	249
\x07	7	LATIN SMALL LETTER I WITH GRAVE	ì	236
\x08	8	LATIN SMALL LETTER O WITH GRAVE	ò	242
\x09	9	LATIN CAPITAL LETTER C WITH CEDILLA	Ç	199
\x0A	10	LINE FEED		10
\x0B	11	LATIN CAPITAL LETTER O WITH STROKE	Ø	216
\x0C	12	LATIN SMALL LETTER O WITH STROKE	ø	248
\x0D	13	CARRIAGE RETURN		13
\x0E	14	LATIN CAPITAL LETTER A WITH RING ABOVE	Å	197
\x0F	15	LATIN SMALL LETTER A WITH RING ABOVE	å	229
\x10	16	GREEK CAPITAL LETTER DELTA	Δ	
\x11	17	LOW LINE	_	95
\x12	18	GREEK CAPITAL LETTER PHI	Φ	
\x13	19	GREEK CAPITAL LETTER GAMMA	Γ	
\x14	20	GREEK CAPITAL LETTER LAMBDA	Λ	
\x15	21	GREEK CAPITAL LETTER OMEGA	Ω	
\x16	22	GREEK CAPITAL LETTER PI	Π	
\x17	23	GREEK CAPITAL LETTER PSI	Ψ	
\x18	24	GREEK CAPITAL LETTER SIGMA	Σ	
\x19	25	GREEK CAPITAL LETTER THETA	Θ	
\x1A	26	GREEK CAPITAL LETTER XI	Ξ	
\x1B	27	ESCAPE TO EXTENSION TABLE		
\x1B0A	27 10	FORM FEED		12
\x1B14	27 20	CIRCUMFLEX ACCENT	^	94
\x1B28	27 40	LEFT CURLY BRACKET	{	123
\x1B29	27 41	RIGHT CURLY BRACKET	}	125
\x1B2F	27 47	REVERSE SOLIDUS (BACKSLASH)	\	92
\x1B3C	27 60	LEFT SQUARE BRACKET	[	91
\x1B3D	27 61	TILDE	~	126
\x1B3E	27 62	RIGHT SQUARE BRACKET	]	93
\x1B40	27 64	VERTICAL BAR	|	124
\x1B65	27 101	EURO SIGN	€	164 *(ISO-8859-15)*

Hex.	Dec.	Character name	Character	ISO-8859-1 DEC
\x1C	28	LATIN CAPITAL LETTER AE	Æ	198
\x1D	29	LATIN SMALL LETTER AE	æ	230
\x1E	30	LATIN SMALL LETTER SHARP S (German)	ß	223
\x1F	31	LATIN CAPITAL LETTER E WITH ACUTE	É	201
\x20	32	SPACE		32
\x21	33	EXCLAMATION MARK	!	33
\x22	34	QUOTATION MARK	"	34
\x23	35	NUMBER SIGN	#	35
\x24	36	CURRENCY SIGN	¤	164 (ISO-8859-1)
\x25	37	PERCENT SIGN	%	37
\x26	38	AMPERSAND	&	38
\x27	39	APOSTROPHE	'	39
\x28	40	LEFT PARENTHESIS	(	40
\x29	41	RIGHT PARENTHESIS	)	41
\x2A	42	ASTERISK	*	42
\x2B	43	PLUS SIGN	+	43
\x2C	44	COMMA	,	44
\x2D	45	HYPHEN-MINUS	-	45
\x2E	46	FULL STOP	.	46
\x2F	47	SOLIDUS (SLASH)	/	47
\x30	48	DIGIT ZERO	0	48
\x31	49	DIGIT ONE	1	49
\x32	50	DIGIT TWO	2	50
\x33	51	DIGIT THREE	3	51
\x34	52	DIGIT FOUR	4	52
\x35	53	DIGIT FIVE	5	53
\x36	54	DIGIT SIX	6	54
\x37	55	DIGIT SEVEN	7	55
\x38	56	DIGIT EIGHT	8	56
\x39	57	DIGIT NINE	9	57
\x3A	58	COLON	:	58
\x3B	59	SEMICOLON	;	59
\x3C	60	LESS-THAN SIGN	<	60
\x3D	61	EQUALS SIGN	=	61
\x3E	62	GREATER-THAN SIGN	>	62
\x3F	63	QUESTION MARK	?	63
\x40	64	INVERTED EXCLAMATION MARK	¡	161
\x41	65	LATIN CAPITAL LETTER A	A	65
\x42	66	LATIN CAPITAL LETTER B	B	66
\x43	67	LATIN CAPITAL LETTER C	C	67
\x44	68	LATIN CAPITAL LETTER D	D	68
\x45	69	LATIN CAPITAL LETTER E	E	69
\x46	70	LATIN CAPITAL LETTER F	F	70
\x47	71	LATIN CAPITAL LETTER G	G	71
\x48	72	LATIN CAPITAL LETTER H	H	72
\x49	73	LATIN CAPITAL LETTER I	I	73
\x4A	74	LATIN CAPITAL LETTER J	J	74
\x4B	75	LATIN CAPITAL LETTER K	K	75
\x4C	76	LATIN CAPITAL LETTER L	L	76
\x4D	77	LATIN CAPITAL LETTER M	M	77

Hex.	Dec.	Character name	Character	ISO-8859-1 DEC
\x4E	78	LATIN CAPITAL LETTER N	N	78
\x4F	79	LATIN CAPITAL LETTER O	O	79
\x50	80	LATIN CAPITAL LETTER P	P	80
\x51	81	LATIN CAPITAL LETTER Q	Q	81
\x52	82	LATIN CAPITAL LETTER R	R	82
\x53	83	LATIN CAPITAL LETTER S	S	83
\x54	84	LATIN CAPITAL LETTER T	T	84
\x55	85	LATIN CAPITAL LETTER U	U	85
\x56	86	LATIN CAPITAL LETTER V	V	86
\x57	87	LATIN CAPITAL LETTER W	W	87
\x58	88	LATIN CAPITAL LETTER X	X	88
\x59	89	LATIN CAPITAL LETTER Y	Y	89
\x5A	90	LATIN CAPITAL LETTER Z	Z	90
\x5B	91	LATIN CAPITAL LETTER A WITH DIAERESIS	Ä	196
\x5C	92	LATIN CAPITAL LETTER O WITH DIAERESIS	Ö	214
\x5D	93	LATIN CAPITAL LETTER N WITH TILDE	Ñ	209
\x5E	94	LATIN CAPITAL LETTER U WITH DIAERESIS	Ü	220
\x5F	95	SECTION SIGN	§	167
\x60	96	INVERTED QUESTION MARK	¿	191
\x61	97	LATIN SMALL LETTER A	a	97
\x62	98	LATIN SMALL LETTER B	b	98
\x63	99	LATIN SMALL LETTER C	c	99
\x64	100	LATIN SMALL LETTER D	d	100
\x65	101	LATIN SMALL LETTER E	e	101
\x66	102	LATIN SMALL LETTER F	f	102
\x67	103	LATIN SMALL LETTER G	g	103
\x68	104	LATIN SMALL LETTER H	h	104
\x69	105	LATIN SMALL LETTER I	i	105
\x6A	106	LATIN SMALL LETTER J	j	106
\x6B	107	LATIN SMALL LETTER K	k	107
\x6C	108	LATIN SMALL LETTER L	l	108
\x6D	109	LATIN SMALL LETTER M	m	109
\x6E	110	LATIN SMALL LETTER N	n	110
\x6F	111	LATIN SMALL LETTER O	o	111
\x70	112	LATIN SMALL LETTER P	p	112
\x71	113	LATIN SMALL LETTER Q	q	113
\x72	114	LATIN SMALL LETTER R	r	114
\x73	115	LATIN SMALL LETTER S	s	115
\x74	116	LATIN SMALL LETTER T	t	116
\x75	117	LATIN SMALL LETTER U	u	117
\x76	118	LATIN SMALL LETTER V	v	118
\x77	119	LATIN SMALL LETTER W	w	119
\x78	120	LATIN SMALL LETTER X	x	120
\x79	121	LATIN SMALL LETTER Y	y	121
\x7A	122	LATIN SMALL LETTER Z	z	122
\x7B	123	LATIN SMALL LETTER A WITH DIAERESIS	ä	228
\x7C	124	LATIN SMALL LETTER O WITH DIAERESIS	ö	246
\x7D	125	LATIN SMALL LETTER N WITH TILDE	ñ	241
\x7E	126	LATIN SMALL LETTER U WITH DIAERESIS	ü	252
\x7F	127	LATIN SMALL LETTER A WITH GRAVE	à	224

www.vaisala.com

M210933EN