NetCrossing Gateway NX2048 and NX8192

Operator's Manual

July 2009

AFAR Communications Inc.
81 David Love Place
Santa Barbara, CA 93117

Tel: +1 805 681 1993
Fax: +1 805 681 1994

NetCrossing Gateway NX2048 and NX8192

Operator's Manual

July 2009

AFAR Communications Inc.
81 David Love Place
Santa Barbara, CA 93117

Tel: +1 805 681 1993
Fax: +1 805 681 1994

$25.00

Customer Service

AFAR provides customer service during normal Pacific Coast business hours and may be reached by voice, fax, or email as follows:

Tel: +1 805 681 1993
Fax: +1 805 681 1994
email: support@afar.net

If you must return the equipment, please contact us for a Return Material Authorization (RMA) number. Equipment should be shipped to:

AFAR Communications Inc.
81 David Love Place,
Santa Barbara, CA 93117
U.S.A.

STATEMENT OF WARRANTY

Afar Communications Inc. products, except as otherwise stated in an applicable price list, are warranted against defects in workmanship and material for a period of one (1) year from date of delivery as evidenced by Afar Communications Inc.'s packing slip or other transportation receipt.

Afar Communications Inc.'s sole responsibility under this warranty shall be to either repair or replace, at its option, any component which fails during the applicable warranty period because of a defect in workmanship and material, provided **purchaser** has promptly reported same to Afar Communications Inc. in writing. All replaced products or parts shall become Afar Communications Inc.'s property.

Afar Communications Inc. shall honor this warranty at its facility in Goleta, California. It is **purchaser**'s responsibility to return, at its expense, the defective Product to Afar Communications Inc. **Purchaser** must notify Afar Communications Inc. and obtain shipping instructions prior to returning any product. Afar Communications Inc. will pay the transportation charges for the return of the Product to **purchaser** but not including any custom clearance fees and other related charges which shall be paid by **purchaser**. If Afar Communications Inc. determines that the Product is not defective within the terms of the warranty, **purchaser** shall pay Afar Communications Inc. all costs of handling, transportation and repairs at the prevailing repair rates.

All the above warranties are contingent upon proper use of the Product. These warranties will not apply (i) if adjustment, repair, or parts replacement is required because of accident, unusual physical, electrical or electromagnetic stress, negligence, misuse, failure of electric power environmental controls, transportation, or abuses other than ordinary use (ii) if the Product has been modified or has been repaired or altered outside Afar Communications Inc.'s factory, unless Afar Communications Inc. specifically authorizes such repairs or alterations; (iii) where Afar Communications Inc. serial numbers, or quality assurance decals have been removed or altered.

Afar Communications Inc. reserves the right to make product improvements without incurring any obligation or liability to make the same changes in Products previously manufactured or purchased.

No person, including any dealer, agent or representative of Afar Communications Inc. is authorized to assume for Afar Communications Inc. any other liability on its behalf except as set forth herein. Afar Communications Inc. hereby disclaims all implied warranties of products including without limitation, all implied warranties of merchantability or fitness for a particular purpose. The warranties expressly stated herein are the sole obligation or liability on the part of Afar Communications Inc. arising out of or in connection with the sale or performance of the products.

In no event will Afar Communications Inc. be liable to **purchaser** for (i) procurement costs; (ii) special, indirect or consequential damages; (iii) any damages resulting from loss of use, data or profits arising out of the use of Afar Communications Inc. products. In no event shall Afar Communications Inc. be liable for any breach of warranty in an amount exceeding the net selling price of any defective Product.

FCC Notice

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Changes or modifications not expressly approved in writing by AFAR Communications Inc. may void the user's authority to operate this equipment. AFAR Communications Inc. can not accept any financial or other responsibilities that may be the result of your use of this information, including direct, indirect, special, or consequential damages. Refer to warranty documents for product warranty coverage and specifics.

TABLE OF CONTENTS

1 PRODUCT DESCRIPTION

1.1 Overview

The Afar NetCrossing™ Gateway allows you to deploy a point-to-point serial synchronous link across a packet switch network. The gateway breaks the continuous serial data stream into fixed size packets, adds the Ethernet or IP framing, and sends them over the packet switch network to a remote gateway. At the remote end, the gateway removes the Ethernet or IP framing and reconstructs the original data stream. The gateways regenerate the clocks and keep both ends synchronized with no bit slips.

The receiving NetCrossing™ gateway buffers a number of incoming packets in order to compensate for the packet delivery jitter introduced by the network. The size of this buffer is configurable to accommodate different amounts of expected jitter. The gateways collect statistics of the network jitter, and can automatically optimize the buffer size for minimal link latency.

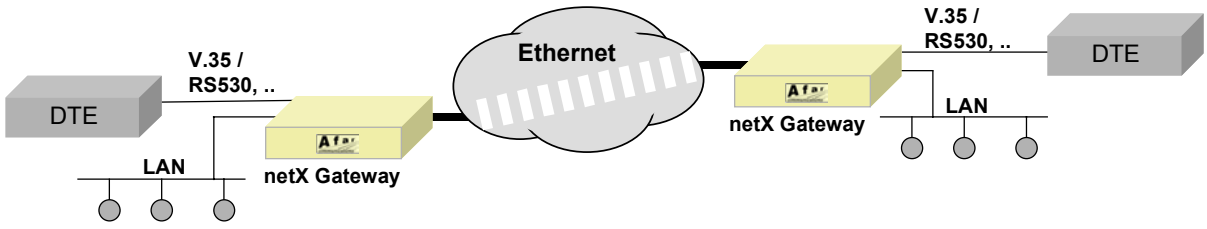


Figure 1.1 – NetCrossing Gateway Typical Application

If the application is to cross a single Ethernet network, the gateways create packets using simple and very efficient SNAP encapsulation. Or you may configure the units to perform full IP/UDP encapsulation, which allows crossing multiple networks.

When crossing a single network the gateways can automatically scout for an unconnected peer and establish a point-to-point connection with minimal configuration required. Configuration and monitoring is performed using a terminal connected to a front panel console port or through the LAN port using Telnet, SNMP or the Afar Ethernet Console program.

In addition to the serial data stream, the gateways include a user LAN Ethernet port. This port implements a transparent learning bridge which only forwards to the network the packets addressed to stations that are not in the local LAN. You can set a limit on the cumulative throughput offered to the network. In this case the gateway gives priority to the serial port and allocates to the user LAN the remaining bandwidth. This is useful if the network port has a throughput limitation imposed by, for example, a radio link.

For wireless applications the NetCrossing™ gateway is designed to work seamlessly with the Afar Wireless Ethernet radios. The gateway provides data, control and power to the radio through a single CAT5 cable. The radio is enclosed in a waterproof enclosure allowing outdoor deployment for improved system performance. In addition, if your application requires multiple wireless links emanating from the same location, the NetCrossing™ gateways can synchronize the transmissions of all the radios such that they do not cause self-interference. This is achieved by simply daisy chaining the SYNC ports of all of the NetCrossing™ gateways. Refer to the Afar Radio literature for more information on this feature.

The NetCrossing Gateway is housed in a tabletop plastic enclosure (see Figure 1.2). It is shipped with an external universal power supply that converts 100-240 VAC into the DC voltage required by the gateway.

The NetCrossing Gateway comes in two different models: in the NX2048 the maximum serial speed is limited to 2.048 Mbps while in the NX8192 the maximum serial speed is 8.192 Mbps.

Figure 1.2 - NetCrossing Gateway front view

1.2 Front Panel

Figure 1.2 shows the NetCrossing Gateway front panel. It includes six green LEDs described in table 1.2 and one DB9 female connector. This connector provides an RS-232 asynchronous port used for maintenance and initial configuration. This connector is wired as a DCE per table 1.1.

Table 1.1 – Console Port Connector (DB9) Pin Assignments

Pin	Signal Name	Abbr.	Direction
2	Receive Data	RD	Gateway to DTE
3	Transmit Data	TD	DTE to Gateway
5	Ground	GND	

Table 1.2 – Front Panel LEDs

LED	Function
Gateway/ Power	Indicates that there is DC power applied to the unit.
Radio/ Power	Indicates that there is current being drawn by the radio connected to the WAN/Radio connector.
Gateway/ Link	"RED": There is currently no link to a serial port of another gateway. "Half second blink ON/OFF": A connection to a serial port of another gateway has been established. The gateways are measuring and adapting to the jitter in the network. "GREEN": The link between the two serial ports is operating normally, and there have been no errors in the link. "AMBER": The link between the two serial ports is working but there has been an error detected since the link was established. Use the "show" command to find the errors and "show clear=1" to restore the LED to green. Momentary Blink: The gateway detected an error in the link
Radio/ Link	If the wide area network connecting the two gateways consists of a radio link using the AFAR radios, this LED indicates the state of the associated radio as follows: "OFF": The gateway has no communications with the local radio "2 second off/ blink ON": the associated radio is in auto SYNC mode and waiting for a transmission from a peer radio or for an sync message (heartbeat) from the gateway. "Half second blink ON/OFF": The associated radio has a heartbeat and is transmitting. However it has not yet received a reply from a peer radio. "Steady ON": The radio has found a peer and an RF link is established between the two radios.
Ethernet/ LAN	Indicates that there is an Ethernet connection on the LAN port. The LED blinks for each packet received.
Ethernet/ WAN	Indicates that there is an Ethernet connection on the LAN port. The LED blinks for each packet received.

1.3 Rear Panel

Figure 1.3 shows the NetCrossing Gateway rear panel. It includes several connectors described in Table 1.3

WARNING

The RJ45 connector labeled "Radio" may include DC voltage in two of the pins. It must not be connected to a LAN as this voltage may damage some LAN cards.

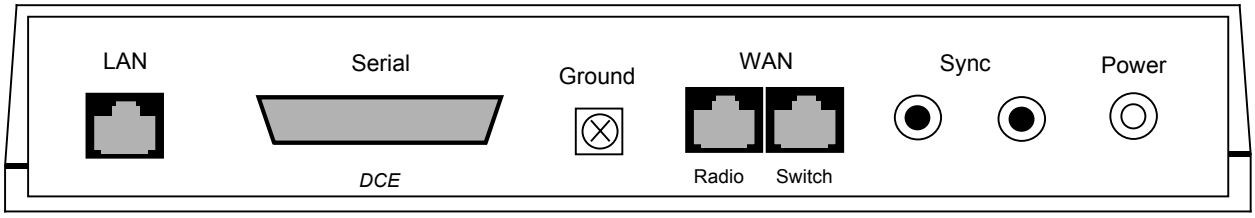


Figure 1.3 - NetCrossing Gateway Rear Panel

Table 1.3 – Rear Panel Connectors

Connector	Type	Function
LAN	RJ45	10 BaseT Ethernet connection to your "Local Area Network". It runs at 10 MHz half or full duplex. See table 1.5 for the pin assignments. The gateway "bridges" this LAN port to the WAN port.
Serial	DB25 female	Synchronous serial data stream. You can program it to several different electrical interface levels as shown in table 1.4. It is wired as a Data Communications Equipment (DCE)..
GND		Ground connection used for surge suppression. We recommend that you connect this to an Earth Ground in order to reduce the possibility of damage to the unit due to transients on external lines
WAN Radio	RJ45	10/100 Base T Ethernet connection to the "Wide Area Network". If you are connecting the Gateway to an AFAR Radio use the connector labeled "Radio". This connector includes voltage to power the radio directly (see table 1.6). In all other cases use the connector labeled "Switch" (table 1.5)
WAN Switch	RJ45	
SYNC	RCA	Synchronization signal used to synchronize multiple AFAR radios, each one connected to its Gateway. The two connectors are wired in parallel and are provided to allow connecting many Gateways in a daisy chain fashion.
Power	Switchcraft	DC Voltage to power up the gateway and optionally one AFAR radio through the WAN/Radio connector.

Table 1.4 - Data Port Connector (DB25) Pin Assignments

		Pin Assignment			
Signal Name	**Abbr.**	**RS-232**	**EIA 530**	**V.35**	**Direction**
Protective Ground	GND	1	1	1	
Transmitted Data	TD	2	(A) 2 (B) 14	(A) 2 (B) 14	DTE to Gateway
Received Data	RD	3	(A) 3 (B) 16	(A) 3 (B) 16	Gateway to DTE
Request to Send	RTS	4	(A) 4 (B) 19	4	DTE to Gateway
Clear to Send	CTS	5	(A) 5 (B) 13	5	Gateway to DTE
Data Set Ready	DSR	6	(A) 6 (B) 22	6	Gateway to DTE
Signal Ground	SG	7	7	7	
Carrier Detect	CD	8	(A) 8 (B) 10	8	Gateway to DTE
Synchronous Transmit Clock	STC	15	(A) 15 (B) 12	(A) 15 (B) 12	Gateway to DTE
Synchronous Receive Clock	SRC	17	(A) 17 (B) 9	(A) 17 (B) 9	Gateway to DTE
Data Terminal Ready	DTR	20	(A) 20 (B) 23	20	DTE to Gateway
Synchronous External Clock		24	(A) 24 (B) 11	(A) 24 (B) 11	DTE to Gateway

For the differential signals you may see some devices identifying the two differential signal polarities as (+) and (-) rather than the standard (A) and (B). When connecting to these devices note that (A) corresponds to the (-) polarity and (B) to the (+) polarity.

Table1 1.5 – "LAN" and "WAN/Switch" Ethernet Connector Pin Assignments

Pin	Signal Name	Abbr.	Direction
1	Ethernet Tx	Tx (+)	Gateway to Ethernet
2	Ethernet Tx	Tx (-)	Gateway to Ethernet
3	Ethernet Rx	Rx (+)	Ethernet to Gateway
4	(not connected)		
5	(not connected)		
6	Ethernet Rx	Rx (-)	Ethernet to gateway
7	(not connected)		
8	(not connected)		

Table 1.6 – "WAN/Radio" Ethernet Connector Pin Assignments

Pin	Signal Name	Abbr.	Direction
1	Ethernet Tx	Tx (+)	Radio to Gateway
2	Ethernet Tx	Tx (-)	Radio to Gateway
3	Ethernet Rx	Rx (+)	Gateway to Radio
4	VDC	DCV (+)	Gateway to Radio
5	VDC	DCV (+)	Gateway to Radio
6	Ethernet Rx	Rx (-)	Gateway to Radio
7	Ground	GND(-)	
8	Ground	GND(-)	

2 THEORY OF OPERATION

The NetCrosing Gateway includes two User ports: one to connect to an Ethernet Local Area Network (LAN), and a second to connect to a Serial synchronous device. The gateway processes the data from these two ports differently, but in general, data from both ports is sent to the Ethernet Wide Area Network (WAN) port. Conversely, data received in the WAN port is processed by the gateway and sent to the LAN or serial synchronous port as appropriate.

2.1 Wide Area Network (WAN)

The Gateway WAN port is connected to a "Wide Area Network", through which the gateway can reach other gateways. The **type** of WAN network connecting the various gateways is an important parameter and should be configured accordingly.

If the WAN network consists of a single Ethernet, possibly including Ethernet Bridges, the network type is classified as a **bridge** network. In this case the gateways can reach each other by sending packets addressed to the other gateway **physical address**. This physical address corresponds to the unit serial number, which is pre-configured at the factory.

If the WAN network includes Internet Protocol (IP) routers that switch IP packets to their final destination, the network type is a **route** network. In this case each gateway must be configured with its own **IP address** and the gateways send the packets to each other's IP addresses. In a routed network the gateways execute the Address Resolution Protocol (ARP) to translate the destination IP address into the physical address of the first router in the path.

The route network is the most generic and you can always operate a bridge network in route mode. However, in bridge mode the packet overhead is considerably smaller. This translates into higher effective throughput which may be important if the WAN network connecting the gateways is limited.

2.2 LAN Port

The Local Area Network (LAN) port can be connected to a user LAN or directly to the Ethernet port of a PC. The gateway implements a self learning "bridging" algorithm that transfers the Ethernet packets, as appropriate, between the LAN and the WAN ports.

Both ethernet ports are configured in "promiscuous" mode, i.e., the gateway examines all the Ethernet packets present in either port. Since these Ethernet packets contain a "source" and "destination" address, the gateway quickly learns the addresses of all the stations that are directly reachable in each network (all the "source" addresses of packets flowing in that port are reachable).

With this information on hand, the gateway examines the destination address of every Ethernet packet received and makes one of the following decisions:

1. If the destination address is the gateway own physical address accept and process the packet.

2. If the destination address is for a station that is reachable in that port, discard the packet.
3. If the destination address is the reachable on the opposite port or is unknown, queue that packet to be sent on the opposite port.

The gateway has capacity to store 500 entries in the Ethernet table. Entries are erased after a certain amount of time to allow for stations to be moved and not show up in two distinct networks. You can control this time-out with the **bridge** command. If the table ever gets full, entries that have been least used are erased to make room for new entries. You can use the **show ethernet** command to display the current list of stations known by the gateway.

The NetCrossing Gateway places the bridged packets from the LAN into a queue. Packets from this queue are then transmitted into the WAN, but in a controlled fashion such that they never delay the serial packets. Once in the WAN however, the combined traffic from the serial port and the LAN port might exceed the WAN capacity. If the equipment between the two gateways (bridges or routers in the WAN) does not distinguish between the LAN and serial traffic, it will discard packets indiscriminately. Therefore a burst of LAN traffic could cause errors in the serial link.

The NetCrossing gateway provides two mechanisms to prevent against such data loss. In a route network using IP encapsulation you can specify the **type-of-service** for the serial packets (using the **udp** command). This will tell any routers on the WAN to give priority to the serial traffic over the LAN traffic. The second mechanism is to specify a maximum WAN **capacity** (with the **wan** command). In this case the gateway will meter the traffic from the LAN such that the combined throughput from the serial and LAN ports never exceeds the specified WAN capacity.

Note that AFAR Radios distinguish between the Gateway LAN and serial packets. If the offered traffic exceeds the radio link capacity, the radios delay, and if necessary discard, the LAN packets before affecting the serial packets. Therefore, when using the AFAR Radios you do not need to use the wan capacity parameter to prevent serial data loss.

2.3 Serial Port

The serial port carries a continuous synchronous data stream. Unlike Ethernet traffic, this type of data is "connection oriented", i.e. the local serial device needs to be linked to another serial device that is connected to a remote gateway elsewhere in the WAN. You can establish this point to point connection between any two gateways connected to the same WAN. Once the connection is established the serial synchronous data flows continuously between the two serial devices as if they were connected by wire.

2.3.1 Connection Setup

You must specify the remote **peer** gateway so that a connection setup can be initiated. If the wan network type is set to **bridge** you can specify the connection in one of two ways:

1. Use the **wan** command to specify the peer serial number.

2. Use the **node** command first to assign a unique two-character alphanumeric ID to each gateway in the network. Then use the **wan** command to specify the ID of the peer gateway.

If the wan network type is set to route, use the **udp** command to specify the IP address of the peer gateway.

A gateway with an assigned peer sends, every other second, a **connect request** packet addressed to that peer (if the peer is specified with the two character ID this packet is broadcast). This connect request packet contains the serial port settings (serial speed and clock source) of the gateway requesting the connection. A gateway that receives a connect request packet responds with a **positive acknowledge** or with a **negative acknowledge** packet depending on whether its configuration is "compatible" with the requesting gateway. The compatibility requirements are as follows:

- The peer address or ID setting of each unit must identify the other as its peer. Alternatively, in bridge mode, one gateway may have its peer address unspecified (set to zero).
- Both units must not be presently connected.
- The **wan network-type** parameter must match in both units (**bridge** or **route**)
- The serial-port setting for the clock source of one gateway must be r**emote** and the other must be either **internal** or **external**.
- If the clock source is external at one gateway, the external clock into that gateway must be running.

Once a gateway that issued the connect request receives a positive acknowledge, the connection is established and the units start exchanging serial data.

In bridge mode you can leave the peer address set to zero. In this case the gateway does not transmit connect request packets. It will, however, accept connect requests from any other gateway as long as the other parameters are compatible. In route mode the peer IP addresses in each unit must identify the other unit as its peer.

If the connection is not established, the command **show status** indicates the reason why the link is not connected. In bridge mode you may also use the **show gateways** command to troubleshoot link problems. In response to this command the gateway sends a probe request packet into the network, causing all visible gateways to respond and include their configuration in the response. The command then displays all visible gateways and indicates which parameters are incompatible.

During the first ten seconds of the connection, the gateways enter a **training** mode, tuning the amount of serial data stored at each end of the link in order to avoid an "underrun" condition. During this training period the front panel Gateway-Link LED flashes green. When that LED turns into a steady green, it indicates that the training period is over and the link is established.

2.3.2 Serial Data Encapsulation

Once a connection is established, a gateway breaks the incoming serial data stream into separate packets. With the default factory setting the packet length is such that each packet contains approximately 2.5 milliseconds worth of data (400 packets per second). For example, with a serial port speed of 128 Kbps, the serial data is broken into packets that are 40 bytes in length. You can

select lower packet rates, all the way down to 10 packets per second, using the **>serial** command. Lower packet rates result in longer packets and add to the latency across the link.

These serial packets are then encapsulated into Ethernet packets and transmitted over the WAN port addressed to the peer gateway. The encapsulation format depends on the WAN **network-type**. In a **bridge** network the encapsulation overhead adds 36 bytes (includes Ethernet Header, CRC etc). In a **route** network the encapsulation adds 56 bytes.

At the receiving side, the gateway removes the packet encapsulation and queues the serial data portion of the packet to be sent out over the serial port.

2.3.3 Jitter Buffer and Link Latency

At 400 packets per second a gateway transmits over the WAN a packet containing serial data every 2.5 milliseconds. If the WAN provided an instantaneous delivery of these packets, the receiving gateway could be sending out, over the serial port, the last bit from the previous packet, when the next packet would arrive providing the next 2.5 ms worth of serial data. In this case the latency in the end to end serial link would be exactly 2.5 ms due to the store and forward delay.

However, the WAN network introduces its own delay, and worst, the delay introduced is not constant for every packet. The variation in the packet delivery time across the WAN is called **jitter**. When a serial packet is delivered late, the receiving gateway might run out of serial data causing an **underrun** error. In order to avoid these errors, the gateways store a certain amount of serial data such that the serial port always has data even when a packet is delivered late. The size of this **jitter buffer** is an important parameter of the gateway configuration. It must be made large enough to absorb the worst case jitter, but the link latency also increases by the same amount.

Sometimes the jitter in the WAN network is hard to predict or varies over time. If the WAN traffic is light when you establish the link, you might set the jitter buffer to a low value, only to find that later in the day, as the WAN traffic peaks, a lot of underrun errors occur.

The gateway keeps statistics on the network jitter and this is a great tool in establishing the optimum jitter buffer size. When you don't know the best jitter buffer value, set it to a high number to avoid underrun errors and let the link run over the peak traffic period of the WAN. Use the **show** command to examine the worst case jitter encountered. Once you know the worst case jitter reduce the jitter buffer size to be slightly above that value.

The total link latency can be approximated as:

Latency = packet_period + WAN_delay + jitter_buffer.

Where:

packet_period is the current packet-rate (displayed with the **show** command)
WAN_delay is the best case delivery across the WAN.

2.3.4 Clock Sources

There are three clock signals in the serial port DB25 connector. The NetCrossing Gateway, wired as a Data Communications Equipment (DCE), drives two of those signals and receives one of them as shown in the table below.

Clock name		Pins	Direction
Synchronous Transmit Clock	STC	(A) 15 (B) 12	Gateway to DTE
Synchronous Receive Clock	SRC	(A) 17 (B) 9	Gateway to DTE
Synchronous External Clock	SEC	(A) 24 (B) 11	DTE to Gateway

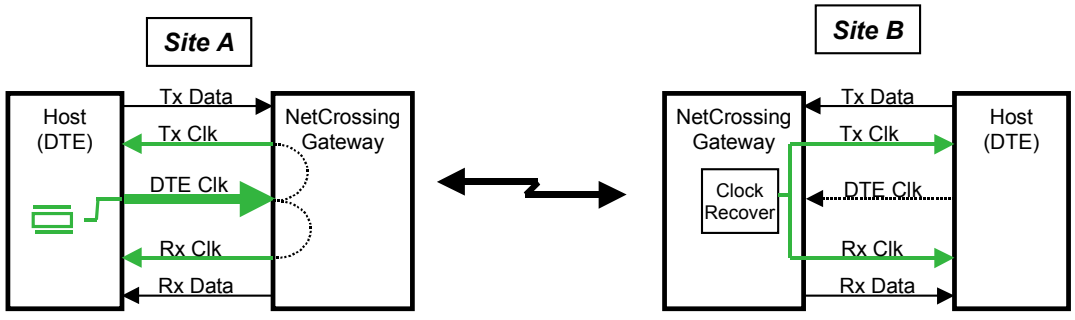
In a serial synchronous connection it is required to have a single source to clock the data stream in both directions. You can specify the clock source for each gateway in one of three different modes, as described below:

Internal: The Tx and Rx clocks are generated by the gateway using a crystal controlled frequency synthesizer. In this case the gateway ignores the External Clock

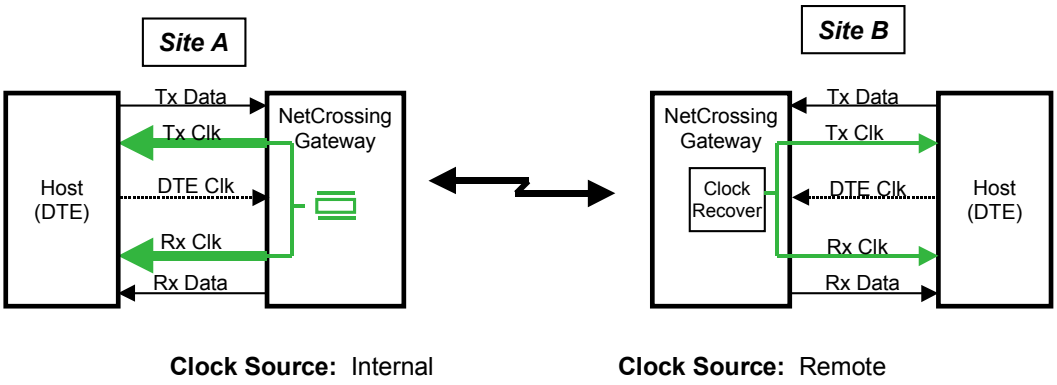
External: The clock is supplied by the external DTE on the SEC line. The gateway loops that clock back into the Tx and Rx clock lines. The gateway also makes a measurement of that clock to pass its value to the remote unit.

Remote: The Tx and Rx clocks are generated in the gateway frequency synthesizer, and once a link is established, they are locked to the clock at the remote gateway.

These three configurations can be combined in only two compatible ways as shown in the following figures.



Clock Source: External **Clock Source:** Remote



2.3.5 Clock Edges

The signals in the serial port connector are inverted by the drivers in the devices. For the case of differential signals the (A) side is inverted (-) while the (B) side is non-inverted (+). The information in this section regarding clock edges refers to the inverted signals as visible on a scope at the DB25 port.

The RxData (output from the gateway) is clocked out on the rising edge of the RxClk. There is a short propagation delay from the rising edge of this RxClk until the RxData changes state. At the host device you can typically use either edge to clock-in the data, but you are safer if you use the falling edge of the RxClk to latch this data.

The gateway latches the TxData (input into the gateway) on the falling edge of the TxClk.

When you select the clock to be external, the gateway replaces the internally generated RxClk and TxClk with this external clock. The external clock therefore shows up at the RxClk and TxClk on the DB25 connector after a short delay, typically around 10 ns.

2.4 Synchronization Port

The two SYNC connectors are used to propagate radio synchronization information between co-located radio/gateway pairs. This feature is only applicable when you use the NetCrossing Gateway in conjunction with Afar radios

In a network consisting of Afar radios and gateways, synchronization information flows between all units through (i) RF links, (ii) Ethernet messages, and/or (iii) the SYNC connectors of the gateways. This synchronization mechanism allows deploying a large network where all radios have their TDD cycles synchronized, eliminating self-generated interference between co-located radios.

When you co-locate multiple radios with each one connected to a gateway, you should daisy-chain the SYNC signal between all gateways. Then use the **tdd** command to set the **synch-mode** of all the co-located gateways to either **auto** or **master**. In any large network only one site should be the source of the timing. Set the **synch-mode** to **master** only at that site. The master gateways will "co-share" the timing such that, even if one gateway fails, the synchronization information continues to flow. . This synchronization feature is explained in detail in the radio manual.

3 UNIT CHECKOUT AND FIRMWARE UPGRADES

If you are a first time user we recommended that you perform an initial check on the bench before a field installation. For this bench check out you need two NetCrossing Gateway units and a crossover Ethernet cable to connect the two units back to back. You may also connect them through an Ethernet switch using two standard Ethernet cables.

In this checkout use the Console port to configure the units. You can also do this using the LAN port connected to a PC and running the program ECON described in appendix C.

3.1 Bench Check Out

1. Connect each NetCrossing Gateway front panel Console Port to a terminal, or a PC running a terminal emulation program. Configure the terminal settings as follows:
   Baud rate: 9600
   Word length: 8 bits
   Parity: none
   Stop bits: 1

2. Connect the WAN port labeled "Swicth" of one NetCrossing Gateway to the same port of the second gateway using an ethernet crossover cable.

3. Connect the Power Supplies of the two units to a power outlet of the appropriate voltage.

4. The terminals should display a banner identifying the unit serial number, hardware and software versions, followed by the command prompt:

   nxg-nnnnn #>

   where nnnnn are the last five digits of the unit serial number (the prompt may be the unit name if that has been pre-configured).

5. Set the first gateway to its factory default configuration by typing the commands:

   > load factory
   > save-configuration

6. Set the second unit to a "compatible" mode by typing the commands:
   > load factory
   > **serial clock=internal**
   > wan peer-sn=NNNNN (numeric digits of the peer unit serial number – no leading zeros)

7. The Gateway-Link LED in the front panel should blink green for approximately 10 seconds and then remain steady green. This indicates that the two serial ports are now connected at the default speed of 128000 bps.

8. The terminal connected to each gateway can be used to further modify the unit's operating parameters. Section 5 describes the command language used to perform those functions.

3.2 Upgrading the Firmware.

3.2.1 Description

The operational firmware for the NetCrossing Gateway is stored in Flash PROM and can be easily updated. The Flash PROM can hold multiple versions of the firmware simultaneously. The table below lists some of the "File Utility" commands used to download and manage the various files stored in Flash PROM. A more detailed explanation for each command can be found in section 5.

File Utility command summary

Command	Description
directory	Lists all files stored in Flash PROM
delete-file filename	deletes the specified file from the directory
download-file path/filename	downloads the specified file from the PC path/filename into the Flash PROM
set-default-program filename	Sets the indicated filename as the default program to run after power up
run-file filename	loads the indicated program into RAM and executes it.

New firmware versions are made available from time to time at the following page in our website:

http://www.afar.net/support.htm

The firmware files are named:

nxgNN_NN.bz (binary zipped file for downloads through the Ethernet port)
nxgNN_NN.dwn (ascii file for download through the serial port, or via Telnet)

where NN_NN is the firmware version number. The website contains instructions for transferring the files into your PC.

A new file can be downloaded into the gateways in one of three ways: (1) Using the "econ" program running in a PC connected to the LAN port of one of the gateways. This is the fastest method and

allows you to download to any other gateway that can be reached through the WAN port (in a bridge WAN network), (2) Using a Telnet session from anywhere on the Internet. This requires the gateway to have been pre-configured with an IP address. (3) Using a terminal emulator program (e.g. HyperTerminal) running on a PC connected through the serial port to the gateway RS-232 console port. This method only allows you to download to that specific gateway.

The next three sessions explain in detail how to download a new file using each method.

3.2.2 Installing new firmware through the Ethernet port

This procedure assumes that the new firmware needs to be installed in both radios of a working link. The upgrade is performed from a single PC connected via Ethernet to one of the two radios. Note that new firmware does not need to be compatible with the firmware currently running. You can still download incompatible firmware and restart the link from a single location.

1. If you have not done so, install the utility program "econ" in the PC. This utility program is distributed with the gateways and can also be downloaded from the website. Please refer to appendix C for instructions on how to install this utility.

2. Make sure the file with the new firmware (file nxgNN_NN.bz) is available in the PC.

3. Start the econsole utility by typing "econ" from a DOS window. Econ will send a "discovery" message and display all the AFAR devices (radios or gateways) that can be seen. Verify that all gateways in the network are listed. Then select the gateways that you wish to upgrade.

4. Issue the command:

   >directory

   to view a list of files stored in Flash PROM as well as the available free space. Verify that the free space in flash PROM is larger than the size of the nxgNN_NN.bz file in the PC. If there is not enough space in Flash PROM delete one of the program files to make up space (use command >delete filename).

5. If the gateway configuration has been password protected, you must first unlock the protection with the command:

   >**unlock enable-configuration=***password*

   (when the configuration is unlocked, the gateway prompt ends with the characters '#>. In locked mode the prompt does not include the '#' character).

6. Issue the command:

   >download *path/*nxgNN_NN.bz

   where *path/* is the directory in the PC where the nxgNN_NN.bz file is stored. The *path/* can be omitted if the file is in the same directory as the ECON program. As the download proceeds econ displays a line showing the current percentage complete.

7. Once the download is complete, issue the command:

   >set-default-program nxgNN_NN

in order to make the new file the default program to run after a reset.

8. Depress the "**F4**" key to log-off the session with the current gateway. "Econ" displays the list of all devices from the initial discovery phase. Select another gateway in the network and repeat steps 4 through 7.

9. Once all the gateways in the network have the new program, log onto each one (using econsole) and issue the command:

   >reboot

   to cause the gateway to restart using the new firmware.

10. Wait at least ten seconds from the moment you entered the reboot command, then press <CR>. Econsole automatically attempts to reconnect to the same gateway. Once a new session with that gateway is reopened issue the command:

    > version

    and check that the gateway is indeed executing the new version.

11. Repeat the previous two steps for each gateway in the network.

Note that the file downloads are executed with serial links in full operation. The only downtime in the link occurs when the gateways are rebooting. Unless the major version of the firmware has changed the gateway configuration is kept intact when a new version is started (see the release notes for details). The downtime for the gateway being restarted, is typically less than twenty seconds.

3.2.3 Installing new firmware using Telnet

Telnet is a protocol that allows you to conduct a remote gateway command session from a local host. The gateway must have been pre-configured with an IP address and be reachable, over the network, from the local host. Refer to section 5 for details on how to configure a gateway IP address and initiate a Telnet session. The Telnet terminal emulation must have the capability of sending an ASCII file to the remote machine. The following description assumes you are using Hyperterminal as the local Telnet terminal emulation.

1. Verify that the new software is available in the local machine. The download software for upgrade via Telnet must have a ".dwn" extension, e.g., nxg01_05.dwn.

2. Initiate a Telnet session with the gateway as described in section 5.

3. If the gateway configuration has been password protected, you must first unlock the protection with the command:

   >**unlock enable-configuration**=*password*

   (when the configuration is unlocked, the gateway prompt ends with the characters '#>. In locked mode the prompt does not include the '#' character).

4. Issue the command:

   >directory

to view a list of files stored in Flash PROM as well as the available free space. Verify that there is enough free space in flash PROM for the new file. The space required will be the size of the nxgNN_NN.dwn file divided by 2.5. If there is not enough space in Flash PROM delete one of the program files to make up space (use command >delete filename).

5. Start the download process by typing:

   >download-file destination=nxgNN_NN method=inline

   where NN_NN file is new version of software being installed.

6. The gateway will return with the following:

   "*Send the file ... if incomplete, end with a line with just a period*"

   When you get this prompt, go to "Transfer-Send Text file…" in Hyperterminal and select the file to be installed. The file must have a ".dwn" extension.

7. After the file is successfully installed issue the command:

   >directory

   to insure that the file has been loaded into memory.

8. Issue the command:

   >set-default-program filename=nxgNN_NN

   where NN_NN file is new version of software being installed.

9. Issue the command:

   >reboot

   to restart the gateway with the new software. Close the Telnet session, wait a few seconds and open a new session with the same gateway.

10. Issue the command:

    >version

    to insure the gateway is running the latest version.

3.2.4 Installing new firmware using the RS-232 serial port

On occasion, it may be necessary to install new firmware using the RS-232 port. This is generally a less desirable method as the download time takes longer and you can only update the gateway that is directly connected to the PC, i.e., remote updates are not possible.

The serial upgrade uses a PC with a terminal emulator. Any emulator can be used, however, it must have the facility to download a text file on demand. In the example below, the emulator used is Windows Hyperterminal.

1. Connect the *NetCrossing Gateway* Console Port to a terminal, or a PC running a terminal emulation program. Configure the terminal settings as follows:
   Baud rate: 9600

Word length: 8 bits
Parity: none
Stop bits: 1

2. Verify that the new software is available in the PC. The download software for the serial upgrade must have a ".dwn" extension, e.g., nxg01_05.dwn.

3. To have the shortest download time possible, set the gateway to use the highest RS-232 speed allowable on the PC. In this example, a download speed of 57600 baud will be used. Set the console speed of the gateway to 57600 baud by issuing the command:

   >console-speed-bps 57600

4. Change the baud rate of the PC to match the gateway. Remember that with Hyperterminal, you must disconnect the session and re-connect before the changes will take effect. Verify the PC communicates with the gateway again.

5. If the gateway configuration has been password protected, you must first unlock the protection with the command:

   >**unlock enable-configuration**=*password*

   (when the configuration is unlocked, the gateway prompt ends with the characters '#>. In locked mode the prompt does not include the '#' character).

6. Issue the command:

   >directory

   to view a list of files stored in Flash PROM as well as the available free space. Verify that there is enough free space in flash PROM for the new file. The space required will be the size of the nxgNN_NN.dwn file divided by 2.5. If there is not enough space in Flash PROM delete one of the program files to make up space (use command >delete filename).

7. Start the download process by typing:

   >download-file destination=nxgNN_NN method=inline

   where NN_NN file is new version of software being installed.

8. The gateway will return with the following:

   "*Send the file ... if incomplete, end with a line with just a period*"

   When you get this prompt, go to "Transfer-Send Text file…" in Hyperterminal and select the file to be installed. The file must have a ".dwn" extension.

9. After the file is successfully installed issue the command:

   >directory

   to insure that the file has been loaded into memory.

10. Issue the command:

    >set-default-program filename=nxgNN_NN

where NN_NN file is new version of software being installed.

11. Issue the command:

    >reboot

    to restart the gateway with the new software. Remember to change the PC Hyperterminal settings back to 9600 baud and disconnect/re-connect the session.

12. Issue the command:

    >version

    to insure the gateway is running the latest version.

3.2.5 Feature upgrades

The *NetCrossing Gateway* has the ability to turn ON or OFF optional features and capabilities. This is done via the use of the "license" command. This command requires a "key" that is specific to a particular gateway serial number and capability. To obtain a feature key, you must supply the specific model number, the serial number, and the feature desired. Please contact your local distributor for a list of optional features available for your gateway.

Refer to Section 5.10 under "license" for the specific use of the license command.

4 COMMANDS

4.1 Configuration techniques

There are three ways to configure the gateway. One uses the RS-232 console port in the unit front panel. This port is always set to operate with the following parameters:

Baud rate: 9600
Word length: 8 bits
Parity: none
Stop bits: 1

This console port allows configuring and monitoring only the local gateway, i.e. you can not monitor and configure any of the remote gateways reachable through the WAN port.

A second configuration method uses the gateway LAN Ethernet port. This approach has the advantage that any gateway reachable across the WAN in a bridge network, can be configured from a single PC.

In order to use the Ethernet connection to configure the gateways the "Ethernet Console Program" (Econsole) needs to be installed at a PC. This PC must be connected to the same LAN as the gateway LAN port. Refer to Appendix C for instructions on the installation of Econsole.

The third configuration method is using Telnet from a remote location. Telnet is explained in more detail in section 6.

After power up the gateway displays a banner identifying the hardware and software versions followed by the command prompt. The default prompt is:

nxg-nnnnn #>

where nnnnn are the last five digits of the gateway serial number. If a node "name" has been assigned to the unit, the prompt will be that name.

The "help" command provides a list of all the commands available. To get more detailed help for a specific command, type "help command-name".

The gateway keeps a history of several of the previously issued commands. Those commands can be viewed by pressing the up-arrow and down-arrow keys on the keyboard. Any of those previously issued commands can then be edited and reentered by pressing the <Enter> key.

4.2 Command syntax

The command interpreter in the NetCrossing Gateway is designed to accommodate both a novice as well as an expert operator. All commands and parameters have descriptive names so that they are easily remembered and their meaning is clear. In order to be descriptive however, those commands

are sometimes long. As the operator becomes familiar with the command language, typing the complete words could become cumbersome. The NetCrossing Gateway command interpreter recognizes any abbreviations to commands and parameter names, as long as they are unambiguous. If an ambiguous command is entered, the gateway will output all possible choices.

Commands have the following generic form:

command parameter=value parameter=value

Following is a brief list of syntax rules:

- Words (for commands, parameters, or values) can be abbreviated to a point where they are unambiguous.

- Some commands or parameters consist of compound words separated by an hyphen. With compound words, the hyphen is optional. Additionally each word in a compound word can be abbreviated separately. For example, the following are all valid abbreviations for the command "save-configuration": "save", "savec" s-c" "sc".

- The parameter and value lists are context sensitive, i.e., in order to solve ambiguities the command interpreter only considers parameters valid for current command, or values valid for the current parameter.

- The arguments "parameter=value" must be entered with no blank spaces on either side of the '=' sign. Those arguments (parameter/value pairs) can be listed in any order.

- Even though parameters can be listed in any order, there is a "natural" order known by the command interpreter. This allows the user to specify parameter values without having to type the parameter names. For example the command

  > **date date=16-may-2003 time=10:32:06**

  can be entered as :

  >**date 16-may-2003 10:32:06**

- Using the preceding rule, for commands that have a single argument, the "parameter name" part of the argument is always optional, i.e., you can enter:

  >*command value*

  For example the command:
  >**save-configuration destination=main**
  can be shortened to any of the following:
  >**save-configuration main**
  >**save main**
  >**save**

- Not all parameters associated with a command need to be specified. Depending on the command, when a parameter is omitted it either assumes a default value or keeps the last value assigned to that parameter.

- For all parameters that accept a numeric value, the number can be entered in decimal or hexadecimal notation. To enter a number in hexadecimal notation precede it with a 0x or 0X. All other numeric values are interpreted as decimal. Example:

  >**serial idle-code=0x7E** (hexadecimal)

The following sections describe the various commands grouped according to their functionality. A summary list of all commands are contained in Appendices A and B.

4.3 Configuration Management Commands

A **gateway configuration** consists of a set of programmable parameters that define the gateway operation with regard to a variety of operating modes. There are five different configurations identified as **current, main, alternate, factory** and **basic**.

The **main** and **alternate** configurations are both stored in non-volatile memory. They can be loaded into the **current** configuration with the **load** command. On power up the gateway loads the **main** configuration from non-volatile memory into the current configuration.

The **current** configuration is the set of parameters currently being used and can be modified by the operator through several commands. This configuration is volatile. If the current configuration has been modified it should be saved using the **save** command. Otherwise the modifications will be lost if power is removed.

The **factory** configuration can not be modified by the operator and is used to return the gateway to the factory default condition. It is useful as a starting point to create a customized configuration.

The **basic** configuration is similar to the factory configuration with the exception that a few parameters are left unchanged when you issue the **load basic** command. These parameters are the radio power on/off mode and the IP parameters. This is useful when you are logged on to a remote unit and need to start from a known configuration. If you were to issue the **load factory** command you might lose contact with the remote unit if, for example, it powers down the radio at the remote site.

The access to change the gateway configuration can be password protected. This password is set by the user with the **change-password** command. Once a password is set, issue the **lock** command to prevent any unauthorized changes to the configuration. Once locked, the configuration can only be modified by issuing the **unlock** command with the correct password.

When the configuration is unlocked, the gateway prompt ends with the characters '#>' to remind the user that the configuration is unlocked. In locked mode the prompt does not include the '#' character. Once a password is set, the gateway will automatically lock the configuration after 10 minutes without any commands being issued.

The configuration management commands are listed below:

change-password

enable-configuration="ASCII string"

This command allows the user to set or change a password used to **lock** and **unlock** access to the commands that change the gateway configuration. The gateway is shipped with no password which allows access to all commands. Once a password is set and the configuration is locked, the password is needed to unlock the access to those commands. After changing the password you should also issue the **save-configuration** command to save the new password in non-volatile memory.

Examples:

>change-password enable-configuration=bh7g8

WARNING

The NetCrossing Gateway is shipped with no password. If the "change-password" command is issued make sure you do not forget the password. Once locked, without a password, the gateway must be returned to the factory to be unlocked.

display-configuration

source= current or ***main*** or ***alternate*** or ***basic*** or ***factory***

Displays all the parameter values for the specified configuration. If the source is not specified it defaults to **current**.

Examples:

> display-configuration factory

> discon

load-configuration

source=main or ***alternate*** or ***basic*** or ***factory***

Loads the specified configuration into the current set of parameters controlling the gateway operation. If no source is specified it defaults to the **main** configuration.

Examples:

> **load-configuration source=factory**
> **load**

lock

This command locks the access to all the commands that can alter the gateway configuration. Once locked use the **unlock** command to regain access to those commands. Note that a password must be set prior to the **lock** command being issued (the gateways are shipped with

no password), otherwise the lock command has no effect. If a password is set, the gateway automatically locks the configuration at the end of 10 minutes with no command activity.

save-configuration

destination=main or ***alternate***

Saves the current set of gateway operating parameters into one of the two non-volatile configurations. If the destination is not specified it defaults to **main**.

Examples:

> **save-configuration destination=alternate**
> **save**

unlock

debug-mode="ASCII string"
enable-configuration="ASCII string"

This command unlocks the access to various commands. The **enable-configuration** password (set with the change-password command) unlocks the various commands listed in this manual that alter the radio configuration. The **debug-mode** is a factory mode used for troubleshooting by customer support.

Examples:

> unlock enable-configuration=bh7g8

4.4 Major Configuration Parameters

These commands change several operating parameters that are part of the NetCrossing Gateway configuration. When entering commands with multiple parameters, if a parameter is not included, that parameter keeps its current value.

bridge

The gateway implements a self learning bridging algorithm that transfers the Ethernet packets, as appropriate, between the LAN and the WAN ports (see section 2.2). In this algorithm the gateway stores all station addresses, and ports (LAN or WAN) where they have been seen. The gateway has capacity to store 500 entries in this Ethernet table. This command sets the timeouts relating to these entries. You can use the **show ethernet** command to display the current list of stations known by the gateway.

station-timeout-sec=5..1800

Sets the time the gateway will retain, in its internal table, Ethernet addresses obtained from the network.

multi-cast-timeout-sec=5..3600

Sets the time the gateway will retain, in its internal table, Ethernet multi-cast addresses obtained from the network. This can not be set to a value below the station-timeout.

Examples:

> **ethernet statio=100 multicast=500**

local-area-network

speed=10hdx or ***10fdx***

Sets the LAN ethernet port speed to 10Mbps half-duplex (10hdx), or 10 Mbps full-duplex (10fdx).

node

name="ASCII string"

Gives the node a meaningful name for further reference. This name will be used as the command prompt. It is also used to identify the node in a variety of commands and displays. The name field can be up to 31 characters with no spaces. If spaces are desired, you may include the whole name in quotation marks.

location="ASCII string"

Optional parameter to define the location of the node. This field is displayed in the "display-configuration" output and also reported through SNMP. This field is used for information only. The location string can be up to 31 characters with no spaces. If spaces are desired, you may include the whole string in quotation marks.

contact="ASCII string"

Optional parameter to define the contact for maintenance purposes. This field is displayed in the "Display-configuration" output and also reported through SNMP. This field is used for information only. The contact string can be up to 31 characters with no spaces. If spaces are desired, you may include the whole string in quotation marks.

id=xx

This is a two-character alphanumeric ID used as an alternate way for establishing a connection between two gateways. You need to be in bridge mode and have both the **node-id** and the **peer-id** defined (using the **wan** command) for the gateways to establish a connection using the ID approach. To delete the ID enter it as 00

Examples:

>node name=bank location="wall street" contact=866-555-1234

radio

power=on or ***off***

Turns **on** or **off** the voltage to power up an external radio connected to the RJ45 connector labeled "radio". When this voltage is present you need to exercise care in not connecting the RJ45 radio connector to any other devices as this extra voltage may damage them.

serial-port

clock-source=internal or ***external*** or ***remote***

Selects the source for the generation of the local Tx and Rx serial clocks used to clock the data in and out of the gateway as follows:

internal: The clocks are generated inside the gateway in its frequency synthesizer circuit, driven from an internal Crystal Oscillator.

external: The Tx and Rx clocks are driven from an external clock source (DTE) on pins 24 and 11 of the DB25 connector.

remote: The Tx and Rx clocks are generated internally but locked to the Tx and Rx clocks of the peer unit.

In order for two gateways to establish a serial link, the clock source in the two units must be set in a compatible fashion. This requires that one unit have its clock source set to remote, and the other to either internal or external.

speed-bps=[3500..8192000] (for the NX-2048 the maximum speed is 2048000)

If the clock-source is set to internal, this parameter selects the serial port speed in bits per second. The Gateway can synthesize a large number of frequencies including all the multiples of 64Kbps and 56 Kbps. If you enter a non-standard frequency the Gateway will synthesize the nearest frequency available (and tell you that your value has been overwritten). The **display-configuration** will show the actual speed being used.

If the clock-source is set to **remote** or **external** this parameter is ignored. In remote mode the clock speed will be set by the remote gateway when the link is established. In external mode, the gateway measures and uses the frequency being supplied by the external equipment.

channel-64kbs=[1..32]

This is an alternate way of specifying the serial port speed, in multiples of 64 Kilobits per second. A value of 4 for example corresponds to 256000 bps.

interface=rs530a or ***rs530*** or ***x.21*** or ***v.35*** or ***rs449*** or ***rs232***

Specifies the electrical interface standard for the signals in the DB25 serial port connector. Refer to section 1 for the pin assignments for each of the standards.

loopback=disable or ***input*** or ***output***

Selects different loopback modes of the serial port used for test purposes as follows:

disable: default value used for normal operation.

input: The Tx data stream input into the gateway is looped back to come out in the Rx data stream. The data stream received from the remote unit is discarded.

output: The Rx data stream output by the gateway is looped back into the Tx data stream. The data stream input from the DTE is discarded. This is equivalent to placing a loopback plug in the DB25 connector.

idle-code=0..255

Specifies the byte to output through the serial port when no other data is available (either when there is no link or in the case of an underrun).

clear-to-send=on or ***off*** or ***remote-rts*** or ***link-state***
data-set-ready=on or ***off*** or ***remote-dtr***
carrier-detect=on or ***off*** or ***remote-rts*** or ***link-state***

These parameters specify different options for the three handshake lines driven by the gateway. When set to **link-state** the gateway asserts those signals when it has a serial connection with a peer gateway.

packets-per-second=10 or ***25*** or ***50*** or ***100*** or ***200*** or ***400***

Specifies a **target** value for the packet rate generated by the gateway. The gateway adjusts the actual packet rate such that the packets do not exceed the maximum Ethernet packet length and are always a multiple of four bytes. Use the ">show" command to find the resulting packet rate and packet size for each setting. Lower packet rates result in longer packets and longer latencies across the link.

time-division-duplex

synch-mode = off or ***master*** or ***auto***

If the network does not consist of AFAR radios set this parameter to **off**. If the network consists of a pair of AFAR radios, this parameter specifies the radio Time Division Duplex (TDD) synchronization mechanism across the radio network. Refer to section 2 of this manual, and the radio manual for a description of the wireless network synchronization features.

cycle-period-ms= 20 or ***40***

This parameter specifies the TDD cycle period of the radios in the network. It is only relevant for a gateway with the synchronization mode set to master. In master mode, if multiple gateways are synchronized together using the SYNC connector, all gateways must have the cycle period set to the same value.

Example:

> tdd sync=master cycle=40

udp

peer-ip-address=<ip address>

When the **network-type** is set to **route**, this parameter specifies the remote gateway IP address for the serial port point-to-point link. After power up, the NetCrossing Gateway transmits a "connect request" packet, every second, addressed to the specified peer until a connection is established. Additionally, upon receiving a connect request from another gateway, a connection will be established only if the request originated from the specified peer.

The peer IP address must be specified at both ends of the link.

port=1..65535

The gateway uses this number in the source and destination port fields of all the UDP packets generated by the gateways. The only reason you would need to alter the default value is if those packets conflict with some other UDP packets in the WAN.

type-of-service=0..255

Specifies the value to put in the **type-of-service** field of the UDP encapsulated serial packets. The default value already requests a higher priority for the serial packets.

wide-area-network

type=bridge or ***route***

This parameter specifies the type of "Wide Area Network" (WAN) between this gateway and its peer. The type of network determines how the gateways encapsulate the serial data into packets as follows:

route – the WAN network includes one or more routers using Internet Protocol (IP) to connect between the two gateways. In this case the serial data is encapsulated in a complete IP/UDP frame, and addressed to the peer IP address (specified with the **udp** command). This encapsulation is the most generic but has more overhead and therefore is less efficient than the bridge encapsulation.

bridge – the WAN network consists of a single Ethernet LAN (which may include bridges). In this case the serial data is encapsulated in a more efficient SNAP packet, and the packets are addressed directly to the peer gateway physical address.

peer-serial-number=0..999999

When the **network-type** is set to **bridge**, this parameter specifies the remote gateway for the serial port point-to-point link. After power up, the NetCrossing Gateway transmits a "connect request" packet, every two seconds, addressed to the specified peer until a connection is established. Additionally, upon receiving a connect request from another gateway, a connection will be established only if the request originated from the specified peer. When specifying the serial number do not enter any leading zeros.

You can also set the peer serial number to 0. In this case the unit does not transmit connect request packets and accepts a connect request from any other gateway.

See also the **peer-id** parameter below as an alternate way of specifying the peer device without using the hard-coded serial number.

peer-id=xx

This is a two-character alphanumeric ID used as an alternate way for establishing a connection between two gateways. You need to be in bridge mode and have both the **peer-id** and the **node-id** defined (using the **node** command) for the gateways to establish a connection using the ID approach. To delete the ID enter it as 00

jitter-ms=0..500

Specifies the average size of the serial buffer at the receiving end of the link used to absorb the jitter in the packet delivery across the network. The default value of zero selects an **automatic** mode described below.

Setting this parameter to a low value reduces the latency in the serial link but increases the probability of underrun errors. If you are unsure of what value to use, start by setting the jitter parameter to a large value. The NetCrossing Gateways keeps track of the worst case jitter ever encountered, which can be examined with the **show** command. After running the link for some time set the jitter parameter to the worst case value plus a small margin.

If during normal operation you experience too many underrun events, you may want to increase this value further.

In order to specify the larger values for the jitter buffer (above 200) you may need to reduce the serial packet rate. Use the command ">serial packet-per-second=XX" to reduce the packet rate below the default 400. However, with very large serial clock speeds the gateway must keep the packet rate high so that it does not exceed the maximum Ethernet packet size. Therefore, the maximum value attainable for the jitter buffer decreases as you increase the serial speed. Use the "show" command to check the actual packet rate and jitter buffer.

Automatic mode: If the jitter value is set to zero the unit performs the process described above automatically: when the link is established the unit sets the jitter buffer to its maximum value (200 ms), and then measures the network jitter. At the end of approximately 12 seconds the unit sets the jitter buffer to an appropriate value based on that measurement. You may examine the final value with the **show** command. Note that the automatic mode bases the final jitter parameter value on only 12 seconds of measurements. If the network jitter varies significantly over time you will need to adjust this value manually.

capacity-kbps=100..20000

If the throughput in the WAN is limited, and the equipment between the two gateways (radios, bridges or routers in the WAN) does not distinguish between the LAN and serial traffic, it will discard packets indiscriminately. Therefore a burst of LAN traffic could cause errors in the serial link.

With this parameter you can specify the throughput capacity of the WAN in kilobits per second. The gateway will first allocate enough of the WAN capacity to the serial data, and limit the traffic from the LAN such that the combined (serial plus LAN) traffic will not exceed the specified capacity.

When you use Afar radios with the gateway, set the capacity to the default of 20000. The Afar radios recognize the packets as coming from the LAN or serial ports and delay or discard the LAN packets that would exceed the radio throughput.

speed=auto-10 or ***10hdx*** or ***10fdx*** or ***100hdx*** or ***100fdx*** or ***auto***

Allows selecting the WAN ethernet port speed between 10/100 Mbps, half / full duplex, or auto negotiate.

The **auto-10**-setting forces the speed to 10Mbps but negotiates the half or full duplex. The **auto** setting negotiates both the speed and duplex to the fastest configuration supported by the other devices on the WAN.

promiscuous=yes or ***no***

This parameter affects the bridging of ethernet packets between the WAN and LAN ports. With promiscuous mode turned on (default) the gateway looks at every packet present on its WAN port to decide whether that packet should be bridged to the LAN port or not.

With heavy traffic on the WAN port the gateway may not have enough time to inspect every WAN packet in this way, which may then cause errors in the serial link. You can turn off the promiscuous mode, which avoids this condition. With promiscuous mode off you can still use the LAN port to configure the gateway, although you lose the bridging between the WAN and the LAN ports.

Examples:

>wan type=bridge peer =170145 jitter=20 speed=auto

4.5 Internet Protocol (IP) Management Commands

The IP Management commands configure the gateway IP protocol parameters which allow the gateway to be monitored and configured through Telnet and SNMP. Refer to section 6 for a more detailed explanation on those two applications.

ip-configuration

address=<ip address>
netmask=<string>
gateway=<ip address>

This command configures the gateway IP address, netmask and gateway. The IP configuration is optional and the gateways are shipped with these parameters left blank. Once the IP configuration has been initialized, the gateways will reply to "ping" packets. The IP configuration is also required in order to use the "ping", "snmp" and "telnet" features.

Example:

> **ipconfig add=207.154.90.81 netmask=255.255.255.0 gateway=207.154.90.2**

ping

destination=<string>
count=0..500
size-bytes=32..1400

This command causes the gateway to "ping" the destination address and display the results. The "ping" packet consists of an ICMP packet with a length specified by the "size-bytes" parameter. The destination is any valid IP address. When the destination host receives the packet it generates a reply of the same size. Upon receiving the reply the gateway displays the round trip delay. This process is repeated until the number of replies reaches the value specified by the "count" parameter (default to 4). A count of zero leaves ping running indefinitely until stopped by the user.

Example:

> **ping 207.154.90.81 count=10 size=100**

snmp

The gateway runs an SNMP agent which allows up to four IP addresses to be specified as valid SNMP managers. This command configures those IP addresses and the type of access allowed. You can issue the command up to four times to specify each separate IP address manager. The gateways are shipped with all entries blank. While no entries are specified, the unit accepts SNMP "get" requests from any IP address with the "public" community. Once one or more entries are specified, the gateway only responds to requests from the specific IP addresses listed. This list of authorized managers is also used for validating Telnet requests.

Refer to section 6 for an overview of Network Management using SNMP and Telnet.

manager=<ip address>

Specifies one valid IP address where the SNMP manager or Telnet session will run.

community=<string>

Any string of up to 9 characters. For SNMP requests the "community" field in the request packet from this IP address must match this parameter. For a Telnet session the username entered when initiating the session from this IP address must match this string. If this parameter is not specified it defaults to "public". Note that you must always enter the "manager" IP address in the same command line that sets the "community" value.

access=g or ***gs*** or ***gst*** or ***gt***

SNMP access type authorized for this IP manager. Specify as any combination of three letters: g (get), s (set) and t(trap). If this parameter is not specified it defaults to "get". Note that you

must always enter the "manager" IP address in the same command line that sets the "access" value.

authentication-traps=0 or 1

Specifies whether an "authentication trap" should be generated if a SNMP request is received that can not be honored (due to invalid IP address, community or access fields). When enabled, all IP managers that have "trap" access will receive this trap.

delete=1..4

Allows deleting one entry in the SNMP table. The number 1..4 refer to the entry number as listed in the "display configuration" report.

Example:

> **snmp manager=207.154.90.81 com=support access=gst**

4.6 Installation and Monitoring Commands

show-table

table=status or ***gateways*** or ***ethernet*** or ***econsole*** or ***ip-stack***
format=counts or ***times***
clear=1 or ***0***

This command displays various tables in different formats as described below. The "clear=1" parameter lets you clear the error counts in the status tables as described below.

status table

This contains miscellaneous information including:

- System start time and current time
- Serial link status, peer, maximum jitter encountered, number of errors and time of last error.
- Serial port speed, packet generation rate and packet size.
- If the Gateway is connected to an Afar radio, displays the radio synchronization status and how much throughput is available for LAN traffic.

The statistics shown in this table (maximum jitter, number of errors, last error) can be cleared by specifying the clear=1 parameter. Note that when the number of errors in the link is non-zero, the "link" LED in the front panel is yellow. By issuing the command "show clear=1" you clear the error count and restore the link LED to green..

gateways table

This table lists all the gateways that are reachable in the WAN network, several of their configuration parameters and whether they are compatible to establish a serial link with this unit. The first entry is always for the unit itself.

ethernet table

This table can be displayed in two formats, "counts" (default) and "times".

>show ethernet

```
Ethernet Stations:
                                                 --Discard--      --Forward--
 #       MAC address   IP address    Port      from       to      from       to
 0 ff-ff-ff-ff-ff-ff                  0         0        0         0      183
 1 00-d0-39-00-2d-cb                 -2         0        0       209      165
 2 00-a0-cc-66-8e-a6   207.154.90.171  1       136       54    139575      172
 3 00-d0-39-00-2d-c3                  0         0        0         0        0
```

>show ethernet times

```
Ethernet Stations:
 #       MAC address   IP address   Port MC  Time added      Idle
 0 ff-ff-ff-ff-ff-ff                   0   29-Nov 16:17:08
 1 00-d0-39-00-2d-cb                  -2   29-Nov 16:17:08   0.01
 2 00-a0-cc-66-8e-a6   207.154.90.171  2   29-Nov 16:17:15   0.00
 3 00-d0-39-00-2d-c3                   0   29-Nov 16:23:41   9.18
```

Both formats list all the ethernet stations attached to either port. The tables list the MAC (Ethernet) address of the station, and, if known, the IP address.

The first entry in the table tracks broadcast traffic while the second entry is always the address of the gateway itself. The "Port" column shows which of the two ports that station was seen, with 1 representing the WAN and 2 the LAN port. The gateway also stores addresses seen as a multicast address and shows those with a port 0.

The "counts" format shows the cumulative number of ethernet packets that have been seen with that MAC addresses in the "source" (from) or the "destination" (to) fields. In bridge mode the gateways are in "promiscuous" mode and look at all the ethernet packets in both networks. The gateways "discard" the packets that are known to be local, but "forward" all other packets to the opposite network. These are accounted separately in the report.

The "times" format indicates whether that entry is for a "multicast" (MC) address, shows the time when the station was added to the table, and how long since that address has been seen. When the "idle" time exceeds the time specified by the "bridge" command, that entry is deleted from the table.

econsole table

The unit sends an e-console discovery packet on the WAN and reports all the replies. These include both gateways and radios that can be reached from the WAN port.

ber-test

Commands the gateway to enter a "Bit Error Test" mode. In this mode the gateway replaces the input bit stream on the serial port with the standard CCITT 511 length pseudo random code. This bitsteam is then packetized and sent over the WAN port. While running the test the

gateway also checks the bit stream arriving through the WAN port and compares it against the same 511 length code. The gateway reports the SYNC errors, SYNC status, number of blocks sent, bit errors and average Bit Error Rate.

This mode only operates at serial speeds below 200 Kbps.

4.7 File Utilities

The NetCrossing Gateway maintains a file system that allows multiple programs to be stored in either non-volatile flash PROM or volatile RAM. New programs can be downloaded into the gateway memory through the auxiliary port, through the Ethernet port, or to a remote gateway across the RF link.

One of the programs in flash PROM is designated as the default program to run after reboot. On power up that program is copied from PROM into RAM and the code runs out of RAM.

Both sections of memory (non-volatile flash PROM and volatile RAM) are segregated into two "directories". The non-volatile flash PROM is called "flash" signifying the flash PROM and the volatile RAM is called "tmp" signifying the temporary status of the program. Use the "directory" command to view the programs loaded and whether they are in non-volatile or volatile memory.

Any program can be invoked with the command "run" without making it the default file. This is useful when upgrading the software over an RF link as a way to ensure that the new code is working correctly before making it the default.

console-speed-bps

baud-rate-bps=9600 or ***19200*** or ***38400*** or ***57600*** or ***115200***

Sets the Auxiliary port of the gateway to the specified baud rate. This setting is not saved in the gateway configuration, the auxiliary port always reverts to 9600 baud on power up.

This command is useful to speed up the download process over the auxiliary port. Before issuing the download command, use this command to change the gateway console speed to the highest baud rate supported by the PC. Then change the terminal settings to match the gateway speed. Issue the download command described below and initiate the transfer at the terminal.

Examples:

>console-speed-bps baud-rate-bps=115200

copy-file

source=filename
destination=filename

Copies the input-file into the output-file. If the memory location is not defined (flash or tmp), the command assumes the flash directory.

Examples:

>copy-file tmp/nxg01_02 nxg01_02

delete-file

filename=filename

Deletes the specified file from RAM or Flash PROM. If the memory location is not defined (flash or tmp), the command assumes the flash directory.

Examples:

>delete nxg01_03

directory

format=short or ***full***

Lists all the files currently stored in flash PROM and RAM, their size, the sectors occupied and the MD5 checksum (full version). It also indicates which of the files is the default program. Files stored in flash PROM have the flash/ prefix. Files stored in RAM have the tmp/ prefix.

Examples:

>dir

download-file

source=path/filename
destination=filename
method=inline or ***binary***

Downloads a program file from a PC to the NetCrossing Gateway.

To download a file through the Ethernet port or across RF links you need to be running the Econsole program on a PC attached to a gateway through the Ethernet port. In this case the program file must be in binary zipped format (with extension **.bz**). The *path/* in the source parameter is the PC directory where the file resides. The program file is transferred to the gateway and is stored in memory under the name specified by the destination parameter. If the destination parameter is omitted, the file will be stored in Flash PROM with the same name as the source. Note that the ".bz" extension is required in the command. The download "method" must be "binary" (which is the default).

Example:

>download C:\load\nxg01_12.bz

download the file nxg01_12.bz from the PC directory C:\load into the unit file flash/nxg01_12

If the download is executed from a terminal connected to the Auxiliary port, the file is in ASCII format and has the extension .dwn. The download method must be "inline". The source parameter is not needed since, after issuing the command, you must initiate the transfer of the file from the terminal.

Example:

>download destination=nxg01_12 method=inline

After issuing the command initiate the file transfer using the terminal facilities.

run-file

filename=filename

Executes the specified file. The file is first copied into RAM and then the program is executed out of RAM. If the gateway is rebooted or power cycled, the gateway reverts back to the program defined as the default program. If the memory location is not defined (flash or tmp), the command assumes the flash directory.

Examples:

>run nxg01_04

set-default-program

filename=filename

Sets the specified file as the default program to be loaded upon reboot or power cycle. Since the default program must reside in flash memory, the "flash/" prefix is assumed and is not required for the command.

Examples:

>sdp nxg01_04

4.8 Event Logging Commands

The *NetCrossing Gateway* keeps track of various significant events in an "event log". This event log holds up to 500 events. The first 100 entries in the log are filled sequentially after power up and are not overwritten. The remaining 400 entries consist of the last 400 events recorded. All events are time-tagged with system time.

Events are classified in different categories from level 0 (catastrophic error) to 7 (debugging).

clear-log

region= all-events or ***reboot-reasons***

This command clears the contents of the system event log from the specified "region". After a code upgrade it is recommended to clear the reboot reasons since the pointer in non-volatile memory pointing to the reason message may no longer be valid.

display-log

region=end or ***tail*** or ***beginning*** or ***all-events*** or ***reboot-reasons***
length=1..500
id=0..200
min-level=0..7
max-level=0..7

This command outputs to the terminal the specified **region** of the event log. The **length** parameter specifies the number of events to output (defaults to 10). The remaining parameters provide filters to leave out specific events. If the **id** parameter is specified, only the event identified by that id will be displayed. The **min-level** and **max-level** settings allow the user to display only the events with the specified category range.

When the region is specified as **tail**, the command displays the last 10 events followed by a blank line, then waits for more events and displays then as they occur. You can press the space bar to exit this mode.

The **reboot-reasons** region of the event log consist of the last four events that that caused the gateway to reboot. These events are stored in non-volatile memory. The time tag in these events is the time the gateway was up since it was rebooted, not the time of day.

Examples:

>display-log region=all

>display-log region=all length=300 min-level=2 max-level=6

max-event

Sets the event severity level that should be saved or displayed. These two parameters are saved as part of the configuration

save=0..7

Only events of the specified level or below will be saved in the event log.

print=0..6

Events of the specified level or below will be output to the console port as they occur.

Examples:

>max-event print=6

4.9 Miscellaneous commands

date

The *NetCrossing Gateway* will set its internal date and time automatically by decoding Network Time Protocol (NTP) packets in the Ethernet LAN. The "zone" parameter specified

with the "date" or "time" command will then be used to display the date/time in local time. The "zone" value is saved as part of the gateway configuration.

If NTP packets are not available, the user can initialize the gateway date and time with either the "date" or "time" commands. The parameters for both commands are identical, but the parameter order is different. The date command can be entered as:

> **date 16-may-2000 10:32:06**

date=day-month-year

Sets the date used by the gateway. The day / month / year parameter may be separated by any valid separator ('-' '/' etc.)

time=hh:mm:ss

Sets the gateway time in hours, minutes and seconds. Use colons to separate the three fields.

zone=zone-code or ***offset***

Sets the time zone to be used by the gateway to translate the NTP time to local time. It can be specified by an offset from GMT (-0800 or +0200 for example), or as a "zone-code". The valid "zone-codes" and the respective offsets are shown below:

Zone	zone code	offset
Pacific Standard Time	PST	-0800
Pacific Daylight Time	PDT	-0700
Mountain Standard Time	MST	-0700
Mountain Daylight Time	MDT	-0600
Central Standard Time	CST	-0600
Central Daylight Time	CDT	-0500
Eastern Standard Time	EST	-0500
Eastern Daylight Time	EDT	-0400
Greenwich Mean Time	GMT	0000

help [command-name]

If no command is specified, displays the complete list of commands. If a command is specified it displays the valid parameter and corresponding values for that specific command.

Examples:

>help monitor-link

history

Displays the previous commands entered.

license

key=< ASCII string>

The "license" command is used to turn ON or OFF a set of optional features or capabilities. The key is a 35-character string combination of ASCII letters, numbers, and hyphens. The key must be input with the syntax as shown in the example below, including hyphens, for the gateway to accept it. The characters can be input as upper or lower case.

After entering the key you must reboot the gateway for the feature, enabled by the key, to take effect.

Each key is unique for a particular gateway serial number and capability, i.e. a key generated to turn ON a capability on one serial number will not work on another gateway.

Example:

>license key=02EL1-ZGZ42-G0000-00C54-81WAJ-C9BEK

logout

Closes the current Econsole session.

reboot

Resets the gateway causing the software to perform a complete start up sequence. This is equivalent to power cycling the gateway off and on.

time

time=hh:mm:ss
date=day-month-year
zone=zone-code or offset

This command is identical to the "date" command explained above except for the order of the parameters. It allows the time and date to be entered as:

> time 10:32:06 16-may-2000

version

Displays the gateway model and software version.

5 NETWORK MANAGEMENT

The NetCrossing Gateways operate as part of a network environment with many devices. Whether operated by an Internet Service Provider (ISP) or the Information Technology (IT) department of a business, there is often a need to supervise and manage the network from a central Network Operations Center (NOC). This chapter describes the features of the *NetCrossing Gateway* that are useful for this purpose.

5.1 Telnet

5.1.1 General

Telnet, which stands for Telecommunications Network, is a protocol that allows an operator to connect to a remote machine giving it commands interactively. Once a telnet session is in progress, the local machine becomes transparent to the user, it simply simulates a terminal as if there was a direct connection to the remote machine. Commands typed by the user are transmitted to the remote machine and the responses from the remote machine are displayed in the telnet simulated terminal.

5.1.2 Starting a Telnet Session

In order to start a telnet session with a gateway you first need to configure the gateway with a unique valid IP address. This is done with the ***ip-configuration*** command described in section 5.6. This initial configuration must be done using either the RS-232 console port or the ECON program.

Once the gateway has an IP address, you must start the telnet application at the local machine and establish a connection with the IP address of the gateway. If the local machine is a PC running Windows, you can start Telnet through Hyperterminal as follows:

1. Start the Hyperterminal application (in a typical Windows installation Hyperterminal can be found from the **Start** button under Programs/Accessories/Communications…)
2. From the **File** menu choose **New Connection**.
3. In the **Name** field enter any name you wish and press the OK button. This will open the "Connect To" window.
4. In the last field, titled "**Connect using**:", select **TCP/IP (Winsock)**. The fields above will change to **Host Address:** and **Port Number:**.
5. In the **Host Address** field, type the IP address of the gateway, then press the OK button.
6. TCP will now attempt to connect to the specified device. If successful the gateway will request a login name with the prompt **login:**
7. Type ***public*** followed by the Enter key

The gateway will now display its prompt command and you may type any commands as described in section 5.

If after entering the ***public*** login name, the terminal displays the message "Login Failed", this may be due to the gateway being configured to be managed from only some specific IP addresses. This is explained in the following section.

5.1.3 Telnet Security

The remote management capability through Telnet opens the possibility for an unauthorized user to login to any gateway accessible through the Internet. The gateway configuration can be password protected with the use of the **lock** and **unlock** commands. If further security is desired you can specify up to four source IP addresses that are authorized to initiate Telnet sessions with the gateway. When configured in this way, the gateway will reject Telnet requests from all IP addresses that are not in the authorized list.

The authorized source IP addresses for Telnet are the same addresses that are authorized to perform SNMP management. They are entered using the ***snmp*** command described in section 5.6 and can be viewed with the ***display-configuration*** command. When this list is empty, you can initiate a Telnet session from any IP address with the login name ***public*.** When this list is not empty, Telnet sessions can only be initiated from the listed hosts. Additionally, for each host, the login name must match the string listed for the ***community*** field.

If you wish to use this security feature you need to know the IP address of the local machine. On a PC running Windows, one way to find its IP address is to open a DOS window and issue the command:

```
>ipconfig
```

5.2 SNMP

5.2.1 Command Line Interface Versus SNMP

Configuration settings on the *NetCrossing Gateway* are displayed and modified using a command line interface, which can be accessed using either the RS-232 console port, the ECONSOLE program, or via a TELNET session.

In a NOC environment, there is a need for an automated monitoring system to collect on an ongoing basis information from devices in the network for three purposes:

1) to build an inventory of all the devices of the network

2) to keep track of all devices on the network and raise alarms when any device becomes unreachable (device failed, link down, etc)

3) to maintain statistics on traffic levels in order to implement usage-based charging, or to determine where congestion exists in the network, so that the network can be expanded to accommodate growth

Command line interfaces are not very suitable for these purposes, and the *NetCrossing Gateway* supports the Simple Network Management Protocol (SNMP) to assist in these tasks. SNMP is a

simple, transaction-based (command/response) protocol, which allows a variety of third-party software products to query network devices and collect data for these purposes.

For a generic introduction to the SNMP protocol, we recommend the book "The Simple Book - An Introduction to Internet Management" by Marshall T Rose (P T R Prentice-Hall, 1994).

5.2.2 What is SNMP?

The SNMP protocol is described in the following documents:

- RFC1157 - Simple Network Management Protocol (SNMP) - ftp://ftp.isi.edu/in-notes/rfc1157.txt
- RFC1155 - Structure and identification of management information for TCP/IP-based internets - ftp://ftp.isi.edu/in-notes/rfc1155.txt
- RFC1213 - Management Information Base for Network Management of TCP/IP-based internets: MIB-II - ftp://ftp.isi.edu/in-notes/rfc1213.txt

SNMP is a specification for the interaction (***protocol***) between the ***SNMP agent*** embedded in a network device, and the ***SNMP manager*** software running on another machine in the network.

The data provided by the SNMP agent in a network device is described by a document called the MIB (Management Information Base). **MIB-II** describes the basic information provided by all devices, and additional documents describe optional extensions for components that may not exist in most devices.

Devices may also provide non-standard MIB groups. In order for a network management system to make use of these extended features, the MIB description must be obtained from the device manufacturer and loaded into the management station.

SNMP data travels in IP packets, using the UDP port 161 for the agent, so in order to use SNMP, the device must have an IP address.

5.2.3 Security Considerations in SNMP

SNMP was designed before the Internet grew commercial, and the original design was not secure. Later versions intended to provide security, but grew cumbersome and complex. As a result, most devices provide secure operation in a non-standard way.

The original SNMP design as embedded in the protocol, assigns network devices to named communities. Any transactions exchanged between the agent and the manager include the name of the community to which they both belong. The agent has a list of which access rights (set, get, trap) it will grant for each community of which it is a member.

In the *NetCrossing Gateway* this has been re-interpreted: The gateway has a list of up to 4 management stations from which it will accept requests, and for each one - identified by its IP address - it is indicated what access rights it is granted, and which community string it must use. Requests from all other sources are ignored. Refer to the ***snmp*** command in section 5.6 for details on how to configure the gateway for management using SNMP..

If no management stations are listed, ***get***-requests with the community ***public*** will be accepted and responded to from any IP address.

5.2.4 Examples of Network Management Systems

Some of the most common network management systems are listed below. All of them provide many similar features, including network status displays showing key devices on a map, where the devices change color if they have alarms, and with provisions for activating a remote paging device if there is a problem.

WhatsUp Professional (Ipswitch Inc)

http://www.ipswitch.com/

USD 1,900 to USD 3,500

SNMPc (Castle Rock Computing, Inc)

http://www.castlerock.com/

USD 1,300 to USD 8,000

InterMapper (DartWare)

http://www.intermapper.com/

USD 500 to USD 8,000

OpenView (Hewlett-Packard)

http://www.openview.hp.com/

USD 3,000 to USD 10,000

The OpenView product line has been revamped; HP is now positioning it not as a turnkey software product, but as a custom adapted application to be bought through a value-added implementation partner.

Nagios (Free Open Source)

http://www.nagios.org/

Free download.

Multi-Router Traffic Graphing

http://www.mrtg.org/

This is a free, open-source software, capacity planning tool.

5.2.5 NetCrossing Gateway Management Information Base (MIB)

The gateway implements only the core MIB-II (i.e. no **enterprises** branch).

A management station will see four interfaces in the ***interfaces group***:

1 - Bridge

2 - LAN

3 – WAN

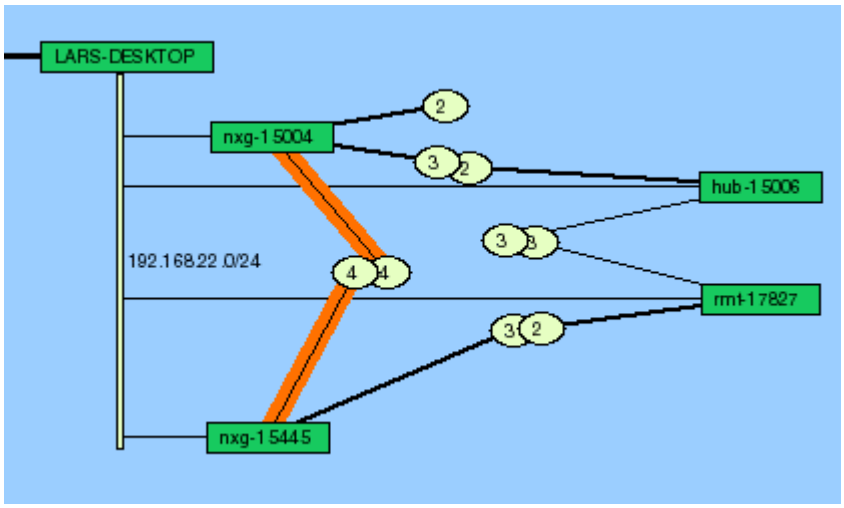
4 – NetCrossing Link

Some network management systems (such as SNMPc) always show all the interfaces; on some other systems (such as InterMapper) you will need to set an option for the device to "show unnumbered interfaces".

The first of the interfaces (Bridge – ***ifIndex==1***) represents the attachment of the SNMP agent to the bridged network. Only IP traffic seen by the embedded host is counted.

The LAN device (***ifIndex=2***) represents the traffic passing through the gateway's LAN port.

The NetCrossing Link (device (***ifIndex==4***) represents the link between two NetCrossing gateways. It will be marked as UP (***ifOperStatus==1***) when the connection between gateways is working. The ***ifSpeed*** field is set to the speed of the serial port, so when the link is up, it is always filled to 100% capacity.



This extract from an InterMapper network map shows two NetCrossing Gateways connected over a radio link. Since both the gateways and the radios are bridging devices, they are all logically connected to the LAN segment 192.168.22.1 shown by the vertical bar on the left.

The physical connectivity actually extends from the PC at the top which acts as a router to connect the 192.168.22.* LAN to the rest of the network, into port 2 of the gateway on top; after bridging through the gateway, it continues from the gateway's WAN port to the radio's LAN port, and then over the RF link between the two radios (port 3 on each radio) to the other gateway.

The logical link between the serial ports of the two gateways is port 4 on each gateway. InterMapper outlines it in orange to show that the link is saturated.

APPENDIX A – Command Summary

This appendix lists all commands organized in the respective functional groups. Parameters that are part of the gateway configuration are identified by having an entry under the "Factory Configuration" heading. When entering a command, if a parameter that is part of the gateway configuration is omitted, the value for that parameter is not modified.

For commands that are not part of the gateway configuration, if a parameter is omitted, the value for that parameter defaults to the value indicated in bold.

Configuration Management Commands

Command	Parameters	Values
change-password	enable-configuration	<string>
display-configuration	source	**current** main alternate basic factory
load-configuration	source	**main** alternate basic factory
lock		
save-configuration	destination	**main** alternate
unlock	enable-configuration	<string>

Major Configuration Parameters

Command	Parameters	Values	Factory Configuration
bridge	station-timeout-sec	5..1800	30
	multi-cast-timeout-sec	5..3600	30
local-area-network	speed	10hdx, 10fdx	10fdx
node	name	(31 character string)	nxg-nnnnn
	location	(31 character string)	
	contact	(31 character string)	
	id	two-char alphanumeric	00
radio	power	on or off	(off)
serial	clock-source	internal, external,remote	remote
	speed-bps	3500...2048000 (NX2048) 3500...8192000 (NX8192)	128000
	channel-64kbs	1..32	
	interface	rs530a, rs530,x.21, v.35, rs449, rs232	rs530
	loopback	disable, input, output	disable
	idle-code	0..255	0x7E
	clear-to-send	on, off, remote-rts, link-state	on
	data-set-ready	on, off, remote-dtr	on
	carrier-detect	on, off, remote-rts, link-state	on
	packets-per-second	10,25,50,100,200,400	400
time-division-duplex	sync-mode	off, master, auto	off
	cycle-period-ms	20, 40	20
udp	peer-ip-address	<ip-address>	0.0.0.0
	port	1..65535	6389
	type-of-service	0..255	192 (0xC0)
wide-area-network	network-type	bridge, route	bridge
	peer-serial-number	0..999999	0
	id	two-char alphanumeric	00
	jitter-ms	0..500	0 (auto)
	capacity-kbps	100..20000	20000
	speed	auto-10, 10hdx, 10fdx, 100hdx, 100fdx, auto	auto

Internet Protocol (IP) Management Commands

Command	Parameters	Values
ip-configuration	address	ip address
	netmask	ip address
	gateway	ip address
ping	destination	ip address
	count	0..500 (def **4**)
	size-bytes	32..1400
snmp	manager	ip address
	community	ASCII string (9 max)
	access	g, gs, gt, gst
	authentication-traps	0, 1
	delete	1..4

Installation and Monitoring Commands

Command	Parameters	Values
show	table	**status** gateways ethernet econsole ip-stack
	format	**count** times
	clear	1 or 0
ber-test		

File Utilities

Command	Parameters	Values
console-speed-bps	baud-rate-bps	9600, 19200, 38400 57600, 115200
copy-file	source	filename
	destination	filename
delete-file	filename	filename
directory	format	**short** full
download-file	source	path/filename
	destination	filename
	method	binary inline
run-file	filename	filename
set-default-program	filename	filename

Event Logging Commands

Command	Parameters	Values	Factory Configuration
clear-log	region	**all-events** reboot-reasons	
display-log	region	**end** tail beginning all-events reboot-reasons	
	length	1..500 (def **10**)	
	id	0…200	
	min-level	0…7 (def: **0**)	
	max-level	0…7 (def: **7**)	
max-event	save	0..7	5
	print	0..7	3

Miscellaneous Commands

Command	Parameters	Values	Factory Configuration
date	date	dd-mmm-yyyy	
	time	hh:mm:ss	
	zone	offset or code	GMT
help	command		
history			
license	key	<35 character string>	
logout			
reboot			
time	time	hh:mm:ss	
	date	dd-mmm-yyyy	
	zone	offset or code	GMT
version			

APPENDIX B - Specifications

WAN Port	
Connector	RJ45 (x 2). One connector with power to the Radio
Speed	100 BaseT, auto negotiate
LAN Port	
Connector	RJ45
Speed	10 BaseT, half or full duplex
Serial Port	
Connector	DB25 female (DCE)
Synchronous Speeds	2.8 Kbps to 2048 Kbps (NX2048) 2.8 Kbps to 8192 Kbps (NX8192)
Interface	RS232, RS530, RS530A, RS449, V.35, X.21
Clock Source	Internal, External, Remote
Console Port	
Connector	DB9 female (DCE)
Speeds	Programmable, 9600 to 115.2 Kbps
Synchronization Port	
Connectors	2 RCA audio
Power Requirements	
Input Voltage	+8 to +28 Volts DC
Input Voltage (AC)	110 VAC or 220 VAC (external supply)
Power Consumption	3.3 Watt
Environment	
Temperature	0 to 55 deg C 32 to 130 deg F
Max. Humidity	90% non-condensing
Mechanical:	
Dimensions	9" (W) x 6" (D) x 1.5" (H) 22.8 x 15.2 x 3.8 cm
Weight	1.0 lb (0.45 Kg).

APPENDIX C – Ethernet Console Program

Short description

The ethernet console program was developed in order to accommodate the remote configuration of a gateway, i.e. the configuration in cases where the physical access to the gateway is not feasible, or it is cumbersome. The software consists of two parts: the client and the server. The client runs on the administrator's PC, while the server runs on the gateway.

The communication is done via a TCP-like protocol. There is an acknowledgment for every packet that is sent, as well as a retransmission mechanism when a packet gets lost.

Each gateway allows multiple sessions, i.e. more than one client can be connected concurrently to the same server (gateway). Nevertheless, for performance reasons, it is not recommended to have more concurrent sessions than they are really needed, and definitely not more than the maximum number which currently is 4.

System requirements

- Win95, Win98, Windows ME, WinNT, Win2000, WinXP
- NetBIOS installed
- WinPCap installed

Note: With regard to Windows NT platform, the code has been tested with versions 4.0, or newer. There is also a Linux beta version

Installation for Windows

In order to install the WinPCap library, if not already installed, just click on the WinPCap.exe. Support and updates for this library can be found at http://netgroup-serv.polito.it/winpcap/. It is strongly suggested to uninstall older versions of the library and reboot the machine before installing the new one. NetBIOS is a software component that comes by default with all Windows system, so you don't have to install it. To start the Econsole, simply open a MS-DOS window and type *econ***.** For available command line arguments, please read the "*input arguments*" section.

Included files

- *win_readme.doc* The file that you are reading
- *econ.exe* The EConsole client
- *WinPCap* The Windows installer for the WinPCap library
- *input_script.txt* A sample input script file, that contains a list of gateway commands.

Input arguments

You can provide the following arguments in the command line, even though none of them is required.

Input file

There are two sources for the input commands: the keyboard, or a text file. The second option is useful when you are running the same set of commands periodically, so you want to avoid retyping them every time you want to execute them. If there is an input file in the command line, then the keyboard will be deactivated and only the function keys will be available. If the specified file cannot be found, the application will be terminated.

example:

C: > econ -i input.txt

Sample input file:
help
 # this is a comment - note that the character # must appear as the fist character
time
date
 # the following is a local command specifying a delay in seconds
. delay 10
time
. delay 1.5
version
logout

As you probably noticed from the above file, all the lines are interpreted as gateway command, unless:

a) They start with the character '#' which implies a comment

b) They start with the character '.' which implies a local command. Currently there is only one local command, namely the *delay < time in secs>*

Important note: All the input scripts should end with the *logout* command. Since all the commands are terminated with the new line character, there must be one command per line and after the final *logout* command you must have an extra empty line.

Output file

When you want to capture the output of a session into a text file, you can pass the filename as an argument. If the file does not exist it will be created, otherwise it will be overwritten.

example:

>econ -o output.txt

Gateway MAC address

If you are interested in a specific gateway, you can pass its MAC address and let the client ignore any response from other gateways. That's very handy when you are always getting connected to the same gateway and you want to avoid the manual selection of a preferred one. Very useful also in case you are using scripts for fully automated procedures.

example:

>econ -r 00:78:24:22:BA:4F

Gateway Serial Number

The same functionality as above (see Gateway MAC address) can be achieved by providing the gateway serial number, instead of the gateway physical address. Note that you should not include the initial UC characters of the serial number (i.e. type *11078* instead of *UC11078*)

example:

>econ -r 11787

Local Physical Address

Even though econsole identifies the PC local physical address automatically, there are some cases in which the user wants to specify the local address on his/her own. These cases usually arise when there are multiple NIC cards with the same names under WinNT operating system. In such case, the econ might pick up the wrong MAC address, and therefore the user should supply manually the physical address as a command line argument.

example:

>econ -m 00:78:24:22:BA:4F

Inverse Screen Colors

You can change the default settings (white texture on black background) by providing the -b option, which will change the settings to black characters on white background.

example:

>**econ -b**

Change the console window size

Currently you can specify two values, either 25 or 50. These values indicate the number of lines of the MS-DOS window.

example:

>**econ -l 50**

Help

Function keys, including F1, are activated after you get connected to a gateway. If you want to get help from the command line, you can use the -h argument.

example:

>**econ -h**

Syntax:

econ <*argument list*>

argument list = *argument list* | *argument* | *{}*

argument = *-o outputfile* | *-i inputfile* | *-r MAC address*

Examples

Let's say you want to read a list of commands from the text file called in.txt, and capture the output to a text file called out.txt. You are also interested only in a specific gateway with MAC address equal to 00:78:24:22:BA:4F. In that case, you will start the EConsole with the following arguments (the arguments order is irrelevent):

>econ -i in.txt -o out.txt -r 00:78:24:22:BA:4F *or*

If you are reading from the keyboard, and you are simply interested in capturing the output of the session, use the following syntax:

>econ -o out.txt

Since no input file was specified, it is assumed that the keyboard will be used for input, and ALL gateways will participate in the discovery process.

Function Keys

Currently there are 6 different function keys.

F1 - Online help - gives a short description of the other function keys and the input arguments

F2 - Active/deactivate diagnostic messages. Initially diagnostic messages are not shown, therefore if you want to see them you should press F2. Diagnostic messages include warnings, and retransmission info in order to get an idea of the connection's speed/integrity. Error messages are always shown.

F3 - Terminates the current session and closes the application.

F4 - Close the session with the current gateway and display the results of the initial discovery phase to allow the user to connect to a new gateway.

F5 - Reverse/Restore screen settings. Initially the screen displays white letters on black background, but you can reverse it to black letters on a white background.

F6 - Increases the console window buffer. This introduces a side bar which enables the user to scroll up and down. Available in Windows NT Only.

Troubleshooting & Updates

Common problems

1. Failed to open adapter

   This usually happens when you haven't installed properly the WinPCap library, or you have and older version of it. Please visit http://netgroup-serv.polito.it/winpcap/ to get the latest version. You should also make sure that your Ethernet adapters are working properly.

2. Cannot find gateway(s) even though they are running properly

   Make sure that:

   - The ethernet cables are OK
   - You are getting connected to the right network segment (i.e. try all ethernet adapters)
   - You are using the right MAC address. The system tries to identify the adapter physical address through some NetBIOS calls in the Win9X case, or some NDIS queries in the WinNT/Win2000 case. If NetBIOS is not installed, the econ will probably use the wrong local host MAC address. Also if there are more than one Ethernet adapter installed with the same name, this might cause problem in the WinNT case.

     Resolution: Use the command line argument to specify the correct physical local address. You can see all the local physical address by executing the *ipconfig -all* command. Example:

     >econ -m 00:78:24:22:BA:4F

3. Find a gateway but not getting connected

   Check if the maximum number of sessions has been reached. The maximum number of sessions on the server side is limited to four, therefore you should NOT connect to the same gateway multiple times if not absolutely necessary. When the number of sessions reaches the limit the gateway will ignore any new discovery messages.

   Another reason might be a unreliable RF link causing a high packet loss. Since during the discovery phase there isn't any retransmission mechanism, it is quite possible that you managed to "see" the gateway, but you weren't able to connect to it, because the connection request packet was lost. In such case, try to connect again.

4. High drop rate - screen freezes momentarily - connection times out

There are two possible causes.

1. The link between the client (PC) and the server (gateway) is very weak. If the packet drop rate is more than 20%, then the connection is problematic.

2. There are multiple sessions opened on the same server. With many concurrent sessions the server response may be noticeably slower. Always close the session gracefully by executing the *logout* gateway command, and not by closing the MS-DOS console. If the *logout* command is not issued the session at the server will remain open for an additional 15 minutes. Use the *list long* command to find out the number of open sessions.

5. If I leave the client inactive for half an hour, and try to type a new command, I get an unable to transfer packet message or I get a "session timeout - application will be closed" message.

An open session times out after 15 minutes of inactivity on the server side, and 30 minutes on the client side.

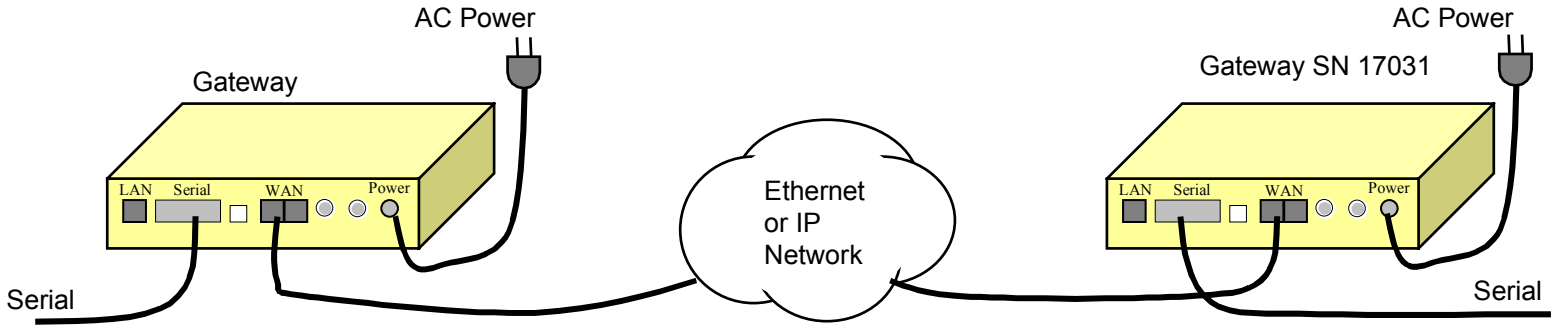
Report a bug & Updates

Please visit htttp://www.afar-inc.com/ for more info.

Acknowledgments

The WinPCap library was obtained from “Politecnico di Torino” and the code is distributed in binary form as part of the Econsole. The following copyright notice applies to that library.

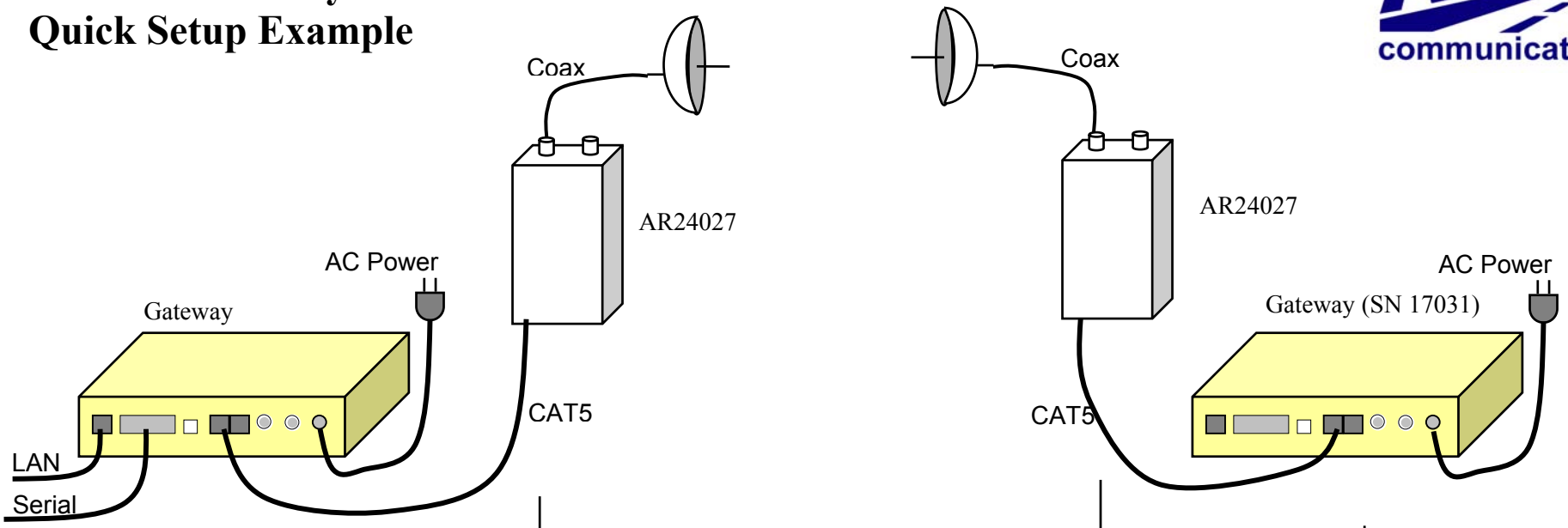
NetCrossing Gateway
Quick Setup Example



>load factory >serial clock=internal >wan peer=17031 (note 1) >save	**Minimal Configuration**	>load factory > save
>serial speed=512000 >serial interface=v.35	**Changing common serial parameters**	>serial interface=v.35
> ber-test	**Checking Link Operation**	> serial loopback=output

Note 1: Enter the actual serial number of the peer gateway

Wireless Serial Synchronous Link
Quick Setup Example



>load factory >radio on >serial clock=internal >wan jitter=30 >wan peer=17031 (note 1) >save	>load factory >node hub >node max-rem=1 >save	**Minimal Configuration**	>load factory >save	>load factory >radio on >wan jitter=30 >save
	>rftr ch=30 >rfrec ch=14	**Changing RF Channels**	>rfrec ch=30 >rftr ch=14	
>serial interface=v.35 >serial speed=512000 >serial clock=external		**Changing Serial Port Settings**		>serial interface=v.35

Note 1: Enter the actual serial number of the peer gateway