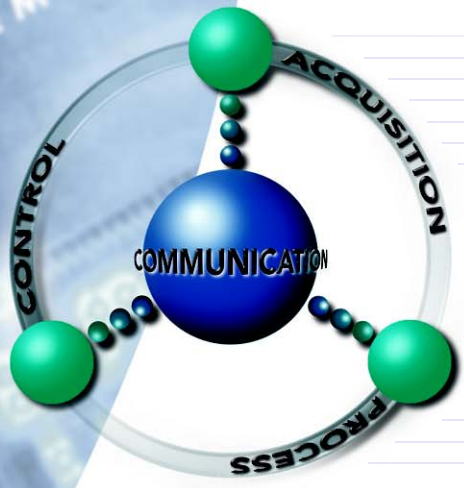
ADCON TELEMETRY

Base Station

Telemetry Gateway A840 and Wireless Modem A440

User Guide

valid for A840 firmware release 3.8.4



SMART WIRELESS SOLUTIONS

ADCON TELEMETRY GMBH
INKUSTRASSE 24
A-3400 KLOSTERNEUBURG
AUSTRIA
TEL: +43|2243|38280-0
FAX: +43|2243|38280-6
http://www.adcon.at

ADCON INTERNATIONAL INC
2050 LYNDELL TERRACE
SUITE 120
CA-95616 DAVIS, USA
TEL: +1|530|7531458
FAX: +1|530|7531054
http://www.adcon.at

Document Release 3.8.4, March 2007

Table of Contents

Chapter 1. Introduction

This manual describes the use of the A840 Telemetry Gateway and A440 Wireless Modem combination. Due to their general nature, either unit can also be used independently, but such uses are beyond the scope of this manual. The manual teaches you how to use the telemetry gateway and the wireless modem as a base station for an Adcon wireless network.

To build a network you need the following:

- one or more A730MD, A733, A733GSM/GPRS, A720, or A723 remote telemetry units (RTUs)
- an A440/A840 base station
- data acquisition and control software, such as Adcon's addVANTAGE 3.x or addVANTAGE 4 Pro software
- to access the A733 you either need a telephone line or an external GSM modem such as the WorldCom M1206

For additional information about the RTUs and the addVANTAGE software, consult the respective user manuals.

The A840 Telemetry Gateway is a low-power, battery-backed device that acts as an interface between an Adcon wireless network and one or more hosts running addVANTAGE or similar data acquisition software. The gateway is based on a powerful 32-

bit processor running the Linux operating system. It has 16-MB Flash EPROM acting as a hard disk and 16 MB RAM. The software can be upgraded in the field.

Several interfaces are available: Ethernet, V34 modem, RS-232 serial, and RS-485 multidrop serial. A built-in rechargeable battery provides the telemetry gateway with approximately 24 hours of operation without mains power (the number of hours depends on the number of RTUs the gateway has to poll).

The A440 Wireless Modem is a low-power, narrow-band data transceiver operating in the 70-cm band. It implements Adcon's low-speed radio protocol and is therefore compatible with all Adcon RTUs. In addition the A440 modem supports a high-speed wireless protocol that will be used by future Adcon devices.

The A440 has an 8-bit Flash-based microcontroller that can also be upgraded in the field.

WARNING This manual is valid only for release 3.8.4 of the gateway's firmware. You can verify that you have the appropriate firmware by logging onto a serial terminal. The following should be displayed on your terminal:

```
Telemetry Gateway A840 Version 3.8.4

(C) 2001-2006 Adcon Telemetry GmbH
```

If you have an older firmware version, you can get a new image file from Adcon's web site (http://www.adcon.at), or by contacting Adcon technical support.

NOTE FOR USA: THIS DEVICE COMPLIES WITH PART 15 OF THE FCC RULES. OPERATION IS SUBJECT TO THE CONDITION THAT THIS DEVICE DOES NOT CAUSE HARMFUL INTERFERENCE.

Conventions

Certain conventions apply in this documentation.

Italics	Indicate the text is variable and must be substituted for something specific, as indicated in the explanation. Italics can also be used to emphasize words as words or letters as letters.
Bold	Indicates special emphasis of the text. Also indicates menu names and items in a window.
`fixed font`	Indicates characters you must type or system messages.
File ›› Save	Indicates menu selection. For example, select the **File** menu, then the **Save** option.
Note	Indicates information of interest. Notes appear **after** the information they apply to.
CAUTION	Indicates that you may get unexpected results if you don't follow the instructions. Cautions appear **before** the information they apply to.
WARNING	Indicates danger to yourself or damage to the device if you don't follow the instructions. Warnings appear **before** the information they apply to.
<value>	Indicates a required input parameter.
[value]	Indicates an optional input parameter.

Chapter 2. System Setup

Package Contents

Before proceeding to the installation of your base station, first verify that you received all of the following components:

- the A840 Telemetry Gateway
- the A440 Wireless Modem
- the 30 m (100 ft) connection cable between the A840 and the A440 device
- a power cord
- a twisted-pair standard Ethernet cable
- a serial null modem cable
- a whip antenna

If any of the above items is missing, contact your dealer.

If you plan to use the gateway's built-in modem (e.g. in conjunction with A733GSM/GPRS RTUs) you will also need a modem cable suitable for your telephone jack (not included).

Installation

The base station has two main components: an indoor unit (the A840) and an outdoor unit (the A440).

Before proceeding with the installation, take a moment to plan your network. First, it is essential to realize the importance of selecting a good location for the base station. You must consider several factors, some of them quite contradictory, when you select this location:

- From a radio perspective, the height of the receiving antenna is essential: the higher the antenna, the greater the range of communication. For the GSM operated RTUs this is not an issue, but you must make sure you have good on-site coverage from your cellular provider.
- The base station should be situated in the same building where the people managing the base station work, or at least spend some of their time.
- Places like cellars, near heat sources, or damp locations are not suitable.
- Geographically it is better to have the base station in the center of the area where the RTUs will be installed (this does not apply to A733GSM/GPRS stations).
- If you plan to use the base station as a server to allow other people to log in and get data, be sure that a telephone line is available exclusively for this use.
- Adcon also recommends that you have a telephone connection next to the PC for when you need technical support.

The communication distance you can achieve is directly proportional to the height of the receiving antenna. The propagation mode of the waves the Adcon system uses is basically the line of sight. Due to the curvature of the earth, on flat terrain, the maximum distance reached depends on the height of the receiver and of the measuring stations' antennas. You can't do much on the remote station side, but you have more options with the base station. Some typical examples of the achieved distances, under various conditions, are shown in Table 1.

The addIT RTU, however, uses low-power technology and was not designed to communicate over large distances. The typical "line-of

sight" distance an addIT can communicate is 800 m (approximately half a mile). This is valid if the partner device is mounted on a 3 m mast (9 ft) and the addIT is mounted on a 30 cm mast (1 ft).

The above figures are estimates based on a great deal of experience with installing this kind of equipment and assume a typical height of 3 m (9 ft) for the antennas of the remote measuring stations. Similarly, ideal physical conditions are assumed for the terrain including a flat, open, non-urban environment.

Table 1. Communication Between Base Station and A730MD/A733

Receiving antenna height	Typical distances achieved
6 m (18 ft)	5 km (3 miles)
10 m (31 ft)	8 km (5 miles)
20 m (62 ft)	16 km (10 miles)
30 m (92 ft)	24 km (15 miles)

What conclusions can be drawn from Table 1? Primarily, you can see the importance of having the base station antenna as high as possible. To gain antenna height, you have several options:

- Build a mast directly on the ground; however, a 30 m (100 ft) mast may not be a practical option.
- Set up a mast on a tall building (of course, it has to be *the* building where you want to have the base station).
- Install the base station in a building that is situated on a hill.

Note that the cable for the outdoor unit delivered with the system is 30 m long (about 100 ft). This means you must locate your indoor unit no more than 30 m from the outdoor unit. Extending the cable is not recommended, because the signal strength loss is significant.

To use the telemetry gateway and wireless modem, you need to do the following:

- Install the outdoor unit.

- Install the indoor unit.
- Initialize the base station.

Installing the Outdoor Unit

Install the outdoor unit on the rooftop of the house where the base station will be located or on top of a nearby mast.

After unpacking the components of the base station, identify the device marked "Wireless Modem A440." This unit has two connectors: one for a whip antenna (also supplied in the package) and a second that accommodates the 30 m (100 ft) connection cable to the indoor unit.

Note: This operation should be performed by a certified electrician. Make sure that the mast on which the A440 Wireless Modem will be installed is properly grounded.

Complete the following steps to install the outdoor unit (Figure 1):

1. Fasten the wireless modem to the aluminum mast (provided by Adcon) using the supplied ring clamp.

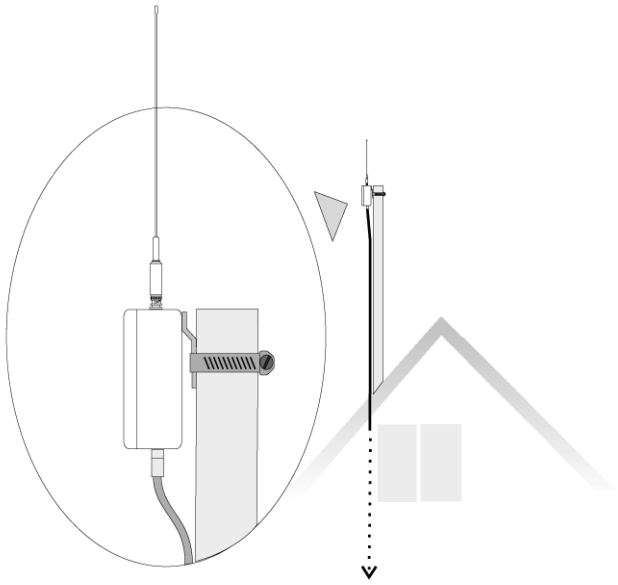


Figure 1. Outdoor Unit Installation

2. Fasten the antenna to the wireless modem.
3. Plug the proper end of the communication cable into the lower connector of the wireless modem.
4. Secure the mast in its place on the roof.
5. Run the cable to the indoor unit.

Note: The cable supplied to connect the outdoor unit to the A840 Telemetry Gateway is 30 m long and cannot be extended. Extending the cable will make your wireless modem inoperable due to the voltage loss on the additional cable length. You can, however, install an outdoor antenna and use a coaxial cable (max. 30 m) between the A440 unit and the antenna, giving you a total of 60 m distance between the indoor unit and the outdoor antenna.

Installing the Indoor Unit

First identify the device marked "Telemetry Gateway A840," which is the indoor unit. Then connect the cables to the gateway as shown in Figure 2.

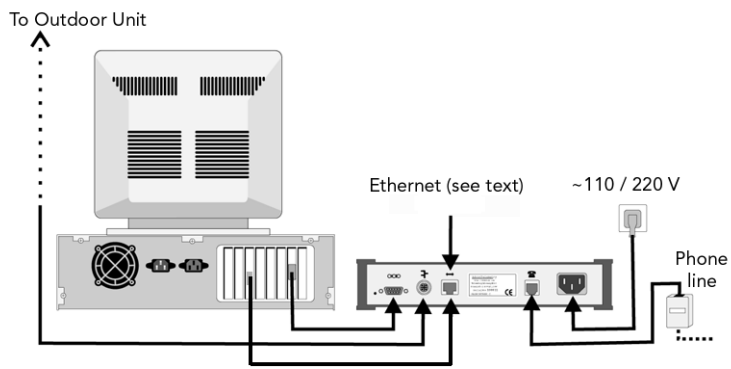


Figure 2. Indoor Unit Installation

If you are using a hub or switch to connect several computers in a local area network (LAN), use the twisted-pair Ethernet cable that Adcon provides to connect the gateway to the hub. If your computer is a standalone PC and is not part of a LAN, you can connect it directly to the gateway, as shown in Figure 2, but you

will need to provide your own "crossover" Ethernet cable, which you can find in most computer stores.

If you plan to use the base station as a standalone unit (that is, without a computer), you need a telephone line that must be connected to the modem jack of the A840 gateway.

Note: Before you operate the base station, Adcon strongly recommends that you charge the internal battery to a level where it can sustain the operation of the gateway. You can do this by plugging the power cable into the A840 device, but disconnecting the serial cable from the A440 outdoor unit for at least six hours.

Initializing and Configuring the Base Station

You can use the gateway in one of the following configurations, depending on your application:

- a new installation
- a replacement for an A730SD with addVANTAGE 3.x
- a replacement for an A730SD and upgrading to addVANTAGE 4 Pro

New Installation

With a new installation, you will need to install the RTUs in the field and configure them in addVANTAGE. Refer to the addVANTAGE software and respective RTU user manuals for additional details.

A730SD Replacement with addVANTAGE 3.x

For the base station to operate properly, you must start the addVANTAGE 3.x software. Make sure that the software is properly configured (the serial port and so forth). Refer to the respective software manual for additional configuration details. If your base station operates standalone (as dumb server), initiate a call through addVANTAGE to configure it.

If you are installing the base station as replacement for an A730SD receiver, just upgrade your addVANTAGE installation to version 3.45 or later (earlier versions don't support the A840 Telemetry Gateway). After starting addVANTAGE, your new base station will be configured automatically.

A730SD Replacement with addVANTAGE 4 Pro

Because the addVANTAGE 4 Pro software communicates only via a TCP/IP network, you must first make sure that the networking is enabled and configured on the gateway. By default the gateways are delivered with the networking enabled, so you should be able to skip this section entirely and go to "Configuring the Gateway Using the Configurator" on page 22.

However, if for any reasons the network has become disabled, follow the steps below to enable it. Note that you might need some information from your network administrator to complete the steps.

Configuring the Network via a Terminal

1. Connect your gateway to the LAN with the twisted-pair CAT5 cable delivered with the product.

Note: If you don't have a LAN and you have only one computer, you must set up a small network. However, if your PC doesn't have the necessary network card, you must first install one and you will need to refer to your PC's user manual to do so.

   If your LAN consists only of your PC and the A840 gateway, you can either:

   - Use a crossover twisted-pair cable (not provided by Adcon) to connect one end of the cable to the gateway and the other to the PC.
   - Use the twisted-pair cable provided by Adcon to connect the gateway to a hub or switch.

2. Connect the gateway to your PC with the serial cable provided (as shown in Figure 2).
3. Start a terminal program (for example, Hyperterminal under Windows, Zterm under Mac OS, or minicom under Linux) and configure it for 19200 baud, no parity, one stop bit, hardware protocol, send LF after CR.
4. Press any key to see the `login` prompt. Log into the gateway as `root` (the default root password is *840sw*).
5. Type **`netconfig`** to display the current configuration on your terminal. Following is an example of a network configuration:

```
Network Configuration
```

```
Network active      : no
IP-address          : 192.168.1.2
Netmask             : 255.255.255.0
Gateway             : 192.168.1.1
Hostname.Domain     : a840
Nameserver IP(s)    : 192.168.1.2
Timeservername or IP:
Name/IP-Address from outside a NATing router : none
Port number of web-server from outside a NATing router
: none


DHCP server running : no

(C)hange/(S)ave/(D)hcp/(Q)uit configuration? [c/s/d/Q]
```

6. The above configuration shows that the network is not enabled. How you enable the network depends on the setup of your LAN:
   - If each computer on the LAN has a fixed address (it can be assigned by a system administrator or a DHCP server), get the IP information from your network administrator. Then type **c** and change the parameters as follows, substituting your own information (press **Enter** at the end of each line to display the next line):

     ```
     Activate Network [no]: y
     IP-address [192.168.1.2]: 221.38.15.75
     Netmask [192.168.1.0/24]: 255.255.255.0
     Gateway [192.168.1.1]: 221.38.15.1
     Hostname.Domain [a840]: my840.example.com
     Nameserver(s) (or none)[192.168.1.2]: 221.38.15.62
     Timeserver (or none) [none]:
     ```

     After you press **Enter** for the Timeserver parameter, the new configuration is displayed:

     ```
     Network Configuration

     Network active      : yes
     IP-address          : 221.38.15.75
     Netmask             : 221.38.15.0/24
     Gateway             : 221.38.15.1
     ```

```
Hostname.Domain     : my840.domain.com
Nameserver IP(s)    : 221.38.15.62
Timeservername or IP:
Name/IP-Address from outside a NATing router : none
Port number of web-server from outside a NATing router
: none


DHCP server running : no

Unsaved changes pending!

(C)hange/(S)ave/(D)hcp/(Q)uit configuration? [c/s/d/Q]
```

Type **s** to save your changes or **q** to quit the configuration. Note that if you type **q**, you will be asked if you want to keep the changes and the default answer is Yes.

Note: If you don't have Internet access, you can leave the gateway and nameserver entries unchanged. The hostname will identify the gateway in your network. However, if you have a name server on your network, you must use the name assigned by your administrator. If you don't have a time server available, either over the Internet or on your LAN, select <none>.

- If you have only a small LAN that has no direct access to the Internet (that is, Internet access is done through a dial-up connection) you should use the DHCP server feature built into the A840 Gateway. In this case, your computers on the LAN will get their IP addresses and other network information from the gateway (to configure your PC to use DHCP, refer to your PC's user manual).

  Type **d** to display the DHCP configuration on your terminal:

```
Network Configuration

Network active      : yes
IP-address          : 192.168.1.2
Netmask             : 255.255.255.0
Gateway             : 192.168.1.1
Hostname.Domain     : 192.168.1.2
```

```
Nameserver IP(s)    : 192.168.1.2
Timeservername or IP:
Name/IP-Address from outside a NATing router : none
Port number of web-server from outside a NATing router
: none
DHCP server running : no (will be started)

Unsaved changes pending!

(C)hange/(S)ave/(D)hcp/(Q)uit configuration? [c/s/d/Q]
```

Now type **s** to update the gateway with the displayed parameter information. You might need to configure your PC's network parameters, but you will have to refer to your PC user guide to do so.

- If the gateway is installed behind a network address translation (NAT/PAT) router and you want to access the gateway from the Internet, type the address and/or the port the gateway should exhibit outside the router . This is the hostname and/or IP address given by your Internet provider. You need a fixed address to be able to use the NAT feature.

You must also remember to configure your NAT router to route port 80 (or whatever port you entered) to the A840 gateway. The gateway will try to figure out if the requests come from the internal network or the Internet and will answer correspondingly.

Note: It is important that your gateway is connected to a network or another PC via the LAN connection before booting it, otherwise the networking hardware will be disabled. Reboot the gateway after attaching it to the network, if necessary, by typing the command **reboot**.

Configuring the Gateway Using the Configurator

You can use the A840 Gateway's built-in graphical tool to configure its functionality. Follow these steps to properly configure your gateway:

1. Configure the TCP/IP parameters.
2. Set the date/time and the time zone.
3. Change the default passwords.
4. Enter a license number.

5. Configure your telemetry network.
6. Configure the modem if your network includes A733GSM/GPRS RTUs.

In order to start and use the Configurator, make sure that your PC and the gateway are connected either directly via a crossover network cable, or via a hub or switch in a LAN.

Configure the TCP/IP parameters

If you have already configured the TCP/IP parameters with the command line interface as described in "Configuring the Network via a Terminal" on page 19, you can skip this section. If you have not configured the parameters, continue reading.

By default, the gateway is delivered with a DHCP server enabled, which means that the server distributes IP addresses to all the machines on the network requesting one. This might be a problem if you install the gateway in a LAN where a DHCP server is already active, so you will have to switch the server off, either via a terminal or via the Configurator. In the latter case, proceed as follows:

1. Configure your PC's TCP/IP networking to request its IP address from a DHCP server. If necessary, consult your PC's user manuals to do so.
2. Point your Internet browser to the address of your gateway. For example, if your gateway was configured as described on page 21, type **`http://192.168.1.2`** in your browser to display the welcome page of the A840 Telemetry Gateway.
3. Before continuing, make sure that all the requirements mentioned on the web page are met by your system (that is, that the Java VM and the Java WebStart technology are installed). When you are sure the system requirements have been met, you can click **Configure**.
4. The WebStart software starts and, after loading the Java application, a message appears warning you that the application requests unrestricted access to your machine and network. The application is signed with an Adcon Telemetry certificate that the WebStart software will not recognize, so you must click **Start** to proceed.
5. You are prompted for a user name and a password (see Figure 3). Use the account `root`, whose default password is `840sw`. If you changed the password, use the new one.

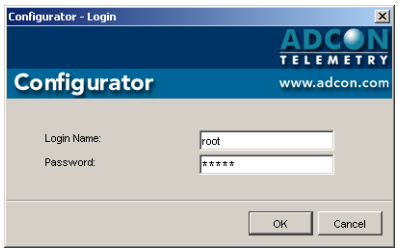


Figure 3. Configurator's Login Window

6. If the login was successful, the Configurator software starts up and by default shows you the actual configuration of your telemetry network: the root device, that is, the gateway itself (the base station), the RTUs (the devices), and their sensors (tags). The Configurator's graphical user interface (GUI) displays something similar to Figure 4.

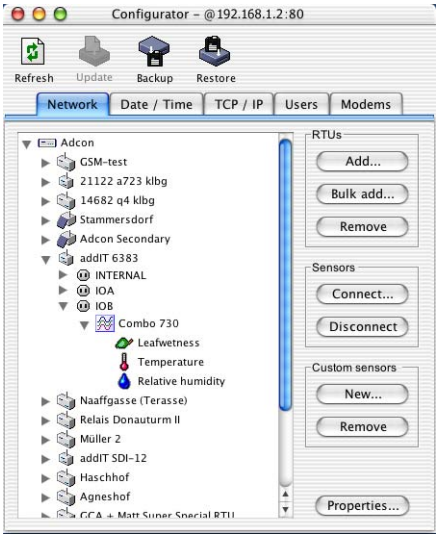


Figure 4. Configurator Startup

7. Select the **TCP/IP** tab. Note that all of the IP parameters are displayed, as shown in Figure 5. Click the **Distribute IP addresses** checkbox to disable this feature for now.

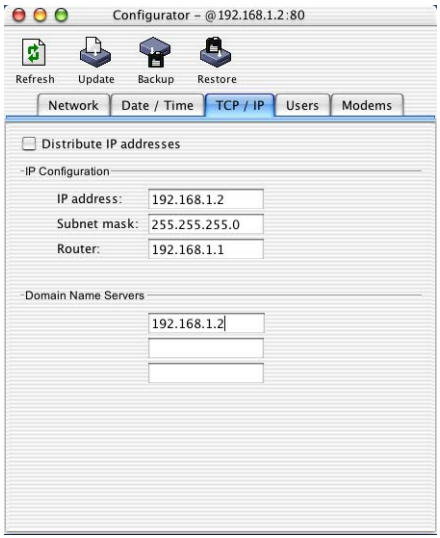


Figure 5. TCP/IP Parameters

8. Click **Update** to confirm the disabling of the automatic IP address distribution. After a short delay, the **Update** button turns gray, signifying that the operation was successful.
9. Configure the IP parameters according to your local network conditions (ask your network administrator if you're not sure how to do this).
10. Click **Update** to confirm your changes.

*Note: If you change the IP address or the netmask, you might lose the connection to the A840 Gateway after clicking **Update**. If this happens, restart the Configurator either via your browser or by double-clicking the link created by WebStart on your desktop (this feature is not available on all platforms).*

Configure the Date/Time and Time Zone

After you have set the TCP/IP parameters, proceed to configuring the date/time and the time zone.

1. Select the **Date/Time** tab to display the window shown in Figure 6.

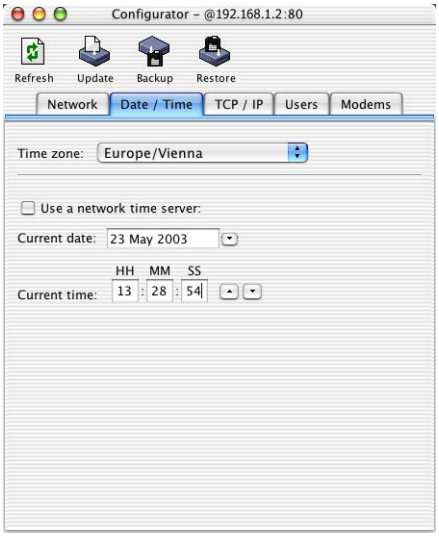


Figure 6. Configure the Date and Time

2. From the **Time zone** combo box, select your local time zone. Selecting the proper time zone is important for such things as automatically switching to and from Daylight Saving Time.
3. Do one of the following:
   - Set the date using the calendar, then set the current time by using one of these methods:
     - Select an edit box (for example, HH for hours) and use the up and down arrows to choose the appropriate time.
     - Type over the numbers in the edit boxes. If you do this, you must press **Enter** in each edit box you change.
   - If you have an NTP time server on your network or over the Internet (that is, your gateway is permanently connected to the Internet), you can select the **Use a network time server** checkbox to activate the time server. In this case the calendar and the time edit boxes are replaced by an edit box where you type the Internet name of the time server (see Figure 7).

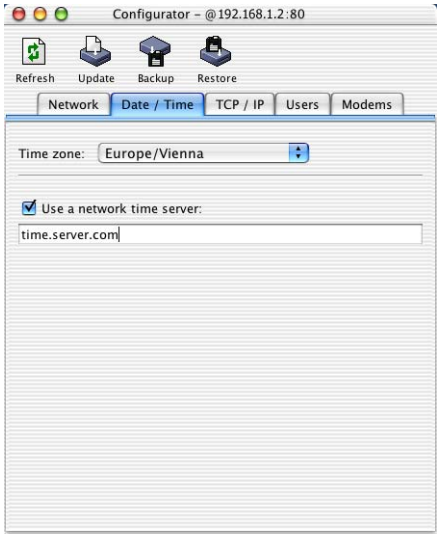


Figure 7. Configure a Time Server

4. After everything is properly configured, click **Update** to save the changes.

Changing Passwords

Your A840 Telemetry Gateway has two accounts: the *root* (administrator) account and the *adv* (addVANTAGE user) account. Both accounts have default passwords. You should change these passwords, especially for the root account, if your gateway is connected permanently to the Internet. Note that you cannot change passwords unless you are logged in as `root` (the default root password is *840sw* and for `adv` the password is *addvantage*; passwords are case-sensitive). If you log in as `adv`, you will see only the Network tab because the user `adv` has restricted rights on the gateway.

Follow these steps to change passwords:

1. Select the **Users** tab.
2. The configurator lists the two accounts *adv* and *root*. To change either of them, select the account and click **Change Password** (you can also double-click the account you want to change). The pop-up window shown in Figure 8 will be displayed.

Figure 8. Changing Passwords

3. Type the new password in the edit box, confirm it by typing it again, and then click **OK** to dismiss the pop-up.
4. Click **Update** to save the changes in the gateway.

You can also change passwords in a terminal session. See "Changing Passwords" on page 55 to do so.

Configuring the Modems

Your gateway contains a built-in modem. This modem can be used either for dial-in access (the "dumb server" functionality) by an application software e.g. addVANTAGE, or as dial-out for calling GSM based RTUs e.g. the A733GSM/GPRS station. For dial-in access, there is no need for any special configuration; however, for dial-out access you may need to configure some or all of the modem parameters.

As an alternative to the built-in modem, an external modem can also be used (e.g. a GSM modem); in this case, the external modem should be connected on the serial port of the gateway.

Note: Only a limited number of external modems are currently supported; some modems may work even if they are not supported. In case of doubt contact an Adcon representative.

To configure the modem parameters, click the **Modem** tab, and then select from the combo box the modem you want to configure. Following is a description of all configurable parameters and their meaning (see Figure 9):

1. In the **Dial** section, you can configure either **Tone** (default) or **Pulse** dialling. Most networks use tone dialling, but if in doubt, ask your telephone provider.
2. If required, enter a **Dialing prefix**; this is mostly needed when your gateway is connected to a private telephone exchange (e.g. a leading "0" before dialing).
3. The **Advanced options** section allows you to change:
    a. The **Initialization command**: the default should be sufficient for most situations; however, you may need to add other modem commands depending on your particular environment. For more details consult the AT command set specification.

Note: If you have a "Worldmodem" built-in your gateway, then remove the "%TCB" command from the initialization command. The information on your built-in modem is provided in a leaflet included in the A840 package. In case of doubt, please consult your dealer.

Figure 9. Modem parameters

*Note: If you use an external GSM modem, you can add **CBST=71,0,1** to the initialization command to tell the GSM modem to use the fully digital V110 mode. This makes setting up a connection much faster than with the standard modem emulation.*

b. The **Modem chat timeout** refers to an internal parameter (the time-out associated with the communication between the A733 microprocessor and the on board GSM module); it should not be changed.

c. The **Cancel the call...** parameter defines how long is the modem allowed to negotiate with its peer to establish a connection; in some cases you may need to increase this value if your peer modem has difficulties to connect. However, a higher value for this parameter might lead to higher connection costs.

Note: The above is valid only in cases of failure: normally the modems connect in typically 40 seconds (10 seconds for GSM modems using the V110 mode), therefore reducing this value will have no effects on the connection costs.

d. The **PIN for SIM** card entry allows you to enter the PIN for the SIM card in the external GSM Modem if there is no valid PIN set. After a successful update, the text in this entry field will change to *accepted* (reading back the PIN setting is not possible due to safety reasons).

Note: This setting is disabled while the modem is in use. You cannot change this setting while you are dialling in over the modem.

e. The **Country Setting** combo-box allows you to configure the country code for the built-in Worldmodem. For the CTR21 or an external modem, this combo-box is disabled.

Note: This setting is disabled while the modem is in use. You cannot change this setting while you are dialling in over the modem.

f. The **Log** combo-box allows you to select between logging only failed calls (default), or all calls. Note that at most the last 100 calls are logged, all older entries are automatically deleted. You can check the log file via the menu **File ▸ Modem log**.

*Note: You can also use a terminal to check the log file via the emu3ap utility by typing the command **calljournal**. Note also that the journal is lost after a reboot.*

In case you changed various modem parameters and you don't seem to make the unit properly dial-out anymore, use the **Restore defaults** button and the factory defaults will be restored.

Entering the license number

Most gateways are delivered with a lock on the maximum number of active RTUs that can be inserted in the network list (typically 5). To increase the maximum number of active RTUs you need a license number from Adcon Telemetry that you have to enter into the gateway (Figure 10):

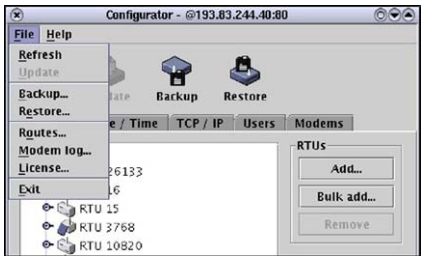


Figure 10. The File ▸ License... menu

- Select the **File ▸ License...** menu. In the dialog box appearing, enter the key.
- Press the **OK** button.

Now your gateway is ready for operation with a larger number of active RTUs.

Configuring your Telemetry Network

After all the basic administrative tasks are done, you need to prepare and configure the gateway for the addVANTAGE 4 Pro software. This requires starting the built-in configurator software and doing one of the following:

- Import a configuration file obtained from addVANTAGE 3.x.
- Configure the Adcon RTU network manually, if for example you don't have a previous addVANTAGE 3.x installation. Since this means starting from scratch, you'll need to refer to

the appropriate hardware and software manuals to configure your network. When you have done so, skip the rest of this section and go to "Administering your Telemetry Network" on page 35.

With your existing addVANTAGE 3.x installation, you will need to export its data. First, however, you need to upgrade your addVANTAGE Manager software to the latest addVANTAGE 3.x release provided on the distribution CD-ROM (currently, addVANTAGE Manager 3.47). Look for the addVANTAGE 3.46 directory to find the setup file for your language.

After you have upgraded addVANTAGE 3.4x, proceed as follows to export your data and configuration information:

1. Start the addVANTAGE Manager.
2. Select the database that you want to have exported, then select **Tools ▸ Export**.
3. Associate the custom sensors (if you have defined any) with an engineering unit.
4. Select the export path and click the **Export only configuration** checkbox to export only the telemetry network's configuration. (Adcon recommends that you do the full export later. See the addVANTAGE 4 Pro user manual for more information.)
5. Click **OK** to start the export operation. When it is finished, you will see a directory named something like *adv3.exp*. In this directory is a file named *config.xml* that your gateway will use as the configuration file.
6. Start the Configurator, if it is not already started, and log in (see Step 2 through Step 5 on page 23).
7. If the login was successful, the Configurator software starts and by default shows you the actual configuration of your telemetry network: the root device, that is, the gateway itself (the base station), the RTUs (the devices), and their sensors (tags). The Configurator's graphical user interface (GUI) displays something similar to Figure 11.

Note: If this is the first time you started the gateway, you will not see any RTUs or sensors displayed, only the root node.

8. Click **Restore** (or select **File ▸ Restore**). A dialog box appears, prompting you to find the directory that has the database

exported from addVANTAGE 3.4x. After navigating to the required directory, select the file *config.xml* and click **OK**.

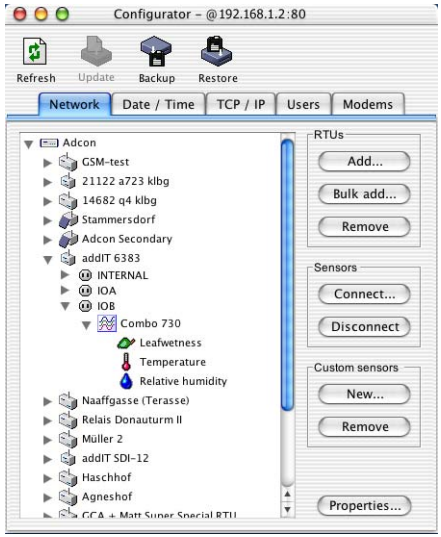


Figure 11. The Configurator GUI

9. Your network configuration, including the sensors and the custom sensors you previously had in addVANTAGE 3.4x and in the addVANTAGE Manager, are now uploaded to your gateway. You can quit the Configurator or proceed to further configuration as described in "Administering your Telemetry Network" on page 35.

Note: If the graphic tree is not changed to reflect the new configuration after the import operation, then probably you have exhausted the maximum number of active RTUs allowed. For more details see also "Entering the license number" on page 31.

After the above steps are completed, your gateway is ready for normal operation. If the A440 Wireless Modem is connected as described in "Installing the Outdoor Unit" on page 16, data from the RTUs in the field will continue to be retrieved and stored into the gateway's internal memory. In addition to the frames required for addVANTAGE 3.4x, new sensor data will also be stored in the format required by the addVANTAGE 4 Pro software.

You can in fact continue to use both addVANTAGE versions at the same time: addVANTAGE 3.4x over the serial line and addVANTAGE 4 Pro over a TCP/IP network. For installing and configuring addVANTAGE 4 Pro as well as some additional tips about migrating from addVANTAGE 3.4x to addVANTAGE 4 Pro, refer to the User Guide supplied with the addVANTAGE 4 Pro CD-ROM.

Configure GPRS usage

Firmware and Routers

The A840 Gateway must have an IMG update to allow GPRS connections. This firmware must be 3.8.0 or higher. In addition; the A840 GPRS feature must be activated with the license key! Look for the string "GPRS" in the license key string.

The new firmware will expect GPRS connections on port 81. Use your router/firewall and NAT/PAT (Network/Port Address Translation) to direct incoming connection requests to port 81 and the private IP of the A840 Gateway.

Configure the A840 for use with GPRS RTU's

Once the A840 firmware has been updated, the RTU must be configured for GPRS. Open the Configurator, create a GSM RTU. Once the device is created. the properties for the new RTU will allow you to select it

GPRSSECRET <secret>

This has to be set (a 32 bit integer number, 0 to 4294967295). This has to be set on the RTU as well as on the gateway. This is a sort of shared secret or password.

Enter the Authorization code; it must EXACTLY match the GPRSSECRET entered in the RTU (see GPRSSECRET). Check the "Close connection after polls" check box.

If you want to change the A840 related configuration of a GPRS-unit via Emu3ap: enter the Authorization code see GPRSSECRET) as phone number (via Emu3ap's phone command), but precede the number by one or two asteriscs (*).

One asterisc means that the IP connection to the GPRS unit will stay open all the time (until an idle timeout occurs, see GPRSTMOUT, or any other condition which disconnects the IP or GPRS connection occurs). Two asteriscs direct the A840 to close the IP connection to the GRPS unit after the poll is done. In this case the next poll will take place when the GPRS unit connects to the A840 the next time. See also GPRSCONNALIGN, GPRSCONNINT, and GPRSTMOUT commands on the GPRS unit in order to select a working configuration.

There are two new commands in Emu3ap, GPRSSTAT which allows you to see if a GPRS unit is connected to the Gateway, and GPRSDISCONNECT to disconnect the TCP connection of the GPRS unit from the Gateway.

The "POLL <id>" command will trigger a poll cycle of the Gateway, if the GPRS unit is connected at the time the command is entered.

Then there is a command CALLJOURNAL, which list the call made to GSM RTUs and connections that were accepted from GPRS RTUs.

Administering your Telemetry Network

The Configurator helps you configure and administer your telemetry network. If you are familiar with the addVANTAGE Manager 3, you can think of the Configurator as its replacement, except that the Configurator also performs some of the tasks previously performed in addVANTAGE 3 itself.

With the Configurator, you can complete all these tasks in one place:

- insert and delete RTUs
- connect and disconnect sensors to and from the RTUs
- add and modify properties of the base station and RTUs
- add, delete, and modify custom sensors
- make backups of and restore the gateway's configuration

Editing the Base Station's Properties

If you need to, you can change some of the Base Station properties, such as its name, frequency, channel spacing, poll interval, and others.

After you have started the Configurator and logged in (as described in Step 2 through Step 5 on page 23), select the root element in the tree (the topmost device) and click **Properties**. You can also right-click the base station and select **Properties** in the pop-up.

Following parameters can be changed:

- The *name* of the server; this has no implications to addVANTAGE or the Internet name of the gateway.
- The network operating *frequency* and the *channel spacing*. These parameters must be identical to those programmed into the RTUs.

WARNING Don't change the frequency and channel spacing of your base station if you have no reason to do so. Apart from the fact that your network may cease to operate properly (you need to perform the same changes on each individual RTU in the network), it might also constitute an infringement to your local or national radio regulations.

- The *poll interval*, in seconds (default 900). The poll interval is the distance in time between two consecutive requests to a certain RTU. You can also set individual poll intervals for each device (see also "Adding New RTUs" on page 38).

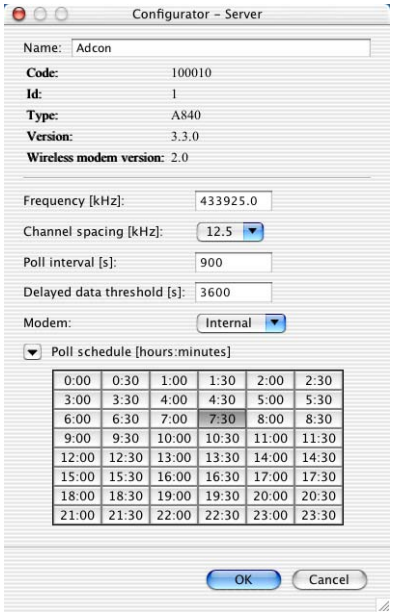


Figure 12. Base Station Properties

- The *threshold* that defines when data is missing from a certain RTU. If no data is collected from a certain RTU for the specified time interval (in seconds), the data from this RTU will first be signalled in addVANTAGE as missing, then when (and if) later retrieved, marked as being delayed (default 3600).

Note: This functionality is supported in addVANTAGE 3.4x only.

- The **Modem** combo-box defines what modem to use for polling the A733GSM/GPRS stations. By default, the built-in modem is used if it is installed. For gateways without internal modem, the selection is restricted to the external modem.
- A *poll schedule* than can be used to poll the RTUs at certain hours of the day (this functionality is displayed after clicking the back arrow left of the **Poll schedule** message). You can select poll times every half an hour. A dark button means that a poll will be made at the specified time of the day. This is a global parameter, however you can set a different poll schedule for each individual RTU (see next section, "Adding New RTUs" on page 38).

Note: The poll schedule is applicable only to the A733GSM/GPRS RTUs.

Adding New RTUs

If you need to add new RTUs, you first need to know their serial number, symbolic names (usually the place where they are installed), and the stations through which they have to be routed, if any. For the A733GSM/GPRS RTUs, you don't need a route, but you will need its data phone number (for more details on the data phone number issue, please consult the *addWAVE User Guide*).

Note: The RTU must be first installed in the field. For information about installing RTUs, consult the respective RTU user manual.

After you have started the Configurator and logged in (as described in Step 2 through Step 5 on page 23), proceed as described below:

1. Click **Add** in the **RTUs** section of the Configurator window to display the window shown in Figure 13.
2. Complete the **Name** and **Code** fields and select the appropriate RTU model (addIT, addWAVE, or A730MD). If you don't know which model the RTU is, you can also select UNKNOWN and the first time the gateway contacts the device, it will update this information. Note, however, that you cannot configure sensors until the RTU has been identified.

Note: A733GSM/GPRS RTUs cannot be automatically identified.

3. To specify a starting date for polling the RTU, select the **Collect data since** combo box to display a drop-down calendar, then select the date. By doing this, you won't have to wait until the base station retrieves data that you don't need from those RTUs with large on-board storages.

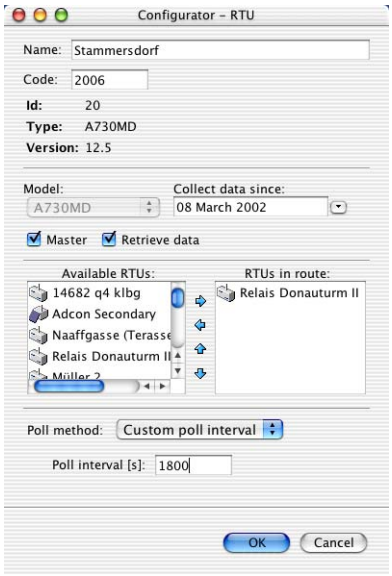


Figure 13. Adding a new A733 RTU

4. If you plan to use the gateway to initialize the RTU's internal real-time clock, click the **Master** checkbox.
5. If you want data from the selected RTU stored on your gateway, click the **Retrieve data** checkbox (if the RTU is used only as a relay station, do not select this option).
6. You can also use this window to set up a route with your new RTU (applicable only if you selected an A733RTU). To do this, select an RTU from the left pane and click the right-arrow button to move it to the right pane. (To remove an RTU from the route, select it in the right pane and click the left-arrow button to move it to the left pane.) To change the order of the relaying stations, select an RTU and click the up or down arrows. Note that the first station in the list is closer to the base station and the last station in the list is closer to the RTU.
7. Enter the **Phone number** of the RTU (applicable only if you selected an A733GSM/GPRS RTU). The phone number is supplied to you by your cellular network operator (for more details, please read also the *addWAVE User Guide*).

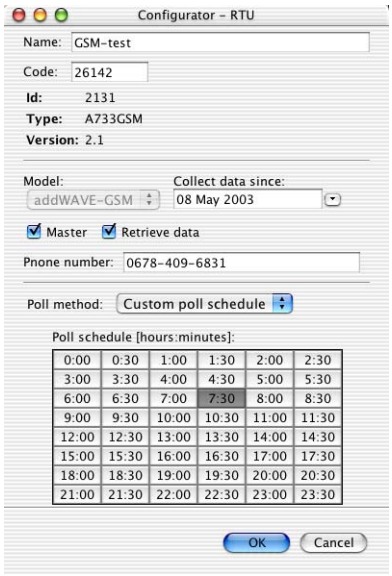


Figure 14. Adding a new A733GSM/GPRS RTU

8. Select the **Poll method**: this can be either:
   a. the default poll interval (as set on the server node, see "Editing the Base Station's Properties" on page 35),
   b. a custom poll interval,
   c. the default poll schedule (as set on the server node).
   d. or a custom poll schedule.

Note: For the standard A733 RTUs, the poll schedule is not applicable; only poll intervals are accepted.

9. When you're finished, click **OK**.

Note: If a message box appears stating that you are not allowed to add the RTU, then probably you have exhausted the maximum number of active RTUs allowed. For more details see also "Entering the license number" on page 31.

10. To update the network with these changes, click **Update** or select **File ▸ Update**.

Special care should be given when programing A733GSM/GPRS RTUs. Polling such an RTU is associated with calling costs, therefore you have to make a reasonable compromise between the

data latency and the number of calls per day in order to keep the costs down. In a typical scenario one or two calls per day would be a good compromise. The call itself takes about two minutes to download data collected for one day, assuming a typical sensor configuration. If you have many sensors and/or they are sampled more often that usual (i.e. once every 15 minutes), then you should set your poll schedule more often. In general, a call should not extend beyond two to three minutes.

Note: The A733GSM/GPRS firmware has a provision to cut the connection after approximately 700 data slots (that corresponds to about one week of data assuming standard sensors); this is also in order to avoid excessive telephone costs.

An efficient way to add RTUs is to use the bulk add feature. In the **RTUs** section of the Configurator, click **Bulk add** to display the window shown in Figure 15.

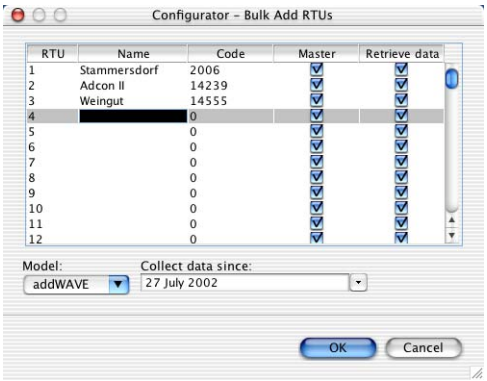


Figure 15. Bulk add RTUs edit box

With this feature, you can create several RTU configurations, adding them to the gateway's list at the same time.

Editing Properties of Existing RTUs

To view or change properties of existing RTUs, select the RTU you want to edit in the Configurator window and then click **Properties**. You can also right-click an RTU and select **Properties** from the pop-up. Most operations described in the previous section can also be performed in the Edit RTU window. After making the

changes, you must confirm them by clicking **Update** or selecting **File ▸ Update**.

Note: If you change the code of an RTU that is also used as a router for other RTUs, the changes will automatically be performed for all the routes affected.

Deleting an RTU

If you decide to remove an RTU from your telemetry network, all you have to do is select the RTU in the Configurator window and click **Remove** in the **RTUs** section. You will be prompted to confirm the deletion; if you choose **Yes**, the device will be removed permanently from the network. To confirm the change, click **Update** or select **File ▸ Update**.

Connecting Sensors

If you connect (or attach) sensors in the field to one or more RTUs, you need to inform your gateway about the type of sensors and the connectors where they are attached.

1. In the Configurator window, expand the RTU tree so that you can see the connector you want to attach a sensor to (you can make multiple selections by using <ctrl> click if several connectors use the same type of sensor).

Note: You can also select multiple RTUs (in addition to connectors) for bulk connecting similar sensors.

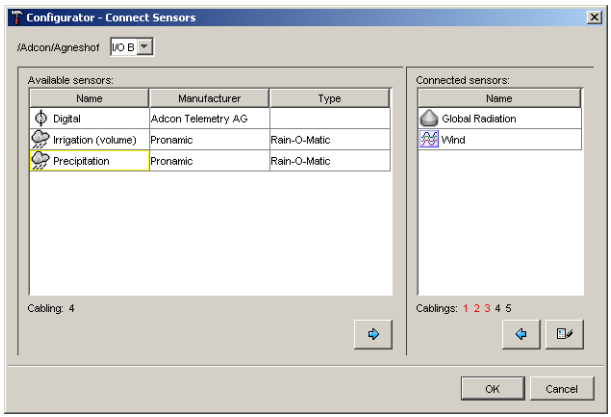


Figure 16. Connecting Sensors

2. Click **Connect** in the **Sensors** section (you can also right-click a sensor and select **Connect Sensors** from the pop-up). The window shown Figure 16 is displayed.From the left pane, select the sensor you want to connect and click the right-arrow button.
3. If you want to edit the name of a sensor already connected, select the sensor, then click the edit button and perform the editing.
4. Click **OK** when you are finished.
5. After connecting the sensors, confirm the changes by clicking **Update** or by selecting **File ▸ Update**.

SDI-12 Sensors

Adcon systems are also compatible with SDI-12 sensors. However, the following limitations apply:

- SDI-12 sensors can be attached only to RTU models A723 and A733 by means of the special SDI-12 adapter (obtainable from Adcon Telemetry GmbH).
- Only a maximum number of 20 values can be collected over the SDI-12 bus; these values can be retrieved either all from one sensor (multiple values sensors), or each one from a different sensor, or any combination of the two.
- The A723 and A733's firmware version must be 2.0 or greater.
- The A440 Wireless Modem's firmware version must be 2.0 or greater.
- The A840 Telemetry Gateway must have software version 3.0 or greater.
- SDI-12 sensor values are not readable via the serial interface of the A840 or via the emulation software (emu3ap). In other words, you cannot use SDI-12 sensors in addVANTAGE 3, but only in addVANTAGE 4 Pro.
- If you use SDI-12 sensors on an RTU, the routing of such an RTU is possible only via A733 RTUs; if you use A730MD RTUs on such routes, data won't be relayed.

To connect an SDI-12 sensor, proceed as follows:

1. In the Configurator window, expand the RTU tree so that you can see the connector you want to attach a sensor to. If the device selected is SDI-12 compatible, an **SDI** connector will be visible in addition to the standard analog **I/O** connectors.

2. Click **Connect** in the **Sensors** section (you can also right-click a sensor and select **Connect Sensors** from the pop-up). The window shown Figure 17 is displayed.

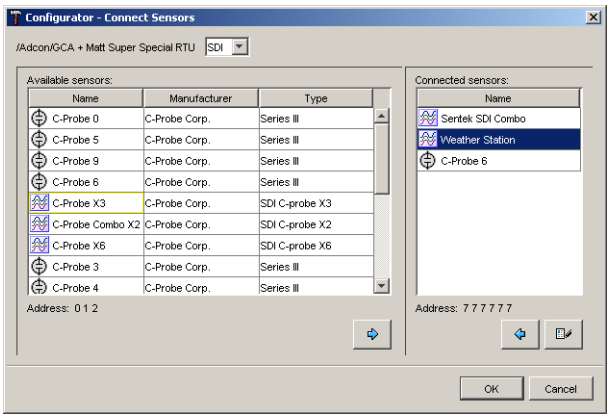


Figure 17. Connecting SDI-12 Sensors

3. From the left pane, select the sensor you want to connect and click the right-arrow button.
4. If you want to edit the name or the SDI address of a sensor already connected, select the sensor, then click the edit button and perform the editing.
5. Click **OK** when you are finished.
6. After connecting the sensors, confirm the changes by clicking **Update** or by selecting **File ▸ Update**.

Disconnecting Sensors

If you want to disconnect (or detach) sensors from one or more RTUs, do the following:

1. In the Configurator window, expand the RTU tree until you locate the connector with the sensor you want to disconnect, then select the sensor.

Note: You can also select multiple RTUs (in addition to connectors) for bulk disconnecting similar sensors.

2. Click **Disconnect** in the **Sensors** section (you can also right-click a sensor and select **Disconnect Sensors** from the pop-up).

3. Click **OK** to confirm the deletion.
4. To confirm the changes, click **Update** or select **File ▸ Update**.

Adding Custom Sensors

If you need to connect a sensor to your RTUs that isn't yet supported by a built-in standard sensor driver, you must first define it in the Configurator.

1. In the **Custom sensors** section of the Configurator window, click **New** to display the window shown in Figure 18.

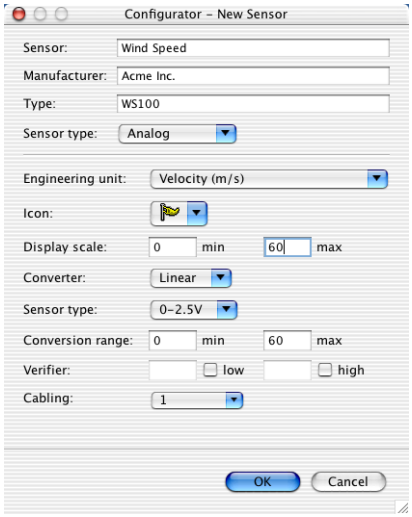


Figure 18. Adding an analog custom sensor

2. Type the default **Sensor** name by which you will recognize your new sensor in the Data Acquisition software.
3. (Optional) Type the **Manufacturer** of the sensor.
4. (Optional) Type specific information about the **Type** of sensor.
5. Select the **Sensor type**: Analog, Pulse Counter, Digital, SDI-12, Combo, or Combo SDI-12. Depending on this selection, the lower window pane will change.
   a. For analog and SDI sensors you will have to select or type:
      - **Engineering Unit**: choose the appropriate engineering unit (EUID) from the combo box. Then

choose an appropriate sensor subclass by selecting the appropriate icon.

Note: If you cannot find a suitable EUID for your sensor, contact Adcon Telemetry.

- **Display scale**, **min** and **max**: type the default values that will be displayed on a trend graph. In most cases these values will be the same as for the **Conversion range** (see below).

b. Specific entries for analog sensors are:

- **Converter**: select **Linear** or **Tabular**. A linear driver needs only two values, a minimum and a maximum, because all values in between are linearly computed by the driver. If you have a nonlinear sensor, you must either select one of the built-in tables or provide your own table with 256 entries (equivalent to an 8-bit accuracy)—in this later case you should select *other...*. The table must be in the form of a text file (similar to the original addVANTAGE 3 files), which will be downloaded to the gateway via the Configurator.
- **Sensor type**: for linear analog sensors, this defines if the sensors deliver a voltage (0-2.5V) or a current (4-20mA using the A502 current-to-voltage converter).

Note: The "4-20mA" setting assumes that you use the A502, which converts a current of 4-20mA to a voltage of 0.4-2.0V. If you want to use a different converter, you have to select "0-2.5V" as sensor type and manually calculate the proper "Conversion range" settings to reflect the physical values the sensor would have for a converted voltage of 0 and 2.5V.

- **Conversion range**, **min** and **max**: the minimum and maximum values delivered by your sensors. The driver computes all the intermediate values from these parameters (it automatically selects the appropriate conversion equation for both 8-bit and 12-bit sensor values).
- **Verifier**: the driver can be instructed to check the input values against some preset thresholds. If the input values are outside the preset thresholds, an invalid sensor value will be signaled.

- **Cabling**: the pin on the connector of the Adcon RTUs that the sensor cable is connected to (refer to the addIT and addWAVE user manuals for a description of the connectors).

WARNING If you plan to add your own sensors to an Adcon RTU, note that the cable between the sensor and the RTU must be shielded, with the shielding properly connected to the casing ground of the RTU. In addition, the maximum cable length should not exceed 10 m (33 ft.).

c. Specific entries for SDI-12 sensors (see Figure 19) are:

- **Address**: type the sensor's SDI-12 address. Note that this is the default address; you can still change the sensor's address when you connect it to an RTU (see also "SDI-12 Sensors" on page 43). The default address is usually 0, but you can get more details from the sensor manufacturer.
- **Method**: this parameter is sensor-dependent and you will need to determine it by consulting the sensor manual. Typically it is 0 (corresponding to the SDI-12 **M0** request); in case of doubt, consult the sensor manufacturer.

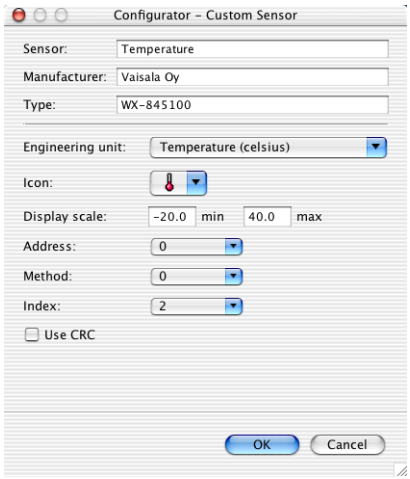


Figure 19. Defining a custom SDI-12 sensor

- **Index**: this parameter is sensor dependent and you will need to determine it by consulting the sensor manual. It represents the position of a sensor value

in the string returned by a multiple values sensor. In case of doubt, consult the sensor manufacturer.

- **Use CRC** checkbox: this instructs the RTU to use CRC request commands over the SDI-12 bus. Before enabling this option, make sure that the sensor supports CRC. In case of doubt, consult the sensor manufacturer.

d. If you chose to generate a new combo sensor (see Figure 20), you will have to provide the individual components:

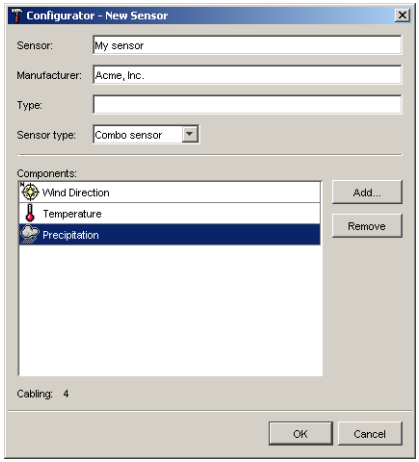


Figure 20. Defining a custom combo sensor

- Click the **Add...** button and select the component sensors you need. When you are finished, click **OK**.
- If you change your mind, you can delete component sensors by clicking the **Remove** button.

  If you generated a Combo sensor, only analog/digital sensors will be shown and if you generated an SDI Combo sensor, only SDI-12 sensors will be shown.

6. After entering/editing all the required values, click **OK**.
7. To confirm the changes, click **Update** or select **File ▸ Update**.

Editing Custom Sensors

To edit custom sensors, select a custom sensor driver and click **Properties**. Most available functions are identical to those offered by the "Add custom sensor" command described on page 45.

Deleting Custom Sensors

If you don't need a custom sensor any more, you can delete it simply by selecting it and clicking **Remove** in the **Custom sensors** section of the Configurator window.

Performing Backups

The Configurator allows you to perform backups of the configuration. Backups are needed if either hardware or software in your gateway fails (the configuration is stored internally in non-volatile memory, so a power failure won't destroy it), or due to an operator error. It is also handy when you want to replace one gateway with another.

To perform a backup, click **Backup** or select **File ▸ Backup**. You are prompted to name the resulting configuration file. Type an appropriate name (for example, the name of the gateway) and click **OK**. The configuration file will be saved on your computer.

Performing Restores

If you lost your configuration, or if you need to replace your gateway, you can simply upload the configuration file into the gateway by performing a restore. To do this, click **Restore** or select **File ▸ Restore**.

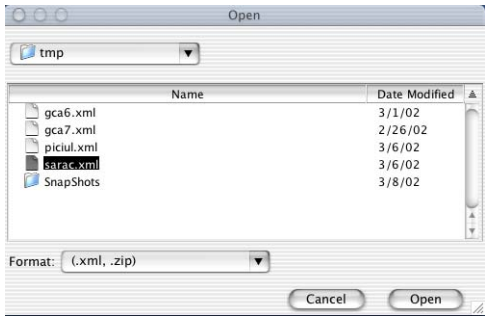


Figure 21. Performing a restore

A window similar to the one shown in Figure 21 is displayed. Browse until you find the configuration file that was previously saved from a backup operation, then select the file. Click **Open** to upload the file to the gateway.

Note: The Restore operation is in fact identical to the Import from addVANTAGE 3.46 operation described in Step 8 on page 32.

About the LED Indicators

The A840 Telemetry Gateway has five LED indicators on its panel. The indicators have a different significance based on whether the device is operating in the usual Linux mode or in hermit mode (for upgrading software and in other unusual circumstances). "Advanced Functions" on page 53 provides more details about the operating modes.

Linux Mode

LAN and ACT

These two LEDs are active only if you have a network cable attached to the gateway and the network section is enabled. In addition, the mains (110/220 volt) must be present, that is, the gateway must not run out of its internal battery. The LAN LED is lit while the Ethernet line is operational, and the ACT LED shows activity on the Ethernet line.

USR and RUN

The USR LED lights when communication takes place over the RS-485 line, that is, to or from the A440 Wireless Modem unit. The RUN LED is currently not used in Linux mode.

PWR

Table 2 summarizes the PWR LED significance:

Table 2. The PWR LED Indicator

Mains	PWR LED	Battery
Present	ON	unknown

Table 2. The PWR LED Indicator

Mains	PWR LED	Battery
Not present	ON 15/OFF 1 (2 sec)	Full
	ON 8/OFF 8 (2 sec)	Half full
	ON 1/OFF 15 (2 sec)	Almost empty

When the mains is not present, the PWR LED blinks at a 2-second rate. The on/off state of the blink shows you how much power is left in the battery. For example, as you can see in Table 2, if the LED is lit for most of the two-second period (on 15/off 1), the battery is full. But if the LED is lit only briefly (on 1/off 15), the battery is almost out of power.

Hermit Mode

In hermit mode (see also "Administrative Tasks at the System Level" on page 53) only the USR, RUN, and PWR LEDs are significant. Table 3 summarizes their significance.

Table 3. USR, RUN, and PWR LED Indicators

State	USR	RUN	PWR
Reset/Power On Reset	ON	ON	ON
Hermit 5 sec. wait	OFF	OFF	ON
Hermit got key	OFF	OFF	2 sec ON/2 sec OFF
Hermit boots	OFF	ON	2 sec ON/2 sec OFF
Hermit failed	OFF	ON	0.5 sec ON/0.5 sec OFF

Chapter 3. Advanced Functions

This chapter describes some advanced functions that you can perform on your new base station. If you find that certain functions are too complicated or you don't understand them, your distributor can do what you need for you.

You can perform these types of operations on your gateway:

- administrative tasks at the system level
- operations at the application level (RTUs pinging, configuring, data checks, and so forth)

Administrative Tasks at the System Level

Because the A840 Telemetry Gateway is based on Linux, many administrative tasks can be performed on it. Linux is a powerful, rich, and exhaustive environment whose description is outside the scope of this manual. You can consult any of the manuals that have appeared lately on the subject. Under normal use you will not have to deal with such tasks, but we have included this information for those rare occasions when something goes wrong or you need to do some minor maintenance such as changing passwords, firmware upgrades, and the like.

Stopping and Starting the Telemetry Gateway

The telemetry gateway has a built-in rechargeable battery that is software controlled. If for some reason you need to stop the telemetry gateway (for example, shipping for service or storing for longer time spans), you need to shut it down by switching off the battery internally, then unplugging the power cable. Proceed as follows:

1. Shut down addVANTAGE and log into the gateway by means of a communication terminal (for example, Hyperterminal in Windows, Zterm or BlackNight on a Mac, or minicom in Linux). Use the standard parameters:
   - 19200 baud
   - 1 stop bit
   - No parity
   - Hardware protocol
   - Send CR after LF.

   A login prompt such as the following appears:

   ```
   Telemetry Gateway A840 Version 3.8
   (C) 2006 Adcon Telemetry GmbH
   A840 login:
   ```
2. At the **`A840 login`** prompt, type **`root`**.
3. At the password prompt, type **`840sw`**.

Note: The default root password is programmed at the factory. For security reasons, Adcon strongly recommends that you change it. To do so, see "Changing Passwords" on page 55.

4. The **>** prompt appears. Type **`halt`**.
5. Several messages appear on your terminal. After you see the last message shown below, you can unplug the unit from the power outlet, because the internal battery is now off.

   ```
   The system is going down NOW !!
   Sending SIGTERM to all processes.
   Terminated.
   Sending SIGKILL to all processes.
   The system is halted. Press Reset or turn off
   power.
   System halted.
   ```

To start the unit, plug the mains cable into the power outlet. After the boot procedure finishes (which takes about 20 seconds), the unit is fully operational.

Changing Passwords

For security reasons, Adcon recommends that you change the password of your gateway. The unit comes from the factory with only two users: `root` and `adv`. The root user is intended only for administrative tasks, while adv is used by the addVANTAGE software. The password for root is *840sw* and for adv it is *addvantage*. It is highly recommended that you change both passwords; however, if you do so, be sure to keep the new passwords in a secure location.

Complete the following steps to change the root password:

1. Log into the gateway as **`root`**, as described in the first step of "Stopping and Starting the Telemetry Gateway" on page 54.
2. At the login prompt, type **`passwd`**.
3. The system prompts you to enter a new password:

```
Changing password for root.
Enter the new password (minimum of 5, maximum of 8 characters).
Please use a combination of upper and lower case letters and numbers.
Enter new password:
```

4. After you enter the new password, the system prompts you to reenter it to be sure that you didn't mistype it. Reenter the password:

```
Re-enter new password:
Password changed.
```

To change the password for the user adv, the steps are the same, except that you start with:

```
passwd adv
```

If you do change the password for the user adv, you must also change it in addVANTAGE. To do this, edit the *agroexp.ini* file. Find the section [Communication], which has the following entries:

```
User=adv
Password=addvantage
```

Do not change the user name, but type whatever new password you assigned to the adv user. The two password strings in the hardware and software must be identical, or addVANTAGE will not be able to download data from the base station.

*Note: Only addVANTAGE 3.45 or higher is compatible with the A840 Telemetry Gateway. If your software is not up to date (check **File ▸ About** in addVANTAGE), download the latest update from Adcon's web server (http://www.adcon.at). The updater will update only version 3.40, so if you have an older version, you'll need to contact your Adcon representative.*

Losing a Password

If you misplace or forget your gateway password, follow these steps to change it with a new password:

1. Open a terminal and connect to the gateway, as described in the first step of "Stopping and Starting the Telemetry Gateway" on page 54.
2. Reset the A840 device by gently inserting a paper clip in the hole on the backplane near the serial line connector (see Figure 22). After you feel a click, remove the paper clip.

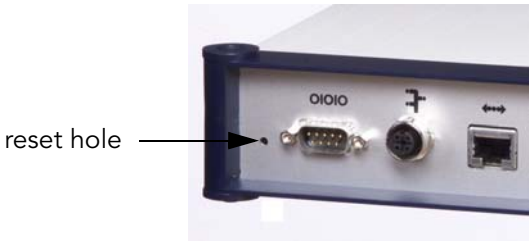


Figure 22. Resetting an A840 Device

3. Carefully follow the messages on the terminal. When you see the "Waiting 5 sec for <ESC>, <DEL>, or <BS> to enter console" message, press one of those keys.
4. At the **hermit>** prompt, type **`linux init=/bin/sash`**.
5. After Linux finishes booting, you will see the **>** prompt. Type the following, ending each line by pressing Enter:

```
stty igncr
cd /etc
/etc/rc.sysinit
```

6. Now you can change the root password as described in Step 2 through Step 4 of "Changing Passwords" on page 55.
7. After you get the "Password changed" message, type **`exec /bin/init`**.

The Linux login prompt is displayed. Now you can log in with your new root password.

Software Upgrades

Both components of your base station have their software stored in Flash EEPROMs. This means that you can upgrade the software at any time and you won't lose any data while the power is switched off.

The A840 Telemetry Gateway has two kinds of software: a bootloader (called *hermit*) and the software proper. Both of them can be upgraded. For free upgrades, check Adcon's web site for the latest files for your gateway.

Upgrading the bootloader

1. Make sure you have the correct file. It should have an .hmt extension (for example, hermit-010626.hmt).
2. Log in as root, as explained in the first step of "Stopping and Starting the Telemetry Gateway" on page 54.
3. Type **`reboot`**.
4. Several messages will appear (the procedure can take some 20 seconds). Wait until you see the following message:

```
Please stand by while rebooting the system.
Restarting system.
Waiting 5 sec for <ESC>, <DEL>, or <BS> to
enter console
```

5. At this point, press the <ESC>, <DEL>, or <backspace> key to enter hermit (the bootloader). The following message will be displayed:

```
Hermit V1.2.3 @19:14:31, Oct 30 2001
hermit>
```

Note: Systems with older hermit versions may display a message stating that any key can be pressed.

6. Now type **upgrade**.
7. The system waits for a file to be sent using the Y-modem protocol. Configure your terminal program to send via the Y-modem protocol, switch your communication program to Send, and send the .hmt file from your computer.
8. After a series of messages, the system asks for confirmation. Press Y. The bootloader will now be replaced.
9. Restart the gateway by typing **linux**.

After Linux finishes booting, the gateway is again ready for use.

To upgrade the system

1. Make sure that you have a new system image (the image file name usually ends with .img).
2. Log in as root, as explained in the first step of "Stopping and Starting the Telemetry Gateway" on page 54.
3. Type **reboot**.
4. Several messages will appear (the procedure can take some 20 seconds). Wait until you see the following message:

   ```
   Please stand by while rebooting the system.
   Restarting system.
   Waiting 5 sec for <ESC>, <DEL>, or <BS> to
   enter console
   ```

5. At this point, press the <ESC>, <DEL>, or <backspace> key to enter hermit (the bootloader). The following message will be displayed:

   ```
   Hermit V1.2.3 @19:14:31, Oct 30 2001
   hermit>
   ```

Note: Systems with older hermit versions may display a message stating that any key can be pressed.

6. Change the serial port speed by typing **set speed 115200**.
7. Change the speed of the communications program to 115200 baud, then press Enter. You'll see the **hermit>** prompt again.
8. Type **upgrade**.

Note: On systems with hermit release 1.2.3 or higher, you can also type `upgrade -y 115200`*, and then change the speed of your terminal to 115200 baud. After starting the Y-modem transfer, the gateway will not prompt you until the upgrade*

procedure is finished, after which you must change the speed of your terminal back to 19200 baud.

9. The system waits for a file to be sent using the Y-modem protocol. Configure your terminal program to send via the Y-modem protocol, switch your communication program to Send, and send the .img file from your computer. The download can take several minutes.
10. After a series of messages, the system asks for confirmation. Press Y. The system software will now be replaced. The procedure can take several minutes.

Note: Do not switch the system off during this time!

11. When the procedure is finished and you see the **`hermit>`** prompt again, restart the gateway by typing **`linux`**.
12. While the system is booting, return your terminal's speed to 19200.

After Linux finishes booting, the gateway is again ready for use.

Operations at the Application Level

From a telemetry point of view, the A840 Gateway has two interfaces to the outer world:

- Through the *emu3ap* application (an emulator of the A730SD receiver). Using this application puts you in an environment almost identical to the A730SD receiver. You can access the emu3ap interface over the serial line or via telnet.
- The addUPI interface (Adcon URL Programming Interface), which is an http-based communication specification. This interface is accessible only over the network (port 80).

The addUPI interface is described in a document that you can request at no cost from Adcon Telemetry GmbH. The current firmware implements the addUPI specification 0.92 (notifications are not yet implemented).

You can access the emu3ap application in either of these ways:

- Log in as user `adv` and you are automatically dropped to an emu3ap shell.
- Log in as `root` and, at the Linux prompt, type **`emu3ap`**.

In either case, you need to press **Enter** until you get a message stating that you are in the emu3ap command line interpreter (CLI).

Note: You can reach the gateway over the serial port or over the built-in modem. If the network is enabled, you can also reach it over Ethernet. You can have more than one emu3ap process running at the same time.

Although the emu3ap software and the A730SD are very similar, there are some slight differences in that some commands were eliminated and other features were added.

WARNING The emu3ap interface is implemented in the A840 Telemetry Gateway only to ensure backward compatibility with older Adcon software. This interface is obsolete and is not recommended for new designs. All new projects should be based on the addUPI interface. The description of this interface is given here only for completeness.

Commands Accepted by the emu3ap Emulator

As with the A730SD, the emu3ap software supports five different classes of commands:

- Configuration (including the SET commands)
- Data
- Administrative
- Direct radio
- Modem/GPRS/GSM

Configuration Commands

This category includes commands that configure various parameters of the emu3ap software.

INSERT

DESCRIPTION

Inserts a new device in the gateway's internal list.

PARAMETERS

The device ID number

```
INSERT <dev>
```

phone number (if the device is capable of GPRS/GSM functionality) please check the PHONE command for further parameters eg. shared secred.

```
INSERT <dev> [phonenumber]
```

RETURNS OK or an error message.

REMARKS A device is a remote measuring station (for example, A730MD, A720, A723, or A733). Once the device has been inserted, various operations can be performed on it. Every 15 minutes, the gateway automatically asks the devices in the list to supply a new slot of data over radio. The 15-minute interval can be changed with the **SET SLOT** command (see page 70).

EXAMPLE

```
insert 2333
OK
```

Insert a device.

```
insert 26137 06641111111
OK
```

To insert a device together with a phone number

REPLACE

DESCRIPTION Replaces one device with another. Data from the original device is not lost, but is associated with the new one.

PARAMETERS The original device ID and the new device ID.

RETURNS OK or an error message.

REMARKS The **REPLACE** command effectively replaces one device with another. Use it when replacing a station in the field. In the following example, device 2333 is replaced by device 2046.

EXAMPLE

```
replace 2333 2046
OK
```

DELETE

DESCRIPTION Deletes a device.

PARAMETERS The device ID.

RETURNS OK or an error message.

REMARKS None.

EXAMPLE

```
delete 2046
OK
```

ROUTE

DESCRIPTION

Sets a route for the target device.

PARAMETERS

The device ID and the route description (composed of the routing devices' IDs). To clear a route, provide only the target device ID (in other words, a null route).

RETURNS

OK or an error message.

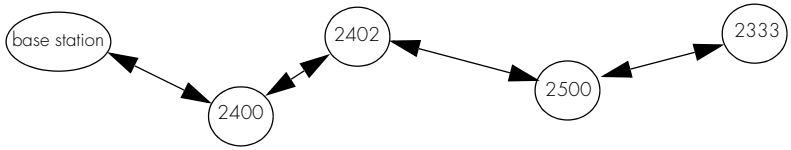
REMARKS

The **ROUTE** command updates the internal descriptors for the specified target device. Normally, after inserting a new device, it is assumed that this device is to be called directly, that is, no routing stations are in between. If a station has to be routed over one or more other stations, the gateway must know the path to be used when calling that station. The route is given as a list of devices, always starting from the gateway and progressing to the end-station (see example below). Note that the number of relay stations is limited to 8, which is in practice more than sufficient. To verify the route a specific device uses to communicate, use the **INSPECT** command (see page 79).

EXAMPLE

```
route 2333 2400 2402 2500
OK
```

In the example above, station 2333 is routed over a path defined by devices 2400, 2402 and 2500. This corresponds to the following topographic situation:



To clear a route, use the same command, but include no parameters for the path:

```
route 2333
OK
```

MASTER

DESCRIPTION

Set/resets the master flag of a specific device.

PARAMETERS

The device ID and the flag value (on/off).

RETURNS

OK or an error message.

REMARKS

When a station has a master flag set, the gateway uses that station for date/time synchronization. It is important to understand that each remote station has an internal real-time clock, used to stamp the stored data. This clock has to be initialized and synchronized, which the gateway does automatically when either of the following occurs:

- A station returns invalid time values (null or out of date).
- The host computer synchronizes the time of the gateway (such as with a **SET TIME** command).

If the remote stations in a network are to be polled by more than one base station, only one of those base stations should be used to synchronize the time of a particular RTU. Otherwise, the remote stations could get confused due to the time differences that might occur between different base stations. Consequently, in a multi-gateway network, only one gateway should have the master flag set for a specific station; all other gateways should treat it as a slave.

To verify whether a device has the master bit set, use the **INSPECT** command (see page 79).

EXAMPLES

```
master 2333 on
OK

master 2333 off
OK
```

ACTIVE

DESCRIPTION

Set/resets the active flag of a specific device.

PARAMETERS

The device ID and the flag value (on/off).

RETURNS

OK or an error message.

REMARKS

Activates/deactivates the poll of a device. If a device has this flag set, then data will be collected over radio from this device; if the flag is reset, the gateway will cease polling it.

EXAMPLES

```
active 2333 on
OK

active 2333 off
OK
```

EXTEND

This command is deprecated in emu3ap. It has been included for compatibility purposes, but always returns OK.

POLLCONF

DESCRIPTION

Get/set poll configuration for a specific device.

PARAMETERS

The device ID and the polling intervall/time.

```
POLLCONF <dev>
```

get poll configuration

```
POLLCONF <dev> INTERVAL [param]
```

set poll configuration

```
POLLCONF <dev> TIMES [param]
```

RETURNS

Pollconfiguration or an error message.

REMARKS

Return of Error 3 means that the configuration is not set.

For GSM functionality poll times should be used, for ISM (normal radio communication) poll interval.

EXAMPLES

```
pollconf 15636 interval 900
OK
```

Sets the interval for the specific device to 900 seconds.

```
pollconf 26137 times 6,14,20,22
OK
```

Sets the poll times for the specific device to 6,14,20 and 22 o'clock.

```
pollconf 15636 interval
Dev: 15636
Poll interval: 900 seconds
OK
```

Gets the poll interval.

```
pollconf 26137 times
Dev: 26137
Poll times: 6,14,20,22
OK
```

Gets polling times.

DEV

DESCRIPTION Gets list of devices.

PARAMETERS None.

RETURNS A list of devices. The list is self-explanatory.

REMARKS None.

EXAMPLES

```
dev
RTU    Notifications pending
---------------------------
52     NONE
15636  PORT+ANLG
3768   NONE
OK
```

NPND

DESCRIPTION Gets list of devices wich have notifications pending.

PARAMETERS None.

RETURNS List of devices and pending notifications.

REMARKS None.

EXAMPLES

```
npnd
RTU    Notifications pending
---------------------------
15636  PORT+ANLG
OK
```

NTF

DESCRIPTION Configure notifications.

PARAMETERS *Note: IOA..D represents die IO range from A to D.*
1..3 represents the range from one to three.

NTF <dev> IOA..D [cabling]

Queries the notification configuration for the given RTU and channel

NTF <dev> IOA..D 5 [ON-or-OFF]

Enables/disables digital port notifications for the given RTU and channel

NTF <dev> IOA..D 1..3 [ON-or-OFF]

Disables analog port notifications for the given RTU and channel

NTF <dev> IOA..D 1..3 ABOVE [limit]

Enables analog port notifications if the analog value is higher than limit for the given RTU and channel

```
NTF <dev> IOA..D 1..3 BELOW [limit]
```

Enables analog port notifications if the analog value is lower than limit for the given RTU and channel

```
NTF <dev> IOA..D 1..3 OUTSIDE [lower-limit]
[upper-limit]
```

Enables analog port notifications if the analog value is outside the limits for the given RTU and channel

```
NTF <dev> IOA..D 1..3 BETWEEN [lower-limit]
[upper-limit]
```

Enables analog port notifications if the analog value is between the limits for the given RTU and channel.

RETURNS

OK or error.

REMARKS

The limit values must be given as physical values as configured in the Configurator application, but without the physical unit, e.g. 37.0 for 37 degrees celsius.

Note: you can neither enable notifications for pulse counters (cabling 4) nor for sensors that are not connected using the Configurator application!

EXAMPLES

```
NTF 1234 IOA 5 OFF
```

Disables digital port notifications on IOA of RTU 1234

```
NTF 5678 IOC 5 ON
```

Enables digital port notifications on IOC of RTU 5678

```
NTF 1234 IOA 1 OFF
```

Disables analog notifications on IOA cabling 1 of RTU 1234

```
NTF 1234 IOA 2 ABOVE 70.0
```

Enables analog notifications if the physical value for IOA cabling 2 is above 70.0 (whatever physical unit the sensor has)

```
NTF 5678 IOC 3 OUTSIDE 12.0 34.0
```

Enables analog notifications if the physical value for IOC cabling 3 is outside the range of 12.0 to 34.0 (whatever physical unit the sensor has)

SENDNTF

DESCRIPTION

To configure how to send notifications from the A840 to you.

PARAMETERS

SENDNTF

Queries the configuration

SENDNTF SYSLOG

Requests notifications to be written to the system log only, which effectively disables sending notifications.

SENDNTF SMS [lock-time] [format] [phone-nr]

Requests notifications to be sent as SMS, at maximum every "lock-time" seconds, to the given phone number. Output format can be NODEID (print the node ID of the sensor causing the notification), TAGNAME (print the sensor name) or PATHNAME (print RTU and sensor name). Parameters not specified use the last (or default) values. Note: the phone number must be given in international format without leading zeroes, whitespace or punctuation characters, for example for country "Austria" (prefix 0043) and provider "A1" (prefix 0664) and the GSM phone number 12345678, the resulting phone number string would be "4366412345678".

SENDNTF SMTP [lock-time] [format] [from-e-mail-addr] [to-e-mail-addr] [server-addr server-port]

Requests notifications to be sent as e-mail over SMTP, at maximum every "lock-time" seconds, using the given e-mail addresses and SMTP server configuration. Note: sending e-mails will only work if the mailserver is configured to accept mails from the gateway sent with the given "from-e-mail-addr" Contact your mailserver administrator to make the appropriate changes if necessary. Ouput format can be NODEID (print the node ID of the sensor causing the notification), TAGNAME (print the sensor name) or PATHNAME (print RTU and sensor name). Parameters not specified use the last (or default) values.

SENDNTF HTTP [lock-time] [base-url]

Requests notifications to be sent as HTTP request, with the URL beginning with "base-url", at maximum every "lock-time" seconds. If you have to use HTTP authentication, you need to specify the user name and password for the HTTP server inside the URL. The general URL syntax is: "http://username:password@hostname:port//path", with the "username", "password" and "port" specifiers being optional. Note: the

password will be transmitted unencrypted since HTTP is used. You should not use the same username/password combination for other purposes.

RETURNS

Ok or error.

REMARKS

None.

EXAMPLES

```
SENDNTF SYSLOG
```

Turns off sending notifications

```
SENDNTF SMS 600 NODEID 4366412345678
```

Requests sending notifications to phone number ++43-664-12345678, the time between two SMS sent is at least 600 seconds. Output format is the node ID.

```
SENDNTF SMS
```

Requests sending notifications as SMS, using the last supplied configuration values

```
SENDNTF SMTP 30 PATHNAME watchdog@adcon.at
trash@adcon.at 192.168.1.2
```

Requests sending notfications as e-mail to trash@adcon.at, using watchdog@adcon.at as the sender e-mail address, the SMTP server at IP address 192.168.1.2 and the default IP port for SMTP servers (port 25). The time between two e-mails sent is at least 30 seconds. Output format is "RTU-Name/Sensor-Name".

```
SENDNTF HTTP 60 http://nowhere.adcon.at/cgi-bin/
notification.cgi
```

Requests sending notifications as HTTP request to the host nowhere.adcon.at, without HTTP authentication.

```
SENDNTF HTTP 60 http://
nobody:nopasswd@nowhere.adcon.at/cgi-bin/
notification.cgi
```

Requests sending notifications as HTTP request to the host nowhere.adcon.at, logging in as user nobody using the password nopasswd.

SET Commands

The **SET** command has many subcommands, because as its name implies, it sets various parameters of the emu3ap application. Most of the **SET** commands have a *get* form, in which only the

subcommand is typed and the emu3ap application returns the requested information.

SET TIME

DESCRIPTION — Sets the internal real-time clock of the emu3ap.

PARAMETERS — The time in the following format: dd/mm/yyyy hh:mm:ss (24-hour clock format).

RETURNS — OK or an error message.

REMARKS — The year may be sent either in two- or four-letter format (for example, 1999 or 99), but the four-letter format is preferred. In addition, the emu3ap also accepts the year 2000 and years thereafter as hundreds, for example, 100 for 2000, 101 for 2001, and so forth. The date/time parameters may be sent with or without leading zeros. The *get* variant **TIME** (with no parameters) returns the current date and time of the device.

EXAMPLE

```
set time 4/6/2001 20:1:0
OK

time
Local time is: Sun Jun 4 20:01:07 2001
OK
```

SET FREQ

DESCRIPTION — Sets the operating frequency and step of the gateway (this parameter is further transmitted to the A440 Wireless Modem).

PARAMETERS — The frequency and step, both in Hz.

RETURNS — OK or an error message.

REMARKS — The *get* variant **FREQ** (with no parameters) returns the actual operating frequency.

EXAMPLE

```
set freq 433925000 25000
OK

freq
Frequency: 433925000, step: 25000
OK
```

In the example above, the gateway plus wireless modem combination operates on 433.925 MHz with channel spacing of 25 kHz.

SET OWNID

This command is deprecated in emu3ap. It has been included for compatibility purposes, but always returns OK.

The radio network ID is programmed at the factory and resides in the A440 Wireless Modem (as for all RTUs, the network ID is the serial number printed on the RTU's label).

SET SLOT

DESCRIPTION — Sets the request rate in seconds (the default is 900, or 15 minutes). Note that the poll time is different from device to device, that is, not all devices will be polled at the same time, but instead based on their insertion time.

PARAMETERS — The request rate in seconds (minimum 10, maximum 10800 seconds, that is, 3 hours).

RETURNS — OK or an error message.

REMARKS — If this parameter is not programmed explicitly, it defaults to 900. The *get* variant **SLOT** (with no parameters) returns the current requesting rate.

EXAMPLE

```
set slot 900
OK

slot
Slot time is 900 seconds
OK
```

SET DELAY

DESCRIPTION — Sets the delay before returning ERROR 15 in case of a temporary radio communication breakdown (see also "GETBLOCK" on page 71).

PARAMETERS — The delay value (minimum 1800, maximum 10800 seconds; in other words, between 30 minutes and 3 hours).

RETURNS — OK or an error message.

REMARKS — If this parameter is not programmed explicitly, it defaults to 3600 seconds (one hour). The *get* variant **DELAY** (with no parameters) returns the current delay value.

EXAMPLE

```
set delay 7200
OK
```

```
delay
Interruption delay is 7200 seconds
OK
```

SET ECHO/NOECHO

This command is deprecated in emu3ap. It has been included for compatibility purposes, but always returns OK. The behavior of the emu3ap software is that ECHO is always on. You can also PING and REQUEST data even for devices that are not shown in the devices list.

Data Commands

This category included commands that return data from the remote stations. The data collected from the stations is stored in the on-board FIFO memory; it can be retrieved based on the station ID and the time stamp. This means that if specific data was retrieved, it can be retrieved again later as long as an appropriate command is given. The data is stored in frames for each station and each time slot (that is, every 15 minutes). The data can be retrieved in any order, each device having its own internal pointers managed by the system. As new data comes in, the old data is overwritten; a "garbage collector" takes care of that. The command that allows this data retrieval is presented below.

GETBLOCK

DESCRIPTION

Returns a block of data found at the current position of the pointer, for the specified device. If a date/time parameter is supplied, **GETBLOCK** searches and positions the internal pointer on that date/time before returning the data block.

PARAMETERS

The device ID and, optionally, the date/time parameter.

RETURNS

A string of data depending on the device type, or an error message. Some important error messages are 14 (no more data) and 15 (radio communication temporary breakdown). The latter means that data might come later if the communication is reestablished.

REMARKS

The number of concatenated frames in a block depends on the frame length (the maximum is 1024 bytes in a block). The bytes are sent without spaces, but with leading zeros if necessary.

EXAMPLE

```
getblock 2006 25/5/2000 15:15:0
1905640F11361509DFC9F8000057AB7F0F1A006702AB7F0F
```

```
01006F021905640F20361509D7C4F8000056A9830F19005B
02A9830F010060021905640F2F361509C9B2F8000056AA80
0F1A006B02AA81100100750219056410023615O9DECAF800
0056AA800F18004F02AA800F0100550219056410113615O9
D8C1F8000056AA7D0F16004D02AA7E0F0100550219056410
20361509D2BBF8000056A87F0F13004002A8800F01004602
190564102F361509DBB2F8000056A97E0F19005102A97F0F
0100590219056411023615O9CAB9F8000056A97C0F1E0045
01A97C0F01004E021905641111361509C6B1F8000055A684
0F1A001F01A6840F0100200119056411203615O9D7C3F800
0055A4870F11001101A4870F01001101190564112F361509
D1C1F8000055A4870F13001601A4870F0000170119056412
02361509DAC3F8000055A5860F12001A01A5860F01001C01
190564121136150 9DBC8F8000055A6830F13002C01A6840F
0100320119056412203615O9D7C5F8000055A5841014001E
01A5840F0100200119056412 2F361509DBC0F8000055A484
0F14001701A4850F0100160119056413023615O9DFCFF800
0055A3860F13001501A38610000015010F
OK
```

To help you understand how the block must be interpreted, the first two frames and the last frame in the block are displayed below (notice the extra carriage returns inserted to help you see the individual frames):

```
1905640F11361509DFC9F8000057AB7F0F1A006702AB7F0F
01006F02
1905640F20361509D7C4F8000056A9830F19005B02A9830F
01006002
...
190564130236150 9DFCFF8000055A3860F13001501A38610
00001501
0F
```

The last byte on the last line is the checksum of the whole block (modulo 256). It is the sum of all the bytes, ignoring the overflows.

Note: The dates for years greater than 1999 are returned in a three-digit format, that is, 100 for 2000, 101 for 2001, and so on.

The data portion of the frame is frame-type dependent, which in turn depends on the RTU that generated it (see "Frame Types" on page 92 for a description of all frame types currently in use). However, separating individual frames from a **GETBLOCK** string is easy if you consider the following:

- Each frame has a header and a data portion.
- Whatever the data portion is, the header has a constant structure and known length (date/time and number of bytes in the data segment).

Thus, to identify the beginning of the next frame you need only parse the date/time and the frame size and then jump to the next frame based on the size of the data portion (adding the frame size

to the current position points in effect to the beginning of the next frame).

GETCONFIG-CSV

DESCRIPTION

Allows the configuration of the text export.

PARAMETERS

The getconfig-csv CGI script accepts the following URL parameters:

`errors=yes|no`

Specifies whether errors are printed as empty fields (default, or errors=no) or printed as error number (errors=yes).

`lines=yes|no`

Specifies whether line numbers are printed (lines=yes) or not (default, or lines=no).

`headers=yes|no`

Specifies whether the header line is printed (default, or headers=yes) or not (headers=no).

`delimiter=string`

Specifies the field delimiter (default is ";"). All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09").

`lineending=string`

Specifies the line ending (default is linefeed). All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09").

`quote=string`

Specifies the string quotation character (default is the double quote). All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09").

`internaltags=yes|no`

Specifies whether internal tags are included in the output (internaltags=yes) or not (default, or internaltags=no).

`rtulist=yes|no`

Specifies whether the RTU list is printed (default, or rtulist=yes) or not (rtulist=no).

`taglist=yes|no`

Specifies whether the TAG list is printed (default, or taglist=yes) or not (taglist=no).

booleanformat=N|T

Specifies whether boolean values are printed as numbers (booleanformat=N) or as strings TRUE, FALSE (default, or booleanformat=T). .

RETURNS

OK or an error message.

REMARKS

To use this command, this feature has to be bought aditionally. A special license key is required.

EXAMPLE

```
getconfig-csv taglist=no
9504;26135;"A733GSM";"RTU 26135 GSM";TRUE;FALSE
9620;27274;"A740";"RTU 27274";TRUE;FALSE
9650;170;"A733";"RTU 170";TRUE;FALSE
OK
```

Requests the RTU list only, the TAG list is not printed.

```
getconfig-csv delimiter=%09
9504 26135 "A733GSM" "RTU 26135 GSM" TRUE FALSE
9620 27274 "A740"    "RTU 27274"     TRUE FALSE
9620 170   "A733"    "RTU 170"       TRUE FALSE
OK
```

Requests both RTU and TAG list, but sets the field delimiter to the tab character instead of the semicolon.

GETDATA-CSV

DESCRIPTION

Returns the text export data.

PARAMETERS

The getata-csv CGI script accepts the following URL parameters:

errors=yes|no

Specifies whether errors are printed as empty fields (default, or errors=no) or printed as #errno (errors=yes).

lines=yes|no

Specifies whether line numbers are printed (lines=yes) or not (default, or lines=no).

headers=yes|no

Specifies whether the header line is printed (default, or headers=yes) or not (headers=no).

delimiter=string

Specifies the field delimiter (default is ";"). All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09").

`lineending=string`

Specifies the line ending (default is linefeed). All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09").

`quote=string`

Specifies the string quotation character (default is the double quote). All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09").

`internaltags=yes|no`

Specifies whether internal tags are included in the output (internaltags=yes) or not (default, or internaltags=no).

`tulist=name[,name]...`

Specifies a comma separated list of RTUs by node ID. All tags of the RTUs specified in this list are output.

`taglist=name[,name]...`

Specifies a comma separated list of TAGs by node ID. This parameter takes precedence over the rtulist parameter.

`timestamp=utc|YYYYMMDDThh:mm:ss`

Specifies the timestamp for the oldest data record to display either as UTC timestamp (seconds since January, 1st 1970 00:00:00 UTC) or as local time in ISO format.

`slots=n`

Specifies the maximum number of returned slots. Values from 1 to 200 are valid.

Note: if the output format is "T" (one table with all tag values), the actual number of table lines can exceed the slot number if the timestamps of the tags differ.

`dateformat=string`

Specifies the format for the date output. The following characters are treated special:

- Y or YYYY for the year with century

- YY for the year without century
- M or MM for the month
- D or DD for the day of month
- h or hh for the hour
- m or mm for the minute
- s or ss for the second
- T or t for the UTC timestamp

All other characters are copied verbatim. All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09").

`timeformat=string`

Specifies the format for the TIME column. If this parameter is ommited, only the DATE column is output. This way, it is possible to output the timestamp either in two columns (one for DATE, one for TIME) or one column (DATE, with time added).

`floatformat=string`

Specifies the format for the floating point values. The following strings are allowed:

- F for the standard format
- E for the n.nnnE+mm format
- e for the n.nnne+nn format
- NdelimiterNNNNN for fixed comma ,where the amount of Ns specifies the minimum digits left and right of the comma, and the delimiter specifies what string to use for the decimal point (for example, if floatformat is set to "N.NNN" the number 1.234567 will be printed as "1.235", but with floatformat set to "N,NNN" it will be printed as "1,235".

`floatdecimalpoint`

Specifies the character (or string) that is used for the decimal point in floating-point numbers (the measurement data). All special characters must be URL-escaped (e.g. for the tab character with ASCII code 0x09, you have to write "%09"). This options is ignored if you specify the floatformat in the NdelimiterNNNNN format!

`headerformat=I|N|S`

Specifies what should be printed in TAG and RTU headers: the node ID (default, or headerformat=I), the full name (headerformat=N) or the first word of the name (headerformat=S).

If you choose headerformat=S, a tag name of, for example, "1234 temperature relay 1" would be printed as "1234".

`outputformat=T|S`

Specifies the format of the output: a single table with a column for each TAG (outputformat=T) or one section with one table per TAG (outputformat=S).

`datequote=string`

Specifies the quote character for the DATE column (same syntax like quote=). Default is the same as the string quote.

`timequote=string`

Specifies the quote character for the TIME column (same syntax like quote=). Default is the same as the string quote.

`statusquote=string`

Specifies the quote character for the STATUS column (same syntax like quote=). Default is the same as the string quote.

`headerquote=string`

Specifies the quote character for the header lines (same syntax like quote=). Default is the same as the string quote.

Note: the following parameters may be used for the "Sections" output format only (outputformat=S):

`sectionheader=string`

Specifies an additional section header. The header can contain the following characters:

- I for the RTU node ID
- N for the full RTU name
- S for the first word of the RTU name
- i for the TAG node ID
- n for the full TAG name
- s for the first word of the TAG name
- {text} to include text as literals

`status=yes|no`

Specifies whether the status of the measurement values is output in its own column (status=yes) or not (default, or status=no).

statusformat=N|T

Specifies whether the status of a measurement value is printed as a number (statusformat=N) or as strings OK, BAD, MISSING (default, or statusformat=T).

bad=yes|no

Specifies whether measurement values with status=bad are output (bad=yes) or not (default, or bad=no).

missing=yes|no

Specifies whether measurement values with status=missing are output (missing=yes) or not (default, or missing=no).

RETURNS Text export data.

REMARKS To use this command, this feature has to be bought aditionally. A special license key is required.

EXAMPLE

```
getdata-csv
rtulist=2&delimiter=%09&timestamp=20040101T00:00
:00
```

Requests all TAG data for the RTU with node ID 2, using the tab character as field delimiter, beginning with January 1st, 2004 at midnight.

```
getdata-csv
rtulist=2,34&timestamp=20040101T00:00:00&datefor
mat=MM/DD/YY hh:mm:ss
```

Requests all TAG data for the RTUs with node IDs 2 and 34, beginning with January 1st, 2004 at midnight, and date output is formatted as month/day/year without century hour:minute:second (a common format for office programs like OpenOffice.org-Calc or Microsoft Excel).

```
getdata-csv
taglist=3,4&timestamp=20040101T00:00:00&dateform
at=MM/DD/YY hh:mm:ss
```

Same as above, but only returns data for the sensors with node IDs 3 and 4.!

SDICHANGED

DESCRIPTION Get/set "SDI configuration changed" flag for a specific device.

PARAMETERS Device ID and "SDI configuration changed" flag.

RETURNS OK or an error message.

REMARKS None.

EXAMPLE
```
sdichanged 44
SDI configuration changes pending for RTU 44
OK
```

Administrative Commands

This category describes the commands that return certain status information.

VER

DESCRIPTION Returns the current version of the emu3ap software.

PARAMETERS None.

RETURNS OK or an error message.

REMARKS None.

EXAMPLE
```
ver
Adcon Telemetry emu3ap, version 3.02
OK
```

TYPE

DESCRIPTION Returns the hardware platform of the emu3ap.

PARAMETERS None.

RETURNS OK or an error message.

REMARKS None.

EXAMPLE
```
type
A7840
OK
```

INSPECT

DESCRIPTION Returns a list of devices and associated information such as the ID of the device, and the date and time of the last stored slot. You can also use this command with a parameter. For example, **INSPECT** *device ID* causes the emu3ap software to return specific information concerning the requested device.

PARAMETERS None or a device ID (two variants).

RETURNS The list of devices or detailed information regarding a specific device (second variant).

REMARKS None.

UPTIME

DESCRIPTION Returns the amount of time the A440 Wireless Modem connected to the gateway has been operational.

PARAMETERS None.

RETURNS The amount of time the wireless modem has been in operation.

REMARKS None.

EXAMPLE

```
uptime
Up 230 day(s), 10 hour(s), 14 minute(s)
OK
```

NOP

This command is deprecated in emu3ap. It has been included for compatibility purposes, but always returns OK. The A840 Telemetry Gateway's software is stored in its internal Flash EEPROM.

HELP

DESCRIPTION Returns all the available commands in the emu3ap software.

PARAMETERS None.

RETURNS A list of commands.

QUIT

DESCRIPTION Exits the emu3ap software. If you were logged in as the user `adv`, you will be returned to the Linux login prompt. Otherwise, you will be returned to a Linux shell.

PARAMETERS None.

RETURNS Nothing.

REMARKS You can use the **EXIT** command with the same result.

TIME

DESCRIPTION Command to determine the current systemtime.

PARAMETERS None.

RETURNS Returns the current time.

REMARKS None.

EXAMPLE
```
time
Local time is: Tue Aug 29 11:28:36 2006
OK
```

UTC

DESCRIPTION Get/Set "wireless network uses UTC" flag.

PARAMETERS
```
UTC [ON|OFF]
```

RETURNS Returns weather the wireless network uses UTC or not.

REMARKS None.

EXAMPLE
```
utc
Wireless network uses UTC
OK
```

POLL

DESCRIPTION Requests gateway to poll a specific device as soon als possible.

PARAMETERS
```
POLL <dev>
```

RETURNS OK or error.

REMARKS Command does not wait for the actual poll.

EXAMPLE
```
poll 44
Dev: 44
Station will be polled in a few minutes...
OK
```

Direct Radio Commands

This category describes several commands that allow the host to communicate directly with the remote stations or other base stations. These commands take a longer time to complete. The exact amount of time depends on how the end-station is routed and how much time it takes the frames to travel from one station to another.

The direct radio commands return by default results such as **OK** or a specific error number. Complete details about the direct radio commands can be found in the user manuals for the respective devices (A730MD, A720, A723, and A733).

DIRECT

DESCRIPTION — Allows to send any availible radio command directly to the specified device.

PARAMETERS — Device Id, wanted command, command parameters

RETURNS — Result of sent radio command.

REMARKS — For details please consult the A73x addWAVE User Guide.

EXAMPLE

```
direct 10820 info
10820 info 255 255 29/8/2006 13:09:19 2.6 2 0 0
84 62 232 6:47 42 65 72 4 900 15 12 0
OK
```

Gets the information from the specific device.

PING

DESCRIPTION — Returns information about the specified device such as the RF levels, the date and time of the internal real-time clock of the station, the software version in the station, and the number of reserved bytes.

PARAMETERS — The RTU's ID.

RETURNS — A string or an error message (for example, ERROR 35 means timeout). The strings and parameters returned depend on the remote device type. Following is an explanation of the information returned:

- Incoming RF is the level recorded by the receiving station.
- Outgoing RF is the level at which the sending station received the requesting station.
- Pout represents the output power of the transmitter while it is sending the answer to the ping frame.
- The date/time represents the actual value of the local real-time clock.
- Ver is the software version of the RTU.
- Clk, stk, and cop contain Adcon-specific information.
- The uptime represents the time the device has operated since the last reset or internal watchdog reset.
- The battery and internal temperature are self-explanatory.

- RSSI (relative signal strength), PMP (power management parameters), and Slot are the current settings of the device.

REMARKS None.

EXAMPLES

```
ping 2006
Type A730MD
Dev 2006: incoming RF 247, outgoing RF 225
Time Fri Jun 2 16:24:04 2000
Ver C5, clk 00, stk FF, cop FF
Reserved bytes 00 00 00 00 00 00 00 00 00 00
OK

ping 6383
Type A720
Dev 6383: incoming RF 255, outgoing RF 255
Time Fri Jun 2 16:24:15 2000
Ver 22, clk 00, stk 00, cop 00
Uptime 396 day(s), 22 hour(s), 33 minute(s)
Batt 6.9 volts, internal temp 31 Celsius
Settings: RSSI 58, PMP 65 72, Slot 900 3
OK

ping 9471
Type A733
Dev 9471: incoming RF 255, outgoing RF 166, Pout:
182
Time Fri Jun 2 16:27:12 2000
Ver 0F, clk 00, stk 00, cop 05
Uptime 143 day(s), 7 hour(s), 44 minute(s)
Batt 7.0 volts, internal temp 27 Celsius
Settings: RSSI 85, PMP 65 72, Slot 900 15
OK
```

REQUEST

DESCRIPTION Returns a slot of data from the remote's FIFO memory.

PARAMETERS The RTU's ID and, optionally, the date/time and/or the *init* string.

RETURNS A string with data or an error message. The strings returned depend on the remote device type (see the examples below). The most common error messages are ERROR 35 (timeout) or ERROR 36 (RF channel in use).

REMARKS None.

EXAMPLE

```
request 2006
Dev 2006: incoming RF 255, outgoing RF 252
Time Thu Nov 2 21:42:21 1995
Data FF 00 00 51 72 DF 10 16 AC 04 01 71 DF 12 16
AC 04 01
OK
```

The command has several variations. If given as in the example above, the most recent slot stored in the station's FIFO is returned. By specifying a date and a time as parameter, the station returns the first slot "younger" than the specified date and time:

```
request 2006 2/11/95 20:0:0
Dev 2006: incoming RF 255, outgoing RF 251
Time Thu Nov 2 20:12:21 1995
Data FF 00 00 51 72 E1 12 0C AF 04 01 72 E1 10 0C
AF 04 01
OK
```

Furthermore, if the **INIT** parameter is also specified after the date and time, the station's real-time clock will be resynchronized with that of the gateway's internal date and time:

```
request 2006 2/11/95 20:0:0 INIT
Dev 2006: incoming RF 255, outgoing RF 251
Time Thu Nov 2 20:12:21 1995
Data FF 00 00 51 72 E1 12 0C AF 04 01 72 E1 10 0C
AF 04 01
OK
```

Note: This is a forced-time initialization of an RTU. In practice it is not needed, because the gateway does it automatically when it is itself initialized by means of the time command (see "SET TIME" on page 69).

An A720 device responds as follows:

```
request 6383
Dev 6383: incoming RF 255, outgoing RF 255
Time Fri Jun 2 17:01:35 2000
Data 4E 00 00 58 A5 60 01 00 00 86 00
OK
```

An A733 device responds as follows:

```
request 9471
Dev 9471: incoming RF 255, outgoing RF 122
Time Fri Jun 2 17:04:42 2000
Data 7F 00 00 00 00 00 00 00 00 59 00 16 AF 6B 20
01 00 00 00 A8 55 AF 13 A0 00 00 00 00
OK
```

Note: The internal FIFO of the A730MD RTUs is limited to 11 slots, which means about 2-3/4 hours of stored data. The A720 series 2 can store up to 240 slots (about 2.5 days), while the A733 RTUs can store up to 1024 slots (a little over 10 days). Older data will be overwritten.

RSETIO

DESCRIPTION

Acts upon the I/O ports of a remote device. It can switch a port to input or output mode and can also switch a port configured as output to a logical one or zero state.

PARAMETERS

The RTU's ID, the value of the data direction register (DDR), and the value of the port itself (DATA). All values are in hex. The DDR

and DATA bits must be OR-ed according to the required state of the ports.

RETURNS The actual state of the port (in hex).

REMARKS This command is deprecated; it is supported only by the A730MD devices and early A720 devices (series 1). New devices use the more advanced **PORT** command.

EXAMPLE

```
rsetio 2003 0 0
DDR 02, REG 5F
OK
```

PORT

DESCRIPTION Acts upon the I/O ports of a remote device.

PARAMETERS The RTU's ID, the byte code, and parameters (for some forms of the command). All parameters are in decimal form (pay particular attention to the byte code because it must be translated to decimal).

RETURNS Status result, depends on command.

REMARKS This command is currently supported only by the A720 (series 2) and A723 (series 3) addITs and A733 addWAVE RTUs.

EXAMPLE

```
port 9473 0
Dev 9473: incoming RF 255, outgoing RF 255
Value: 255, Error level: 0
OK
```

ANALOG

DESCRIPTION Acts upon the analog input ports of a remote device.

PARAMETERS The RTU's ID, the byte code, and parameters (for some forms of the command). All parameters are in decimal form (pay particular attention to the byte code because it must be translated to decimal).

RETURNS Status result, depends on command.

REMARKS This command is currently supported only by the A723 (series 3) addITs and the A733 addWAVE RTUs.

EXAMPLE

```
analog 9473 0
Dev 9473: incoming RF 255, outgoing RF 255
Average method: 1 2 3 4 5 6 7 5 5 5 5 5 0 0 0 0
Error level: 0
OK
```

B

DESCRIPTION — Issues a broadcast request frame that forces all listening devices to answer.

PARAMETERS — None.

RETURNS — A list of devices that answered.

REMARKS — None.

EXAMPLE

```
b
From 2003: incoming RF 255, outgoing RF 255
From 2646: incoming RF 188, outgoing RF 141
From 9474: incoming RF 255, outgoing RF 255
From 2419: incoming RF 255, outgoing RF 254
From 2464: incoming RF 255, outgoing RF 255
From 10843: incoming RF 255, outgoing RF 255
OK
```

RB

DESCRIPTION — Forces the specified remote device to issue a broadcast request frame that forces all listening devices to answer. By subsequently using the **RBLIST** command, a list of all the stations heard from can be retrieved.

PARAMETERS — The ID of the remote.

RETURNS — OK or error.

REMARKS — Currently only the A723 addIT and the A733 addWAVE devices support this command. At least 12 seconds must elapse between the time you issue the **RB** and **RBLIST** commands.

EXAMPLE

```
rb 9473
Error level: 0
OK
```

RBLIST

DESCRIPTION — Returns a list of all the stations heard from after a broadcast request.

PARAMETERS — The ID of the remote.

RETURNS — A list of the stations heard from.

REMARKS — Currently only the A723 addIT and the A733 addWAVE devices support this command. At least 12 seconds must elapse between the time you issue the **RB** and **RBLIST** commands.

EXAMPLE

```
rblist 9473
Dev 9473: incoming RF 255, outgoing RF 255
Last Broadcast: Mon Jun 19 12:50:31 2000
201: 255
4446: 255
2622: 255
6127: 255
10820: 255
11127: 255
2646: 255
2008: 255
OK
```

RSET

DESCRIPTION

Issues remote SET commands, which set parameters in the RTUs. Not all the RTUs support these commands, most notably the A730MD devices don't support it.

PARAMETERS

The ID of the remote, the set-command, and its parameters.

RETURNS

OK or an error message.

REMARKS

Currently only the A720 (series 2), A723 (series 3) addIT, and the A733 addWAVE devices support this command. The following remote set commands are supported: ID, SLOT, PMP, FREQ, FDEV, and SST. Not all devices support all commands.

EXAMPLE

```
rset 11123 slot 600 10
OK
```

XCONF

DESCRIPTION

This command transmits command strings for commands, which are suited for this mode of operation, to the targetted RTUs. Allowed commands are: CALC, COMP, COND, LC, MSTR, NPND, OC, PC, PS, SWITCH.

PARAMETERS

```
XCONF <dev> [params]
```

RETURNS

The reply string and the commands success or error code is returned.

REMARKS

Detailed information about parameters and output formats please infer from the A740 addNODE User Guide.

EXAMPLE

```
xconf 52 lc ?
52 XCONF lc 128 0 0 0
OK
```

XDATA

DESCRIPTION This command requests data for a list of logical channels for given timestamps.

PARAMETERS
```
XDATA <Tsync> <flags> <Tlast> <last channel> <max values> <nr of channels> <channels ...> [ <max packet size> [ <Tto> ] ]
```

RETURNS The requested data if available is returned in the specified format.

REMARKS Detailed information about parameters and output formats please infer from the A740 addNODE User Guide.

EXAMPLE
```
xdata 59 0 0 1092746115 0 255 0
59 XDATA 0 0 1092746115 0 255 0
[...]
OK
```

MODEM/GSM/GPRS Commands

MODEM

DESCRIPTION get/set which modem to use

PARAMETERS Get which modem is used, set if internal or external modem should be used.

RETURNS Whether internal or external Modem is used.

REMARKS None.

EXAMPLE
```
modem
Modem: external
OK
```

DIAL

DESCRIPTION dial out to device

PARAMETERS device ID.

RETURNS Call status parameters. The list is self explanatory.

REMARKS None.

EXAMPLE
```
dial 26137

Dev:      26137
Modem:    external
```

```
Phone:    06641111111
Key:      123456789
Start:    Fri May  5 11:44:38 2006
Duration: 15
OK
```

HANGUP

DESCRIPTION hang up call to device

PARAMETERS Device ID and shared secret

RETURNS A list of status parameters. The list is self-explanatory.

REMARKS The shared secret is needed to hangup the call.

EXAMPLE

```
hangup 26127 123456789

Dev:      26127
Modem:    external
Phone:    06641111111
Start:    Fri May  5 11:44:38 2006
Duration: 38
OK
```

MDMSTAT

DESCRIPTION Display modem state

PARAMETERS None.

RETURNS A list of current status parameters. The list is self-explanatory.

REMARKS Get only.

EXAMPLE

```
mdmstat
Dev:      0
Modem:
Phone:
OK
```

CALLJOURNAL

DESCRIPTION Display the call journal.

PARAMETERS None.

RETURNS List of calls in the call journal. The list is self-explanatory.

REMARKS Get only.

EXAMPLE

```
calljournal
Start--- Time--  Code Phone------- Status  Cause
20060430 17:00:02  7 30150 89.144.206.167 OK NORMAL
20060430 17:30:02  8 30150 89.144.206.167 OK NORMAL
20060430 18:00:02 15 30150 89.144.206.167 OK NORMAL
OK
```

PHONE

DESCRIPTION get/set phone for device

PARAMETERS

```
PHONE <dev>
```

get phone

```
PHONE <dev> 123
```

set GSM mode, set phone number

```
PHONE <dev> *123
```

set GPRS mode, set shared secret "123" connection will be permanent

```
PHONE <dev> **123
```

set GPRS mode, set shared secret "123"
connection will be closed by gateway after polling the device

RETURNS Phone number of the specified device.

REMARKS For field use please consider to choose a strong enough shared secret.

EXAMPLE Set:

```
phone 26137 06641111111
OK
```

Get:

```
phone 26137
Dev: 26137
Phone number: 06641111111
OK
```

GPRSSTAT

DESCRIPTION Returns GSM/GPRS module related status information.

PARAMETERS None.

RETURNS A list of current status parameters. The list is self-explanatory.

REMARKS GET only.

EXAMPLE
```
gprsstat
RTU    Address          Port   NPND  POLL  TIME
-------------------------------------------------
34990  172.16.0.66      45964  0     0        43693
OK
```

GPRSDISCONNECT

DESCRIPTION Close existing GPRS connection to device.

PARAMETERS Device ID.

RETURNS OK or error.

REMARKS None.

EXAMPLE
```
gprsdisconnect 34990
OK
```

Error Messages

The emu3ap software might return any of several error messages. Some are self-recoverable after a retry and others are simply warnings. These messages are returned instead of the **OK** prompt.

- 1 — Nonexistent command
- 2 — CLI buffer overflow
- 3 — Internal error
- 5 — Invalid or missing parameter
- 10 — Device not found
- 11 — Device already exists
- 13 — No more space for device descriptors
- 14 — No more records for the requested device
- 15 — Remote device currently unavailable (radio connection temporarily interrupted)
- 18 — Reserved device numbers (that is, 0)
- 20 — Incorrect time supplied, conversion not possible

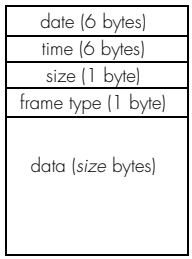
- 30 — Error receiving a frame (cyclic redundancy check and others)
- 31 — An unexpected frame type was received
- 32 — False length (type does not fit its length)
- 35 — Radio timeout when receiving (no answer after a request
- 36 — RF channel in use

Frame Types

The information in this section is intended to programmers writing their own routines to interface with the A840 Telemetry Gateway.

The A840/A440 combination can communicate with various types of RTUs. The distinctive frame type of each RTU makes no difference to the application software on the A840. The software stores the frame in the memory and retrieves it when needed—whatever its length is. But it is important for the software on the host to correctly interpret the frames returned by the gateway.

The data is retrieved from the gateway using the **GETBLOCK** command (see page 71). All frames have certain common elements: the date/time, the number of following bytes (including the frame type), the frame type, and the data portion. Clearly, the only difference between the various frames is the data portion (the size and type are also different from frame to frame). Following are descriptions of the frames that are currently in use.



The A730MD frames

The A730MD uses frame type 9.

Type 9

SIZE

21 (including the type byte).

FORMAT

```
struct tlg_type9 {
    BYTE      RF_LevelIn;
    BYTE      RF_LevelOut;
    BYTE      DigiByte;
    BYTE      PulseCounter0;
```

```
    BYTE      PulseCounter1;
    BYTE      BatteryLevel;
    BYTE      Analog1;
    BYTE      Analog2;
    BYTE      Analog3;
    BYTE      Analog4;
    BYTE      Analog5;
    BYTE      Analog6;
    BYTE      Analog7;
    BYTE      Analog8;
    BYTE      Analog9;
    BYTE      Analog10;
    BYTE      Analog11;
    BYTE      Analog12;
    BYTE      Analog13;
    BYTE      Analog14;
};
```

DESCRIPTION

All values are 8 bits and must be converted accordingly, depending on the sensor connected. Figure 23 shows the placement of various inputs on the connectors.

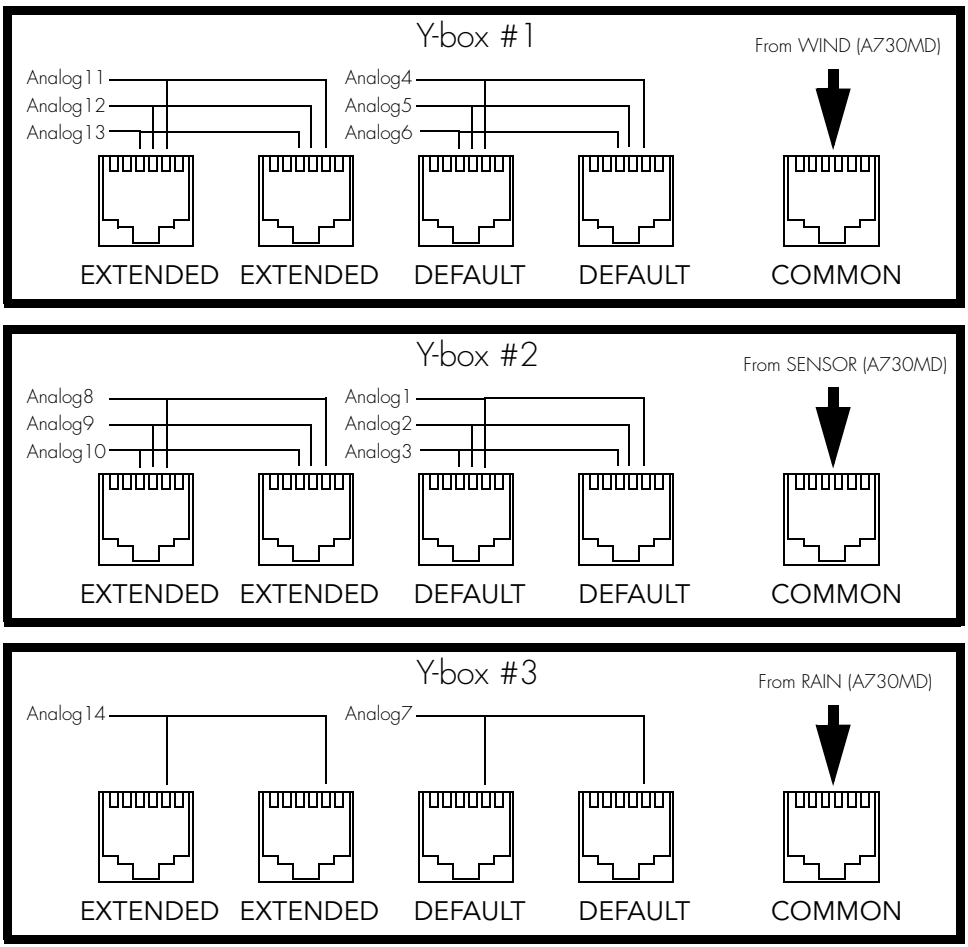


Figure 23. A730 Connector Inputs

Note: If no Y-boxes are installed, the values sampled on channels 1 to 7 will be duplicated on channels 8 to 14. The drawing above does not imply that all the Y-boxes must be in place. In fact, adding the third Y-box makes little sense because only one analog input can be duplicated on the RAIN connector.

The RF level is non-linear and results from a table (which you can get from Adcon as an ASCII file). The value of the battery can be computed as follows:

$$Batt_{[V]} = \frac{BatteryLevel \bullet 20}{255}$$

The digibyte is essentially the reflection of several inputs or internal status bits on the A730MD; its structure is described below:

B7	B6	B5	B4	B3	B2	B1	B0
S.C.	AUX:4	N.U.	RAI:3	RAI:4	AUX:2	AUX:3	AUX:5

S.C. = Solar Cell status N.U. = Not used RAI = Rain AUX = Aux

For more details, refer to the user manual for the A730MD device (addVANTAGE A730).

The A720 (addIT) Frames

The addIT uses a reduced frame type (38). No adapter use for extending the inputs is possible.

Type 38

SIZE

14 (including the type byte).

FORMAT

```
struct tlg_type38 {
    BYTE      RF_LevelIn;
    BYTE      RF_LevelOut;
    BYTE      DigiByte;
    BYTE      PulseCounter0;
    BYTE      PulseCounter1;
    BYTE      BatteryLevel;
    BYTE      Analog1;
    BYTE      Analog2;
    BYTE      Analog3;
    BYTE      Analog4;
    BYTE      Analog5;
    BYTE      Analog6;
    BYTE      Reserved;
```

```
};
```

DESCRIPTION

All values are 8 bits and must be converted accordingly, depending on the sensor connected. Figure 24 shows the placement of various inputs on the connectors.

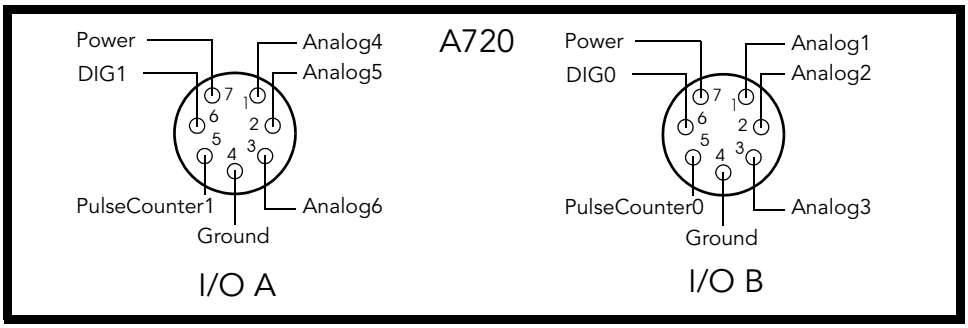


Figure 24. A720 Connector Inputs

The RF level is non-linear and results from a table (which you can get from Adcon as an ASCII file). The value of the battery can be computed as follows:

$$Batt_{[V]} = \frac{BatteryLevel \bullet 20}{255}$$

The digibyte is essentially the reflection of several inputs or internal status bits on the A720; its structure is described below:

B7	B6	B5	B4	B3	B2	B1	B0
S.C.	res	res	res	res	res	DIG1	DIG0

S.C. = Solar Cell status res = reserved

For more details, refer to the user manual for the A720 device.

The A723 (addIT series 3) Frames

The addIT series 3 implements currently two frame types, depending on the compatibility mode flag: 38 (described on page 95) and 39. For more details about the compatibility flag, refer to the A720 series user manual.

There is, however, a fundamental difference between frame 39 and the previously described data frames. The A723 samples the

analog values with 10-bit resolution and stores them as 12-bit values. Also, the A723 contains two 16-bit pulse counters, that is, it has more data to send. Due to the limited space available and to minimize the radio traffic, the frames are slightly compressed, in that six 12-bit values are packed in 9 bytes.

Type 39

SIZE

22 (including the type byte).

FORMAT

```
struct tlg_type37 {
    BYTE      RF_LevelIn;
    BYTE      RF_LevelOut;
    BYTE      DigiByte;
    WORD      PulseCounter0;
    WORD      PulseCounter1;
    BYTE      BatteryLevel;
    BYTE      Analog[9];
};
```

DESCRIPTION

The analog values are 12 bits and must be converted accordingly, depending on the sensor connected, while the pulse counters are 16-bit values. Only the RF and battery levels are 8-bit values. **`Analog[9]`** is an array of 9 unsigned bytes that is the result of packing the six 12-bit values. These are the values returned by the internal A/D converter from the respective I/O connectors.

Note: The integers (16-bit values) are sent using the big endian convention, that is, first the most significant byte and then the least significant byte.

The packing mechanism is shown in Figure 25.

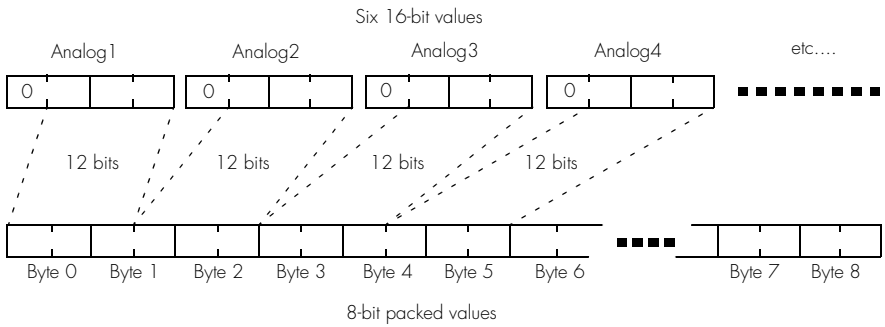


Figure 25. Type 39 Frame Compression

Note: Only Analog1 to Analog6 are packed; the 16-bit Pulse Counters are not.

Figure 26 shows the placement of various inputs on the connectors.

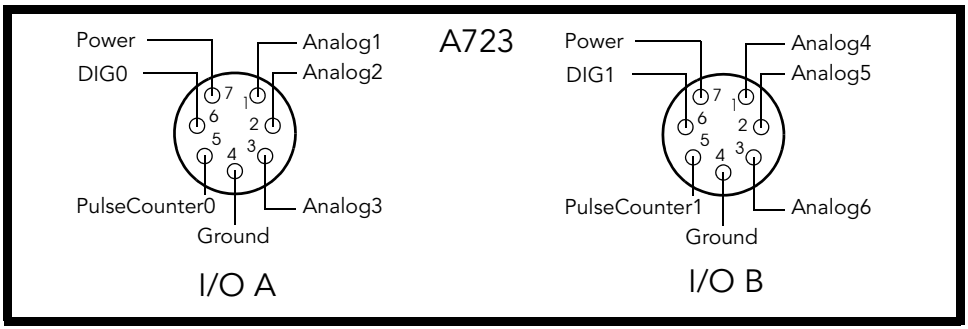


Figure 26. A723 Connector Inputs

The RF level is ⏩linear and results from a table (which you can get from Adcon as an ASCII file). The value of the battery can be computed as follows:

$$Batt_{[V]} = \frac{BatteryLevel \bullet 20}{255}$$

The digibyte is essentially the reflection of several inputs or internal status bits on the A723; its structure is described below:

b7							b0
S.C.	res	res	res	res	res	DIG1	DIG0

S.C. = Solar Cell status res = reserved

For more details, refer to the user manual for the A723 device.

The A733 (addWAVE) Frames

The addWAVE currently implements only one frame type (37). The A733 samples the analog values with 10-bit resolution and stores them as 12-bit values. Also, the A733 contains built-in logic to sample 12 different analog inputs, as well as four 16-bit pulse counters—all in all, substantially more data. Due to the limited

space available and in order to minimize the radio traffic, the frames are slightly compressed, in that twelve 12-bit values are packed in nine 16-bit words.

Type 37

SIZE

31 (including the type byte).

FORMAT

```
struct tlg_type37 {
    BYTE      RF_LevelIn;
    BYTE      RF_LevelOut;
    BYTE      DigiByte;
    WORD      PulseCounter0;
    WORD      PulseCounter1;
    WORD      PulseCounter2;
    WORD      PulseCounter3;
    BYTE      BatteryLevel;
    BYTE      Analog[18];
};
```

DESCRIPTION

The analog values are 12 bits and must be converted accordingly, depending on the sensor connected, while the pulse counters are 16-bit values. Only the RF and battery levels are 8-bit values. **`Analog[18]`** is an array of 18 unsigned bytes that is the result of packing the twelve 12-bit values. These are the values returned by the internal A/D converter from the respective I/O connectors.

Note: The integers (16-bit values) are sent using the big endian convention, that is, first the most significant byte and then the least significant byte.

The packing mechanism is shown in Figure 27.

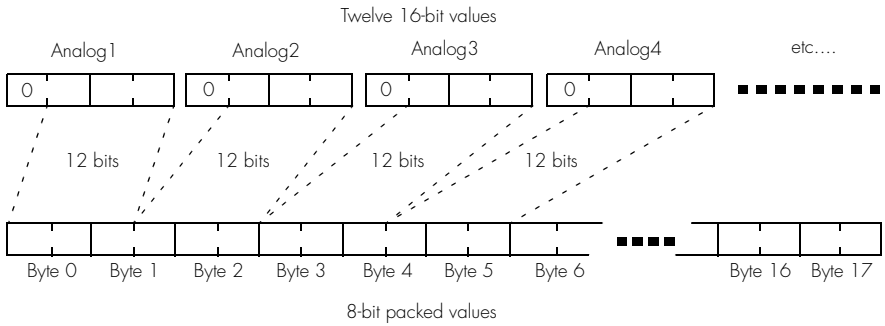


Figure 27. Type 37 Frame Compression

Note: Only Analog1 to Analog12 are packed; the 16-bit Pulse Counters are not.

Figure 28 show the placement of various inputs on the connectors.

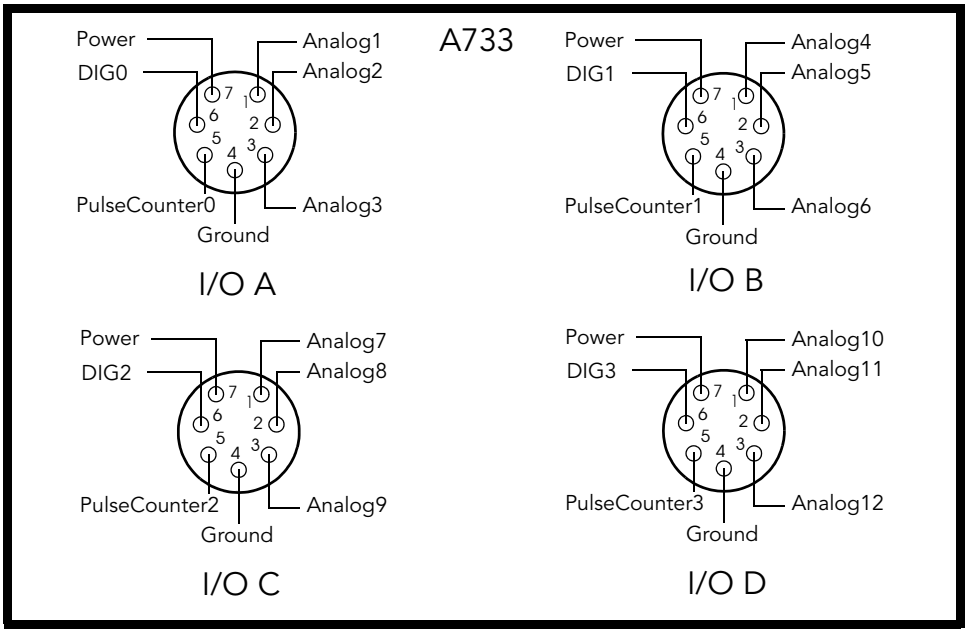


Figure 28. A733 Connector Inputs

The RF level is not linear and results from a table (which you can get from Adcon as an ASCII file). The value of the battery can be computed as follows:

$$Batt_{[V]} = \frac{BatteryLevel \bullet 20}{255}$$

The digibyte is essentially the reflection of several inputs or internal status bits on the A733; its structure is described below:

b7							b0
S.C.	res	res	res	DIG3	DIG2	DIG1	DIG0

S.C. = Solar Cell status res = reserved

For more details, refer to the user manual for the A733 device.

Frame Parsing Example

To better understand how the retrieved frames must be processed, this section provides an example of such processing. Let's suppose that we used the **GETBLOCK** command and retrieved a block of frames of type 37. After parsing the block, we separate it into individual frames as described in "GETBLOCK" on page 71. Because all the frames are treated identically, we will show you how to interpret only one of them from the block. The frame looks something like this:

```
02066411312A1F25FF7B7F0000000000000000590016AC6B
0000000000A6D5E81C40000000000206
```

Separating the date/time, frame type and size, and the data segment, we get this:

```
020664: Date (2/6/00)
11312A: Time (17:49:42)
1F: Size (31 bytes)
25: Frame type (37)
FF7B7F0000000000000000590016AC6B0000000000A6D5E8
1C4000000000: Data
```

Using the description of frame 37, we can parse the data segment. To show this more easily, we will rewrite the frame with spaces inserted between different elements:

```
FF 7B 7F 0000 0000 0000 0000 59 001 6AC 6B0 000
000 000 A6D 5E8 1C4 000 000 000
```

Now we can easily map the elements onto the frame 37 structure:

```
FF: RF in
7B: RF out
7F: Digibyte
0000: PulseCounter0
0000: PulseCounter1
0000: PulseCounter2
0000: PulseCounter3
59: BatteryLevel
001: Analog1
6AC: Analog2
6B0: Analog3
000: Analog4
000: Analog5
000: Analog6
A6D: Analog7
5E8: Analog8
1C4: Analog9
000: Analog10
000: Analog11
000: Analog12
```

The last step is to convert the analog values to actual engineering units. This is easily accomplished if we know what sensor is connected to each input of the RTU, and its conversion equation. Let's assume that a temperature sensor is connected to the Analog7 input. Then:

$$Temp_{[^\circ C]} = \frac{2669 \cdot 100}{4095} - 40 = 25{,}17$$

Note that 2669 is A6D converted to decimal while the -40 was necessary because the standard Adcon temperature sensor has a range from -40 to 60 °C.

Similarly, the battery level can be computed as follows:

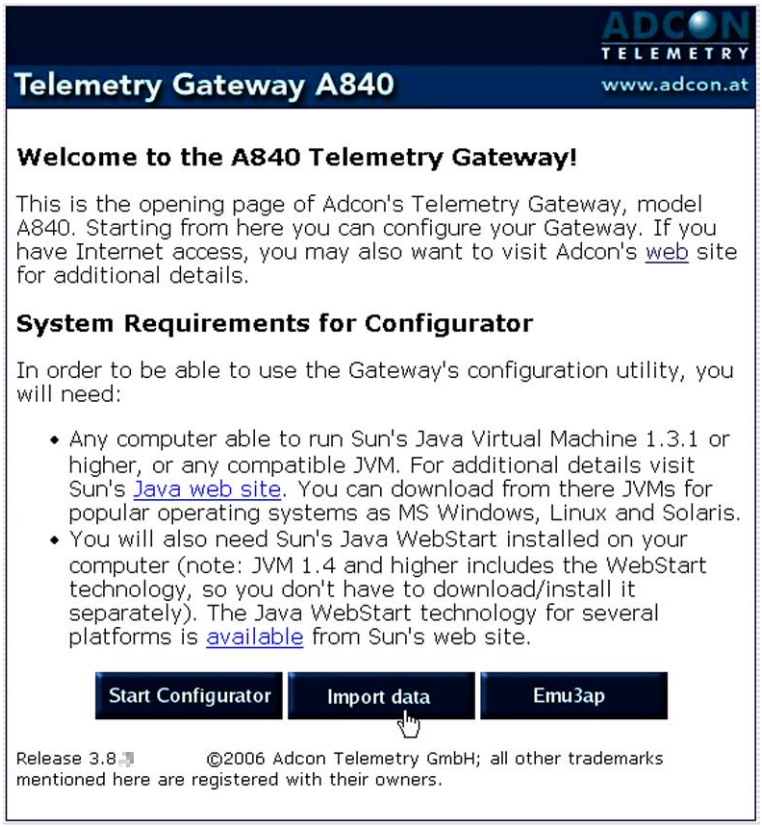
$$Batt_{[V]} = \frac{89 \cdot 20}{255} = 6{,}98$$

Note here that we had to deal with an 8-bit value (59 hex = 89 decimal), so the divider is 255, while the previous example was based on a 12-bit conversion and the divider was 4095.

Data Import

Note: The data import to your A840 Telemetry Gateway is only possible when using files exported by a A510 Data Display

To perform a data import please connect to your A840 Telemetry Gatway using your webbrowser and selecting the "Import data" option on the welcome page of your A840 Telemetry Gateway Configurator.

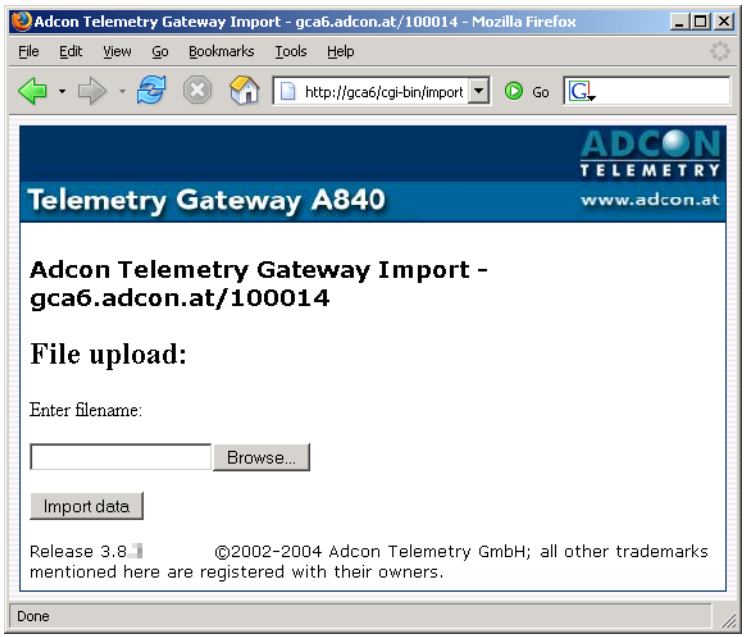


After entering your A840 Telemetry Gatway username and password correctly into the autentification promt you will be forwarded to the data import interface.

The data import interface

In the data import interface you can choose your desired data file either by entering the filename or by browsing your computer for the file.

The data is imported to your A840 Telemetry Gatway by clicking the "Import data" button.



Chapter 4. Appendix

Free and Open-Source Software

The A840 Telemetry Gateway contains third-party software packages which were made available to the public by their respective copyright owners under license terms which grant additional rights beyond mere usage of the software, such as the right to get, modify and redistribute the sourcecode of these packages.

These software packages are commonly called *Free and Open-Source Software* (FOSS). If you are interested in further informations and have Internet access, you might want to visit the following websites:

- The Free Software Foundation (http://www.fsf.org/)
- The Open Source Initiative (http://www.opensource.org/

The sourcecode to these software packages and the toolchain needed to build the corresponding binaries are available as downloadable archives on Adcon's web site (http://www.adcon.at) or on digital storage media (contact Adcon's Support).

However, there are a few things you have to consider:

- Some software packages installed on the gateway are neither derived from nor licensed as FOSS like the bootloader, the data logging andretrieval software and the Configurator application. These packagesare not available in sourcecode form and may not be redistributedwithout explicit written permission of their respective copyright owners!
- Adcon provides support only for the A840 firmware packages distributed by Adcon running on the gateway hardware. Modified software or running the A840 firmware packages on other hardware is not supported by Adcon.
- Most software packages distributed with the gateway were modified by Adcon which means that they differ from the version available on the project's web sites. The authors of the original versions are not responsible for the packages distributed by Adcon and usually will not provide support for any packages other than those they distribute themselves.
- The software packages include disclaimers of liability in their license terms. Read them carefully, especially if you want to use, modify and/or distribute the software packages yourself.

Third-party Software

hermit 1.2 - ARM bootloader

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation.

q.v. Appendix II GPL-v2

linux 2.4 - Operating system kernel

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation.

q.v. Appendix II GPL-v2

glibc 2.1.6 - GNU C Library

The GNU C Library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

q.v. Appendix II LGPL-v2

This product includes software developed by Harvard University and its contributors.

Copyright (C) 1991-2005 Free Software Foundation, Inc.

Portions Copyright (C) 1991 Regents of the University of California.

Portions Copyright (C) 1993 by Digital Equipment Corporation.

Portions Copyright (C) 1984, Sun Microsystems, Inc.

Portions Copyright (C) 1991,1990,1989 Carnegie Mellon University

Portions Copyright (C) 1990, 1993, 1994, 1995, 1996, 1997

Sleepycat Software.

Portions Copyright (C) 1995, 1996

The President and Fellows of Harvard University

zlib - General purpose compression library

Copyright (C) 1995-2004 Jean-loup Gailly and Mark Adler

expat - XML parser library

Copyright (c) 1998, 1999, 2000

Thai Open Source Software Center Ltd and Clark Cooper

Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006

Expat maintainers.

libmodem - Modem handling library

Copyright (C) 1994-2001 Riccardo Facchetti

Portions Copyright (C) 2030 Adcon Telmetry GmbH

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

q.v. Appendix II GPL-v2

md5 - Message digest algorithm

The algorithm is due to Ron Rivest. This code was written by Colin Plumb in 1993, no copyright is claimed. This code is in the public domain; do with it what you wish.

Equivalent code is available from RSA Data Security, Inc. This code has been tested against that, and is equivalent, except that you don't need to include two pages of legalese with every copy.

ghttpd - GazTek HTTP daemon

Copyright (C) 1999 Gareth Owen <gaz@athene.co.uk>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

q.v. Appendix II GPL-v2

tinylogin - Login program

Copyright (C) 1999 by Lineo, inc. Portions Copyright 1995 by Wietse Venema.

TinyLogin may be used and distributed under the terms of GNU General Public License with addition of some special terms.

q.v. Appendix II GPL-v2

snarf - HTTP download utility

Copyright 1998 Zachary Beane <xach@xach.com>

This program is licensed to you under the terms of the GNU General Public License.

q.v. Appendix II GPL-v2

putsms - SMS sending utility

Copyright (C) 2000 Stefan Frings

This program is free software unless you got it under another license directly from the author. You can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation. Either version 2 of the License, or (at your option) any later version.

q.v. Appendix II GPL-v2

ppp - Point-to-Point protocol daemon

Parts of this product are available under the terms of the GNU General Public License as published by the Free Software foundation.

q.v. Appendix II GPL-v2

This product includes software developed by the University of California, Berkeley and its contributors.

This product includes the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" Copyright (C) 1990, RSA Data Security, Inc.

Portions Copyright (c) 1985, 1986, 1989 The Regents of the University of California.

Portions Copyright (c) 1993 The Australian National University.

Portions Copyright (c) 1989 Carnegie Mellon University.

Portions Copyright (c) 1995 Pedro Roque Marques

Portions Copyright (c) 1991 Gregory M. Christy.

Portions Copyright (c) 1995 Eric Rosenquist, Strata Software Limited.

Portions Copyright (C) 1999 Tommi Komulainen <Tommi.Komulainen@iki.fi>

ntpclient - Network Time Protocol client

ntpclient is Copyright 1997, 1999, 2000, 2003 Larry Doolittle, and may be freely used and copied according to the terms of the GNU General Public License, version 2.

q.v. Appendix II GPL-v2

mini_sendmail - send mails to an SMTP server

Mgetty+Sendfax was written by Gert Doering, with many contributions from the Internet.

Vgetty was written by Klaus Weidner, Marc Eberhard, Marc Schaefer and also many contributions from the Internet community.

The whole package, starting with version 1.1.22, is distributed under the GNU GPL license, found in the accompanying file 'COPYING'.

q.v. Appendix II GPL-v2, mini_sendmail

busybox - UNIX commandline utilities

Original release code (unless otherwise noted) Copyright 1995, 1996 Bruce Perens <bruce@pixar.com>

mkswap Copyright 1991 Linus Torvalds

tiny-ls(ls) Copyright 1996 Brian Candler <B.Candler@pobox.com>

tarcat, loadkmap, various fixes, Debian maintenance Copyright 1998 Enrique Zanardi <ezanardi@ull.es>

more(v2), makedevs, dutmp, modularization, auto links file, various fixes, Linux Router Project maintenance Copyright 1998 Dave Cinege <dcinege@psychosis.com>

mini-gzip(gzip), mini-netcat(mnc) Copyright 1998 Charles P. Wright <cpwright@villagenet.com>

Tons of new stuff as noted in header files Copyright (C) 1999,2000,2001 by Lineo, inc. and written by Erik Andersen <andersen@lineo.com>, <andersee@debian.org>

Busybox may be used and distributed under the GNU General Public License.

q.v. Appendix II GPL-v2

bash - command line interpreter

Copyright (C) Free Software Foundation, Inc

GNU bash is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

q.v. Appendix II GPL-v2

cu - Taylor UUCP

Copyright (C) 1991, 92, 93, 94, 1995 Ian Lance Taylor

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

q.v. Appendix II GPL-v2

isc-dhcpd - ISC DHCP daemon

Copyright (c) 2004-2005 by Internet Systems Consortium, Inc. ("ISC")

Copyright (c) 1995-2003 by Internet Software Consortium

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND ISC DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL ISC BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Internet Systems Consortium, Inc.

950 Charter Street

Redwood City, CA 94063

<info@isc.org>

http://www.isc.org/

wu-ftpd - File transfer protocol daemon

wu-ftpd comes under the terms of the WU-FTPD SOFTWARE LICENSE q.v. Appendix II WU-FTPD Software License

gdbserver - server stub of the GNU debugger

This program is part of the gdb project.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

q.v. Appendix II GPL-v2

procps - The /proc file system utilites and library

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

q.v. Appendix II GPL-v2

sash - Stand-alone shell

Copyright (c) 2002 by David I. Bell

telnetd

This product includes software developed by the University of California, Berkeley and its contributors.

Copyright (c) 1989 Regents of the University of California. All rights reserved.

License texts

WU-FTPD Software License

Use, modification, or redistribution (including distribution of any modified or derived work) in any form, or on any medium, is permitted only if all the following conditions are met:

1. Redistributions qualify as "freeware" or "Open Source Software" under the following terms:

a. Redistributions are made at no charge beyond the reasonable cost of materials and delivery. Where redistribution of this softwarenis as part of a larger package or combined work, this restriction applies only to the costs of materials and delivery of this software, not to any other costs associated with the larger package or combined work.

b. Redistributions are accompanied by a copy of the Source Code or by an irrevocable offer to provide a copy of the Source Code for up to three years at the cost of materials and delivery. Such redistributions must allow further use, modification, and redistribution of the Source Code under substantially the same terms as this license. For the purposes of redistribution "Source Code" means all files included in the original distribution, including all modifications or additions, on a medium and in a form allowing fully working executable programs to be produced.

2. Redistributions of Source Code must retain the copyright notices as they appear in each Source Code file and the COPYRIGHT file, these license terms, and the disclaimer/limitation of liability set forth as paragraph 6 below.

3. Redistributions in binary form must reproduce the Copyright Notice, these license terms, and the disclaimer/limitation of liability set forth as paragraph 6 below, in the documentation and/or other materials provided with the distribution. For the purposes of binary distribution the "Copyright Notice" refers to the following language:

Copyright (c) 1999,2000 WU-FTPD Development Group. All rights reserved.

Portions Copyright (c) 1980, 1985, 1988, 1989, 1990, 1991, 1993, 1994 The Regents of the University of California.

Portions Copyright (c) 1993, 1994 Washington University in Saint Louis.

Portions Copyright (c) 1996, 1998 Berkeley Software Design, Inc.

Portions Copyright (c) 1983, 1995, 1996, 1997 Eric P. Allman.

Portions Copyright (c) 1998 Sendmail, Inc.

Portions Copyright (c) 1989 Massachusetts Institute of Technology.

Portions Copyright (c) 1997 Stan Barber.

Portions Copyright (c) 1997 Kent Landfield.

Portions Copyright (c) 1991, 1992, 1993, 1994, 1995, 1996, 1997 Free Software Foundation, Inc.

Use and distribution of this software and its source code are governed by the terms and conditions of the WU-FTPD Software License ("LICENSE"). If you did not receive a copy of the license, it may be obtained online at http://www.wu-ftpd.org/license.html

4. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes software developed by the WU-FTPD Development Group, the Washington University at Saint Louis, Berkeley Software Design, Inc., and their contributors."

5. Neither the name of the WU-FTPD Development Group, nor the names of any copyright holders, nor the names of any contributors may be used to endorse or promote products derived from this software without specific prior written permission. The names "wuftpd" and "wu-ftpd" are trademarks of the WU-FTPD Development Group and the Washington University at Saint Louis.

6. Disclaimer/Limitation of Liability:

THIS SOFTWARE IS PROVIDED BY THE WU-FTPD DEVELOPMENT GROUP, THE COPYRIGHT HOLDERS, AND CONTRIBUTORS, "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 WU-FTPD DEVELOPMENT GROUP, THE COPYRIGHT HOLDERS, OR CONTRIBUTORS, 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 DAMAGES.

7. USE, MODIFICATION, OR REDISTRIBUTION, OF THIS SOFTWARE IMPLIES ACCEPTANCE OF ALL TERMS AND CONDITIONS OF THIS LICENSE.

GPL-v2

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights.

These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains

a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a

"work based on the Program" means either the Program or any derivative work under copyright law:

that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an

announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of

Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed

under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License.

However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not

signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation

excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR

DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by he Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software

Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

LGPL-v2

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your

freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any

associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensurethat, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library. In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy. This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you

accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work

under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library includingwhatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and

all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software

Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

mini_sendmail

mini_sendmail - accept email on behalf of real sendmail

Copyright 1999 by Jef Poskanzer <jef@acme.com>. All rights reserved.

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.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``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 OR CONTRIBUTORS 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.

glibc

This file contains the copying permission notices for various files in the GNU C Library distribution that have copyright owners other than the Free Software Foundation. These notices all require that a copy of the notice be included in the accompanying documentation and be distributed with binary distributions of the code, so be sure to include this file along with any binary distributions derived from the GNU C Library.

All code incorporated from 4.4 BSD is distributed under the following license:

Copyright (C) 1991 Regents of the University of California.

All rights reserved.

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. [This condition was removed.]

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``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 REGENTS OR CONTRIBUTORS 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.

The DNS resolver code, taken from BIND 4.9.5, is copyrighted both by UC Berkeley and by Digital Equipment Corporation. The DEC portions are under the following license:

Portions Copyright (C) 1993 by Digital Equipment Corporation.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the document or software without specific, written prior permission.

THE SOFTWARE IS PROVIDED ``AS IS'' AND DIGITAL EQUIPMENT CORP. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

The Sun RPC support (from rpcsrc-4.0) is covered by the following license:

Copyright (C) 1984, Sun Microsystems, Inc.

Sun RPC is a product of Sun Microsystems, Inc. and is provided for unrestricted use provided that this legend is included on all tape media and as a part of the software program in whole or part. Users may copy or modify Sun RPC without charge, but are not authorized to license or distribute it to anyone else except as part of a product or program developed by the user.

SUN RPC IS PROVIDED AS IS WITH NO WARRANTIES OF ANY KIND INCLUDING THE WARRANTIES OF DESIGN, MERCHANTIBILITY AND FITNESS FOR A PARTICULAR PURPOSE, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE.

Sun RPC is provided with no support and without any obligation on the part of Sun Microsystems, Inc. to assist in its use, correction, modification or enhancement.

SUN MICROSYSTEMS, INC. SHALL HAVE NO LIABILITY WITH RESPECT TO THE INFRINGEMENT OF COPYRIGHTS, TRADE SECRETS OR ANY PATENTS BY SUN RPC OR ANY PART THEREOF.

In no event will Sun Microsystems, Inc. be liable for any lost revenue or profits or other special, indirect and consequential damages, even if Sun has been advised of the possibility of such damages.

The following CMU license covers some of the support code for Mach, derived from Mach 3.0:

Mach Operating System

Copyright (C) 1991,1990,1989 Carnegie Mellon University

All Rights Reserved.

Permission to use, copy, modify and distribute this software and its documentation is hereby granted, provided that both the copyright notice and this permission notice appear in all copies of the software, derivative works or modified versions, and any portions thereof, and that both notices appear in supporting documentation.

CARNEGIE MELLON ALLOWS FREE USE OF THIS SOFTWARE IN ITS ``AS IS'' CONDITION. CARNEGIE MELLON DISCLAIMS ANY

LIABILITY OF ANY KIND FOR ANY DAMAGES WHATSOEVER RESULTING FROM THE USE OF THIS SOFTWARE.

Carnegie Mellon requests users of this software to return to

Software Distribution Coordinator

School of Computer Science

Carnegie Mellon University

Pittsburgh PA 15213-3890

or Software.Distribution@CS.CMU.EDU any improvements or extensions that they make and grant Carnegie Mellon the rights toredistribute these changes.

The file if_ppp.h is under the following CMU 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. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY CARNEGIE MELLON UNIVERSITY AND CONTRIBUTORS ``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 UNIVERSITY OR CONTRIBUTORS 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.

The following license covers the files from Intel's "Highly Optimized Mathematical Functions for Itanium" collection:

Intel License Agreement

Copyright (c) 2000, Intel Corporation

All rights reserved.

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

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* 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.

* The name of Intel Corporation may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "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 INTEL OR CONTRIBUTORS 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.

The files inet/getnameinfo.c and sysdeps/posix/getaddrinfo.c are copyright

(C) by Craig Metz and are distributed under the following license:

/* The Inner Net License, Version 2.00

The author(s) grant permission for redistribution and use in source andbinary forms, with or without modification, of the software and documentationprovided that the following conditions are met:

0. If you receive a version of the software that is specifically labelled as not being for redistribution (check the version message and/or README), you are not permitted to redistribute that version of the software in any way or form.

1. All terms of the all other applicable copyrights and licenses must be followed.

2. Redistributions of source code must retain the authors' copyright notice(s), this list of conditions, and the following disclaimer.

3. Redistributions in binary form must reproduce the authors' copyright notice(s), this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.

4. [The copyright holder has authorized the removal of this clause.]

5. Neither the name(s) of the author(s) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY ITS AUTHORS AND CONTRIBUTORS ``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 AUTHORS OR CONTRIBUTORS 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.

If these license terms cause you a real problem, contact the author.