Configuration Reference Manual

EDACS®
Data Gateway

ERICSSON

TABLE OF CONTENTS

INTRODUCTION

This manual is a guide to configuring the Ericsson EDACS® Data Gateway (EDG™) for proper operation. In particular, it describes the command set and syntax for the *SYSTEM.TXT* file, which is a special file located in the /cnfg directory of the EDG hard drive. Examples of *SYSTEM.TXT* files for various system configurations are included in the *EDACS Data Gateway Installation and Maintenance Manual* (LBI-38962B).

The *SYSTEM.TXT* commands have been designed to allow a quick and easy setup for users with a simple system configuration, yet provide the necessary power to users who are designing very complex systems. A general knowledge of IP is assumed on the part of the reader.

There is a DOS based utility to test the syntax of the *SYSTEM.TXT* file called *SYSCHECK.EXE*. This utility should be included in the configuration diskette. Both *SYSCHECK.EXE* and the EDG output a file giving current configuration information, *system.rpt*. For the EDG, *system.rpt* is located in the /cnfg directory.

Note there is no revision A of this document.

USING THE EDG CONFIGURATION FILE

The EDG configures itself upon startup based on the contents of the *SYSTEM.TXT* configuration file. Without this file, the EDG will not boot. It is in ASCII format and can be created and modified with any ASCII based text editor.

NOTE

The EDG diskette drive only supports double sided / high density (1.44 MByte) 3.5 inch diskettes. Do not use double density (720 KByte) diskettes.

GENERAL RULES

The following are some general rules about creating and editing the *SYSTEM.TXT* file.

- *SYSTEM.TXT* can be automatically loaded from a diskette when the EDG is rebooted.
- The *SYSTEM.TXT* command parser is not case sensitive. Lower case characters are automatically read as upper case characters. However, file names are an exception to this rule. The EDG file system is UNIX based and, therefore, all file names are case sensitive. Files names are discussed further below.
- Unless specifically stated otherwise, any number of spaces or tab characters may be freely placed within the file as field separators. Spaces and tabs are equivalent to the parser, and will be referred to as a space character in the remainder of this manual.

- If a space character is required by a command's syntax, this will be explicitly stated in the command description.

- The EDG only recognizes tabs, line feeds, carriage returns, DOS end of file (CTRL-Z), and printable characters in *SYSTEM.TXT*. Any other characters cause a syntax error.

- The fixed disk and diskette drives are referred to as "01.02" and "01.01", respectively, when using a full path name.

- All file and directory names on the hard disk follow a UNIX format, are case sensitive, must begin with an alphabetic character, and may contain up to 12 characters.

- All file and directory names on the diskette drive follow a DOS format. Names are in upper case, and may not exceed eight characters, plus a three character extension, separated by a period.

- When using a file's full path name, separate the drive name, directories, and file name with the forward slash character ("/"). An example is "01.02/subdir1/subdir2/FILE.EXT".

The *SYSTEM.TXT* file contains the statements that configure all the system parameters that allow the EDG to function properly. There are basically three types of statements that are entered in this file: commands, table entries, and headings. Commands are used to configure a particular system parameter, table entries contain data that is used for various EDG tables, and headings identify groups of related commands or table entries.

♦ A heading statement can have one of two possible forms:
[heading]
[heading *field_value***]**

The **heading** value is a predefined text string containing no space characters. The brackets, '[' and ']', which identify that this is a heading statement, must be entered. Space characters may be entered before and after the opening and closing brackets, if desired.

For those heading statements that require it, *field_value* must be entered, which is a variable value within the range and in the format specified in the heading statement description. The **heading** and *field_value* must be separated by one or more space characters.

Heading statements are followed by zero or more commands or table entries. Heading statements that are not applicable for a particular configuration may be omitted from the file. Groups of commands and table entries can contain blank lines.

All heading statements, with their associated command blocks, may be placed in any order within the file.

- Each command statement has the following syntax:
  keyword *field_values*

  The **keyword** value is a predefined text string containing no space characters. The *field_values* are one or more values entered by the user, in the format and within any range specified in the command description. One or more space characters must separate **keyword** and the first field value. Multiple field values are also separated by one or more space characters.

  Each command statement must be entered on a separate line in the file and may only appear under its associated heading statement. Any command that is not related to the command block's heading will be ignored and will generate a warning. Commands may be placed in any order within the command block.

- Each table entry is composed of one or more fields, where a field is a value entered by the user in the format and within any range specified in the table description. Table entries may be placed in any order within the table, with one entry per line.

- The comment character, '#', may be inserted freely within the file. Any text to the right of the comment character is ignored by the parser. If a comment is to be on the same line as a heading or command statement, one or more space characters must precede the '#' character.

- All required numeric values may be entered in decimal or hexadecimal format. A hexadecimal format is identified by appending the character 'h' or 'H' to the value. The hex characters 'a' through 'f' may be entered in upper or lower case. For example, all of the following represent the same value:
  1ab9h, 1ab9H, 1Ab9H, 1AB9h

- Some fields require an IP address value. This address must consist of four separate numeric values, with each value separated by a period, and no space characters. An address may be entered in decimal or hexadecimal format, with the hexadecimal format as described previously. If using decimal format, the range for each of the four values is 0 - 255, while the range for each of the values using a hexadecimal format is 00 - FF. Note that leading zeros may be used for each of the four values, or may be omitted. For example, all of the following represent the same address:
  143.001.04.013, 143.1.004.13, 8F.01.4.dh, 8f.01.04.0dH

- Some fields allow a numeric range. A range consists of either a single numeric value, or two numeric values separated by a '-' character. Space characters are permitted before and after the '-'. The first value must be numerically less than the second value.

- Some fields allow a numeric list. A list is composed of one or more numeric values, each separated by a comma. Space characters are permitted before and after each comma. The values may be entered in the list in any order.

This manual uses the following conventions for configuration file command descriptions.

Convention	Usage
italics	A field in italics indicates a variable that you must replace with a value.
BOLD	A word in boldface letters indicates a key word that must be used. You may enter the keyword in upper or lower case letters.
OPTION	Items shown in non-boldface uppercase characters are a specific option that can be selected. You can use them in upper or lower case in the file.
A|B|C	When options appear with a vertical bar separator, you must choose between one of the options.

On the following page is a summary of the available SYSTEM.TXT commands and headings. Each heading is followed by either its command block, containing all available commands for that heading, or its table entry format. Detailed descriptions and examples for each command and heading follow the summary page.

COMMAND SET SUMMARY

[Board *board_number***]**
HDI_Port_Hosts *port_number host_list*
HDI_Port_Parity ODD | EVEN | NONE
Label "*board_label***"**
Load *loadfile*
Port_Direction *port_number* IN | OUT | BI | OFF
TSI_Baud 19200 | 9600
TSI_EDACS_ID *unit_id*
TSI_Hosts *host_list*
Type CAP | TSI_Master | TSI_Slave *master_board_number* | HDI

[EDACS_Network]
EDACS_IP_Network_ID *network_address*
Label *label*

[IP]
CAP_Default_Gateway *ip_address*
CAP_Ext_Address *ip_address*
Host_Name *host_name*
Int_Board_Address *board_number ip_address*
Int_Network_ID *ip_network_id*
Routing_Address *ip_address*

[System]
EDACS_Err_Retries *max_count delay*
EDACS_Busy_Retries *max_count delay*
Max_FTP_Sessions *session_count*
Max_Msgs *msg_count*
Max_TELNET_Sessions *session_count*
Msg_Timeout *timeout_value*
Outbound_Msg_Delay *delay_value*
Transport_Layer_Protocol *protocol_type_code*

[Device_Config_Table]
id_type id_range start_ip_address BREN_configuration network_layer
.
.
id_type id_range start_ip_address BREN_configuration network_layer

COMMAND SET SUMMARY *CONTINUED*

[Internodal_EDG_Route_Table]
edg_route_addr *next_gateway*
.
.
edg_route_addr *next_gateway*

[Internodal_Group_Repeat_Table]
id_type id_range
.
.
id_type id_range

[IP_CAP_Ext_Routing_Table]
HOST | NET *destination next_gateway*
.
.
HOST | NET *destination next_gateway*

COMMAND DESCRIPTIONS

[Board] Command Heading

Format: [Board *board_number*]

The commands under this heading define a board and its associated parameters within the EDG. Each board is assigned a unique identification number which is entered in the *board_number* field. Board numbers must be contiguous, starting at the number "1", which must be always be assigned to the CAP board.

[Board] HDI_PORT_HOSTS Command

Syntax: **HDI_Port_Hosts** *port_number host_list*

Requirement: Valid for HDI type boards only. This command is required for each active port (i.e. a port for which the Port_Direction command was stated with a direction of IN, OUT, or BI).

Default Value: None.

Description: This command is used to assign EDACS host IDs to a given HDI port.

port_number must be a value in the range 1 - 4, corresponding to port 1 through port 4 on the board.

host_list contains one or more EDACS host IDs, with each ID separated by a comma. A host ID is any value in the range 1 - 63. Data messages destined for a given RDI host will only be routed through those HDI output ports with the destination host ID in its host list. Any messages from an RDI host received through an HDI port will have its source host ID defined as the lowest valued host ID assigned to that port.

[Board] HDI_PORT_HOSTS Command: *Continued*

Comments: A host ID may be assigned to more than one port on the same HDI or multiple HDIs. If a host ID is assigned to two or more HDIs, output messages will be rotored to each of the HDIs. At the individual HDI level, output messages are also rotored through all output ports to which the destination host ID is assigned.

Every host ID assigned to an HDI must be assigned to a master TSI. This may be done explicitly using the **TSI_Hosts** command (which allows you to designate to which TSI each host ID is to be assigned), or it will be done by default by the EDG during startup.

Every host ID assigned to an HDI port must be mapped to an IP address under the **[Device_Config_Table]** heading. This may be done explicitly, or it will be done by default by the EDG during startup. In the default case, the IP address assigned to the host ID will be 200.0.0.XX, where XX is the host ID. Note this default will be changed to 192.168.200.XX per RFC 1597 with Release 4.

Each HDI port will automatically be defaulted a port direction of BI, if not explicitly defined with the **Port_Direction** command.

[Board] HDI_PORT_HOSTS Command: *Continued*

Example: There are two HDIs in the system, with four ports active on the first HDI, and only port 2 active on the second HDI. Port 1 on HDI #1 will process messages for host IDs 1 - 5, ports 2 & 3 on HDI #1 will process messages for host ID 10, and port 4 on HDI #1 and port 2 on HDI #2 will process messages for host ID 2. The following commands would be entered for HDI #1:

HDI_Port_Hosts 1 1, 2, 3, 4, 5
HDI_Port_Hosts 2 10
HDI_Port_Hosts 3 10
HDI_Port_Hosts 4 2

The following command would be entered for HDI #2:

HDI_Port_Hosts 2 2

If no IP addresses were assigned explicitly to the host IDs in the **[Device_Config_Table]**, the following addresses would be assigned by default during startup:

Host ID 1: 200.0.0.1
Host ID 2: 200.0.0.2
Host ID 3: 200.0.0.3
Host ID 4: 200.0.0.4
Host ID 5: 200.0.0.5
Host ID 10: 200.0.0.10

If no port directions were assigned explicitly to the port 1 - 4 on HDI #1 and port 2 on HDI #2, then they would be defaulted to BI.

[Board] HDI_PORT_PARITY Command

Syntax: **HDI_Port_Parity** ODD | EVEN | NONE

Requirement: Valid for HDI type boards only. This is an optional command.

Default Value: Defaults to EVEN parity.

Description: This command is used to indicate the type of parity to be used on all ports of a single HDI for serial communications.

The type of parity must be one of the three values specified above, either ODD, EVEN, or NONE.

This command applies to all ports of the HDI board for which it is specified. However, each HDI board in the system may be defined with its own parity type.

Example: There are two HDIs in the system. The first HDI is to use even parity on all its ports. The second HDI is to use no parity on all its ports. Since even parity is the default, the HDI_Port_Parity command does not have to be specified for the first HDI. However, the following command would need to be entered for the second HDI:

HDI_Port_Parity NONE

[Board] LABEL Command

Syntax: **Label** "*label*"

Requirement: Optional for all boards

Default: A text string consisting of the board number and type will be assigned, e.g. "BD 2 TSI Mas", "BD 1 CAP".

Description: This allows a user defined text string of up to 12 case sensitive characters to be assigned to the board. It is only referenced in messages to the terminal and in any EDG generated reports.

label may contain any printable ASCII character, including space characters, and must be enclosed by double quotation marks. Labels need not be unique. As many characters as will fit on a single line may be entered as the *label*, but only the first twelve will be read and saved.

Example: Board number 2 is a TSI Master type. No Label command is entered for it. Its default label will be "BD 2 TSI Mas".

Example: It is desired to assign the label "Ralph the TSI wonder dog" to the current board. The following command would then be entered:

Label "Ralph the TSI wonder dog"

However, since only the first 12 characters are saved, the actual label assigned to this board would be "Ralph the TS".

[Board] LOAD Command

Syntax: **Load** *filename*

Requirement: Optional.

Default: None

Description: This command tells the system loader which application file to load onto the board.

filename should be the full pathname of the file to be loaded, and should not be enclosed in quotation marks. The full pathname should include the device volume number, the directory path, and the file name. Note that all names are case sensitive.

Example: The hard drive volume number is 01.02. The diskette drive volume number is 01.01.

The EDG loader automatically copies any application files found on a diskette to the /loads directory on the EDG hard drive, with the filename in all capital letters. Therefore, to load the CAP application code in this case, the following command would be entered:

Load 01.02/loads/CAP.SX

[Board] PORT_DIRECTION Command

Syntax: **Port_Direction** *port_number* IN | OUT | BI | OFF

Requirement: Valid for TSI_MASTER, TSI_SLAVE, and HDI type boards only. At least one port on each board must be assigned a non-OFF direction, i.e. a direction of IN, OUT, or BI.

Default: For a TSI_MASTER or TSI_SLAVE, a BI direction is assigned to any port for which this command is not explicitly stated. For an HDI, a BI direction is assigned to any port for which this command is not explicitly stated and an **HDI_Port_Hosts** command is given; an OFF direction is assigned to any port for which this command is not explicitly stated and an **HDI_Port_Hosts** command is not given.

Description: This command defines the direction of message flows for a given port on the board.

port_number must be a value in the range 1 - 4, corresponding to port 1 through port 4 on the board.

The direction must be one of the four possible text strings as shown above:

- **IN** - Only input data to the EDG is accepted through this port; no output data will be routed through it even if no other ports are available.
- **OUT** - Output data may be sent through this port. However, it does not prevent the EDG from accepting data received through the port.
- **BI** - Input and output messages may be routed freely through the port.
- **OFF** - A port that is not being used should be set to this value, although any port that is not explicitly defined a direction will be set to an OFF status.

Example: Ports 1 and 2 of the current HDI board are to be active, with port 1 to only receive messages, while port 2 may transmit and receive messages. The following commands would be entered:

Type HDI
Port_Direction 1 IN
Port_Direction 2 BI
Port_Direction 3 OFF
Port_Direction 4 OFF
HDI_Port_Hosts 1 20
HDI_Port_Hosts 2 20

Note that the 2nd command is not necessary, since an HDI port that is not assigned a direction and is assigned hosts via an **HDI_Port_Hosts** will be set to BI.

[Board] PORT_DIRECTION Command: *Continued*

Example: Ports 2 and 3 of the current TSI board are to be active, with both ports to be transmitting and receiving messages. Ports 1 and 4 are not to receive or transmit messages. The following commands would be entered:

Type TSI_Master
Port_Direction 1 OFF
Port_Direction 4 OFF

Note that the ports 2 and 3 are automatically assigned a BI direction.

[Board] TSI_BAUD Command

Syntax: **TSI_Baud** 9600 | 19200

Requirement: Valid for TSI Master type boards only. It is an optional command.

Default: The default is a 19200 baud rate.

Description: This command sets the baud rate at which the master TSI communicates with the **D**ata **I**nterface **M**odule (**DIM**) located in the EDACS IMC/CEC.

Example: There are 2 master TSIs in the system. The first one communicates with the IMC/CEC at 19200 baud. The second one communicates at 9600 baud. Since the default baud rate is 19200, no command is required for the first TSI to set its baud rate. However, the following command would be entered under the **[Board]** heading for TSI #2:

TSI_Baud 9600

[Board] TSI_EDACS_ID Command

Syntax: **TSI_EDACS_ID** *unit_id*

Requirement: Required for all TSI Master type boards only. Not valid for other board types.

Default: None.

Description: Assigns an EDACS Logical Unit ID to a TSI Master, which is required by the IMC. The unit ID assigned to a TSI Master must be unique among all other unit IDs assigned to all other entities within the EDACS system, including other TSI Masters, radios, consoles, etc.

unit_id is a value in the range 64 - 16382. Normally, allowable EDACS unit IDs are in the range 1 - 16382. However, the EDG requires that a TSI may not be assigned one of the possible host IDs in the range 1 - 63.

Comments: The EDACS System Manager does not currently support an EDG TSI as a valid type when entering a unit ID in the database. However, to avoid potential EDACS system problems, any unit ID assigned to a TSI Master should be entered in the System Manager database to avoid inadvertent assignment to another entity. This could be done by adding it as a radio type, but assigning a label such as "EDG TSI" for reference.

Example: There are 2 master TSIs in the system. TSI #1 is assigned unit ID 600, while TSI #2 is assigned unit ID 601.

The following command would be entered under the **[Board]** heading for TSI #1:

TSI_EDACS_ID 600

The following command would be entered under the **[Board]** heading for TSI #2:

TSI_EDACS_ID 601

[Board] TSI_HOSTS Command

Syntax: **TSI_Hosts** *host_list*

Requirement: Valid for TSI Master type boards only.

Default: When network layer RDTs or radios have been defined (i.e. there are one or more RDT or radio entries entered under the **[Device_Config_Table]** heading with the *network_layer* set to V0 | V1) and no **TSI_Hosts** command is specified, host IDs 49 - 63 are automatically assigned to all master TSIs in round robin fashion. Also, any host IDs found in the **[Device_Config_Table]** that have not been assigned to a master TSI by the user, will be assigned in round robin fashion as well, whether network layer RDTs or radios are defined or not.

Description: This command is used to assign EDACS host IDs to all active ports of a TSI Master board. The IMC receives this information and will send messages destined for a given host ID only to the TSI Master to which the host ID is assigned.

host_list contains one or more EDACS host IDs, with each ID separated by a comma. A host ID is any value in the range 1 - 63.

The same host ID may not be assigned to more than one master TSI.

If only non-network layer RDTs are being used (i.e. all RDT entries have a NONE under the *network_layer* column of the **[Device_Config_Table]** heading), the following must also be done for every host ID from 1 - 63 assigned to a TSI:

- it must be mapped to an IP address under the **[Device_Config_Table]** heading.
- it must be assigned to an HDI port, using the **HDI_Port_Hosts** command,
  or its IP address must be entered as the *destination_address* in an **[IP_CAP_Ext_Routing_Table]** HOST table entry,
  or the network portion of its IP address must match the *destination_address* of an **[IP_CAP_Ext_Routing_Table]** NET table entry,
  or the network portion of its IP address must match the IP network ID assigned to the CAP external network, via the **[IP] CAP_Ext_Address** command,
  or the network portion of its IP address must match the IP network ID assigned internally to the EDG, via the **[IP] Int_Network_ID** command.
  or the network portion of its IP address must match the IP network ID assigned to the EDACS network, via the **[EDACS_Network] EDACS_IP_Network_ID** command.
  or the **[IP] CAP_Default_Gateway** command is specified.

If both non-network layer RDTs and network layer RDTs are being used, only one host ID from 1 - 63 must be mapped to an IP address, but all mapped IDs must follow the above rules.

[Board] TSI_HOSTS Command: *Continued*

Example: Non-network layer RDTs are in use and there are two TSI masters in the system. TSI #1 will process messages destined for host IDs 1-4, while TSI #2 will process messages destined for host IDs 5-9.

The following command would be entered for TSI #1:

TSI_Hosts 1,2,3,4

The following command would be entered for TSI #2

TSI_Hosts 5,6,7,8,9

[Board] TYPE Command

Syntax: **Type** CAP | TSI_SLAVE *master_board_num* | TSI_MASTER | HDI

Requirement: Required for TSI_MASTER, TSI_SLAVE, and HDI boards

Default: None

Description: This command defines the board's type. One of three possible text strings shown above must be entered.

master_board_num is required only for a TSI_SLAVE board type. Every TSI slave board in the EDG must be linked to a board defined as a TSI master. Therefore, the board number of the master to which the slave is linked must be entered in this field.

Example: The current board is to be the CAP board. The following command would then be entered:

Type CAP

Example: The current board is to be a TSI Slave type. Board number 2 in the system is a TSI Master type, and is to be this board's master. The following command would then be entered:

Type TSI_SLAVE 2

[EDACS_Network] Command Heading

Format: [EDACS_Network]

This heading is used to define an EDACS IMC/CEC (abbreviated as IMC) in the system. There must always be one EDACS_Network defined, as every master TSI in the EDG is connected to the IMC.

[EDACS_Network] EDACS_IP_NETWORK_ID Command

Syntax: **EDACS_IP_Network_ID** *ip_network_id*

Requirement: Optional command.

Default: 140.1.0.0. This will be changed to 172.16.0.0 for Release 4 of the EDG.

Description: Assigns an IP network ID to the current EDACS IMC network being defined.

Each IMC/CEC has associated with it a single IP network ID with all EDACS RDT, radio, RDT group, and radio group IDs assigned unique IP addresses within that network. The actual mapping of these IDs to IP addresses is defined in the table under the **[Device_Config_Table]** heading.

It is important to note that each IP address assigned to an EDACS RDT, radio, RDT group, or radio group ID must have the same network ID portion, but a unique host ID portion. The EDG will use this information to build its routing tables.

ip_network_id is a 32-bit value, in an IP address format, that obeys the rules for IP network ID values.

[EDACS_Network] EDACS_IP_NETWORK_ID Command: *Continued*

Comments: The class of the network ID defines the available IP addresses that may be assigned to RDTs, radios, RDT groups and radio groups on the EDACS network for this IMC. :
If class A: (first octet of network ID is in range 0 - 127), the host ID portion of the address occupies the lower 24 bits.
If class B: (first octet of network ID is in range 128 - 191), the host ID portion of the address occupies the lower 16 bits.
If class C: (first octet of network ID is in range 192 - 223), the host ID portion of the address occupies the lower 8 bits.
No other network classes are allowed within the EDG.
Note that the host ID bits may not be set to all ones or all zeros.

Example: A class B network ID, 150.0.0.0, is to be assigned to the EDACS Network. The following command would then be entered under the **[EDACS_Network]** heading:

EDACS_IP_Network_ID 150.0.0.0

The available addresses would then be 150.0.0.1 through 150.0.255.254 (address 150.0.0.0 would be the "all zeros" host ID case, while 150.0.255.255 would be the "all ones" host ID case, both of which are illegal).

Example: An EDACS network has been defined, i.e. **[EDACS_Network]**. No IP network ID has been explicitly specified for this EDACS network, so the default value assigned will be 140.1.0.0.

[EDACS_Network] LABEL Command

Syntax: **Label** "*label*"

Requirement: Optional for all EDACS networks.

Default: A text string consisting of "EDACS NET".

Description: This allows a user defined text string of up to 12 case sensitive characters to be assigned to the EDACS Network. It is only referenced in messages to the terminal and in any EDG generated reports.

label may contain any printable ASCII character, including space characters, and must be enclosed by double quotation marks. Labels need not be unique. As many characters as will fit on a single line may be entered as the *label*, but only the first twelve will be read and saved.

Example: It is desired to assign the label "IMC KEY MOUSE". The following command would then be entered under the **[EDACS_Network]** heading:

Label "IMC KEY MOUSE"

However, since only the first 12 characters are saved, the actual label assigned to this EDACS network would be "IMC KEY MOUS".

[IMC] Command Heading

Format: [IMC *imc_number*]

This heading is obsolete with Release 3. It is replaced by the **[EDACS_Network]** command heading. Please reference this heading instead.

[IP] Command Heading

Format: [IP]

This heading's command block contains commands that are used in setting up IP specific values within the EDG. Some commands are board specific, while others affect the overall EDG.
Almost all **[IP]** commands described here are optional, with system defaults normally used in all but the more complex system configurations. See the command descriptions below and the examples given for more details.

[IP] CAP_DEFAULT_GATEWAY Command

Syntax: **CAP_Default_Gateway** *ip_address*

Requirement: Optional command.

Default: None

Description: Defines the default gateway address to which packets will be sent if all of the following conditions are met:

- The packet's destination address is not on the CAP's external network.
- The packet's destination address is not on the EDG internal network.
- The packet's destination address is not in the CAP's routing table.

ip_address is the 32-bit address value, in an IP address format, to be assigned as the default gateway. This address must exist on the CAP's external network.

Comments: If a default gateway is not specified, any packet that meets all the conditions listed in the description above will be discarded.

Example: The CAP's external network ID is 10.0.0.0. Any packet received by the CAP with an unknown destination is to be sent to the address 10.34.78.6 on the CAP's external network. The command entered under the **[IP]** heading would be:

CAP_Default_Gateway 10.34.78.6

[IP] CAP_EXT_ADDRESS Command

Syntax: **CAP_Ext_Address** *ip_address*

Requirement: Optional if used in a RDI host only configuration, required otherwise.

Default: 1.0.0.1 if RDI host only configuration.

Description: Assigns an IP address to the external (i.e. ethernet) port of the CAP. This value is significant only if the EDG is to be connected to an external IP network.

ip_address is the 32-bit address value, in an IP address format, to be assigned as the external CAP address. The network ID portion of *ip_address* must match the ID of the network to which it is connected, while the host ID portion of *ip_address* must be a unique number on that network.

Comment: The CAP external network ID is also derived from this value, based on the network class of the address:
If class A: (first octet of network ID is in range 0 - 127), the network ID is the first octet value, followed by zeros in the remaining three octets.
If class B: (first octet of network ID is in range 128 - 191), the network ID is the first and second octet values, followed by zeros in the remaining two octets.
If class C: (first octet of network ID is in range 192 - 223), the network ID is the first, second, and third octet values, followed by a zero in the fourth octet.
No other network classes are allowed within the EDG.

Example: **CAP_Ext_Address** 129.039.12.6

This command specifies that the external address assigned to the CAP is 129.039.12.6. Since this is a class B address, the CAP external network ID will be set to 129.39.0.0.

[IP] HOST_NAME Command

Syntax: **Host_Name** *name_string*

Requirement: Optional command.

Default: EDG

Description: Assigns a name to the EDG. The host name is used by the *status* shell command.

name_string is a case sensitive string of 1 to 31 alphanumeric characters, with no embedded spaces. The string should not be enclosed in quotation marks.

Example: "Division_8_EDG_System_1" is to be assigned as the IP host name of the EDG. The following command would be entered under the **[IP]** heading:

Host_Name Division_8_EDG_System_1

[IP] INT_NETWORK_ID Command

Syntax: **Int_Network_ID** *ip_network_id*

Requirement: Optional command.

Default: 199.0.0.0. This will change to 192.168.100.0 for Release 4 of the EDG.

Description: Assigns an IP network ID to the EDG internal network. All devices that communicate directly on the same network have the same unique network ID as part of their IP address. In this case, the network is the EDG backplane, and the devices are the CAP, TSI, and HDI boards.

ip_network_id is a 32-bit value, in an IP address format, that obeys the rules for IP network ID values.

Example: A class C network ID, 201.0.0.0, is to be assigned to the EDG internal network. The following command would then be entered under the **[IP]** heading:

Int_Network_ID 201.0.0.0

[IP] INT_BOARD_ADDRESS Command

Syntax: **Int_Board_Address** *board_number ip_address*

Requirement: Optional command.

Default: The network ID portion of a board's address is set to the **Int_Network_ID** value, while the host ID portion is set to the board's number plus one.

Description: Defines the internal IP address to be assigned to each board in the EDG.

board_number is the number of the board to which the address is to be assigned.

ip_address is the 32-bit address value, in an IP address format, to be assigned to the board.

Comments: All internal IP addresses assigned to boards must contain the same internal network ID value. The host ID portion of the address must be within the range of allowable values as determined by the network class:
If class A: (first octet of network ID is in range 0 - 127), the host ID portion of the address occupies the lower 24 bits.
If class B: (first octet of network ID is in range 128 - 191), the host ID portion of the address occupies the lower 16 bits.
If class C: (first octet of network ID is in range 192 - 223), the host ID portion of the address occupies the lower 8 bits.
No other network classes are allowed within the EDG.
Note that the host ID bits may not be set to all ones or all zeros.

Example:

Internal_Network_ID	192.16.4.0	
Internal_Board_Address	1	192.16.4.1
Internal_Board_Address	2	192.16.4.8

In this example, the internal network ID defines a class C network, which means that the upper 24 bits of every internal board address must be the same value, namely 192.16.4. Since only the lower 8 bits are to be used for host IDs, the addresses that may be assigned to EDG boards may only be in the range 192.16.4.1 to 192.16.4.254. Addresses 192.16.4.0 and 192.16.4.255 may not be assigned to a specific device since to do so would violate IP addressing rules. Here, board 1 is assigned the address 192.16.4.1, while board 2 is assigned the address 192.16.4.8.

[IP] INT_BOARD_ADDRESS Command: *Continued*

Example: All the addresses in the class C network, 197.0.0.0, are available for use within the internal EDG network. There are 3 boards in the system. In this case, the internal board addresses may be used, with only the network ID specified under the **[IP]** heading:

Int_Network_ID 197.0.0.0

The following addresses would then be assigned automatically:
Board 1: 197.0.0.2
Board 2: 197.0.0.3
Board 3: 197.0.0.4

[IP] ROUTING_ADDRESS Command

Syntax: **Routing_Address** *ip_address*

Requirement: Optional command. The command is only needed when the EDG's IP addresses are being set manually.

Default: The network ID portion of the address is set to the **Int_Network_ID** value, while the host ID portion is set to 1.

Description: This is a unique IP address that is required by the EDG in order to perform proper routing. In an Internodal Data configuration, this is the address that other EDG use to communicate with the EDG.

ip_address is a unique 32-bit address value, in an IP address format. It must contain the EDG internal network ID, and the host ID portion of the address must be within the range of allowable values as defined by the internal subnet mask value.

See the **Int_Network_ID** and **Int_Board_Address** commands.

Example: The internal network ID is to be 210.3.2.0, which is a class C network. Only addresses 210.3.2.20 through 210.3.2.23 are available for use. There are 2 boards in the system. The following commands would be entered under the **[IP]** section:

Int_Network_ID	210.3.2.0
Routing_Address	210.3.2.20
Int_Board_Address 1	210.3.2.21
Int_Board_Address 2	210.3.2.22

Example: There are 3 boards in the system. The internal network ID is to be 210.3.2.0, which is a class C network. All addresses in the network are available for use. The following command is entered under the **[IP]** section:

Int_Network_ID 210.3.2.0

The Routing Address will be assigned a default IP address of 210.3.2.1. Board 1 will be assigned a default IP address of 210.3.2.2, board 2 will be assigned 210.3.2.3, and board 3 will be assigned 210.3.2.4.

[System] Command Heading

Format: [System]

This heading's command block contains commands that are used in setting up global system parameters.

[System] EDACS_ERR_RETRIES Command

Syntax: **EDACS_Err_Retries** *max_count delay*

Requirement: Optional command. Its value is used to control the number of retries and the delay period between them on a case of a transmission failure.

Default: 0 for *max_count*, 0 (in tenths of a second) for *delay*.

Description: This command is used to specify the maximum number of retransmissions that a TSI board should attempt for a message when it fails to be transmitted successfully. It also delays this retransmission by a configurable amount of delay.

max_count is the maximum number of retransmission the TSI board should attempt from 0 to 100.

delay is the number of tenths of seconds in the range 0 to 1000.

Comments: The counts and delays kept for each message are independent of the **EDACS_Busy_Retry** command. Therefore, a message could be retried up to a *max_count* number of times specified in the **EDACS_Err_Retry** command for failed messages and a *max_count* number of times specified in the **EDACS_Busy_Retry** command for system busy messages, or any lesser combination thereof.

Example: **EDACS_Err_Retry** 2 40

This command indicates that the EDG should retry a failed message up to 2 times with a delay of 4 seconds between retries.

[System] EDACS_BUSY_RETRIES Command

Syntax: **EDACS_Busy_Retries** *max_count delay*

Requirement: Optional command. Its value is used to control the number of retries and the delay period between them on a case of a transmission failure due to a system busy.

Default: 3 for *max_count*, 20 (in tenths of a second) for *delay*.

Description: This command is used to specify the maximum number of retransmissions that a TSI board should attempt for a message when it fails to be transmitted successfully due to a system busy. It also delays this retransmission by a configurable amount of delay.

max_count is the maximum number of retransmission the TSI board should attempt from 0 to 100.

delay is the number of tenths of seconds in the range 0 to 1000.

Comments: The counts and delays kept for each message are independent of the **EDACS_Err_Retry** command. Therefore, a message could be retried up to a *max_count* number of times specified in the **EDACS_Busy_Retry** command for system busy messages and a *max_count* number of times specified in the **EDACS_Err_Retry** command for failed messages, or any lesser combination thereof.

Example: **EDACS_Busy_Retry** 2 40

This command indicates that the EDG should retry a message up to 2 times with a delay of 4 seconds between retries.

[System] MAX_FTP_SESSIONS Command

Syntax: **Max_FTP_Sessions** *session_count*

Requirement: Optional command.

Default: 2

Description: This command is used to specify the maximum number of FTP sessions that may be simultaneously active in the EDG from remote hosts.

session_count is a value in the range 0 to 4. A value of 0 indicates that no remote hosts may access the EDG via FTP.

Example: **Max_FTP_Sessions** 1

This command indicates that only one remote host may access the EDG via FTP. If a second host tries initiate an FTP session, it will be denied.

[System] MAX_MSGS Command

Syntax: **Max_Msgs** *msg_count*

Requirement: Optional command.

Default: 100

Description: This command is used to specify the maximum number of outbound messages that may be queued on a TSI Master or HDI. If the maximum count is reached, any new outbound messages received by that board will be discarded until the message count drops below the maximum. The EDG uses this parameter to give priority to messages with multiple fragments and to protect against running out of memory.

msg_count is the number of messages in the range 10 to 1000.

Comments: When a message is discarded due to this maximum message count being exceeded, an ICMP "Source Quench" message is sent back to the message source.

This command is used to fine tune the EDG system performance (also see the **Msg_Timeout** command).

If boards are running out of memory, this count should be lowered. Boards that are out of memory will continue to process messages, but in a degraded mode. Inbound messages will be discarded and fragments for partially completed outbound messages are no longer accepted.

If the majority of outbound messages are small and are arriving at a high rate, then this value can be increased for higher throughput.

If there are multiple boards of the same type, a board may reach its maximum and stop accepting new outbound messages even if the other boards are not filled to capacity. The other boards will continue to accept new outbound messages though.

Example: **Max_Msgs** 200

This command indicates that there is a maximum of 200 outbound messages that can be queued at an individual TSI Master or HDI.

[System] MAX_TELNET_SESSIONS Command

Syntax: **Max_TELNET_Sessions** *session_count*

Requirement: Optional command.

Default: 2

Description: This command is used to specify the maximum number of TELNET sessions that may be simultaneously active in the EDG from remote hosts.

session_count is a value in the range 0 to 4. A value of 0 indicates that no remote hosts may access the EDG via TELNET.

Example: **Max_Telnet_Sessions** 3

This command indicates that only 3 remote hosts may access the EDG via TELNET at the same time. If a fourth host tries initiate a TELNET session, it will be denied.

[System] MSG_TIMEOUT Command

Syntax: **Msg_Timeout** *timeout_value*

Requirement: Optional command.

Default: 30 seconds

Description: This command specifies the maximum amount of time, in seconds, that a message may be active in the EDG (either on an HDI or TSI) before being discarded.

The time-out period begins when the EDG receives the first fragment of a message.

timeout_value is the number of seconds in the range 15 to 1000.

Comments: **This parameter should be less than the time out value used in the Transport Layer**. If the Transport Layer times out a message and resends it before the EDG times out the message, both copies of the message could be queued at the EDG and sent to the radio. This causes unnecessary congestion and wastes resources.

When the EDG times out a message, an ICMP "Fragment reassembly time exceeded" message is sent back to the message source.

If a transport layer is handling retransmissions, the following guidelines should be used to set the transport layer time-out value:

With RDI hosts not using ACK2 and all IP hosts, set it to a minimum of *timeout_value* plus five seconds.

With RDI hosts using ACK2, set it to a minimum of *timeout_value* plus 15 seconds.

With RDI hosts using a 45 second time-out for the ACK2 return (as defined in the RDI specification), *timeout_value* must be set to 30 seconds.

This command is used to fine tune the EDG system performance along with the **Max_Msgs** command.

The "time to live" value in the IP header is not used to time-out a message since this is a hop count, rather than an actual time value.

Example: **Msg_Timeout** 40

This command indicates that a time-out value of 40 seconds is to be applied separately to each message destined for output from a TSI or HDI. If the full message has not been transmitted 40 seconds after receiving the first fragment of the message, any and all remaining fragments will be discarded.

[System] OUTBOUND_MSG_DELAY Command

Syntax: **Outbound_Msg_Delay** *delay_value*

Requirement: Optional command.

Default: 300 milliseconds

Description: This command is used to specify the minimum delay to be introduced by the EDG between consecutive data calls to the same destination radio on the IMC network. The delay is computed as the time between the end of one data call and the beginning of the next data call.

delay_value is the number of milliseconds in the range 0 to 15000, and in increments of 10, i.e. 0, 10, 20, 30, ... 14990, 15000.

Comments: This forced delay is required due to the call setup and data processing time requirements at the receiving radio and RDI. Without such a throttling mechanism at the EDG, data loss would increase, due to two factors:

- Some radio models require a minimum amount of setup time between the end of one call and the beginning of the next call.
- Even after the radio has transmitted all the data to the RDI and is ready to accept the next call, the RDI may still be transmitting the data to the RDT.

The amount of delay to be specified here is dependent upon the particular radio model being used, and whether simulation of a full duplex link is desired between the EDG and any given RDT.

For one way outbound data traffic, i.e. data being sent out from the EDG, as of the writing of this document, the following values were successfully used for the radio model indicated:

FMD	100 (for unit data call sizes < 350 bytes)
	200 (for unit data call sizes > 350 bytes)
	600 (for group data calls)
MTD	200 (for unit data calls)
	600 (for group data calls)
Orion	0 (for unit data calls)
	100 (for group data calls)

[System] OUTBOUND_MSG_DELAY Command: *Continued*

The delay value can also be used to simulate a full duplex link, i.e. two way simultaneous traffic, between the EDG and the RDT. This is accomplished by making the delay long enough for the RDT to respond to a received message before the next message is sent from the EDG, as well as allowing the message response to be received by the EDG before the number of transmit retries on the next outbound message have been exhausted.

If throughput is not an issue, a good value to use is 7500 ms plus the amount of time it takes to process the message at the RDT. If throughput is an issue, the delay could be reduced to as low as 1000 ms. However, this may increase the outbound data error rate since the EDG may try to transmit the next outbound message before the inbound message response is received in full. Although the EDG will retry a message transmission three times, at two second intervals, before finally giving up and discarding the message, it is not desirable to effectively "waste" the available retries due to a delay value that is too small.

The delay value is more of a work around to a possible data congestion problem, rather than a solution. The best way to avoid congestion is to have the application software on the host or RDT slow down its message output.

It should be further noted that, in the case where the same radio is receiving both group and unit data calls consecutively, this delay factor will not help prevent the loss of data. This is due to the fact that the group call will have a destination ID or address that is different from that of the unit call, and the EDG does not track which radios are assigned to which groups.

Example: **Outbound_Msg_Delay** 100

This command indicates that for two or more consecutive data calls with the same destination ID or address, the TSI will wait a minimum of 100 milliseconds after the end of one call before it attempts the next call to the same radio.

[System] TRANSPORT_LAYER_PROTOCOL Command

Syntax: **Transport_Layer_Protocol** *protocol_type_code*

Requirement: Optional command. Its value is referenced only when non-network layer RDTs are being used.

Default: 17, which is the protocol type code for UDP IP packets.

Description: When non-network layer RDTs are used, the message arrives at the TSI, from the RDT, with no IP header. However, all messages that pass through the EDG must have an IP header, which the TSI will create in this situation. One of the fields in this header is the protocol type code field. The value defined by this command is the value placed in that field by the TSI.

protocol_type_code is a value in the range 0 to 255. There are many standard protocol type codes defined, including the following:

1 ICMP
6 TCP
8 EGP
17 UDP

Example: **Transport_Layer_Protocol** 6

This command indicates that the TCP protocol type code will be placed in the IP header created by the TSI when non-network layer RDTs are in use.

[System] TSI_ANTI_BIASING Command

Syntax: **TSI_Anti_Biasing** TRUE | FALSE

Requirement: This command is obsolete for Release 3. It is now configured by the *BREN_configuration* entry in the **[Device_Config_Table]**. Please reference this command.

[Device_Config_Table] Table Heading

Format: [Device_Config_Table]

This heading precedes a table of entries that will configure characteristics for a number of different devices. One such function is to map various types of EDACS IDs to IP addresses. The EDG requires a unique IP address to be mapped to every RDT, radio, RDT group and radio group ID involved in data communications. This is necessary since every network data message sent through the EDG must have a source IP address, identifying the message sender, and a destination IP address, identifying the desired recipient. The source address allows the recipient to respond back to the sender, while the destination address is used to provide the correct routing through the EDG and any external IP networks. The device is also configured as using a network layer or not. Finally, the device is designated as implementing BREN or not.

There are no commands associated with this heading, only one or more single line entries. The entry format is described below.

[Device_Config_Table] Entry

Entry Syntax: *id_type id_range start_ip_address BREN_configuration network_layer*

[Device_Config_Table] Table Entry: *Continued*

Description: Each entry in this table maps either a single EDACS ID or a block of EDACS IDs to a single IP address or block of IP addresses, respectively. It also configures BREN for the device and configures the network layer header version, if one is used. There is no limit to the number of entries that may be placed in this table, i.e. every possible EDACS ID may be assigned to a separate entry. Each entry consists of two to three required fields depending on the type of entry.

id_type consists of a string identifying the type of EDACS ID being mapped. Valid values include:

- **RDT** - EDACS Radio Data Terminal ID
- **RDT_group** - EDACS Radio Data Terminal group ID
- **radio** - EDACS radio unit ID for radio resident applications.
- **radio_group** - EDACS radio group ID for radio resident applications.
- **host** - EDACS Host ID. Although a host ID is actually a unit ID in the range 1 - 63, it must be distinguished, within the EDG, from an RDT unit ID assigned to a radio. This is necessary for building of internal routing tables, and other error checking algorithms.

id_range is a single host, RDT or radio ID or range of host, RDT or radio IDs to be mapped to a single IP address or range of IP addresses, respectively. A range is entered in the from:

start_id - end_id

RDT and radio IDs may be in the range 1 - 16382, and RDT and radio group IDs may be in the range 1 -2047.

start_ip_address is a single value in IP address format. If the preceding id_range field is a single value, then *start_ip_address* is the IP address assigned to the ID value. If, instead, *id_range* is a range of values, then *start_ip_address* identifies the beginning of a contiguous block of addresses to be assigned to the IDs. RDTs, radios, RDT groups, and radio groups not local to the EDACS Network should **NOT** have an entry.

BREN_configuration is an optional string value for RDT and radio units on the EDACS Network. Valid values are ON | OFF. It should **NOT** be entered for RDT or radio groups, nor for RDT and radio units not local to the EDACS Network, nor for hosts.

[Device_Config_Table] Table Entry: *Continued*

Description: *network_layer* is an optional string value for RDT, radio, RDT group, and radio group IDs on the EDACS Network. Valid values are V0 | V1 | NONE. The device must specify NONE if it does not use the network layer. RDTs, radios, RDT groups, or radio groups not local to the EDACS Network should **NOT** have an entry. Hosts should also **NOT** have an entry for this field.

Requirement: Optional.

[Device_Config_Table] Table Entry: *Continued*

Default: If **any** RDT, radio, RDT group, or radio group entries are entered, then **no** defaults will be created for any of those types. Otherwise, the following defaults will be created for all entries:

RDT entry records will be generated for all units from 64 to 16382 that are not used as TSI EDACS IDs. The *start_ip_address* is Host ID 0.64 on the EDACS IP Network. The default *BREN_configuration* is ON. The default *network_layer* is V0.

radio entry records will be generated for all units from 64 to 16382 that are not used as TSI EDACS IDs. The *start_ip_address* is Host ID 72.64 on the EDACS IP Network. The default *BREN_configuration* is OFF. The default *network_layer* is V1.

An RDT group entry record will be generated for groups 1 to 2047. The *start_ip_address* is Host ID 64.0 on the EDACS IP Network. There is no default *BREN_configuration*. The default *network_layer* is V0.

An RDT group entry record will be generated for groups 1 to 2047. The *start_ip_address* is Host ID 136.0 on the EDACS IP Network. There is no default *BREN_configuration*. The default *network_layer* is V1.

Furthermore, the *start_ip_address* is only required for the first EDACS network entry in the **[Device_Config_Table]**. Successive entries on the EDACS network (i.e. RDTs, radios, RDT groups, and radio groups) may leave this field blank, which will default the IP address to continue sequentially from the last IP address used in the previous entry on the EDACS network.

An RDT entry with no *BREN_configuration* will be defaulted a value of OFF. A Radio entry with no *BREN_configuration* will be defaulted a value of ON. RDT groups and radio groups have no default *BREN_configuration*.

An RDT or RDT group entry with no *network_layer* will be defaulted a value of V0. A Radio or radio group entry with no *network_layer* will be defaulted a value of V1.

If no host entries are entered, the following defaults will be created:

A host entry record will be generated for each host ID assigned to with an **[Board] HDI_Port_Hosts** command. The *start_ip_address* is the host ID on IP Network 200.0.0.0, this will change to 192.168.200.0 during Release 4 of the EDG.

There is no default *BREN_configuration* or *network_layer* for host entries.

[Device_Config_Table] Table Entry: *Continued*

Comments: Every host entry must be assigned to a TSI Master through the **TSI_Hosts** command. This may be done explicitly, or it will be done by default by the EDG during startup. The EDG will assign hosts in round-robin fashion to TSI Masters.

Example: The following IP mapping must be performed:

- Host IDs 1 - 10 are to be assigned a block of IP addresses beginning at address 200.0.0.1. Note there are no *BREN_configuration* or *network_layer* entries.
- radio ID 1000 is to be assigned address 130.2.0.1, using BREN and network layer header V1.
- radio ID 1001 is assigned the next default address (130.2.0.2). The device uses BREN and is assigned the default radio network layer header (V1).
- RDT ID 1000 is to be assigned address 130.2.0.10. The device is assigned the default RDT BREN configuration (OFF) and does not use a network layer header (NONE).
- RDT ID 1001 is to assigned the next default address 130.2.0.11. The device is assigned both the default BREN configuration (OFF) and network layer header (V0).
- RDT IDs 5000 - 5999 are to be assigned a block of IP addresses beginning at 130.2.1.0, without using BREN and using network layer header V0.
- radio IDs 5000 - 5999 are to be assigned a block of IP addresses beginning at 130.2.128.0, using BREN and network layer defaults (ON and V1).
- All EDACS RDT group IDs (1 - 2047) are to be assigned a block of IP addresses beginning at 130.2.128.0, using network layer defaults (V0).

The table would then appear as follows:

[Device_Config_Table]

host	1-10	200.0.0.1			# Host IDs
radio	1000	130.2.0.1	ON	V1	# Special radio
radio	1001		ON		# Special radio
RDT	1000	130.2.0.10		NONE	# NNL RDT
RDT	1001				# Special RDT
RDT	5000-5999	130.2.1.0	OFF	V0	# RDTs units
radio	5000-5999	130.2.128.0			# radio units
RDT_group	1-2047	130.2.128.0			# All possible RDT groups

[Internodal_EDG_Route_Table] Table Heading

Format: [Internodal_EDG_Route_Table]

This table is only used when the Internodal Data feature is enabled. It provides an EDG with a list of the other EDGs in an Internodal Data configuration. It also tells the EDG the next gateway to use to reach each of the EDGs.

There are no commands associated with this heading, only one or more single line entries. The entry format is described below.

[Internodal_EDG_Route_Table] Entry

Entry Syntax: *edg_route_addr* *next_gateway*

Description: *edg_route_addr* is a remote EDG's Routing Address.

next_gateway is the IP address of the next gateway that should be used to reach the remote EDG's Route Address. If both EDGs are on the same external network, then *next gateway* should be the remote EDG's CAP_Ext_Address. Otherwise *next gateway* should be the address of the router that this EDG should use to reach the remote EDG. The *next_gateway* address must contain the same network ID as the local EDG's CAP_Ext_Address.

Requirement: Optional.

Default: No other EDGs defined.

Comments: Note that the host ID bits of the *edg_route_addr* may not be set to all ones or all zeros (i.e. the *edg_route_addr* may not be a network id).

Example: An Internodal Data system contains three nodes, EDG_1, EDG_2, and EDG_3. EDG_1 and EDG_2 are on the same network. The router 140.1.5.9 connects EDG_1 to the rest of the internet containing EDG_3. EDG_1 would require the following Internodal_EDG_Route_Table:

[Internodal_EDG_Route_Table]

201.0.0.1	1.0.0.5	# Reach EDG 2 via EDG 2's CAP Ext. Addr.
202.0.0.1	140.1.5.9	# Reach EDG 3 via Router 140.1.5.9

[Internodal_Group_Repeat_Table] Table Heading

Format: [Internodal_Group_Repeat_Table]

This table contains a list of RDT and radio group ID types and IDs. If an EDG that is in an Internodal Data configuration receives a RDT or radio group data call, the EDG sends the RDT or radio group call out on its local node and also sends the RDT or radio group data call to all of the other EDGs. When an EDG receives a repeated RDT or radio group call, the EDG will check its Internodal_Group_Repeat_Table. If the RDT or radio Group ID (GID) is in the table, the EDG will repeat the call on its local node.

There are no commands associated with this heading, only one or more single line entries. The entry format is described below.

[Internodal_Group_Repeat_Table] Entry

Entry Syntax: *id_type id_range*

Description: *id_type* consists of a string identifying the type of EDACS ID. Valid valued include:

- **RDT_group** - EDACS Radio Data Terminal group ID
- **radio_group** - EDACS Radio group ID.

id_range is a single RDT or radio group ID or range of RDT or radio group IDs. A range is entered in the form:
start_id - end_id
RDT and radio group IDs may be in the range 1 - 2047.

Requirement: Optional.

Default: No group calls will be repeated on this node.

Example: An Internodal Data system contains three nodes, EDG_1, EDG_2, and EDG_3. EDG_1 does not have an Internodal_Group_Repeat_Table. EDG_2's table appears as follows:

[Internodal_Group_Repeat_Table]
radio_group 50
radio_group 100 - 250
RDT_group 100 - 250

If a data call to RDT group 175 is received on EDG_3, the call will be sent out on EDG_3's node, and repeated on EDG_2. EDG_1 will not send the call out on its node. The radio will know this call is for the RDT with group ID 175 and not for the radio with group ID 175.

[IP_CAP_Ext_Routing_Table] Table Heading

Format: [IP_CAP_Ext_Routing_Table]

This heading precedes a table of entries that make up the external routing table for the CAP board. Each entry in the table defines the correct routing for any messages destined for an external host that is not on the CAP's local external network, i.e. the host's network ID is not the same as the CAP external network ID.

There are no commands associated with this heading, only one or more single line entries. These entries are concatenated to the CAP's internal routing table, which is built automatically at startup. The entry format is described below.

NOTE

The total number of entries in the CAP's routing table, which consists of the external entries defined here, plus the number of CAP internal routing table entries, plus any CAP default gateway entry, may not exceed 1024.

The number of CAP internal routing table entries will be equal to the number of unique host IDs assigned to HDIs, plus four (one each for the EDG backplane Network Interface (NI), the foreplane NI, the loopback address 127.0.0.1), and the EDACS Network.

[IP_CAP_Ext_Routing_Table] Entry

Entry Syntax: HOST | NET *destination_address next_gateway*

Description: The *destination_address* type is specified using one of the two values shown above. If it refers to an individual destination, the HOST keyword should be specified. If it refers to a network, the NET keyword should be specified.

destination_address is the IP address of an external host, or an external network, that is not on the CAP's local external network, but can be reached via a device, i.e. a gateway, that is on the CAP's local external network.

next_gateway is the IP address of a host that is on the CAP's local external network through which messages may be routed to reach a destination on another network. Note that any next_gateway address must contain the same network ID as that of the CAP external address.

[IP_CAP_Ext_Routing_Table] Entry: *Continued*

Requirement: Optional.

Default: No Additional routes defined.

Example: The CAP's external network ID is 10.0.0.0. A gateway with address 10.4.3.2 exists on this network through which the network 144.6.0.0 may be reached. This same gateway may reach network 144.7.0.0. Another gateway with address 10.20.2.79 exists on this network through which the single host device 6.1.1.16 may be reached. The table would then appear as follows:

[IP_CAP_Ext_Routing_Table]

NET	144.6.0.0	10.4.3.2
NET	144.7.0.0	10.4.3.2
HOST	6.1.1.16	10.20.2.79

[IP_Map_ID_Table] Command Heading

Format: [IP_Map_ID_Table]

This heading is obsolete with Release 3. It is replaced by the **[Device_Config_Table]** command heading. Please reference this heading instead.

[IP_Map_ID_Table] Entry

Entry Syntax: *id_type id_range start_ip_address*

Requirement: This command is obsolete with Release 3. It is replaced by the *id_type, id_range* and *start_ip_address* entries in the **[Device_Config_Table]**. Please reference this command instead.

[Network_Layer_RDT_Table] Table Heading

Format: [Network_Layer_RDT_Table]

This heading is obsolete with Release 3. It is replaced by the *network_layer* entry in the **[Device_Config_Table]**. Please reference this command instead.

[Network_Layer_RDT_Table] Entry

Entry Syntax: *id_type id_range*

Requirement: This command is obsolete with Release 3. It is replaced by the *network_layer* entry in the **[Device_Config_Table]**. Please reference this command instead.

Ericsson Inc.
Private Radio Systems
Mountain View Road
Lynchburg, Virginia 24502
1-800-592-7711 (Outside USA, 804-592-7711)

Printed in U.S.A.