FieldServer Utilities Manual

APPLICABILITY & EFFECTIVITY

This manual provides instructions for the following FieldServer products:

	Description
FS-X20 Series	FieldServer
FS-X30 Series	FieldServer
FS-X40 Series	FieldServer

Effective for all systems manufactured after April 2013

Kernel Version:	6.10
Document Revision:	0

TABLE OF CONTENTS

1 UTILITIES OVERVIEW

1.1 Ping Utility (RuiPing)

The RuiPing Utility is used to:

- Ascertain the existence of working FieldServers
- Monitor a FieldServer's healthy system operation
- Check a FieldServer's system information

1.2 Remote User Interface (RuiNet)

The RuiNet Utility is used to:

- Transfer files (configuration, firmware, etc) to and from a FieldServer
- Monitor a working FieldServer's internal data and parameters
- Change or update a FieldServer's internal data and parameters
- Delete files on a FieldServer
- Change the FieldServer's IP address
- Restart a FieldServer

1.3 Serial and Snapshot Capture Utility (RuiDebug)

The RuiDebug Utility is used to capture information about a FieldServer to assist in problem diagnosis. RuiDebug is useful for troubleshooting low-level serial communications when standard troubleshooting options have been exhausted. RuiDebug captures all data traffic on a given serial port in either hex or ASCII format in a log file which can be analyzed for inconsistencies by anyone familiar with the communications protocol in question. RuiDebug also captures information about the FieldServer being monitored in a separate log file (RuiDebug.log). The snapshot option available with RuiDebug also captures connection statistics in RuiDebug.log.

The information captured includes

1. FieldServer status Information
2. Connection Statistics
3. Error Messages
4. Data Array Contents

Snapshot

5. Port Logs

The RuiDebug Utility **cannot** capture a log of Ethernet based message contents.

1.4 FST_Diag Utility

The FST_Diag Utility connects to a FieldServer, collects information about the FieldServer setup and current processes, downloads the configuration, zips all information into one file and reports back to FieldServer support with the information.
The FST_Diag Utility **cannot** capture a log of Ethernet based message contents. Refer to Enote0063 on the supplied flash drive for information on how to do this.

1.5 Internet Address Configurator (RuiBoot)

The RuiBoot application can be used to monitor the FieldServers on a LAN, automatically allocate IP addresses, alert the user to duplicate IP addresses and perform other Utility functions such as setting the MAC address of the Ethernet adapter on the FieldServer. This Utility is typically only required if advanced functions are needed or if old firmware that does not support IP change via RuiNet is being used.

2 PC REQUIREMENTS

2.1 Hardware

The Utilities work over an Ethernet network and require a TCP/IP enabled PC with a network card supporting 10/100Mbit/s Ethernet. The PC and FieldServer can be connected via an established network or directly using a cross-over cable.

2.1.1 Connecting to a FieldServer over the Internet

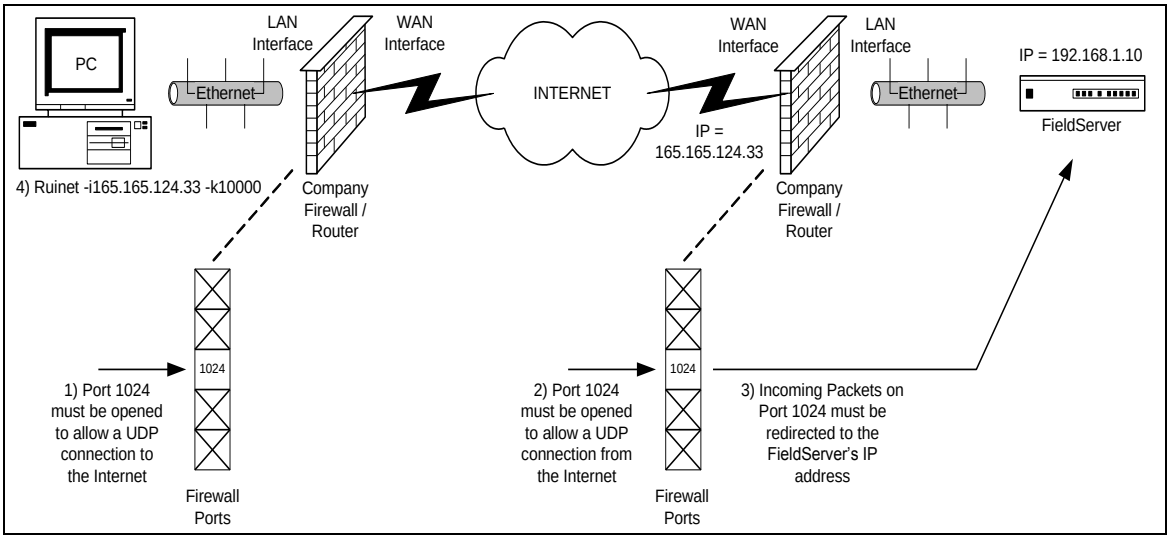


Figure 2.1 – Diagram showing Connection over the Internet

Notes

1. The PC firewall must allow UDP connections to port 1024 on the WAN (Wide Area Network) interface/Internet.
2. The network firewall must allow UDP connections from the WAN interface/Internet to port 1024.
3. The network firewall must redirect network packets from port 1024 to the FieldServer's IP address.

2.2 Software

The Utilities run under any of the following operating systems:

DOS, Windows 95 (SP2 upwards), Windows 98, 2000, NT, XP

2.3 Installation and Setup

- The utilities are loased on the flash drive shipped with the FieldServer. They can be accessed from the start menu once they have been downloaded onto the computer: ***|Start|Programs|FieldServer Utilities|***
- The Ruinet PC and the FieldServer have to be setup with an IP address on the same subnet .
- The Ruinet PC's TCP/IP settings may have to be changed to successfully connect to a specific FieldServer if there is more than one FieldServer on the network.
- If a PC is used on an already established network, it is better to change the FieldServer's IP address than the PC's IP address. Please refer to section 5.14.1 to change the FieldServer's IP address.

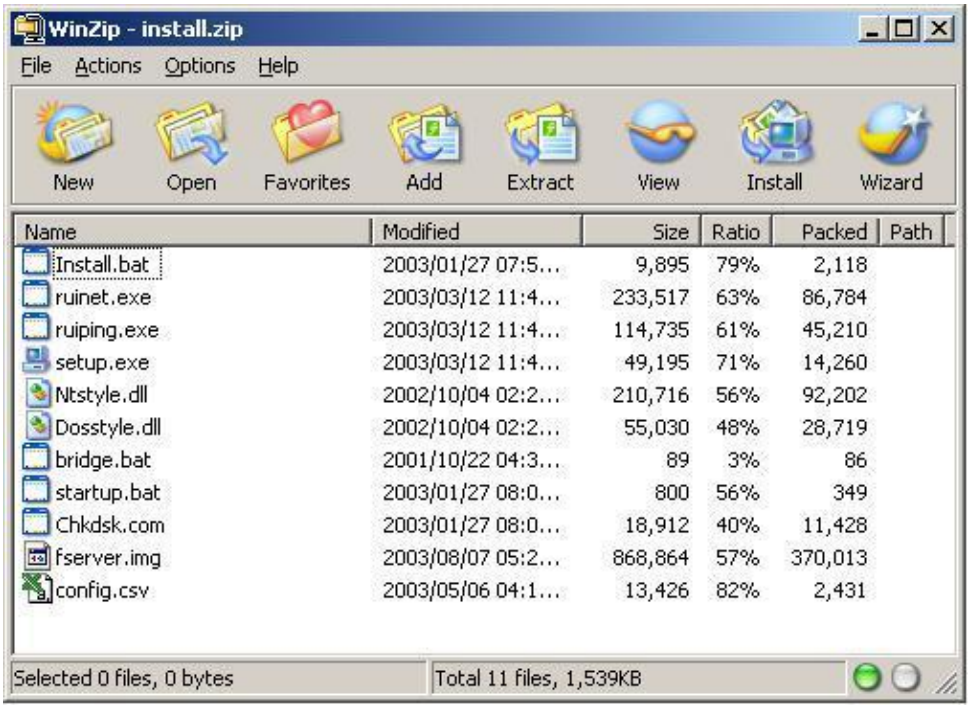
3 INSTALLING NEW FIRMWARE ON A FIELDSERVER

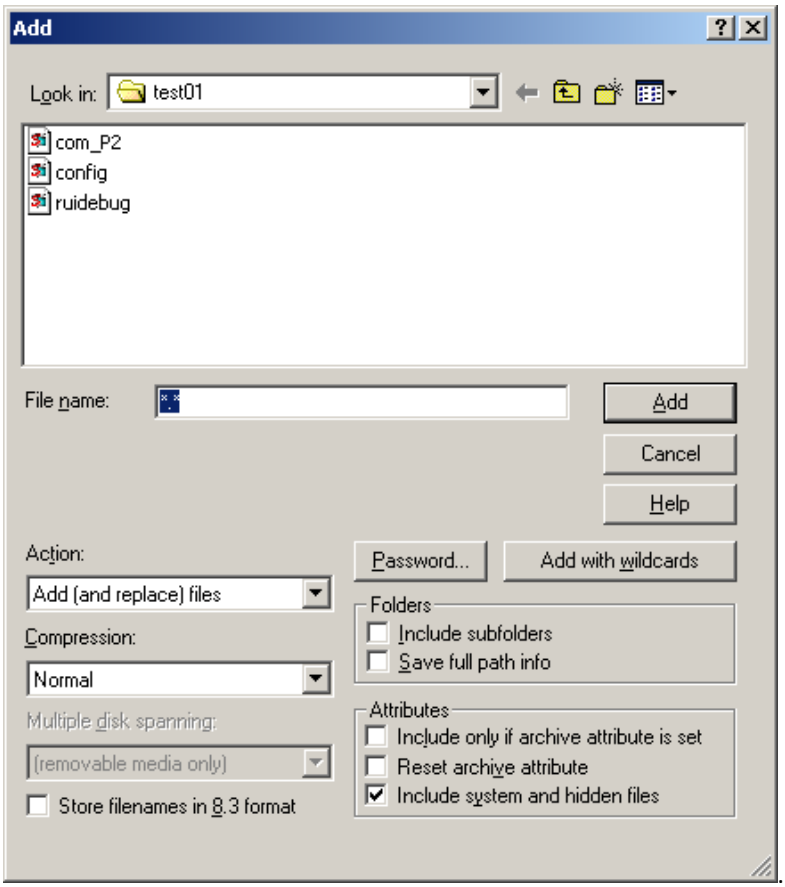
New firmware is supplied as a zip file called install.zip under special circumstances where a firmware update is required. It is not part of the standard shipment.

- Open the install.zip file. This can be done directly from your mail client.
- If you are using WinZip, press “Install”. Alternatively extract the files to a temporary folder and run the setup.exe program.
- If there is only one FieldServer on your network, then the firmware install procedure will start automatically after pressing “Install”
- If there are multiple FieldServers on your network, select the FieldServer that requires the new firmware to be installed from the menu shown.
- To download a configuration file other than the one included in the Install.zip file do the following:

3.1 Using Winzip:

- Rename the file you want to download to the FieldServer to config.csv
- Click on “Add”





- Browse and Select the config.csv file you want to send to the FieldServer
- Click “Add”
- Click “Install” to install the config.csv file to the FieldServer.

3.2 Not using WinZip:

- Copy the config.csv file to the Install.zip

Notes

- The csv file to be added must be named config.csv
- If the Install.zip file contains multiple csv files, only one will be downloaded to the FieldServer.
- If config.csv, primserv.csv and secdserv.csv are present then config.csv will be downloaded.
- If config.csv is not present then primserv.csv will be downloaded and if config.csv and primserv.csv are not present then secdserv.csv will be downloaded.

4 PING UTILITY (RUIPING)

RuiPing is a Utility that "pings" a specific FieldServer or all FieldServers on the network. A ping is a general message that requests a reply from a FieldServer. The reply indicates the existence of a FieldServer on the network and contains information of use to the RuiPing operator or to batch files using RuiPing. FieldServers will respond to the 'Ping' application provided with Microsoft Windows, but there is no distinction between FieldServers and other devices that respond to the ping.

Run RuiPing from the Start Menu: ***Start|Programs|FieldServer Utilities|Ping Utility*** or using Run, Command, RuiPing -i1.2.3.4

The screen-shot below shows the typical layout of information presented by RuiPing:

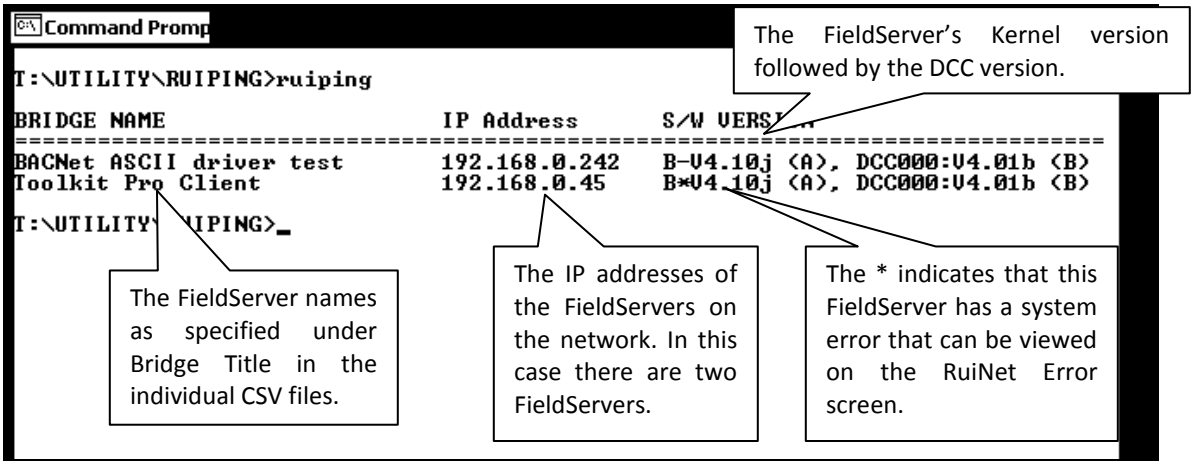


Figure 4.1

RuiPing will continue pinging FieldServers every 3 seconds until the Q key is pressed or will exit when conditions specified by command-line switches have been met.

4.1 Manually checking the network for Duplicate IP Addresses

- Connect the FieldServer to the network and run RuiPing. Verify that the IP address chosen for the FieldServer is active
- Disconnect the FieldServer. Run the Windows Ping Utility (in DOS, type: ping <IP Address>). Make sure that nothing responds to the chosen IP address.

For further information on pinging specific FieldServers, refer to Appendix B.

5 REMOTE USER INTERFACE (RUINET)

The notes in this section explain how to navigate the menu system provided by RuiNet and provide information on the contents of the various screens.

5.1 A – Connecting to a FieldServer

Since RuiNet can only work with one FieldServer at a time, it is necessary to target the FieldServer of interest. (Note that it is possible to run multiple instances of RuiNet at the same time). If RuiNet is run without specifying a target it will provide a list of the FieldServers on the network. Pick the required FieldServer from the list -

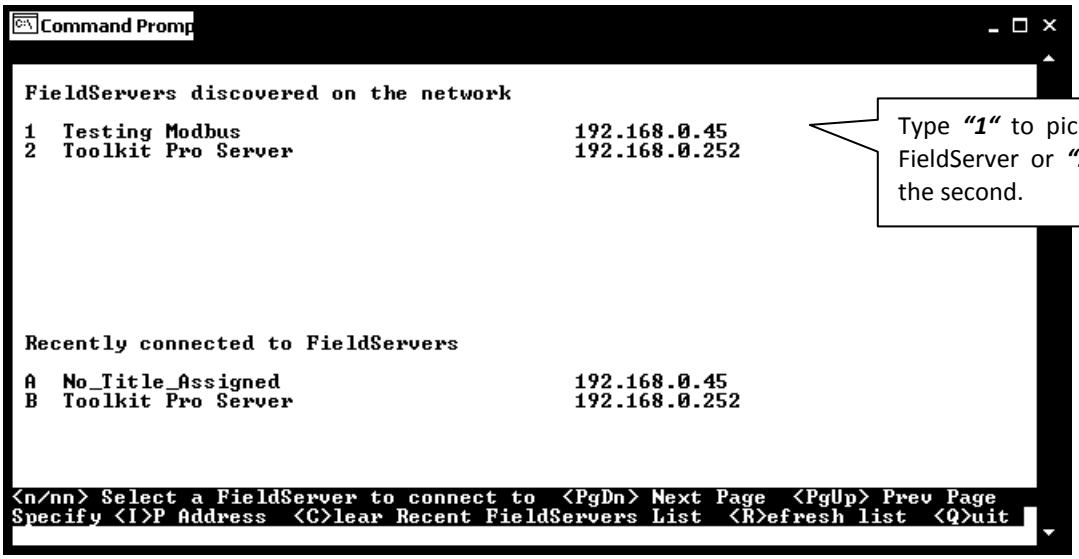


Figure 5.1

If a FieldServer is not selected, RuiNet will wait about 10 seconds and then automatically connect to the FieldServer last connected.

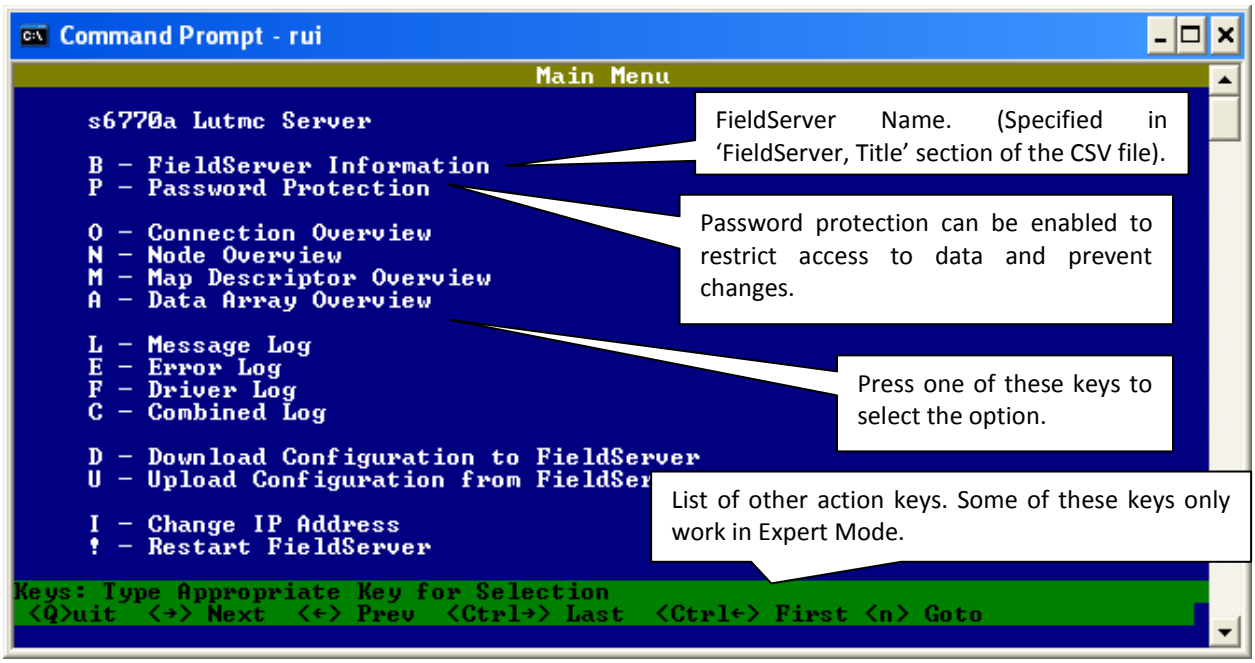


Figure 5.2

5.2 B - FieldServer Information

FieldServers were previously known as bridges, hence the use of the B key to access the screen displaying general information about a FieldServer. The B Screen consists of two aspects (settings and status). Use the spacebar to toggle between the two aspects. The settings screen is depicted below. The various parameters are described more fully in the tables that follow.

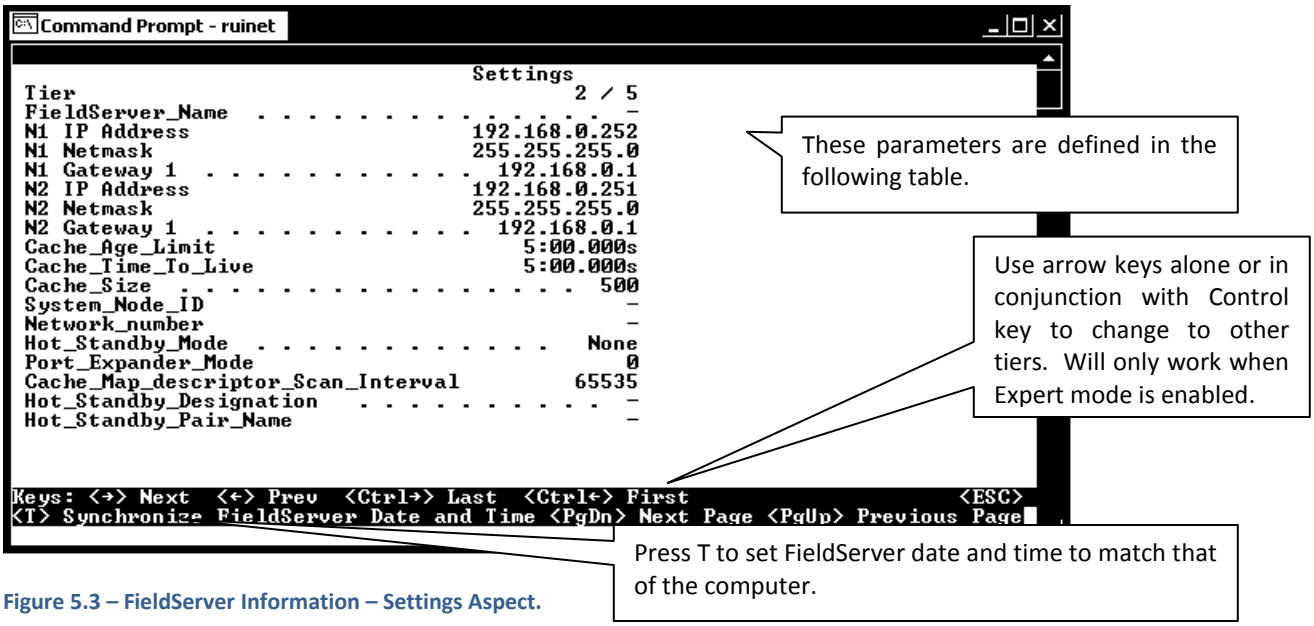


Figure 5.3 – FieldServer Information – Settings Aspect.

5.2.1 FieldServer Information – Settings Aspect

Parameter	Description
Tier	FieldServers have the ability to run as "multiple" FieldServers on one platform. To differentiate between the different running applications, each of the applications is referred to as a Tier with a specific name.
FieldServer_Name	A name by which a FieldServer is identified - need not be unique.
N1 IP Address	The IP address of the N1 Ethernet Adapter.
N1 Netmask	Netmask of N1 Ethernet Adapter.
N1 Gateway 1	The IP address of the gateway that N1 Ethernet messages use if the destination IP is not found on the local network.
N2 IP Address	The IP address of the N2 Ethernet Adapter.
N2 Netmask	Netmask of N2 Ethernet Adapter.
N2 Gateway 1	The IP address of the gateway that N2 Ethernet messages use if the destination IP is not found on the local network.
Cache_Age_Limit	Maximum age of data in a cache Map Descriptor for immediate response to poll. Default 5 minutes. See Appendix C.1.5
Cache_Time_To_Live	The time that the FieldServer maintains the port expanded polling in Port Expander Mode. Default 5 minutes. See Configuration Manual for more information.
Cache_Size	The maximum number of cache Map Descriptors allowed.
System_Node_ID	Use is driver dependent. Generally used to identify the FieldServer as a Node when it is configured as a Server.

Parameter	Description
Network_number	Displayed where a protocol requires the FieldServer to be assigned a network number (e.g. BACnet).
Hot_Standby_Mode	Where specified, this parameter defines the behavior of the standby FieldServer in Hot Standby mode. In Mode 1 the FieldServer is completely passive; in Mode 2 the standby FieldServer polls the connected devices through alternate communication paths. Refer to the Configuration Manual for more information.
Port_Expander_Mode	Indicates whether the port expander mode function is enabled or not.
Cache_Map_Descriptor_Scan_Interval	Default is two seconds. If the value 65535 is displayed, then this is an error and it indicates that there is no setting.
Hot_Standby_Designation	Primary or Secondary. On boot the primary tries to become the active and the secondary tries to become the standby FieldServer. This behavior may be different if the so called secondary FieldServer gets re-booted first.
Hot_Standby_Pair_Name	A name by which a pair of FieldServers configured as a Hot Standby pair is known. When one of a pair boots, it broadcasts a message with its pair name in order to try and locate the other FieldServer that forms the hot standby pair.

5.2.2 FieldServer Information – Status Aspect

Parameter	Description
Tier	FieldServers have the ability to run as "multiple" FieldServers on one platform. To differentiate between the different running applications, each of the applications is referred to as a Tier with a specific name.
Driver Configuration	The part of the FieldServer firmware that contains the drivers ordered. Each combination of drivers is known as a DCC.
DCC Version	A DCC version number is allocated to each DCC. This version increases with changes/updates to drivers. Tech support are able to track a DCC version to determine what features of each driver are available and what bugs may have been present in a particular version.
Kernel Version	The version number of the kernel. The kernel is that part of the firmware that provides support and resources to the individual drivers. Tech support may require this number.
BIOS Version	The version number of the FieldServer's BIOS. This seldom changes.
Data Points Used	Each FieldServer has a combination of drivers and a maximum number of data points that may be managed. A data point is an element of a Data Array with a responsible Map Descriptor. Responsible Map Descriptors are Client side, active and almost always read Map Descriptors. Example: A CSV file configured with a RDBC Map Descriptor with a length of 100 may use 100 data points. If the number of points used exceeds the maximum then the FieldServer will continue to operate for 24 hours and then shutdown. See Enote024 for details on point count.
Data Points Max	The file slots.ini controls the maximum number of data points. Only use slots.ini supplied with the FieldServer or by tech support. The user cannot edit or generate this file. The default for an X40 is 1000 points. The default for an X20 is 500 points. The FieldServer bridge ID must be provided to tech support for them to generate a new slots.ini file.

Parameter	Description
Cycles Now	Number of times the FieldServer executes all its software per second. This number will change continuously.
Cycles Max	The maximum value since the FieldServer started.
Cycles Min	The minimum value since the FieldServer started.
Avg Cycle Time	The average time in milliseconds the software took to complete since last restart. Cycle timers are only started after the CSV files have been loaded.
Min Cycle Time	The minimum/maximum time in milliseconds the software took to complete. Use the R key to reset this value.
Max Cycle Time	
Cache Age Ave[1]	The average data age of the cache blocks currently in existence.
Cache Age Max[1]	The maximum data age of the cache blocks currently in existence
Cache Age Max Ever[1]	The maximum data age of cache blocks that existed on the FieldServer since startup, i.e. the oldest that cache data ever got.
Cache usage (RDB)[1]	The number of active cache blocks reading data at the current time.
Cache usage (WRB)[1]	The number of active cache blocks writing data at the current time.
Memory Blocks	The number of memory blocks reserved by drivers and other system functions.
Last Time Rebooted	The time that the FieldServer was last restarted. New FieldServers are shipped without the time or date set and hence the value shown here is meaningless until the FieldServer time is synchronized with the computer's.

5.3 P - Enable/Disable Password Protection

The P screen allows the administrator to restrict access to data and the ability to make changes. Password protection is disabled by default.

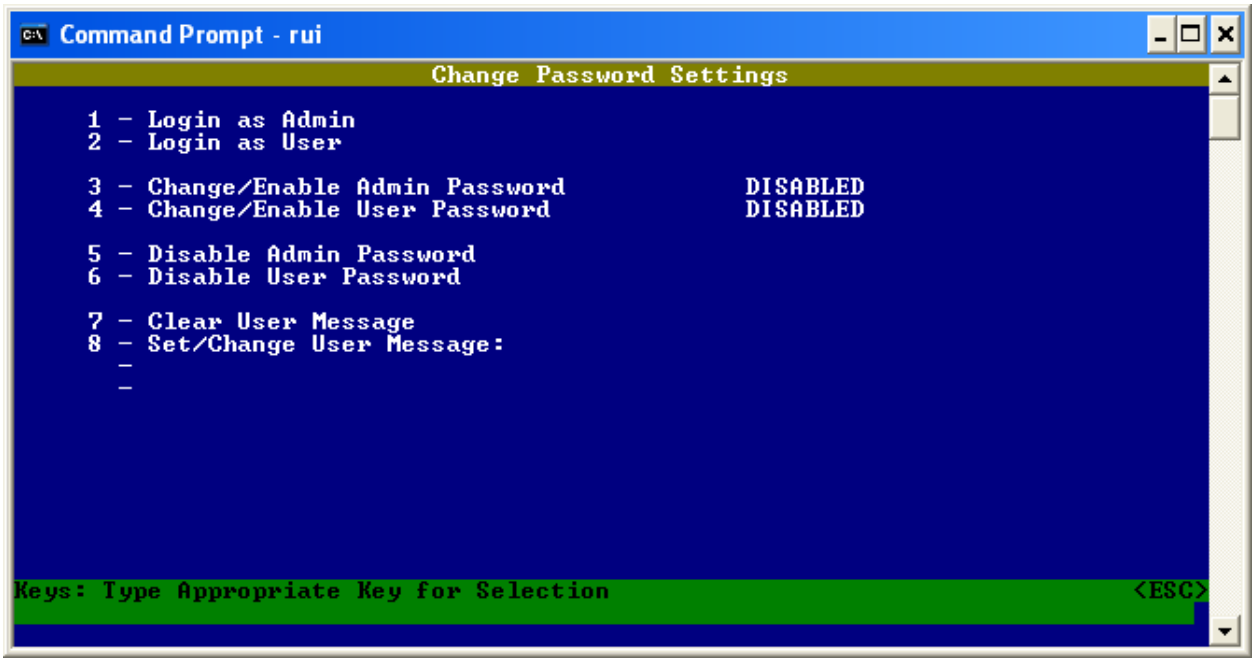


Figure 5.4

[1] Cache blocks are temporary Map Descriptors created by the FieldServer to achieve certain objectives such as a write through or port expansion. Because they are temporary this number rise and fall is not visible. The read cache blocks persist until they expire after the Cache_Age_To_Live time has expired.

Parameter	Description
Login as Admin	Admin privileges allow full access to all screens and changes to information.
Login as User	User privileges allow read only access, i.e. browsing between screens, but no changes can be made.
Change/Enable Admin Password	Administrator password enabled – protects access. Once the protection is enabled, "Password Screen" (Figure 5.5 will be the only screen visible until logged in as User or Admin using options '1' or '2'
Change/Enable User password	Administrator can create user account.
Disable Admin Password	Protection removed by administrator
Disable User Password	Remove the User Account

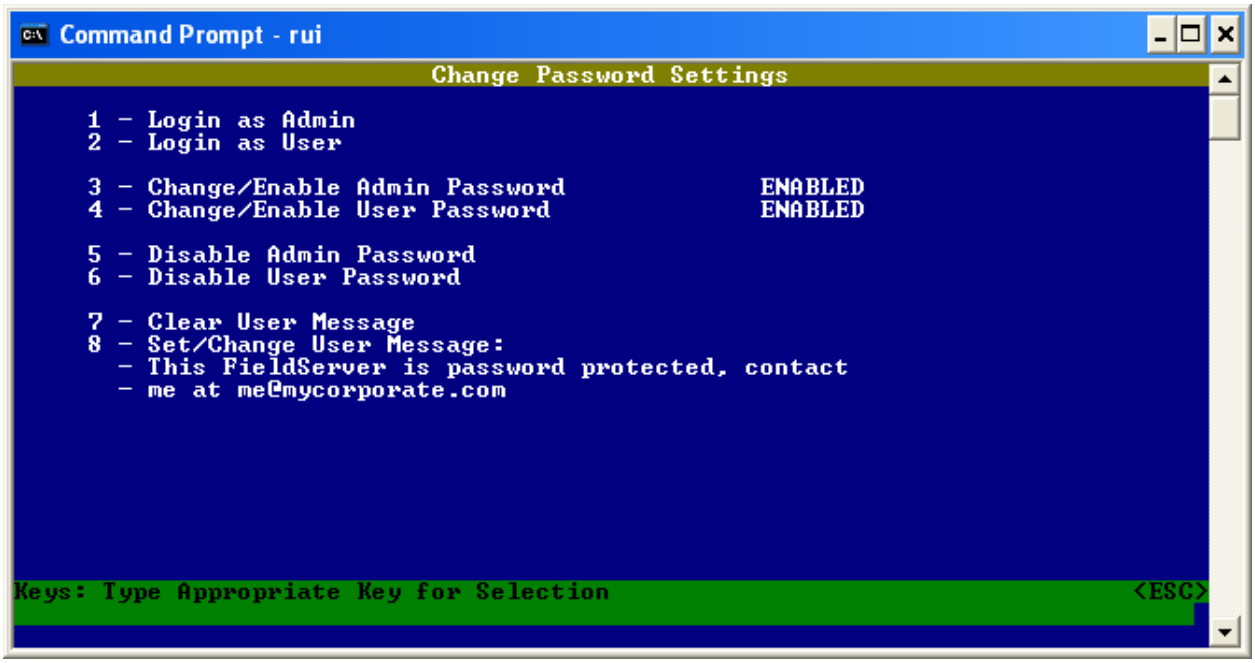


Figure 5.5 – Password Enabled Screen

5.4 O - Connection Overview

This screen supplies information on communication between the FieldServer and remote devices. A number of aspect screens are available, and some of the aspect screens have more than one page. Use the space bar to toggle between aspects and the PgUp, PgDn keys to toggle between pages of the same aspect. The Connection Overview and Settings Aspect screens are depicted below. The various parameters are described more fully in the tables that follow.

Connection number. - Type number to go to detail for that connection, e.g. type 01 to navigate to detail for connection port N1

Every attempt is made to standardize the way in which drivers report stats but in some drivers unavoidable differences exist. In general, however, Rx Msg and Tx Msg correspond closely when connection is healthy.

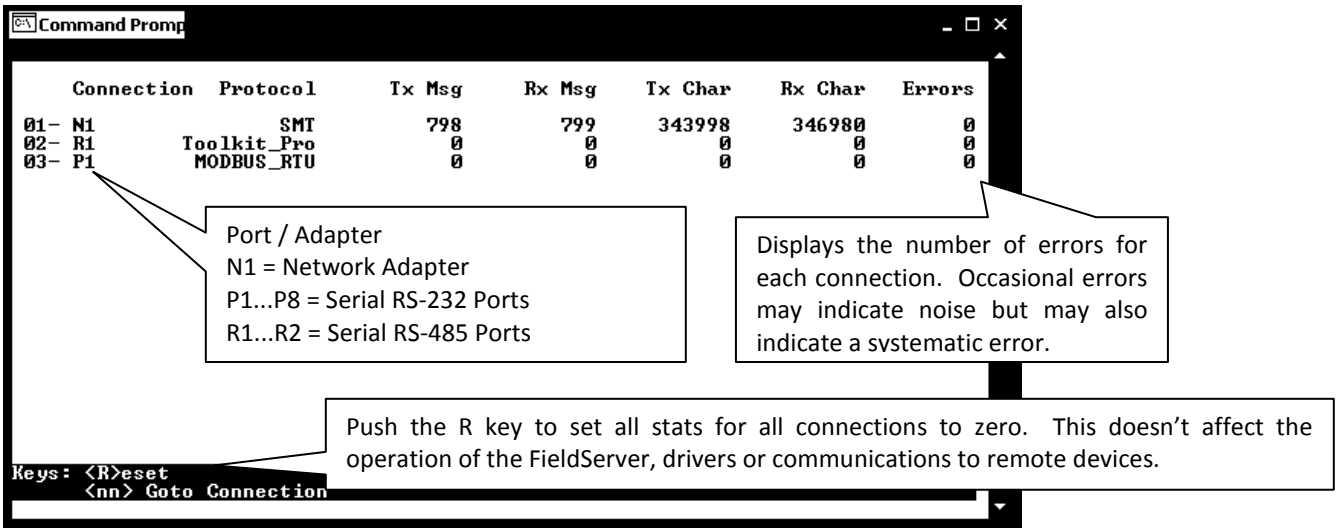


Figure 5.6 – Connection Overview Screen

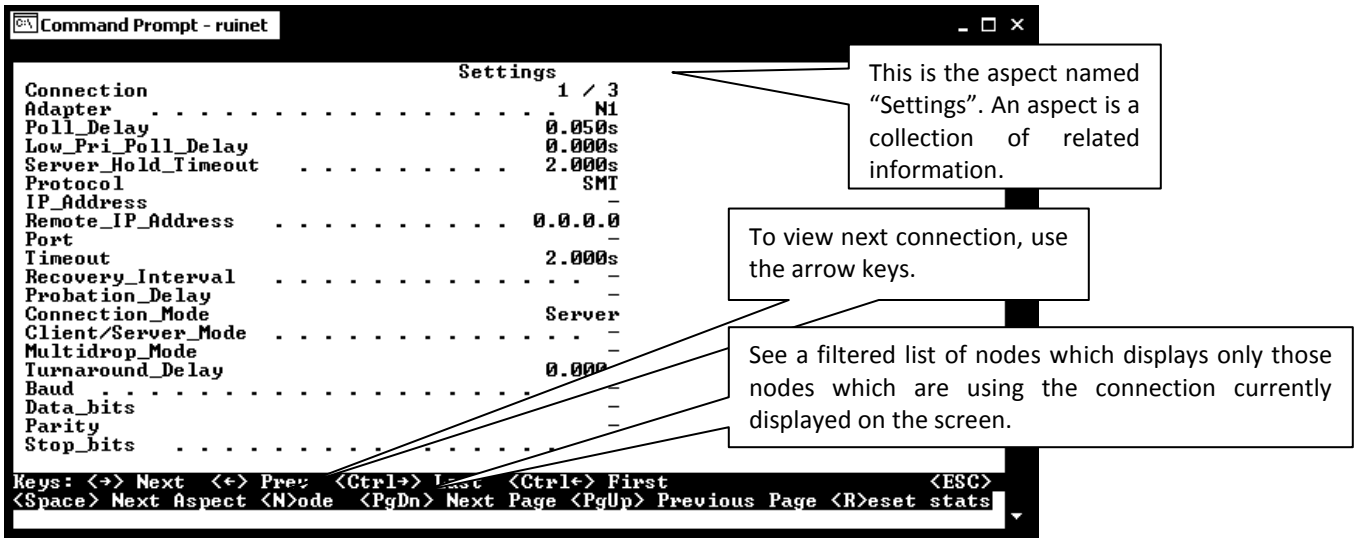


Figure 5.7 – Connection Overview – Settings aspect – Page 1

5.4.1 Connection Overview – Settings Aspect.

Parameter	Description
Connection	The current connection out of the total number of connections.
Adapter	An adapter ID is displayed if the driver is an adapter driver.
Poll_Delay	The minimum amount of time that must pass between one Client Map Descriptor completing its task and the next Client Map Descriptor being serviced. Refer to Driver Manuals.
Low_Pri_Poll_Delay	The poll delay used for lower priority Map Descriptors.
Server_Hold_Timeout	When an upstream device polls the FieldServer, and the data is unavailable or too old, the driver generates a poll to the downstream device for fresh data, (port expansion). The Server_Hold_Timeout defines the time available for this transaction to complete before an error is returned. The default is 2 seconds.
Protocol	The name of the protocol used by this connection.
IP_Address	An IP address for the connection if applicable.

Parameter	Description
Remote_IP_Address	A remote IP address for the connection if applicable.
Port	Identifies the port for a serial driver. P1... P8 or R1...R2
Timeout*	The timeout defined for the connection. Default 2 seconds. See Appendix C.1 for further information.
Recovery_Interval	The time after a node goes off-line before the driver tries to poll the device again. Default 30 seconds.
Probation_Delay	The length of time communication needs to be re-established for before an offline Client node is marked on-line again. Default 1 minute.
Connection_Mode	Server (passive) or client (active).
Client/Server_Mode	Optional setting to force a connection mode. Possible values are Server, PLC, Hot_Standby, Hot_Standby_Data, Client_Only, Diagnostic.
Multidrop_Mode	Indicates whether Multidrop mode is enabled or not. Refer to Appendix C.1.3 for more information.
Turnaround_Delay	The delay that the driver imposes between receiving a poll and sending a response. The default is 5ms for serial drivers.
Baud	Connection baud rate used by serial drivers. Some drivers override this value because the communication protocol allows the baud rate to be changed by the master.
Data_bits	The number of data bits used for serial communication by the connection. The values are 7 or 8.
Parity	The parity mode used by the connection to detect communication errors. Values are even, odd or none. The default is none.
Stop_Bits	The number of stop bits used for serial communication by the connection. The values are 1 or 2.
Line_drive_on	Time a serial driver using RS-485 will wait before driving the communications line after receiving permission. Default is 1milisecond.
Line_drive_off	Time that a communication line using RS-485 is actively driven before being released, after the last bit has been sent. Default 1milisecond..
IC_timeout	Time a driver will wait between receiving the first and second bytes of a message before generating an IC Timeout. Default 0.5 seconds. See Appendix C.1.4 for more information.
IP_port	Determined by specific driver or protocol used. See Driver Manual.
Remote_IP_Port	
Max_Master	
Max_Info_Frames	
Connection_Type	
Application	

5.4.2 Connection Overview - Status Aspect

Parameter	Description
Connection	The current connection being displayed out of the total number.
Timer	The use of this variable to determine timeouts is driver dependent and has no consistent meaning. In passive drivers this variable may not have been used at all. In simple poll response drivers this variable should decrease from its maximum (equal timeout value) to zero.

5.4.3 Connection Overview - Statistics Aspect

Connection statistics are a roll-up of all the statistics maintained by the Nodes and Map Descriptors using the connection. For example: If three Map Descriptors use a single connection then each time a message is sent for each Map Descriptor, the Map Descriptor, Node and connection statistics are all updated. Thus response time statistics are more meaningful when viewing individual nodes and Map Descriptors.

Parameter	Description
Connection	The current connection being displayed out of the total number
Client Read Msg sent	The number of read messages sent by a driver acting as a Client. May include messages sent to connect to the Server.
Client Read Msg recd	The number of responses received to read messages sent by a driver acting as a Client. For most drivers this statistic quals the number of messages sent.
Client Write Msg sent	The number of write messages sent by a driver acting as a Client. May include messages sent to connect to the Server.
Client Write Msg recd	The number of responses received to write messages sent by a driver acting as a Client. For most drivers this statistic equals the number of messages sent.
Client Passthru messages	This statistic relates to port expansion. Messages that are unrecognized (unsupported) are passed through the FieldServer without the contents being considered.
Client Passthru Msg sent	The number of unrecognized messages passed through the FieldServer.
Client Passthru Msg recd	The number responses to unrecognized messages passed through the FieldServer.
Client Broadcast msg	The number of broadcast messages sent.
Client Bytes Sent	Number of bytes contained in messages sent by the driver acting as a Client. May include bytes of messages used to connect or login to the remote device.
Client Bytes Recd	Number of bytes contained in messages received by the driver when acting as a Client - typically responses to messages sent. May include bytes of messages used to connect or login to the remote device.
Server Msg recd	The number of messages received by a driver acting as a Server. May include non-data messages such as connection or login requests and port supervision messages.
Server Msg sent	The number of messages sent by a driver acting as a server - typically responses to messages received. May include responses to non-data messages as above.
Server Bytes Sent	A count of the bytes sent by the driver when acting as a Server in response polls. May include bytes contained in non-data messages.
Server Bytes Recd	A count of the bytes received by the driver when acting as a server. May include bytes contained in non-data messages.
Cache - Hits	The number of times a cache Map Descriptor containing current data has been polled. See Appendix C.1.5
Cache - Misses	The number of times a cache Map Descriptor containing outdated data has been polled. See Appendix C.1.5
Cache - Created	The number of times a cache Map Descriptor containing outdated data has been polled resulting in the creation of a new cache Map Descriptor. See Appendix C.1.5
Cache - Bumped	The number of expired cache blocks. See Appendix C.1.5
Pex Write Thru	Writes are never cached - the external device is put on hold while the FieldServer resends the message to the PLC and waits for the response. When the FieldServer receives the response from the PLC it in turn responds to the external device. This

Parameter	Description
	operation is counted as a PEX Write thru.
Server Response Max	The maximum time that the FieldServer has taken since the last reset to receive a message from an external device, poll the PLC and respond to the external device. The SCADA Hold Timeout parameter should be set higher than this limit
Server Response Avg	This is the average time that the FieldServer has taken since the last reset to receive a message from an external device, poll the PLC and respond to the external device. Reconfigure if response times are slow.
Link Control	This parameter is applied differently in different drivers. Please refer to the appropriate Driver Manuals. The Drivers that currently make use of this paramater are: AB-TCP, GE-SRTP, EST3, Modbus Plus.
Who-Is-Router-To-Network	BACnet specific – see Driver Manual.
I-Am-Router-To-Network	
Reject-Msg-To-Network	
Messages Reconstructed	Metasys®[2] Specific – see Driver Manual.
Unsupported Property	The driver encountered an unsupported property in a message.
Unsolicited Messages Recd	A message was received without the driver polling for it.
Single Write	A write data message containing a single data element.
Single Item Read	A read data message requesting a single data element.
Block Write	A write message containing a block of data elements.
Block Read	A read message requesting a block of data elements.
Sequence Error	Messages containing sequence numbers received out of sequence.
Data Object Startup	Data requested from a node in start-up mode.
Expedite Read	The number of Map Descriptors given the highest priority to complete a read/write first.
Expedite Write	
Fasttrack Read	The number of Map Descriptors given a higher priority to complete a read/write as soon as possible.
Fasttrack Write	
Fasttrack Overrun Read	The number of times the buffer holding fasttrack read/write Map Descriptors overflowed.
Fasttrack Overrun Write	
Max Read Response Time	Maximum/Minimum/Average time in seconds that passed before a response was received to a read message sent by a driver acting as a Client.
Min Read Response Time	
Avg Read Response Time	
Max Write Response Time	Maximum/Minimum/Average time in seconds that passed before a response was received to a write message sent by a driver acting as a client.
Min Write Response Time	
Avg Write Response Time	
Max Passthru Response Time	Maximum/Minimum/Average time in seconds to receive a response to a poll sent using the Passthru mechanism.
Min Passthru Response Time	
Avg Passthru Response Time	
TCP Conn Lost	Number of times a TCP connection was lost or interrupted.
TCP Send Failed	Number of times a message sent on TCP connections failed.

[2] Metasys® is a Trademark of Johnson Controls Inc

5.4.4 Connection Overview - Error Statistics Aspect

Parameter	Description
Connection	The current connection being displayed out of the total number.
PEX No slave	PEX No Slave is intended for a multidrop context where it shows that messages are being ignored because the corresponding slave is not configured on our device. See Driver Manual for further information.
Server Overruns	A message arrived from the upstream device while the server port was on hold. Increase the timeout setting in the external device.
Server Hold Timeouts	If an upstream device requests data from a cache where the data is too old then the data will be refreshed by polling from the downstream device. The response was not received in time.
Timeouts	A remote device never responded to the FieldServer poll. Either the device is not responding, or one of the timeout parameters is set too low. Ensure that the device is online and addressed correctly and if necessary increase the relevant timeout parameter in the CSV file.
Checksum Errors	External influences e.g. electrical noise corrupted the data. Check that communication cables are shielded, not too long and do not run past power cables.
Protocol Errors	An external device responded with unexpected or unknown messages. Consult the relevant driver manual.
Noise	Corrupted or garbage bytes on a communications line.
Bad Length	A message that looks Ok, but is of the incorrect length.
Bad Node	A no-response/error response from an addressable external device.
Bad Function	E.g. an external device is written to that does not support writes.
No Start	Communications to an external device could not be started.
PLC exception	A variation of a NAK message produced by some protocols.
NAK	A NAK message is received in response to a poll.
Streaming	Data seems to be continuously arriving from an external device. This could be due to a mismatch in baud rates. Streaming errors are typically produced when: 1) The transmitter sends more data than the FieldServer can process. 2) An unexpectedly long message is received or messages have been corrupted so that the end of a previous message cannot be detected. 3) A message longer than the driver expects has been received.
Premature	A response from an external device arrived before it was expected, implying that it is not the response for the poll the FieldServer has just sent. This could occur when there are time delays in communication networks which contain bridges and routers which may delay messages for longer than expected. Increase the timeout parameter for the connection to eliminate these errors.
Preamble	Characters preceding a message were dropped.
IC Timeouts	Too much time between receiving successive bytes in a message.
Address Errors	A driver tried to address a wrong device or a wrong block of data within an external device.
Data Object offline	A poll was received for a node that is offline.
Node Offline	An external device node is offline in response to a driver or external device trying to access it.

Parameter	Description
Msg Ignored	Messages received but unable to be processed - normally because the driver has not implemented functionality for that message.
Sys Cleared	The number of times that Data Arrays were cleared after a system-normal type message was received from a device (e.g. fire alarm panel)
Squelch TX[3]	The number of bytes received during the squelch timing period started when RTS is asserted.
Squelch RX[1]	The number of bytes received during the squelch timing period started when RTS is de-asserted.
Cache Failed	FieldServer unable to create a cache block due to memory shortage or inability to find a downstream node.
Segmentation Not Supported	The received message was segmented but the driver does not support re-assembling segmented messages.
Passthru Retries[4]	Produced when a Passthru poll is busy on the downstream side, and an identical poll (retry) is received on the upstream side.
Passthru Overruns[2]	Produced when a Passthru poll is busy on the downstream side and a different poll (not a retry) is received on the upstream side.
Passthru Early Retries	A Passthru Retry, which occurs when the upstream retry is received before the 1st downstream Passthru poll has been sent.
Passthru Normal Retries	A Passthru Retry, which occurs when the upstream retry is received after the 1st downstream Passthru poll has been sent.
Passthru Early Overruns	A Passthru Overrun, which occurs when the upstream retry is received before the 1st downstream Passthru poll has been sent.
Passthru Normal Overruns	A Passthru Overrun, which occurs when the upstream retry is received after the 1st downstream Passthru poll has been sent.
Passthru Early Overruns Fails	This statistic is currently not used.
Passthru Normal Overruns Fails	
PWT Expired	The Passthru Window Timer expired before the downstream Passthru poll could be sent.

5.4.5 Connection Overview - API Aspect

These statistics are produced by the FieldServer kernel's API (Application Programmer Interfaces). They are intended for advanced users only.

Press the 1 key or the 2 key to change the filter on these stats.

1 = TCP API

2 = Ethernet API (Default)

Parameter	Description
Connection	The current connection being displayed out of the total number
TX bytes	Number of bytes sent.

[3] Information is available in ENOTE19. Enotes are available on the flash drive shipped with the FieldServer.
[4] Currently only produced by the Metasys® driver

Parameter	Description
TX packets	Number of packets sent.
TX dropped	Number of packets not sent for reasons that include the transmit buffers being full.
TX errors	Errors during transmission such as hardware errors.
RX IP fragments	The number of IP fragmented packets received.
Collisions	Number of collisions - the network is too busy.
TX abort errors	The NIC layer reports this error. Should be zero.
TX carrier errors	
TX heartbeat errors	
TX window errors	
Max TX buffers used	The highest value for the used transmit buffer count since the FieldServer was restarted. An extremely busy network may have a max of 7 or 8 but values this high are unusual.
Tx buffers in use	Current transmit buffers in use.
Tx broadcast	The number of broadcast messages sent.
RX bytes	Number of bytes received by the API.
RX pkts total	Number of packets received by the API.
RX pkts for us	The number of received packets destined for transmission.
RX pkts NOT for us	The number of received packets destined for transmission not meant for the API.
RX dropped	The number of received packets which were dropped for reasons such as the Ethernet queue being full.
RX errors	The NIC layer reports this error. Should be zero.
RX length errors	
RX overflow errors	
RX crc errors	
RX frame errors	
RX frame errors	
RX buffer full	
RX ring buffer error	
RX IP type	Number of messages using IP protocol received.
RX ARP type	Number of messages using ARP protocol received.
RX BACNET type	Number of messages using BACnet protocol received.
RX 802_3 type	Number of messages using 802_3 Ethernet protocol received.
RX UNKNOWN type	Number of messages using other Ethernet protocol received.
Frag buf overrun	IP defrag on the receive side. The buffer is (about 80kb) is full.
RX Broadcast	Number of packets received in broadcast.
Exception 1	The Ethernet packet type could not be identified
Exception 2	An Ethernet packet received (i.e. addressed to the FieldServer Ethernet address) was addressed to another IP address - either an Ethernet broadcast was sent with a specific IP address, or another device has an incorrect ARP table.
Exception 3	A UDP packet of length >1500 was received which cannot be handled by the FieldServer.
Exception 4	The Ethernet TX interrupt handler was kick started.
RX global timeout	This statisticis currently not used.
RX local timeout	
RX IP not for us	The number of received IP packets not meant for the FieldServer.
Max RX buffers used	The maximum number of receive buffers used since restart.

Parameter	Description
Rx buffers in use	The number of receive buffers currently in use.

5.5 N - Node Overview

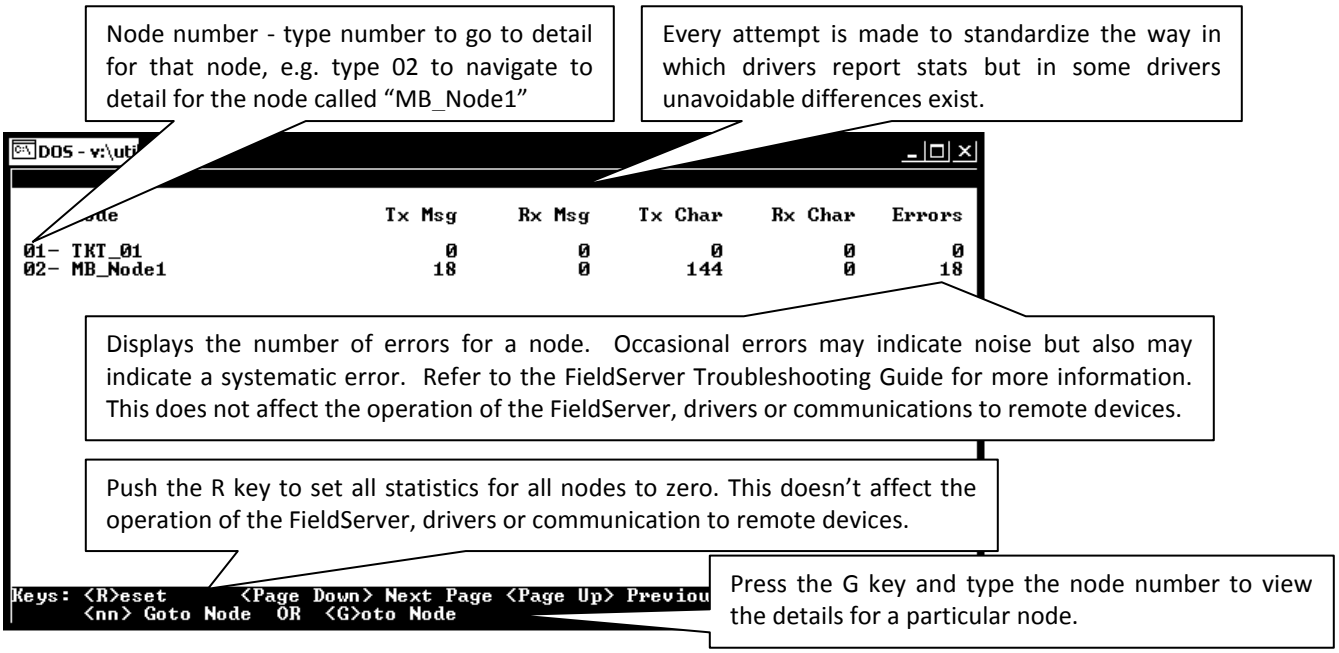


Figure 5.8 – Node Overview Screen.

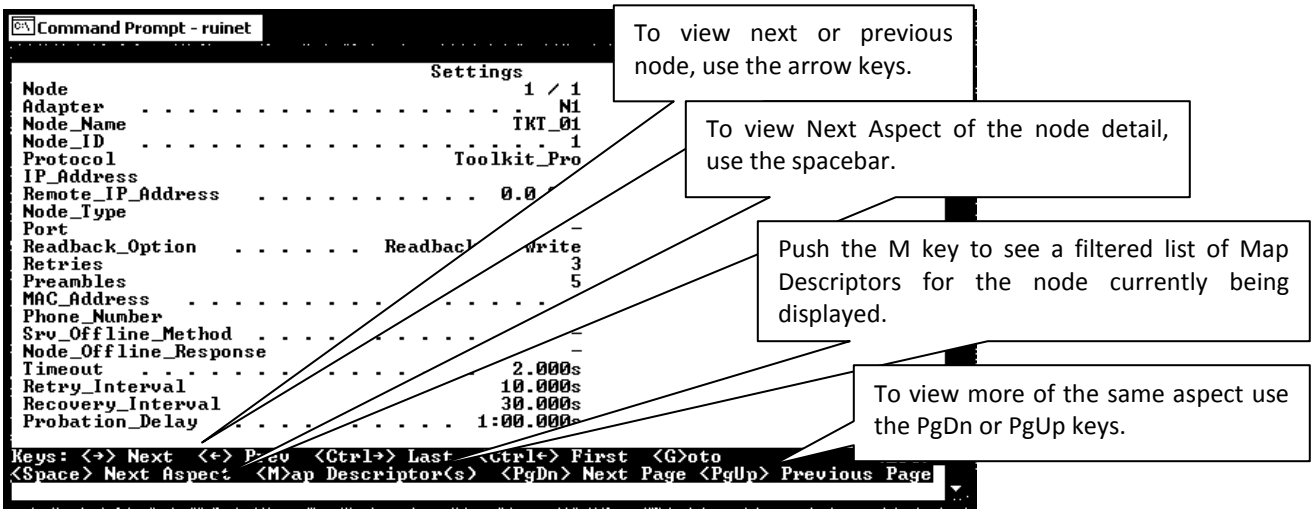


Figure 5.9 – Node Overview – Settings Aspect Page 1

5.5.1 Node Overview - Settings Aspect

Parameter	Description
Node	The current node being displayed out of the total.
Adapter	The adapter name where a node is connected to an adapter.
Node_Name	The node name specified in the CSV file.
Node_ID	The station number or address of the node. The actual meaning is dependent on the driver and protocol – refer to the Driver Manual.

Parameter	Description
Protocol	The protocol being used to update the data for that node. Refer to the Driver Manual
IP_Address	The IP address used by this node.
Remote_IP_Address	The remote IP address used by this node
Node_Type	Specified in the configuration file as the PLC_Type. - Consult the driver manual for additional information.
Port	Port number for a serial connection.
Readback_Option	After a write has been executed in a write through operation, the FieldServer schedules the read Map Descriptor to poll again in order to immediately read back the data that was written. The default is 'Readback_on_Write". Other options are "None" and "Expire_Current_Data"
Retries	Tells the driver how many times to retry a poll before considering the node to be offline. The default is 3.
Preambles	Counts data bytes received before a valid message, but not forming part of a valid message, e.g. a message fragment.
MAC_Address	Currently this field is not used. It is intended to allow drivers to resolve an IP address by giving a MAC address.
Phone_Number	This field is intended for modem support. Currently disabled.
Srv_Offline_Method	The method used on the node to decide if it is to be considered offline.
Node_Offline_response	The type of response the Server side of the driver sends when it finds the Server node to be offline.
Timeout	The timeout specified for the node. Refer to Appendix C.1.
Retry_Interval	The amount of time in seconds that the driver should wait before retrying a poll after a timeout has occurred
Recovery_Interval	The time in seconds after a node goes off-line before the driver tries to poll the device again. – default 30seconds.
Probation_Delay	The length of time communication needs to be re-established for before an offline Client node is marked on-line again. Default 1 minute.
Network_Number	Network station number used on this node.
Server_Name	An alternate to specifying the IP address. Typically used when the user wants two nodes to talk to each other. When specified, the FieldServer sends out a broadcast with the server name and uses the reply to fill in the IP address for the node. Until the reply has been received all polling for the node is disabled. The server name given should correspond to the pair_name specified in the remote FieldServer's bridge settings.
Alias_Node_ID	This is used to distinguish between different nodes connected to the FieldServer when a PLC does not support the allocation of different None_ID's. Each node is given a different alias. Upstream devices poll the Alias_Node_ID and the FieldServer routes the poll to the correct PLC which is polled using the Node_ID.
Ports_on_PLC	For hot standby operation. This field is used to control which port on a PLC to poll.

5.5.2 Node Overview - Status Aspect

Parameter	Description
Node	The current node being displayed.
Node Status	For a Client node. Online, offline, disabled or probation. Probation means that the node is in transition from offline to online. The node was offline, a poll has succeeded but the probation timeout has not expired so the node has not been returned to online yet. If the node isn't a

Parameter	Description
	client node then it is reported as server.
Node Mode	Client or Server
Retry State	The state number of the node retry state engine – for FieldServer developers.
Used Retries	The total number of retries since start-up
Recoveries	The number of times that the driver has gone from offline to online.
Active R/W on Startup	This is an important indication if the driver appears not to be polling. Displays yes or no. If a Map Descriptor with function = "ARS" (Active Read on Startup) is found then this field reports as 'yes'. ARS Map Descriptors are scheduled to occur when a node is still offline and are only used once. They are intended to establish a connection or to log into a remote device. No other Map Descriptors are polled until the ARS Map Descriptors complete normally.

5.5.3 Node Overview - Operating Statistics Aspect

Node statistics are a roll-up of all the statistics maintained by the Map Descriptors which belong to the node. For example, if three Map Descriptors belong to a single node, then each time a message is sent for each Map Descriptor, the statistics for the Map Descriptor, the node and the connection are updated.

Parameter	Description
Node	The current node being displayed.
Client Read Msg sent	Refer to Section 5.4.1 for a description. On this screen the statistic count applies to the node only.
Client Read Msg recd	
Client Write Msg sent	
Client Write Msg recd	
Client Passthru Msg sent	
Client Passthru Msg recd	
Client Broadcast msg	
Client Bytes Sent	
Client Bytes Recd	
Server Msg recd	
Server Msg sent	
Server Bytes Sent	
Server Bytes Recd	
Cache - Hits	
Cache - Misses	
Cache - Created	
Cache - Bumped	
PEX Write thru	
Server Response Max	
Server Response Avg	
Link Control	
Messages Reconstructed	
Unsupported Property	
Unsolicited Messages Recd	
Single Write	
Single Item Read	
Block Write	
Block Read	

Parameter	Description
Sequence Error	
Data Object Startup	
Expedite Read	
Expedite Write	
Fasttrack Read	
Fasttrack Write	
Fasttrack Overrun Read	
Fasttrack Overrun Write	
Max Read Response Time	
Min Read Response Time	
Avg Read Response Time	
Max Write Response Time	
Min Write Response Time	
Avg Write Response Time	
Max Passthru Response Time	
Min Passthru Response Time	
Avg Passthru Response Time	
TCP Conn Lost	
TCP Send Failed	

5.5.4 Node Overview - Error Statistics Aspect

Parameter	Description
Node	The current node being displayed.
PEX No slave	Refer to Section 5.4.3 for a description. On this screen the count applies to the node only.
Server Overruns	
Server Hold Timeouts	
Timeouts	
Checksum Errors	
Protocol Errors	
Noise	
Bad Length	
Bad Node	
Bad Function	
No Start	
PLC exception	
NAK	
Streaming	
Premature	
Preamble	
IC Timeouts	
Address Errors	
Data Object offline	
Node Offline	
Msg Ignored	
Sys Cleared	

Parameter	Description
Squelch TX	
Squelch RX	
Segmentation Not Supported	
Passthru Retries	
Passthru Overruns	
Passthru Early Retries	
Passthru Normal Retries	
Passthru Early Overruns	
Passthru Normal Overruns	
Passthru Early Overrun Fails	
Passthru Normal Overrun Fails	
PWT Expired	

5.6 M – Map Descriptor Overview

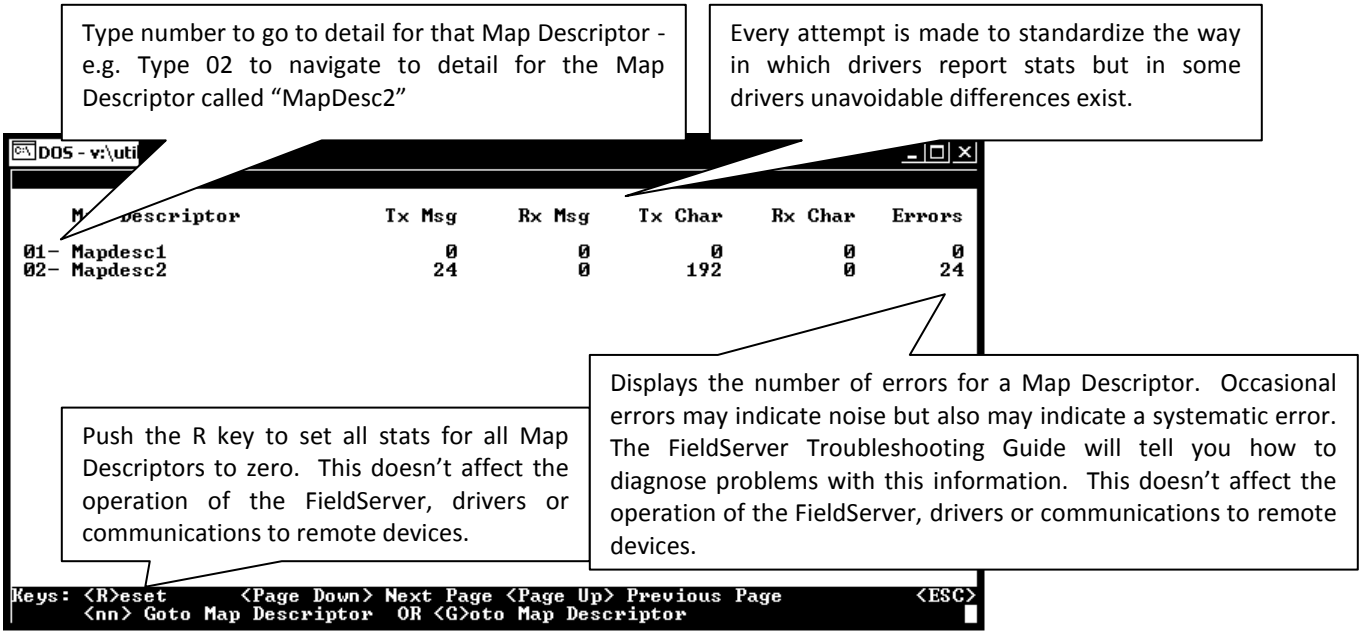


Figure 5.10 – Map Descriptor Overview Screen

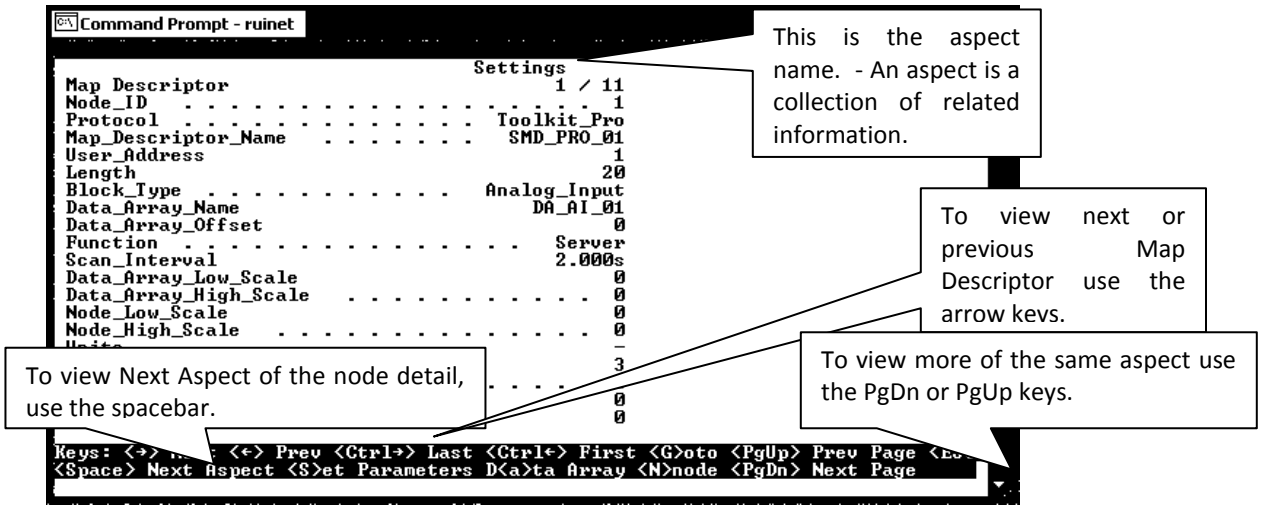


Figure 5.11 – Map Descriptor Overview – Settings Aspect

5.6.1 Map Descriptor Overview - Settings Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total
Node_ID	The Node ID used by this Map Descriptor when the driver builds read or write messages..
Protocol	The protocol used by this Map Descriptor.
Map_Descriptor_Name	Used to identify a Map Descriptor by name.
User_Address	Allows a Map Descriptor to address remote device data at a specific start memory location.
Length	Alows a Map Descriptor address a number of remote device data locations from the start address.
Block_Type	Used by some drivers to indicate the data format used to pack a block of data, e.g. when reading a block of 4 bytes from a PLC and the Block_Type is Word, the incoming data will be interpreted as 2 words.
Data_Array_Name	The name of the Data Array where information will be stored to and retrieved from by the Map Descriptor.
Data_Array_Offset	The offset into the Data Array where data should be stored on reads or retrieved from on writes.
Function	The Map Descriptor function can be mainly read or write with a number of variations of each. Refer to the FieldServer Configuration Manual for other functions.
Scan_Interval	When using continuous Map Descriptor functions such as RDBC, this is the time a Map Descriptor will wait before polling for data again.
Data_Array_Low_Scale	Used in the scaling of data values before storing them or before sending them in write messages. Consult the Driver manual to determine whether the driver supports scaling.
Data_Array_High_Scale	
Node_Low_Scale	
Node_High_Scale	
Units	Used to specify engineering units to interpret data if used. Will display a dash if not used.
Network	Used by some drivers as a network number.
Sector	Used by some drivers as a sector number for rack addressing.

Panel	Used by some drivers as a panel number for rack addressing.
Card	Used by some drivers as a card number for rack addressing.

5.6.2 Map Descriptor Overview – Status Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total.
Scan Timer	Value of scan timer used by driver for this Map Descriptor. This is for debugging purposes only and has no specific meaning.

5.6.3 Map Descriptor Overview - Operating Statistics Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total.
Client Read Msg sent	Refer to Section 5.4.1 for a description. On this screen the statistic count applies to the Map Descriptor only.
Client Read Msg recd	
Client Write Msg sent	
Client Write Msg recd	
Client Passthru Msg sent	
Client Passthru Msg recd	
Client Broadcast msg	
Client Bytes Sent	
Client Bytes Recd	
Server Msg recd	
Server Msg sent	
Server Bytes Sent	
Server Bytes Recd	
Cache - Hits	
Cache - Misses	
Cache - Created	
Cache - Bumped	
PEX Write thru	
Server Response Max	
Server Response Avg	
Link Control	
Messages Reconstructed	
Unsupported Property	
Unsolicited Messages Recd	
Single Write	
Single Item Read	
Block Write	
Block Read	
Sequence Error	
Data Object Startup	
Expedite Read	
Expedite Write	
Fasttrack Read	
Fasttrack Write	

Parameter	Description
Fasttrack Overrun Read	
Fasttrack Overrun Write	
Max Read Response Time	
Min Read Response Time	
Avg Read Response Time	
Max Write Response Time	
Min Write Response Time	
Avg Write Response Time	
Max Passthru Response Time	
Min Passthru Response Time	
Avg Passthru Response Time	
TCP Conn lost	
TCP Send Failed	

5.6.4 Map Descriptor Overview - Error Statistics Aspect

Parameter	Description
Map Descriptor	The current Map Descriptor being displayed out of the total number of Map Descriptors.
PEX No slave	Refer to Section 5.4.3 for a description. On this screen the statistic count applies to the Map Descriptor only.
Server Overruns	
Server Hold Timeouts	
Timeouts	
Checksum Errors	
Protocol Errors	
Noise	
Bad Length	
Bad Node	
Bad Function	
No Start	
PLC exception	
NAK	
Streaming	
Premature	
Preamble	
IC Timeouts	
Address Errors	
Data Object offline	
Node Offline	
Msg Ignored	
Sys Cleared	
Squelch TX	
Squelch RX	
Segmentation Not Supported	
Passthru Retries	
Passthru Overruns	

Parameter	Description
Passthru Early Retries	
Passthru Normal Retries	
Passthru Early Overruns	
Passthru Normal Overruns	
Passthru Early Overruns Fails	
Passthru Normal Overruns Fails	
PWT Expired	

5.7 A – Data Array Overview

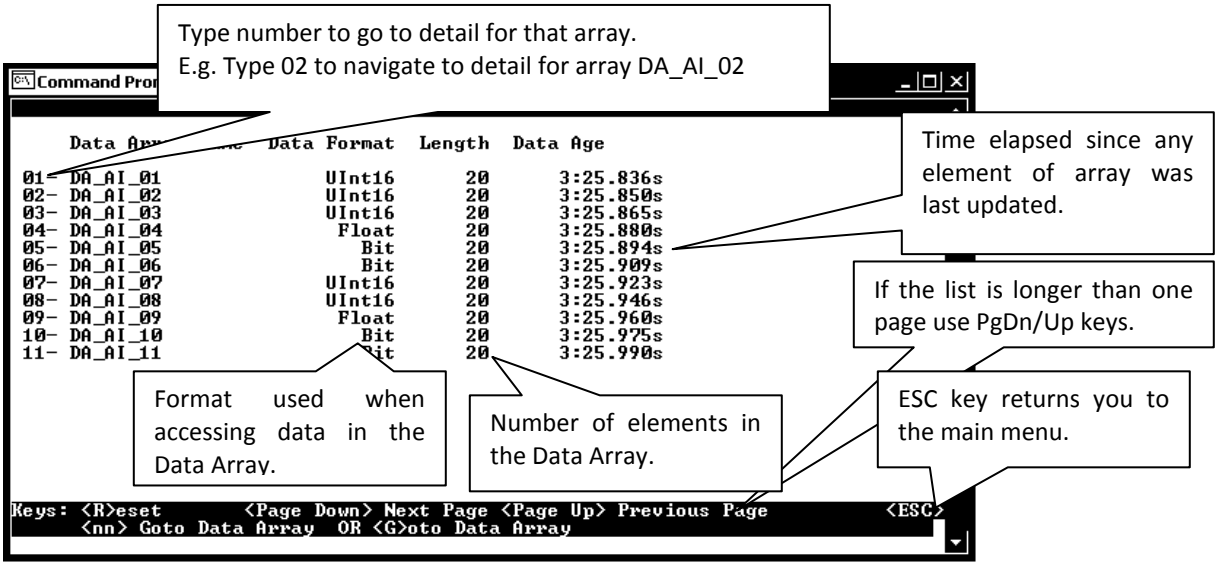


Figure 5.12 – Data Array Overview Screen

5.7.1 Data Array Detail Screen

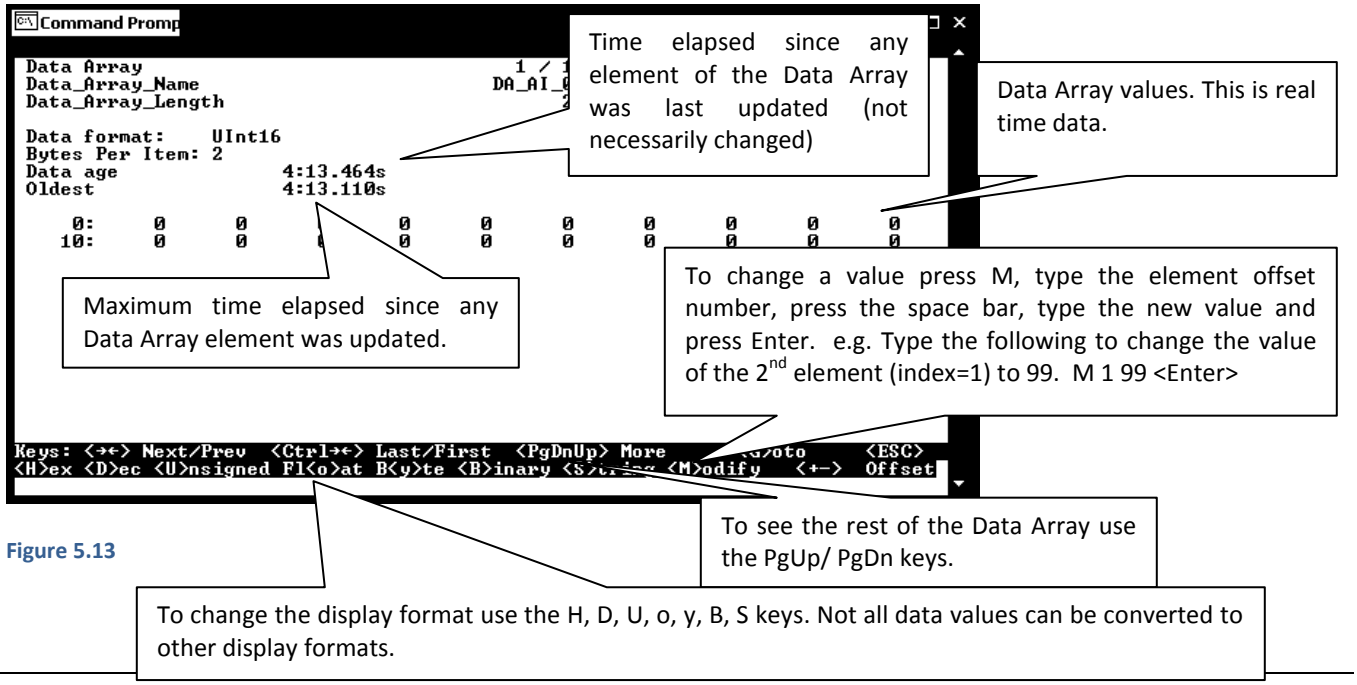


Figure 5.13

5.8 L – Message Log

This screen shows normal informational type messages from the operating system

Error! Objects cannot be created from editing field codes.

Figure 5.14

5.9 E – Error Messages

Other messages – non critical but may assist in troubleshooting.

The error screen will show error messages printed by the FieldServer kernel which should normally have a SDO meaning. Driver messages for older drivers (pre 2003) also send their messages to the E screen. The error screen is a circular buffer which can hold a limited number of lines of information. Once full, as each new line is added to the bottom of the buffer, the top line is removed.

Kernel Messages - informational messages for FieldServer use, normally non critical.

Error! Objects cannot be created from editing field codes.

Figure 5.15

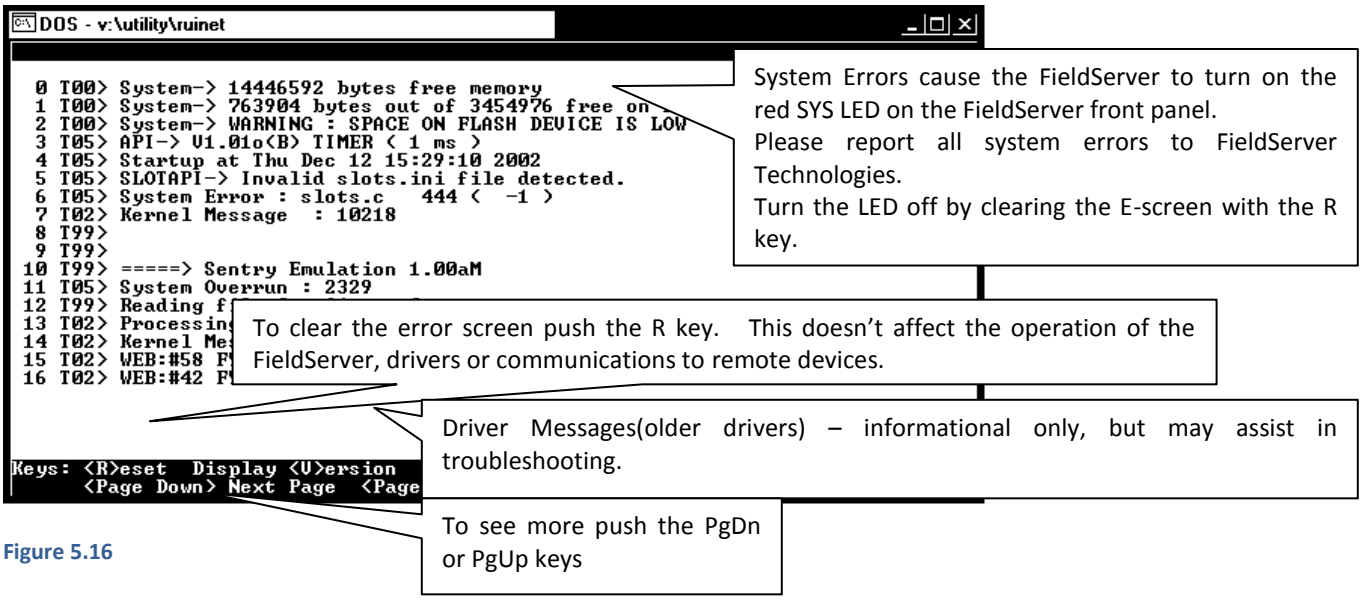


Figure 5.16

5.10 F – Driver Messages

Informational type messages from protocol drivers, eg. Modbus or BACnet are printed on this screen.

Error! Objects cannot be created from editing field codes.

Figure 5.17

5.11 C – Combined Log

All messages from the E, F and L screens are displayed on this screen

Error! Objects cannot be created from editing field codes.

Figure 5.18

5.12 D – Download Configuration to FieldServer

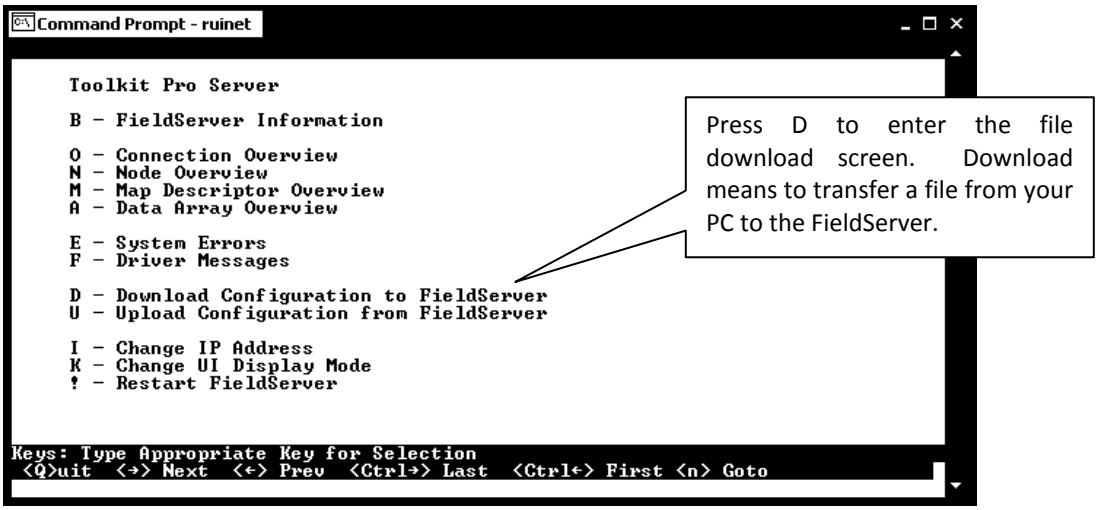


Figure 5.19

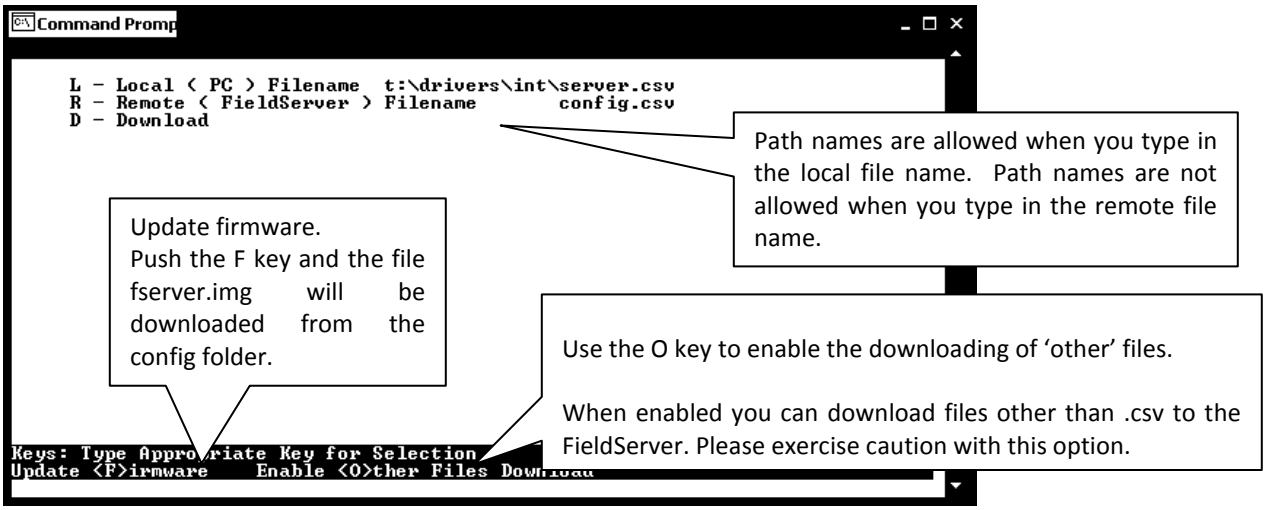


Figure 5.20

5.12.1 Procedure for Downloading a File from a PC to a FieldServer

- From the main menu select ***"D"*** to go to the download screen.
- Specify the local file name by selecting ***"L"***, type the file name and press ***<Enter>***
- If the remote filename is incorrect, select ***"R"***, type the file name and press ***<Enter>.***
- Begin the download by selecting ***"D".***

5.13 U - Upload Configuration from FieldServer

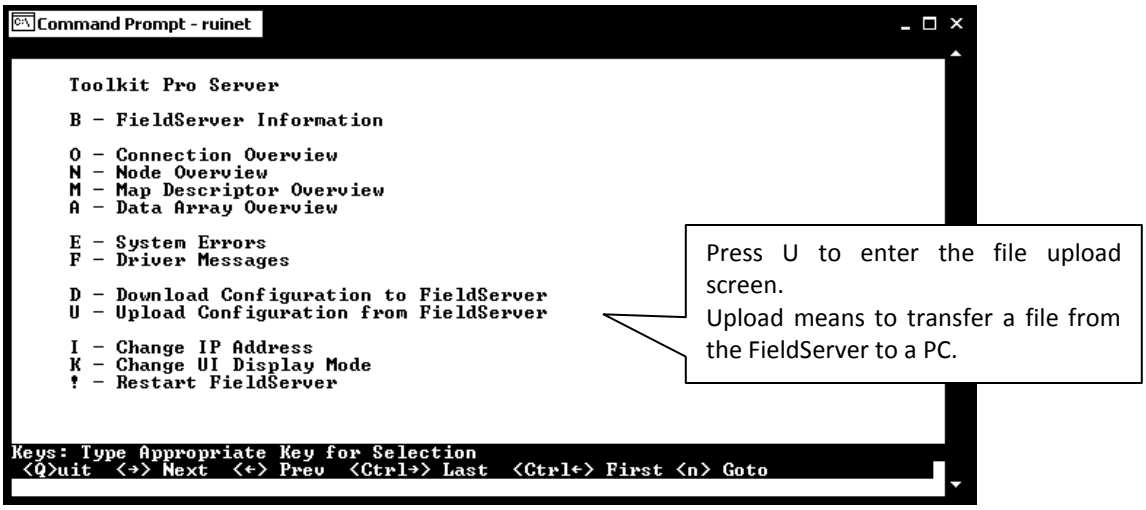


Figure 5.21

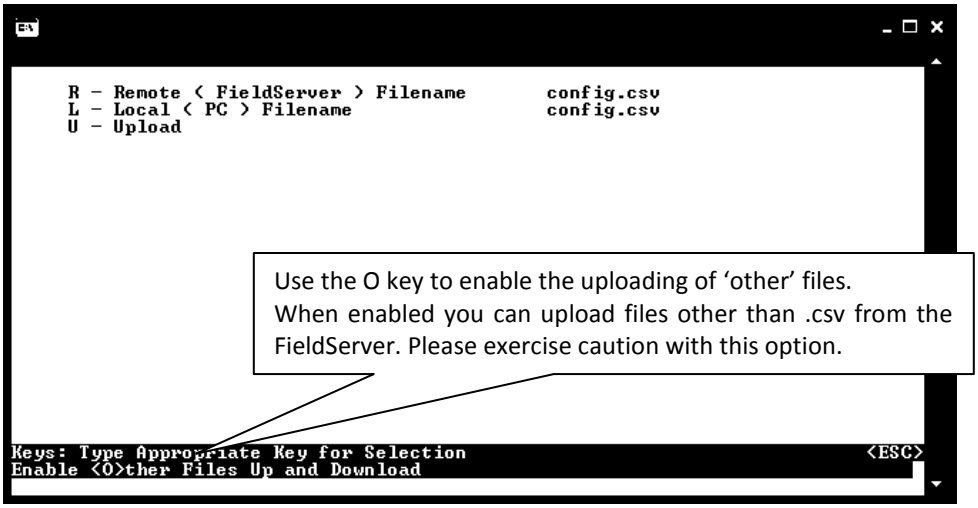


Figure 5.22

5.13.1 Procedure to Upload a File from a FieldServer to a PC

- From the main menu, select ***"U"*** to go to the Upload screen.
- Select ***"R"***, type in the remote file name (File on the FieldServer) and press ***<Enter>***
- If the local file name is incorrect, select ***"L"***, type the name and press ***<Enter>.***
- Begin the upload by pressing ***"U".***
- When the upload is completed, the uploaded file may be opened with one of the listed editors. Note that the editors are not supplied with RuiNet and must be loaded on your machine to work effectively.
- See Appendix B.2.10 for more information

5.14 I - Change IP Address

From the main menu, press ***"I"*** to enter the Edit IP Address Settings menu.

5.14.1 Procedure to Change a FieldServer's IP Address

- Press ***"1"*** to modify the IP address of the N1 Ethernet adapter or ***"6"*** to modify the IP address of the N2 Ethernet adapter (if available on the FieldServer).
- Type in a new IP address in the format 192.168.2.X[5] and press ***<Enter>.***
- If necessary, press***"2"*** or ***"7"*** and change the netmask.
- Restart the FieldServer.

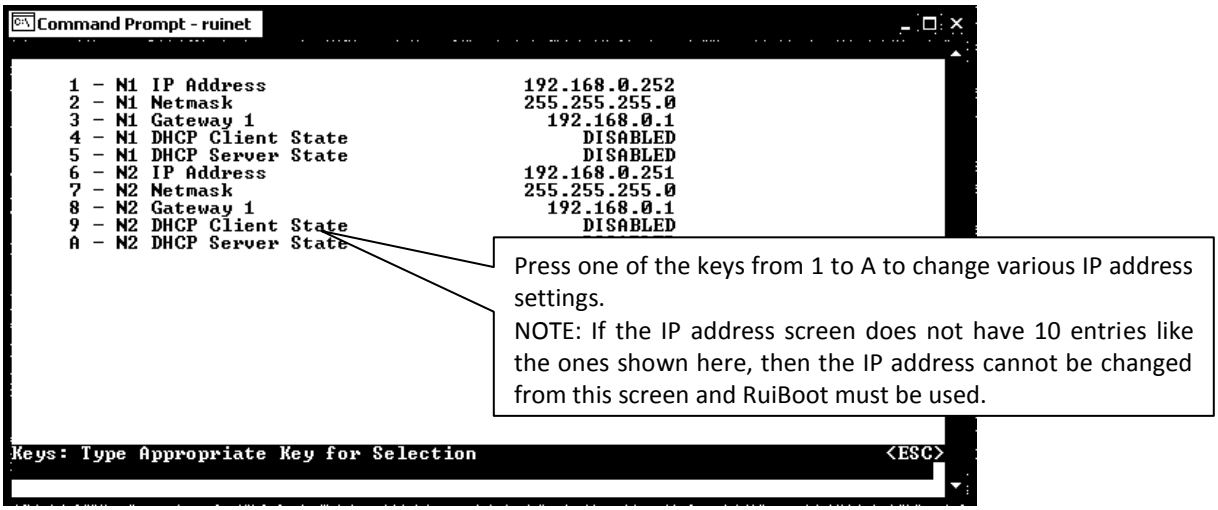


Figure 5.23

Common IP address ranges and netmasks:
Class A – IP (1.0.0.1 to 126.255.255.254), Netmask (255.0.0.0)
Class B – IP (128.0.0.1 to 191.255.255.254), Netmask (255.255.0.0)
Class C – IP (192.0.0.1 to 223.255.255.254), Netmask (255.255.255.0)

Do not use broadcast IP address ending on 255. This IP address is reserved as a target IP address when a device wants to send a message to all other devices on the network.

5.14.2 Obtain the IP address using the DHCP Client

The FieldServer can obtain its IP address from a DHCP Server using the built-in DHCP Client. The network's DHCP Server must be setup correctly for this to work. Please contact your network administrator for assistance.

- Enable the DHCP Client on Ethernet port N1 and/or N2 by pressing ***"4"*** or ***"9"***
- Press ***"Y" <Enter>*** to enable or ***"Esc"*** to escape. ***"N"*** will disable the DHCP client
- Press any key to continue and escape back to the main menu. Press***"!"*** to restart the FieldServer
- Press ***"Q"*** twice to quit from RuiNet and the discovery screen
- Run RuiNet again to connect to the FieldServer

If the FieldServer cannot contact a DHCP Server, the FieldServer will use the last good IP address it was configured with.

[5] "X" is a number that is specific to a specific FieldServer

5.14.3 Obtain the IP address using the FieldServer's DHCP Server

The FieldServer can be used as a DHCP Server to allocate an IP address to a PC or Laptop that you want to use to connect with RuiNet to the FieldServer.

- Enable the DHCP Server on Ethernet port N1 and / or N2 by pressing ***"5"*** or ***"A"***
- Press ***"Y" <Enter>*** to enable or ***"Esc"*** to escape. ***"N"*** will disable the DHCP Server
- Press any key to continue and escape back to the main menu. Press***"!"*** to restart the FieldServer

When connecting a PC or Laptop to the FieldServer on port N1 or N2, be sure to enable the DHCP Client function on the PC or Laptop. The following screen-shot from the Network Connection's properties on a Windows XP PC shows the TCP/IP settings needed to enable the PC's DHCP Client:

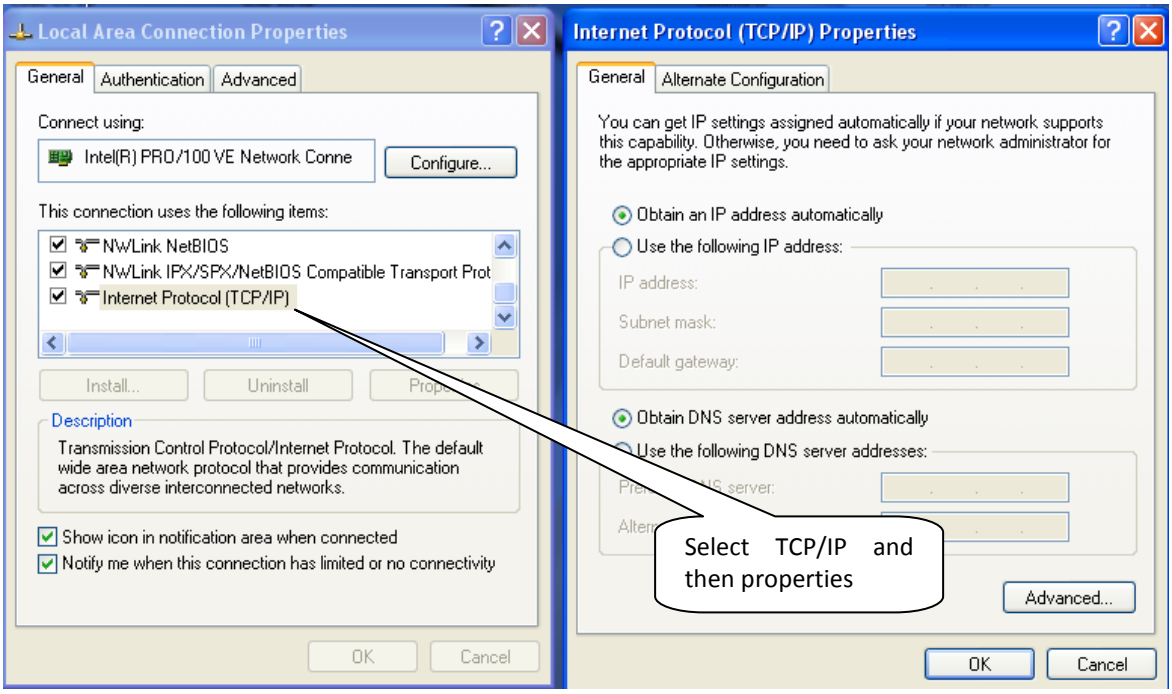


Figure 5.24

Simply restart the PC while connected to the FieldServer and it will obtain an IP address from the FieldServer.

NOTE: Some implementations of the Microsoft DHCP Client will not use an IP address lease offered by the FieldServer until its previous lease has been released. The way to release and renew a lease on a PC is to open a command prompt window and use the following commands (syntax may be different on Windows other than XP):

```
ipconfig /release *
ipconfig /renew *
```

The FieldServer checks every few minutes for the existence of other DHCP Servers and will disable itself if any are detected. A message on the RuiNet E screen will indicate when the DHCP Server has been disabled.

5.15 K - Change UI Display Mode

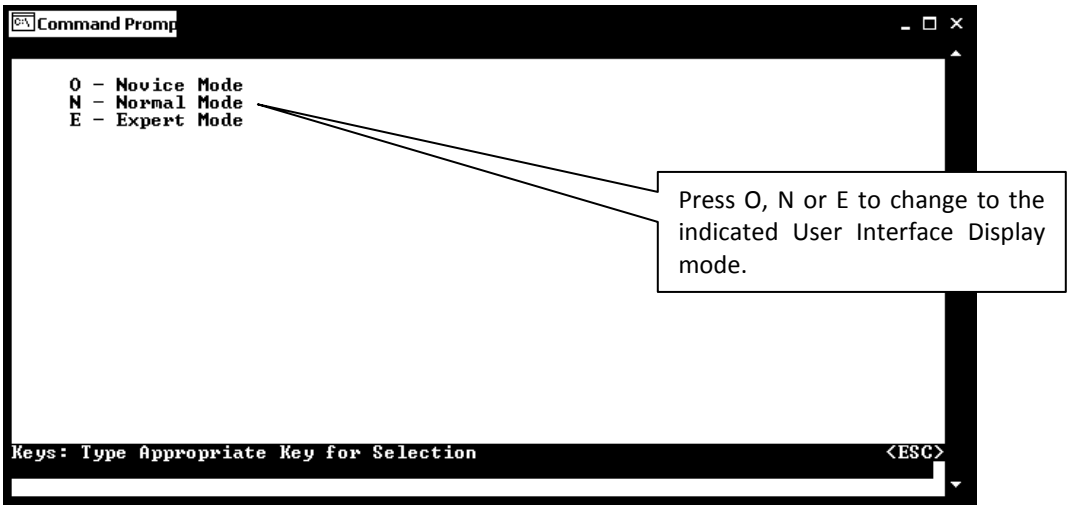
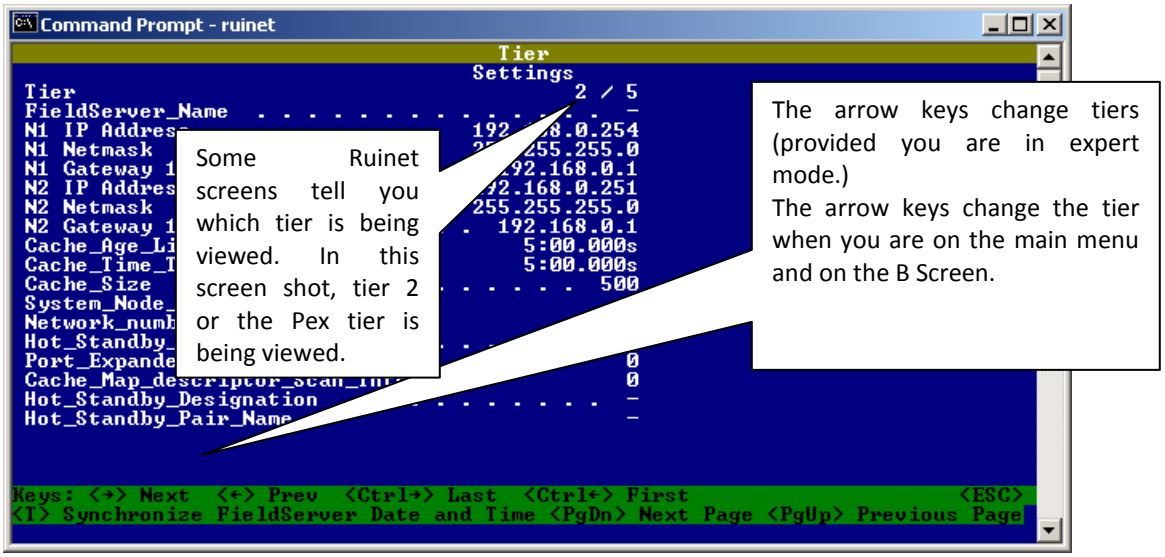


Figure 5.25

The Expert Mode allows viewing of the different FieldServer tiers. Novice and Normal modes prevent viewing the FieldServer tiers.

A tier is a virtual FieldServer – a software construction on which the various drivers are executed. Most applications only use one tier (called "Tier 2" or "Tier B" or "The Pex Tier") but some emulations and complex configurations use multiple tiers.

When Ruinet connects to a FieldServer it will always connect to the Pex Tier (Tier 2). The arrow keys allow the user to change tier but only if in the expert mode. If not in expert mode then the arrow keys have no effect.



NOTE: Normal and Expert modes are intended for the use of FieldServer personnel only. They provide no improved application functionality, and the added functions in these modes are neither documented, nor supported by FieldServer Technical Support. Consequently, it is strongly advised that these modes are not enabled.

5.16 Restart FieldServer

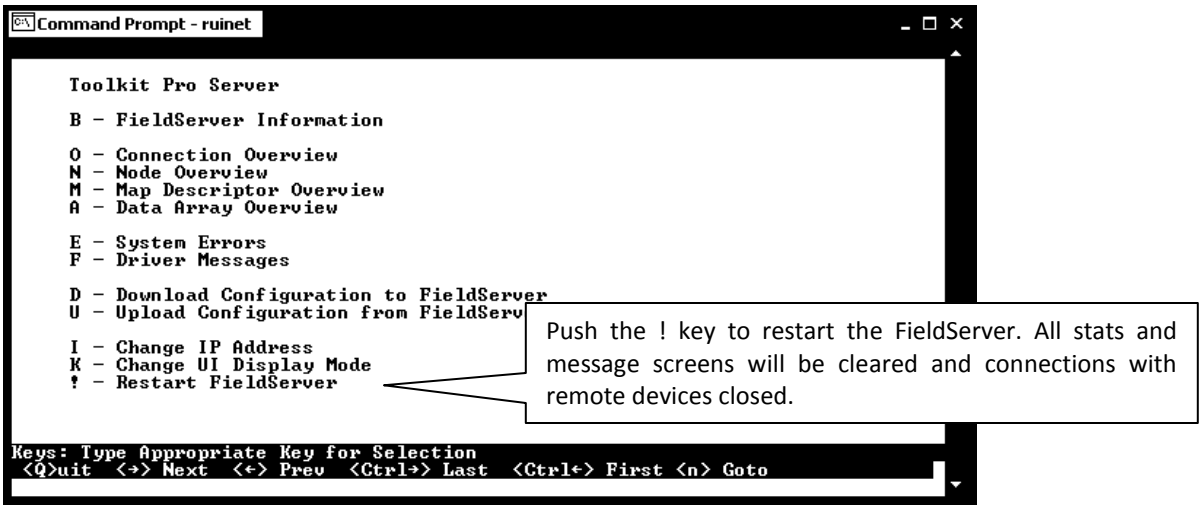


Figure 5.26

RuiNet will display the main interactive menu again after the FieldServer has restarted.

6 SERIAL AND SNAPSHOT CAPTURE UTILITY (RUIDEBUG)

The notes that follow document the use of RuiDebug in advanced mode when taking logs from a command line prompt. For standard operation, access these utilities via the Start Menu in Windows, i.e.:

Start|Programs|FieldServer Utilities|Tools|Snapshot (for taking snapshots) OR
Start|Programs|FieldServer Utilities|Tools|Serial Capture (for taking a serial capture log)
Refer also to Appendix B.3.

6.1 Log a Serial Port

Ports P1 … P8 and R1 and R2 can be logged. Using [t] tells RuiDebug to record timing information and to record the log in ASCII format. The logs produced by these command lines will record the bytes sent and received in hexadecimal format. FieldServer tech Support is able to convert these logs to ASCII.

```
RuiDebug –zp8[t]
RuiDebug –zr1[t]
```

P8 is the port being logged

R1 is the port being logged

- Allow some time to pass (typically 10 minutes), or, if taking a log to diagnose a problem that occurs when a specific event occurs, generate that event and wait.
- Use the 'Q' key to quit RuiDebug.
- Use a text editor to open the following files and check that a log was recorded.
- Browse through the **RuiDebug.log** file and check that that a snapshot was recorded. A snapshot always begins with lines similar to the following.

```
FieldServer Snapshot at Fri Nov 22 10:07:22 2002
```

- Browse through the **com_xx.log.** (xx is the port being logged, for P3 the file name will be com_p3.log). file and check that a log was recorded. The log always begins with lines similar to the following.

```
PORT_LOGGING started on Wed Jan 09 14:03:54 2002
```

- E-mail all the log files and the FieldServer configuration file (CSV file) to FieldServer Technical Support. If necessary, compress the log files using a ZIP Utility.

6.2 Monitor a FieldServer

- Initiate RuiDebug and select the required FieldServer from the list on the screen.
- Ensure that monitoring continues for sufficient time to generate the incident.
- Use the 'Q' key to quit
- Use a text editor to open the following file and check that a log was recorded.
- Browse through the **RuiDebug.log** file and check that that a snapshot was recorded. A snapshot always begins with lines similar to the following.

```
FieldServer Snapshot at Fri Nov 22 10:07:22 2002
```

- E-mail all the log files and the FieldServer configuration file (CSV file) to FieldServer Technical Support. If necessary, compress the log files using a ZIP Utility.

7 FST_DIAG

Before using this Utility, please make sure of the following conditions:

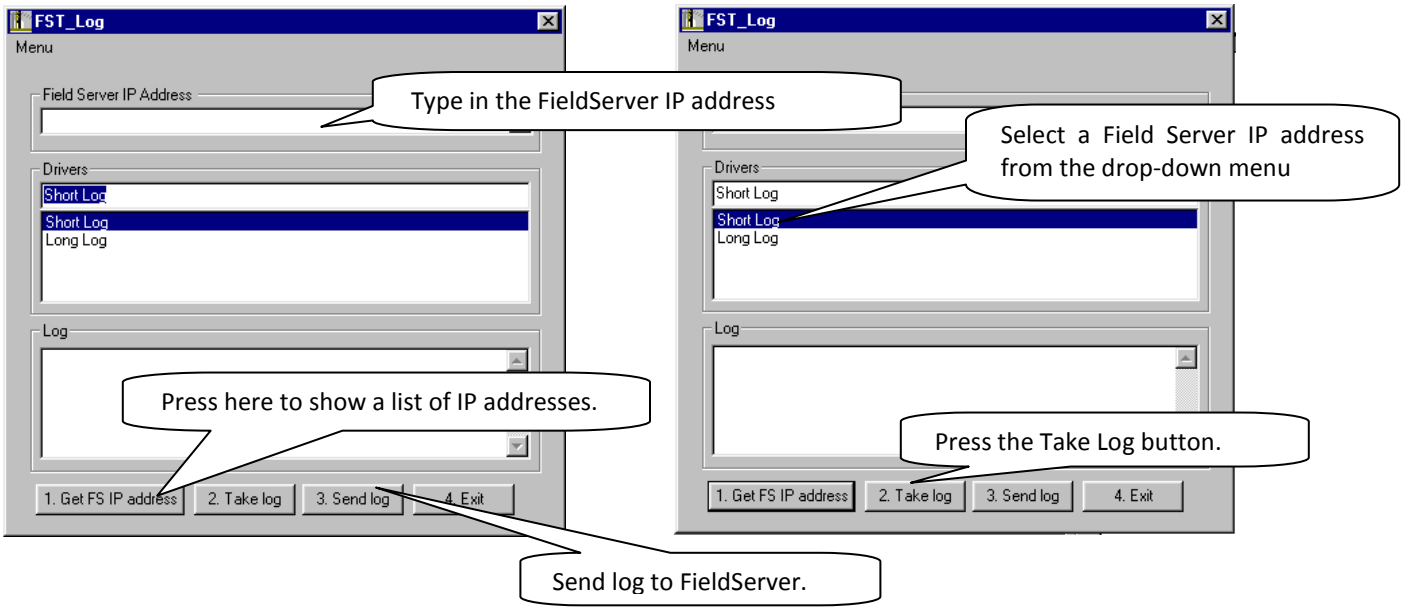
- The computer is on the same subnet as the FieldServer. Refer to Enote 0026 on the flash drive.
- The computer is connected directly to the same network as the FieldServer, not through a VPN or other remote networking software.
- All Ethernet cards on the computer have been deactivated. For laptops especially, ensure that separate wireless Ethernet cards are deactivated.
- Antivirus software is not preventing the NIC card from operating properly.

If there are Ethernet protocols (such as BACnet_IP, Ethernet_IP, Modbus_TCP) running on the computer it is advisable to capture an Ethernet log too. This process is not automated. Start the Ethernet log before pressing 'Take Log' and end the Ethernet log after the message 'Now Send Log'. Refer to Enote 0063 on the flash drive.

7.1 Using the FST_Diag

Step 1: Select a Field Server IP Address.

The IP address can be entered manually or selected using the Utility.

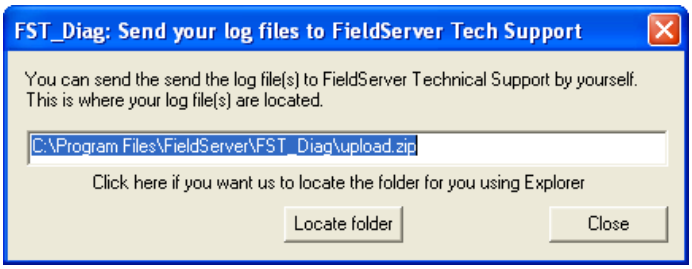


Step 2: Take a Log

Press the Take Log button. While the Utility runs a few DOS prompts will flash across the monitor. Don't click or type anything in to these DOS prompts. This step may take a few minutes depending on the chosen Log Type and computer speed. When the Utility is finished you will be presented with a log of events that have occurred.

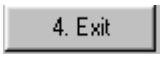
Step 3: Send Log

Click the "Send Log" button located near the bottom of the dialog. The following dialog should appear.



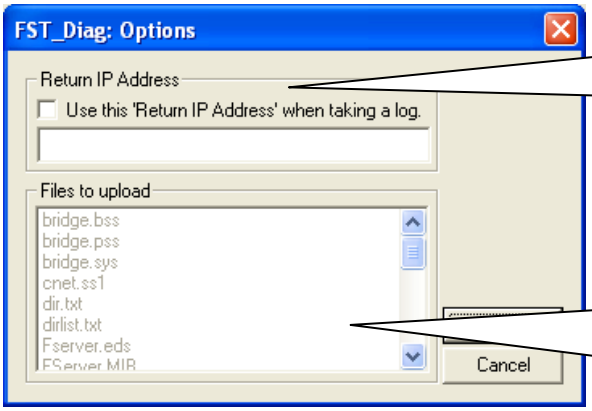
Push the 'Locate Folder' button to launch explorer and have it point directly at the correct folder. The file upload.zip must be sent to Tech Support.

Step 4: Close the Program

Press the exit button when the log is completed 

7.1.1 Options – Return IP address

The Menu selection 'Options' presents the following dialog:.

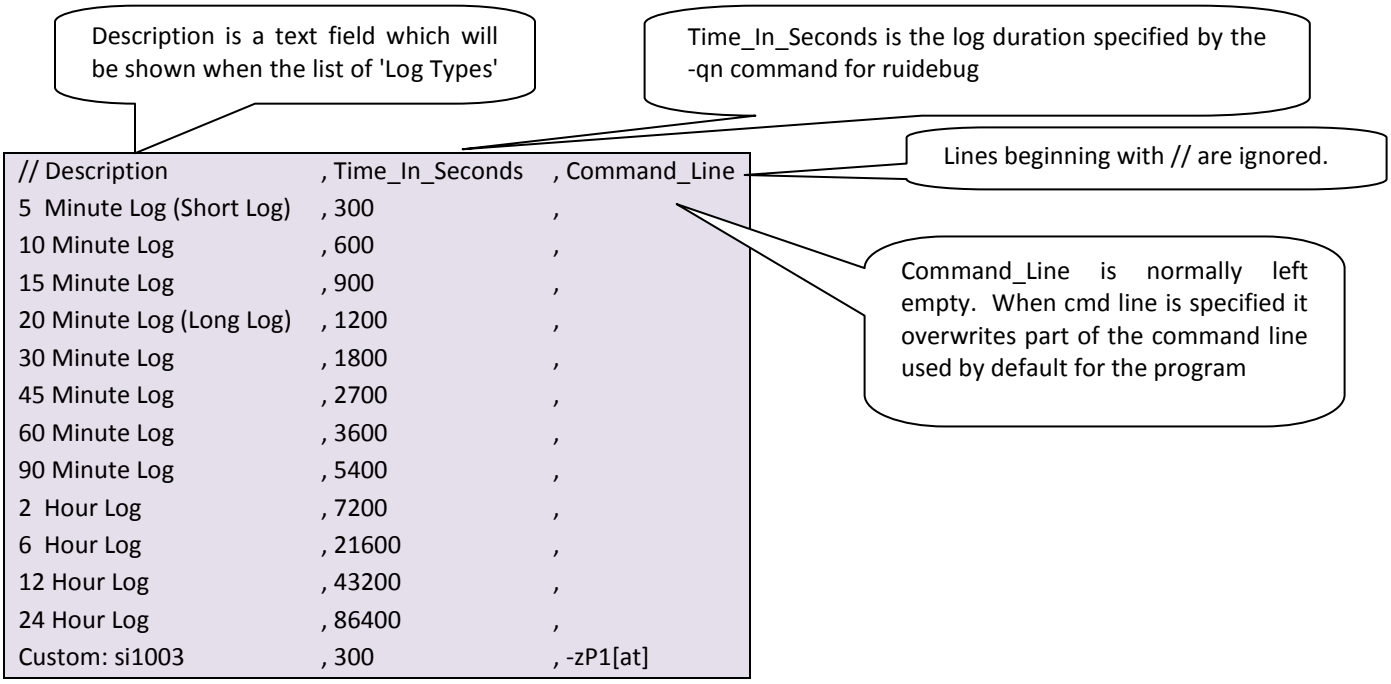


Leave the checkbox and edit field blank unless directed by FST Tech Support personnel. When a valid IP is specified and the checkbox is checked then the Utility adds the following to the command line of Ruidebug: "-mReturn_IP_Address"

This is a list of files found in upload_list.ini which is used to configure the utility.
The utility tries to upload these files.
The list cannot be edited within the FST_Diag application but it can be edited externally. Each time FST_Diag is re-installed, all changes are lost.

7.1.2 Configuring the "Log Type" List

In the install folder an Ascii file named config.ini controls the contents of the list named 'Log Types'. The default contents of this file are as follows:




```
// Description              , Time_In_Seconds   , Command_Line
5  Minute Log (Short Log)   , 300               ,
10 Minute Log               , 600               ,
15 Minute Log               , 900               ,
20 Minute Log (Long Log)    , 1200              ,
30 Minute Log               , 1800              ,
45 Minute Log               , 2700              ,
60 Minute Log               , 3600              ,
90 Minute Log               , 5400              ,
2  Hour Log                 , 7200              ,
6  Hour Log                 , 21600             ,
12 Hour Log                 , 43200             ,
24 Hour Log                 , 86400             ,
Custom: si1003              , 300               , -zP1[at]
```

This file can be edited using a text editor.

7.1.3 Troubleshooting

Error Message	Action
TakeLog:#1 Err. Cant execute Utility (Ruinet)	Check the install directory of FST_Diag[6] for Ruinet.exe. If it is not there please re-install this Utility.
TakeLog:#11 Err. Cant execute Utility (Ruidebug)	Check the install directory of FST_Diag[6] for Ruidebug.exe. If it is not there please re-install this Utility.
TakeLog:#12 Err. Cant execute Utility (Zip)	Check the install directory of FST_Diag[6] for Zip.exe. If it is not there please re-install this Utility.
TakeLog:#13 Err. Cant execute Utility (FST_ping)	Check the install directory of FST_Diag[6] for FST_ping.exe. If it is not there please re-install this Utility.
Get FS IP Address not working.	Check the install directory of FST_Diag[6] for FST_Ping.exe If it is not there please re-install this Utility. If it is there, then it is not working - type in your FS IP address manually.
Error on sending email to FieldServer support.	Check the install directory of FST_Diag[6] for blat.exe. If it is not there please re-install this Utility.

[6] default is c:\temp\fst_log\

7.1.4 Contents of Upload.zip

Filename	Description
Upload.bat	This is bridge.bat as found on the FieldServer.
Upload.f	If bridge.bat calls cb8menu with a Forth file then this is that file
Upload0.csv	If bridge.bat calls cb8menu with a config file then this is that config file. If bridge.bat calls cb8menu with a Forth file then this is the 1st config file loaded by the Forth file. i.e. It is the 1st file found in the .f file preceded by a "read_csv" command.
Upload1/2/3…..csv	If bridge.bat calls cb8menu with a Forth file then this is the 2nd and subsequent config file(s) loaded by the Forth file. i.e. It is the 2nd/3rd… file found in the .f file preceded by a "read_csv" command.
Snap.log	This is produced by Ruidebug. It is a simple snapshot.
Upload.log	This is the log file created when Ruidebug starts logging the ports. It contains the snapshot before and after the port log too.
Com_*.log	These are the serial port log files.
*.sys	Device Drivers
*.ini	Various initialization files
.xif, Lon.cfg	Lonworks files
*.mib	SNMP files
*.gse, *.ss1	Profibus files
*.bat	Various bridge script files

8 INTERNET ADDRESS CONFIGURATOR (RUIBOOT)

Whenever a FieldServer starts up, it broadcasts a message over the Ethernet displaying its current IP Address and waits to be reassigned a new IP address if there is a program listening to the request. RuiBoot will reassign an IP address and restart the FieldServer. RuiBoot responds to the **first** FieldServer to boot and can only allocate the IP address to one FieldServer in this mode. If in doubt, ask your network administrator to allocate an IP address for you to use.

When used in this simple mode, RuiBoot does not check for duplicate IP addresses and a manual check needs to be undertaken. Refer to Section 4.1 for more information. .

- Power down the FieldServer
- Start RuiBoot
- Power up FieldServer
- Wait for completion. This may take up to 60 seconds.

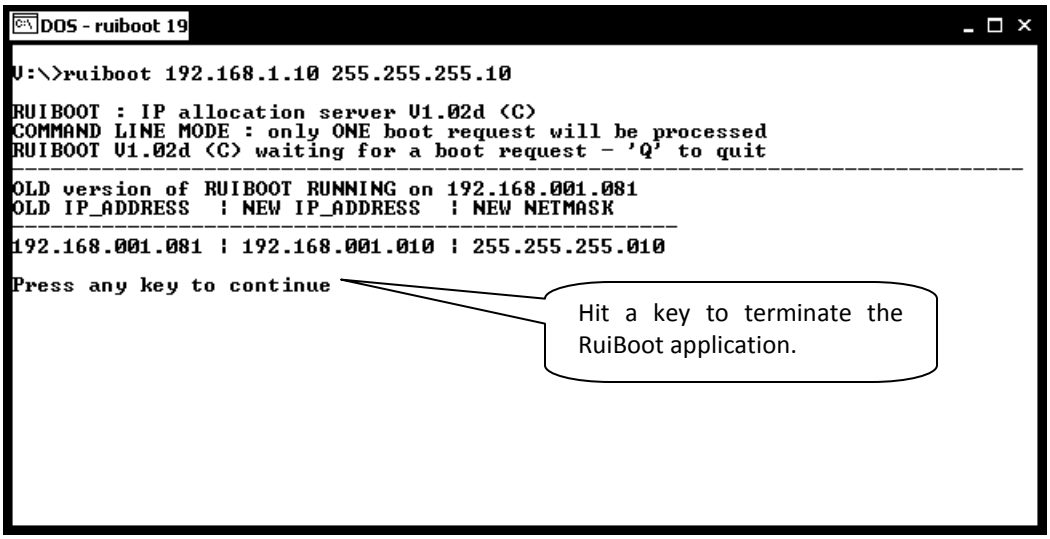


Figure 8.1

8.1 The INI file

A factory configured IP Address value is stored in this file. If RuiBoot fails (server does not exist), then the FieldServer uses the IP Address in the INI file for the Primary address. The applicable INI filename is FS_TCP.INI. On old versions of the configuration, the file was called SMCTP.INI. SMCTP.INI should be renamed FS_TCP.INI or deleted entirely if the FS_TCP.INI file exists on the FieldServer.

Appendix A. Troubleshooting[7]

Appendix A.1. General Configuration

- If the Ethernet cable is damaged or incorrect, connection to the FieldServer will not be able to be established. In this case the connection lights on the FieldServer and/or PC will not be illuminated.
- Connection will not be able to be established if the Ethernet card and/or protocol setup is incorrect. In this case Ruiping probably won't work either. Contact the Systems Administrator to have an Ethernet port set up correctly.
- The Utilities will not work if there are two FieldServers with different names but the same IP address. Run RuiNet (see Section 5) – these will show red in the list of FieldServers. Disconnect one FieldServer from the network and change the other FieldServer's IP address to a different value using RuiNet.
- If a FieldServer is on a different subnet to the PC, and there is more than one FieldServer on the network, then it will not be possible to use broadcast mode to cross subnets. Disconnect the other FieldServers and use broadcast mode, or change the IP address of the "orphan" FieldServer to put it in the same subnet and then reconnect the other FieldServers.
- Ensure that the firewall or router is not blocking UDP port 1024.
- If the FieldServer is connected to the host computer's network on adapter N2 then the SMT protocol must be defined for the N2 adapter in the config.csv file. Add the following lines to the file if it is not possible to connect to N1 instead:

```
Adapters
Adapter     , Protocol
N2          , SMT
```

Appendix A.2. File Download

- Unless a path is specified for the local file name then the Utilities expect to find the local file in the same folder as that in which the Utility was launched or in the folder set as the working directory (as specified in the Windows shortcut). If the file cannot be found then an error is reported.
- The local and remote file names must meet the DOS 8.3 file name format requirements.
- There must be sufficient flash disk space on the FieldServer.
- If the download of files like fserver.img (the firmware file) or config.csv (the configuration file) fails before completion then the next time the FieldServer is restarted the FieldServer will try and run with corrupt firmware or a corrupt configuration. If the firmware is corrupt then future downloads may not work at all and the user may need to follow a recovery procedure. See the Troubleshooting manual for further information.

[7] Please note that the Troubleshooting information presented here is specific to the Utilities. For connection and other troubleshooting information, please refer to the Troubleshooting Manual.

Appendix B. Command Line Switches

Appendix B.1. Ping Utility Command Line Switches

Appendix B.1.1. Ping FieldServer by IP Address -i<ip_address>

Ping a specific FieldServer by specifying its IP address.

```
RuiPing –i192.168.1.44
```

RuiPing will ping the FieldServer with the specified IP address.

Appendix B.1.2. Ping FieldServer by Name -n<FieldServer_Name>

Ping a specific FieldServer by specifying its name. The FieldServer name is configured in the CSV file under the Bridge, Title section. Enclose the FieldServer name in "" if the name contains spaces.

```
RuiPing –n"Modbus Server"
```

RuiPing will ping the FieldServer with the bridge title Modbus Server.

```
RuiPing –nServer
```

RuiPing will ping the FieldServer with the bridge title Server.

Appendix B.1.3. Ping FieldServer by Hot Standby Pair Name -p<Pair_Name>

Ping a specific FieldServer Hot Standby pair by specifying its name. The FieldServer Hot Standby Pair name is configured in the CSV file under the Bridge, Hot_Standby_Pair_Name section. Enclose the Pair name in "" if the name contains spaces.

```
RuiPing –p"Modbus H-Pair"
```

RuiPing will only ping the FieldServers that are in a hot standby configuration with name Modbus H-Pair.

```
RuiPing –pPair
```

RuiPing will only ping the FieldServers that are in a hot standby configuration with name Pair.

Appendix B.1.4. Ping list of FieldServers from file -f<filename>

Ping only FieldServers that are specified in the indicated file. The FieldServers to ping may be specified by IP address or by FieldServer name.

```
RuiPing –flist.txt
```

RuiPing will only ping the FieldServers that are specified in the file called list.txt

The following is an example of the contents of a ping list file:

```
192.168.0.242
Modbus Server
192.168.0.244
192.168.0.233
Bacnet Client
```

Appendix B.1.5. Set ping repeat rate -r<seconds>

By default, RuiPing pings a FieldServer or FieldServers, depending on switch usage, every 3 seconds. This switch can be used to increase or decrease the time between pings. A value of 0 will cause one ping before Ruiping exits.

```
RuiPing –r10
```

RuiPing will ping all FieldServers on the network every 10 seconds.

```
RuiPing –i192.168.0.101 –r0
```

RuiPing will ping a specific FieldServer once and then exit.

Appendix B.1.6. Use Quiet Mode-q

This command line switch is useful in batch files and prevents RuiPing from printing any output to the user screen. Typically used with the –r0 switch to check how many FieldServers are present on the network, examining RuiPing's exit code.

```
RuiPing –q –r0
```

RuiPing will ping a specific FieldServer once and then exit. The exit code can be used in a batch file when calling RuiPing to check if the FieldServer is present on the network.

Sample batch file:

```
@echo off
RuiPing –q –r0
if %errorlevel% EQU 2 goto single_fs
if %errorlevel% EQU 1 goto no_fs
if %errorlevel% EQU 0 goto many_fs
:single_fs
echo One FieldServer on network.
goto end
:no_fs
echo No FieldServers on network.
goto end
:many_fs
echo Many FieldServers on network.
:end
```

Appendix B.1.7. Use Analyze Mode -a<filename>

This switch instructs RuiPing to enter analyze mode. In this mode a menu is presented that may be used to effect certain operations. A ping list filename similar to when using the –f switch must be specified with the analyze mode switch. RuiPing will only effect the indicated operations on the FieldServers that are specified in the indicated file. The FieldServers to ping may be specified by IP address or by FieldServer name.

```
RuiPing –alist.txt
```

RuiPing will enter analyze mode and will only ping the FieldServers that are specified in the file called list.txt RuiPing records statistical information during analyze mode and saves it to a text file called pingstat.

Sample ping list file:

```
192.168.0.242
Modbus Server
192.168.0.244
192.168.0.233
Bacnet Client
```

Appendix B.1.8. Display FieldServer Identification Number -b

RuiPing -b — RuiPing will display the identification numbers of all FieldServers that reply to it.

Appendix B.1.9. Display FieldServer Hot Standby Pair Names -c

When two FieldServers are used in hot standby mode, they are assigned a pair name that is different from the individual name used for each FieldServer. This switch causes RuiPing to display the hot standby pair name of the FieldServers.

RuiPing -c — RuiPing will display the pair name of FieldServers that are used in a hot standby mode

Appendix B.1.10. Display FieldServer Ethernet ID -e

This switch can be used to let RuiPing print out the Ethernet ID or MAC Address of the Ethernet ports on the FieldServer(s). This switch can be used in conjunction with the `-1 -2` and `-3` switches to only display the Ethernet IDs of specific Ethernet ports. When used alone without specifying `-1 -2` or `-3`, only the N1 network port's Ethernet ID will be displayed.

RuiPing –e or RuiPing –e -1 — RuiPing will display the Ethernet IDs of the N1 network ports on all FieldServers

RuiPing –e -2 — RuiPing will display the Ethernet IDs of the N2 network ports on all FieldServers

RuiPing –e -3 — RuiPing will display the Ethernet IDs of the N1 & N2 network ports on all FieldServers

Appendix B.1.11. Do not display heading -0

This switch can be used to prevent RuiPing from displaying a heading for the list of FieldServers that replied to the ping. This may be useful when capturing RuiPing's output.

RuiPing –0 — RuiPing will not display a heading for the ping results.

Appendix B.1.12. Display N Network Ports

When using this switch, RuiPing will only display FieldServers that responded on the N1 network port. Some FieldServers may be using both N1 and N2 ports and this switch can be used to prevent a single FieldServer be shown twice in the list or to just test the N1 network ports.

RuiPing -1 — RuiPing will display FieldServers that replied using the N1 network ports

RuiPing -2 — RuiPing will display FieldServers that replied using the N2 network ports

RuiPing -3 — RuiPing will display FieldServers that replied using the N1 and/or N2 network ports

Appendix B.1.13. Display Program Version -v[e]

RuiPing –v — RuiPing prints version information and exits.

RuiPing –ve — RuiPing prints extended version information and exits.

Appendix B.1.14. Display Help -h

Prints a list of command line switches. If this command line switch is used then all others are ignored.

```
RuiPing –h
```

RuiPing prints out all options and then exits

Appendix B.1.15. Display exit codes -x

This switch may be used for debugging or other purposes to let RuiPing display the actual exit code it returns to the command interpreter. This is useful when using RuiPing in batch files. The switch must be used with the `-r0` switch

```
RuiPing –x –r0
```

RuiPing will display one of the following exit codes (except codes 9 and 10 which has to be enabled with –t) when it exits:

Exit codes :

0	Multiple FieldServers were detected
1	No FieldServers were detected
2	Exactly one FieldServer was detected
3	Same as 2 but FieldServer NOT on our network
7	User quit from program
8	Program system error
9	Pinged FieldServer has panics or broadcast panics detected
10	Pinged FieldServer has no panics and no broadcast panics

Exit codes – 0, 1, 2, 3	RuiPing must be used with the –r0 switch so it can exit after determining the number of FieldServers on the network. It must not be used with any of the –i, -n, -p, -f switches since these switches target a specific FieldServer.
Exit code – 7	This exit code will only be used when the user quits from RuiPing by pressing the Q key.
Exit code – 8	A program system error occurred in RuiPing. This should be reported to FieldServer technical support.
Exit codes – 9, 10	RuiPing must be used with the –t switch when testing for FieldServer system errors or panics. The –r switch has no effect since RuiPing will exit immediately after receiving a reply from any FieldServer. RuiPing must be used with the –i, -n, -p or –f switch for this test to be meaningful. When a FieldServer did not reply to RuiPing after the test time has passed (600 seconds when just using just –t or e.g. 10 seconds when using –t10), it will exit with code 1 (No FieldServers were detected).

Appendix B.1.16. Enable system error test exit codes -t<seconds>

This switch enables the system error test exit codes. RuiPing can be used to specifically check if a specific FieldServer has system errors or not. When using this switch without specifying a time limit for the test, the default time is 10 minutes or 600 seconds, after which RuiPing will exit if it received no replies. RuiPing will exit after receiving one reply and will use exit codes 9 and 10 when exiting to indicate whether system errors were found on the FieldServer that replied.

```
RuiPing –i192.168.0.
```

RuiPing will ping the indicated FieldServer for 600s. If no replies are received it will exit with code 1. If it receives a reply it will exit immediately using codes 9 or 10

Appendix B.1.17. Using RuiPing exit codes in a batch file

RuiPing returns a value or exit code to indicate the success of its operations. This return value can be used in a dos batch file by utilizing the 'errorlevel' batchfile variable.

The following comparisons on the errorlevel or exit code from RuiPing can be done in a batch file:

```
if %errorlevel% EQU value action
if %errorlevel% NEQ value action
if errorlevel value action
```

Checks if errorlevel is equal to a value

Checks if errorlevel is not equal to a value

Checks if errorlevel is greater than or equal to a value

The following example illustrates the use of RuiPing in a batch file to check if a specific FieldServer has system panics or not. The batch file may be saved as panic.bat and will be used by typing panic 192.168.0.242 or any other IP address.

```
@echo off

if c%1 == c goto user_prompt

RuiPing -i%1 -q -t10

if %errorlevel% EQU 9  goto panics
if %errorlevel% EQU 10 goto no_panics
if %errorlevel% EQU 1  goto no_fs

echo Error, unexpected exit code %errorlevel%
goto exit

:no_fs
echo FieldServer not found
goto exit

:panics
echo FieldServer has panics
goto exit

:no_panics
echo FieldServer has no panics
goto exit

:user_prompt
echo Error, specify a FieldServer IP address e.g. 1.2.3.4
goto exit

:exit
```

Appendix B.2. Ruinet Command Line Switches

Appendix B.2.1. Connecting to a FieldServer using IP Address -i

Specify the IP address using the ***-i*** switch.

```
RuiNet –i192.168.2.45
```

RuiNet will connect to the FieldServer with IP address 192.168.2.45

Appendix B.2.2. Connecting to a FieldServer using Name -y

Specify a FieldServer's name with the ***-y*** switch

```
RuiNet –yModbus_Test
```

RuiNet will connect to the FieldServer with the specified name

Appendix B.2.3. Broadcast Mode -ib

RuiNet can be used in broadcast mode to poll all IP addresses on a network. This feature is not recommended under normal operating conditions and should only be used when there is just one FieldServer on the network.

```
RuiNet –ib
```

RuiNet will use broadcasting to connect to a FieldServer

Appendix B.2.4. The Most Recently Connected FieldServer -p

If the ***-p*** command line switch is used, RuiNet will connect to the FieldServer that was connected previously. The most recently connected to FieldServer will be shown on top of the list of "Recently connected to FieldServers" on the RuiNet list of FieldServers screen. This command line switch is useful in batch files.

```
RuiNet –p
```

Sample batch file:

```
RuiNet –zdefault.htm –m0
RuiNet –p –zzone1.htm
RuiNet –p –zzone2.htm
RuiNet –p –zzone3.htm
RuiNet –p –zzone4.htm
```

No FieldServer targeted but –p ensures that the FieldServer selected by the user when the previous command was executed is the FieldServer that is connected to with this command.

Appendix B.2.5. Disable Auto Connect Mode –m0

If the target FieldServer is not specified then RuiNet either auto connects to the only FieldServer or presents a list of FieldServers and auto connects to the most recently connected FieldServer or to the first FieldServer in the list after 10 seconds. The ***–m0*** switch which disables the auto connect function.

```
RuiNet –m0
```

Ensures presentation of a list and auto connection does not occur.

Appendix B.2.6. Restart a FieldServer -b

Tells RuiNet to restart a FieldServer. No confirmation to restart is requested.

```
RuiNet –b
```

Restarts the FieldServer that RuiNet auto connects to.

```
RuiNet –i192.168.2.81 –b
```

Restarts a specific FieldServer.

Appendix B.2.7. Startup Screen -x

Directs RuiNet to display a particular screen on connection to a FieldServer. Specify the switch and the start-up screen number.

Screen Description		Start up number
System Errors	(E Screen)	-x1
Driver Messages	(F Screen)	-x2
Connection Overview	(O Screen)	-x3
Node Overview	(N Screen)	-x4
Map Descriptor Overview	(M Screen)	-x5
Data Array Overview	(A Screen)	-x6
FieldServer Information	(B Screen)	-x7

RuiNet –x3

When RuiNet connects the connection overview screen will be displayed.

Appendix B.2.8. Transferring files –l, -f

When a file is transferred, its local file name (on host computer) and its target (FieldServer-side) file name must be known for RuiNet to work correctly. The local file name is specified with the ***–l*** switch and the target filename is specified with the ***–f*** switch. RuiNet will fill in a default target file name if possible.

If transferring a file with the extension .CSV from a host computer to a FieldServer then RuiNet will use the default file name config.csv unless the ***–f*** command line switch was used to specify a different target file name.

RuiNet -fconfig.csv –lc:\temp\config.csv –u1

Transfers the file config.csv from the FieldServer to the host computer where it will be saved as config.csv in the temp folder of the C drive

Appendix B.2.9. Downloading Files –u0

This command is used to transfer files from a PC to a FieldServer. In the following examples the target FieldServer is not specified. If there is a single FieldServer on the network then the files will be transferred to that FieldServer. If more than one FieldServer is connected to the network then RuiNet will provide a list and the transfer will begin once a FieldServer is selected from the list.

RuiNet –ldelivery.csv –u0

Transfers the file config.csv from the computer to the FieldServer where it will be saved as config.csv.

RuiNet -ldefault.htm –u0

Transfers the file default.htm from the computer to the FieldServer where it will be saved as default.htm.

RuiNet –lc:\temp\delivery.csv –fdelivery.csv –u0

Transfers the file delivery.csv from host computer's temp folder on the C drive to the FieldServer where it will be saved as delivery.csv

RuiNet –lconfig.csv –u0

Transfers the file config.csv from host computer to the FieldServer where it will be saved as config.csv. RuiNet expects that the file config.csv exists on the host computer and is located in the same folder as the folder where RuiNet is executed. If a file is not transferred correctly then RuiNet will continue to retry until the transfer is successful

Wildcards are allowed on download filenames. RuiNet will present a list of the files marked for download and a list of the files already on the FieldServer for you to decide whether to proceed with the download. The "list download files" prompt can be overridden with a ***–c*** switch.

```
RuiNet –ltest*.csv –u0
```

Transfers all files with filenames starting with test and ending with .csv to the FieldServer.

Appendix B.2.10. Uploading Files –u1

This command is used to transfer a file from the FieldServer to the PC. In the following example the target FieldServer is not specified. If there is a single FieldServer on the network then the files will be transferred from that FieldServer. If more than one FieldServer is connected to the network then RuiNet will provide a list and the transfer will begin once the required FieldServer is selected.

```
RuiNet -fconfig.csv –u1
```

Transfers the file config.csv from the FieldServer to the host computer where it will be saved as config.csv in the folder from which RuiNet was executed

Appendix B.2.11. Forcing a Download -o

When downloading a file that already exists on the FieldServer, RuiNet checks if the file to be downloaded is identical to the file already on the FieldServer. If the files are identical, RuiNet responds that a download is not necessary. This behavior can be overridden and the download forced by using the ***–o*** command line switch.

```
RuiNet –lconfig.csv –u0 –o
```

RuiNet downloads config.csv regardless of whether the file on the FieldServer is identical.

Appendix B.2.12. Number of File Transfer Tries -n

If a file is not transferred correctly then RuiNet will continue to retry until the transfer is successful. This can be avoided by specifying the maximum number of retries that may be attempted.

```
RuiNet –lconfig.csv –u0 –n10
```

RuiNet will try up to ten times to successfully download the file config.csv to a FieldServer.

Appendix B.2.13. Set FieldServer Side File Name for a transfer -f

RuiNet will always use the FieldServer file name of config.csv for any CSV file downloaded unless an altenate file name is specified using this switch. If a file is downloaded with the name "fserver.img", RuiNet will automatically convert this to "cb8menu.exe" on the FieldServer.

A path in the FieldServer Side file name cannot be set.

```
RuiNet –lconfig.csv –fbackup.csv –u0
```

RuiNet will download config.csv from host PC to the FieldServer and save it as backup.csv on the FieldServer.

Appendix B.2.14. Set Local File Name for a Transfer -l

If transferring a file with the same name from host computer to a FieldServer, omit this switch. If the file is to have a different name or if the file is to be saved to a specific drive and/or folder then use this switch.

```
RuiNet –lconfig.csv –fbackup.csv –u1
```

RuiNet will upload backup.csv from the FieldServer and save it as config.csv on the host PC.

Appendix B.2.15. Help -h

Prints a list of command line switches. If this command line switch is used then all others are ignored.

```
RuiNet –h
```

RuiNet prints out all options and exits.

Appendix B.2.16. Create a Log File -a

Instructs RuiNet to make a log file containing progress information.

```
RuiNet –lconfig.csv –u0 –alog.txt
```

RuiNet creates a log file called log.txt on host PC and logs the progress and results of the config.csv file download to the log file.

The log file contains time and date stamped entries that describe the actions performed by RuiNet. The log file is used by FieldServer technical support and developers for debugging purposes. An excerpt from a typical log file is shown below:

```
07/30/03 20:34:09 Doing rui_send_command with cmd = 182
07/30/03 20:34:09 Rui_version = 41610400
07/30/03 20:34:29 Calling Download_with_repeat (config.csv, config.csv, 1, 0, 0, 1024
07/30/03 20:34:29 Setting tx_rui = 3000
07/30/03 20:34:29 Getting Rui_get_file_size for = config.csv
07/30/03 20:34:29 Restoring tx_rui = 2000
07/30/03 20:34:29 Rui_get_file_size = 4259
```

Appendix B.2.17. Version Information -v, -ve

```
RuiNet –v
```

RuiNet prints version information and exits.

```
RuiNet –ve
```

RuiNet prints extended version information and exits.

Appendix B.2.18. Skip date and time check on connect -m1

If the system date and time of the machine running RuiNet differs by more than 5 minutes from that of the FieldServer you will be prompted to update the FieldServer. This behavior can be overridden using the ***–m1*** switch.

```
RuiNet –m1
```

Ensures update FieldServer date and time prompt does not occur.

Appendix B.2.19. Delete a File -z

Delete a file on the FieldServer. Use this switch with extreme caution as no opportunity for confirmation to delete is provided. If RuiNet connects, the file will be deleted.

```
RuiNet –zdefault.htm
```

Deletes the file default.htm on the FieldServer that RuiNet connects to.

```
RuiNet –i192.168.2.81 –zdefault.htm
```

Deletes the file default.htm on a specifically targeted FieldServer.

Appendix B.2.20. Synch FieldServer Time with the Computer's Time -g

The FieldServer's real time clock is maintained with a super-capacitor. The capacitor discharges after about 30 days. A consequence of this is that often when a user first receives their FieldServer the time is not correctly set.

The FieldServer date and time can be synchronized with host computer's date and time by using the ***–g*** switch.

```
RuiNet –g
```

Sets a FieldServer's date and time to that of the host computer.

Appendix B.2.21. Set Timeout -k

This is the time RuiNet waits for a response before deciding that 1) a FieldServer connection has been lost or 2) A FieldServer cannot be connected to. The default is 2 seconds, and this is normally adequate unless connecting to FieldServers on remote networks or over the internet when it might be necessary to make the timeout longer. Timeout is set in Miliseconds..

RuiNet –k10000 –i192.168.2.X[8]

Tells RuiNet to try and connect to a specific FieldServer and sets the timeout to 10 seconds.

Appendix B.2.22. -downloadPassword xxxxxxxxxxxxxxx

Where xxxxx is an unencrypted password. The password must match the one encrypted into the ruilogin.ini file loaded on the FFP or FieldServer.

Some firmware on FFP's and FieldServer's requires a password to download to the device. The firmware will reply demanding a password when Ruinet attempts the download.

The download function attempts, in the following priority order, to locate the password:

1. Searches for ruilogin.ini in Ruinet's working folder. If found, the file is assumed to contain an encrypted password. Typically use the same file as loaded on the device.
2. Looks to see if a password was set by a command line parameter.
3. User is prompted to enter the password manually.

Once the password has been used to download one file to the device, the password need not be used again until after the device has been rebooted.

Passwords cannot be validated ahead of time.

Appendix B.2.23. Run in Test Mode

This option is for FieldServer testing purposes only. It activates various test actions when RuiNet is used. It currently causes checksum errors during a file download to test if retries are working.Advanced Topics

Appendix B.3. Serial & Snapshot Capture Utility Command Line Switches

Appendix B.3.1. Target IP Address -i

Target a specific FieldServer by specifying its IP address.

RuiDebug –i192.168.1.81
RuiDebug –i192.168.1.81 –zp1[at]

If no FieldServer exists at the specified address the following screen will be displayed:

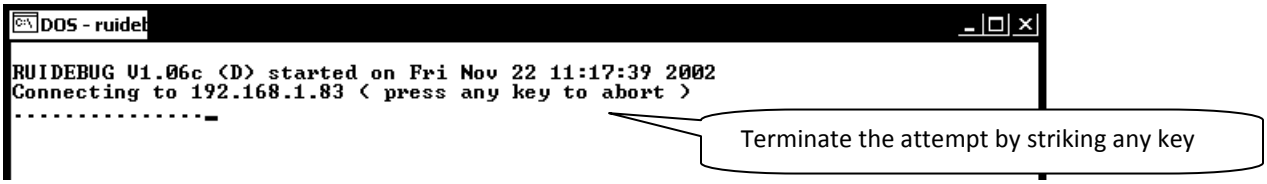


Figure 8.2

[8] X is a variable and it's value is specific to a particular FieldServer.

Appendix B.3.2. Target FieldServer Name -y

Target a FieldServer by specifying its name. Use "" when the name contains spaces.

```
RuiDebug –yServer
RuiDebug –y"Server 01"
```

Appendix B.3.3. Monitor a FieldServer by broadcast (unknown IP) -ib

Use to connect to a FieldServer on the same physical network, but on a separate sub-net. Under this option, RuiDebug broadcasts its messages and all FieldServers will receive the message and try to reply. Only use this option in cases where there is one FieldServer on the physical network

```
RuiDebug –ib
```

Appendix B.3.4. Set self termination tim -q.

After RuiDebug has been running for the specified self termination time it will terminate and close the logs. RuiDebug always self terminates. The default self termination time is one hour. Specify the termination time in seconds, minutes or hours.

```
RuiDebug –qs60
RuiDebug –qm5
RuiDebug –qh8
```

Runs for 60 seconds

Runs for 5 minutes

Runs for 8 hours

Appendix B.3.5. Version Information -v

Use this command line switch to tell RuiDebug to print its version number.

```
RuiDebug –v
```

Appendix B.3.6. Help -h

Print a list of command line switches. If this command line switch is used then all others are ignored.

```
RuiDebug -h
```

Appendix B.3.7. Change the default log file name -a

Specify an alternate name for the log file (Default is RuiDebug.log)

```
RuiDebug –aMYLOG.txt
```

Changes the log file name from RuiDebug.log to MYLOG.txt

Appendix B.3.8. Echo to console -e

All the information captured in RuiDebug.log is echoed on the screen while it is happening.

```
RuiDebug -e
```

Appendix B.3.9. Listen Only (Suppress Snapshots) -s

Normally RuiDebug takes a snapshot when it starts and when it exits. In addition, RuiDebug also takes various snapshots on a periodic basis (period is displayed while RuiDebug is running). Suppress these by using the ***–s*** switch. You can always toggle the listen mode by using the ***"S"*** key while RuiDebug is running.

```
RuiDebug -s
```

In this example RuiDebug will not take starting, periodic and terminating snapshots

Appendix B.3.10. Limit the log to recording errors only -r

This switch has no effect. Do not use it.

Appendix B.3.11. Reset The FieldServer statistics and errors -x

Clears the error screen and clears all nodes and connection stats. If the error screen contained any System Error Messages these are cleared too as is the Red 'Sys' LED on the front of the FieldServer. Note that RuiDebug terminates immediately when this switch is used.

```
RuiDebug -x
```

Appendix B.3.12. Record driver messages in a special log -l

This switch is the equivalent of the M key used during interactive mode. Some drivers send information and error messages to RuiDebug while RuiDebug is connected to the FieldServer. This switch instructs RuiDebug to capture these messages in a file called **msg.log.**

```
RuiDebug -l
```

Appendix B.3.13. Set log Level -k

Sets the verbosity level of the messages described in Appendix B.3.12. There are three verbosity levels: (1, 2, 3). 3 is the most verbose.

```
RuiDebug –l –k3
```

This sample enabled driver message logging and sets the verbosity level to 3.

The verbosity of messages can also be set using the ***"K"*** key while in interactive mode.

Appendix B.3.14. Take a single snapshot and terminate -n

Tells RuiDebug to take a single snapshot and then terminate. Most other command line switches are ignored when the –n switch is used.

```
RuiDebug -n
```

Appendix B.3.15. Add Map_Descriptors to the snapshot -o

Tells RuiDebug to add Map Descriptor information to each snapshot.

```
RuiDebug –o
```

Captures all Map Descriptor information to the log file.

```
RuiDebug –oc
```

Captures only Map Descriptors of cache blocks to the log file.

```
RuiDebug –os
```

Captures only Map Descriptors declared in the configuration file to the log file

Appendix B.3.16. Log a serial port (P1..P8, R1/2) -z

Instructs RuiDebug to capture serial port data. Only Ports P1...P8 and R1/R2 can have their received and transmitted messages captured.

The captured data is appended to a file named according to the port being logged.

P1... P8	com_p1.log... com_p8.log
R1, R2	com_r1.log, com_r2.log

Once enabled on the command line, suspend the capture using the ***D*** key in interactive mode and then re-enable the capture using the ***Z*** key.

The log can record the following information

- Each byte received and transmitted.
- The time that the direction changes between receive and transmit.
- The time in milliseconds between each byte captured.

RuiDebug –zP1	Captures data on Port P1. The captured data will be logged in hexadecimal (default mode) and no inter-byte timing information will be recorded (default.)
RuiDebug –zP1[a]	Captures data on Port P1. The captured data will be logged in Ascii and no inter-byte timing information will be recorded (default.)
RuiDebug –zP1[t]	Captures data on Port P1. The captured data will be logged in hexadecimal (default mode.). Inter-byte timing information will be recorded.
RuiDebug –zP1[at]	Captures data on Port P1. The captured data will be logged in ASCII and inter-byte timing information will be recorded.

Appendix B.3.17. Redirect Responses from FieldServer (Internet Logging) -m

If the FieldServer you wish to connect to using RuiDebug is on another network you can work around this to a limited extent using the ***–m*** switch.

The routers on each network must be set up to forward port 1790 to the appropriate computer and FieldServer. The ***–i*** switch is used to specify the public IP of the remote router and the ***–m*** switch to specify the public IP of the network on which the RuiDebug computer is running.

```
RuiDebug –i63.171.24.69 –m66.23.1.192 zP1
```

In this example the FieldServer is on another network. The public IP of the remote network is 63.171.24.69. The public IP of the local network is 66.23.1.192.

Example of Setup Required for Internet Logging

The following example is based on the configuration of a LinkSys Router.

Consider a computer with IP address 66.23.171.192 directly connected to the internet while a FieldServer with IP address 192.168.1.102 is on a network connected to the internet via a LinkSys Router. The router is configured as shown below.

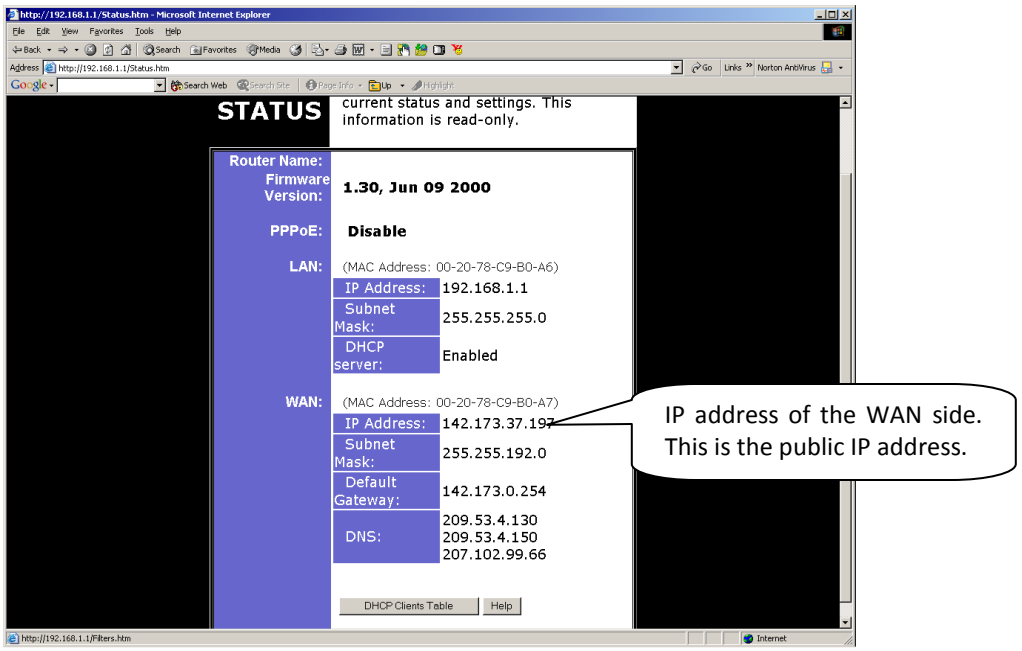


Figure 8.3

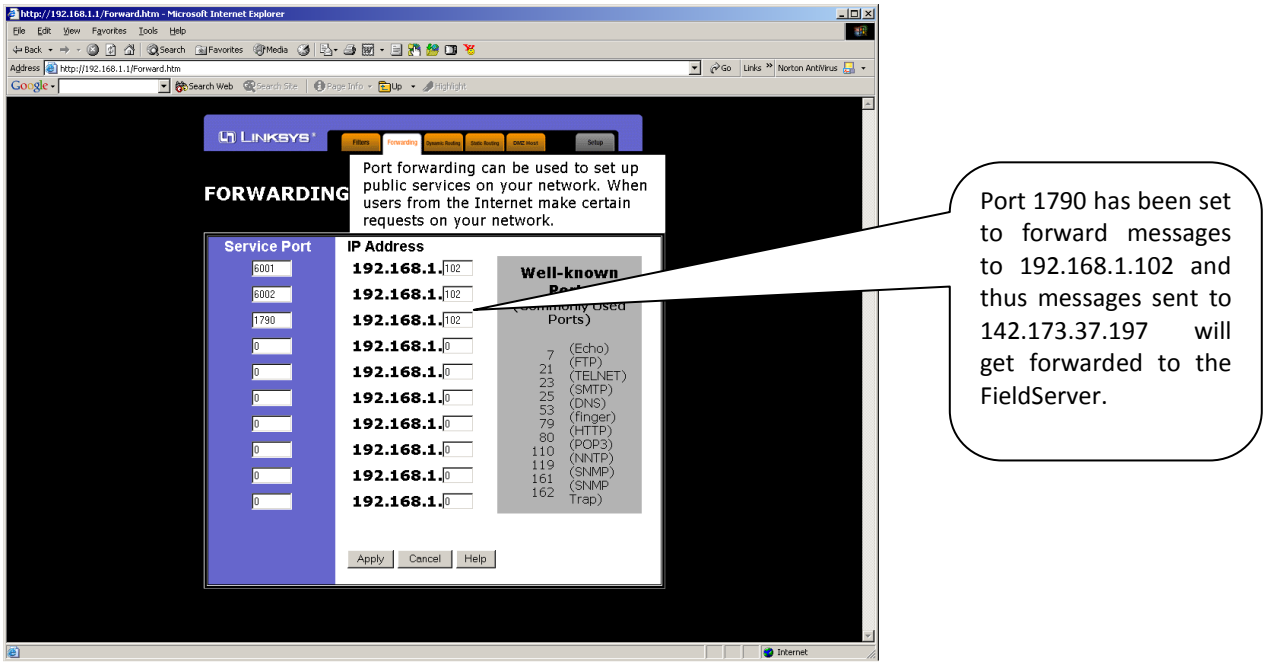


Figure 8.4

When the FieldServer responds it sends its reply back to the source IP which is the IP of the computer (66.23.171.192) directly connected to the internet and so there are no problems.

If the computer running RuiDebug is connected to the internet via a router, port forwarding needs to be set up on this router too. Route ports 1024 and 1790 to the IP of the RuiDebug computer. Say that the WAN side IP address (public IP address) of your router is 71.31.1.2 then run RuiDebug with the following command line.

```
RuiDebug –i142.173.37.197  -m71.31.1.2
```

Appendix B.4. RuiDebug interactive mode

You are able to interact with RuiDebug while it is running.

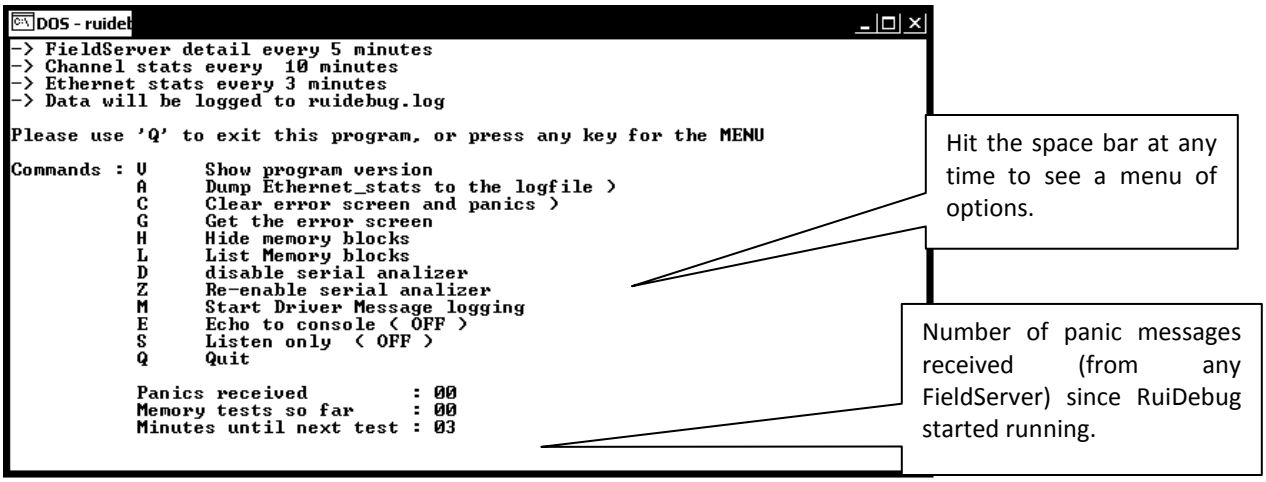


Figure 8.5

Command	Description
V - Show Program version.	Prints the RuiDebug revision information on the screen.
A - Dump Ethernet Stats.	Causes RuiDebug to take an immediate snapshot of the Ethernet statistics and add it to the log. This is in addition to the periodic Ethernet statistics snapshot taken by RuiDebug.
C - Clear Error Screen.	Clears the error log ('E' Screen') on the FieldServer. If the Error log contained System Error Messages (panics) these are also cleared.
G : Dump the Error Screen to the Log	Adds the contents of the E Screen to the log. The screen contents are added to the log with each press of the *"G"* key.
H : Hide memory Blocks	On the press of the *"H"* key, all memory blocks allocated at the time are hidden. This facilitates troubleshooting on a single FieldServer.
L : List memory Blocks	Produces a list of memory blocks and the names and line numbers of the functions that created them. The list is added to the log file.
D : Disable Serial Logging	Use this key to temporarily suspend the capture of a serial port log. The logging is suspended until you use the *Z* key.
Z : Enable Serial Logging	Use this key to enable serial port logging. The key has no effect if you did not specify serial port logging on the command line with the *–z* command line switch.
M : Driver Message Logging On/Off	Some drivers are able to send messages directly to RuiDebug when it is running.
E: Echo to Console On/Off	Toggles the echo state between on and off. When Echo is on, everything sent by RuiDebug to the log file is echoed on the screen. (It does not echo the serial port messages) Changing the echo state does not affect what gets recorded in the log file.
S : Listen Only (Stop Periodic status and statistics captures)	When in listen mode, RuiDebug stops capturing FieldServer statistics and status information on a periodic basis. Pushing the S key toggles the mode. Toggling the mode to listen only is the equivalent of using the –s option on the command line.
Q : Quit	Terminates RuiDebug. A final snap shot may be added to the log before the program returns to the command line if it wasn't in listen only mode when you quit.

Appendix B.5. Internet Address Configurator Command Line Switches

Do not use the advanced features of the RuiBoot application unless directed to by a support engineer at FieldServer Technologies. The advanced mode is generally used as part of the FieldServer production process and in this context is used to allocate IP addresses and Mac addresses to newly manufactured units.

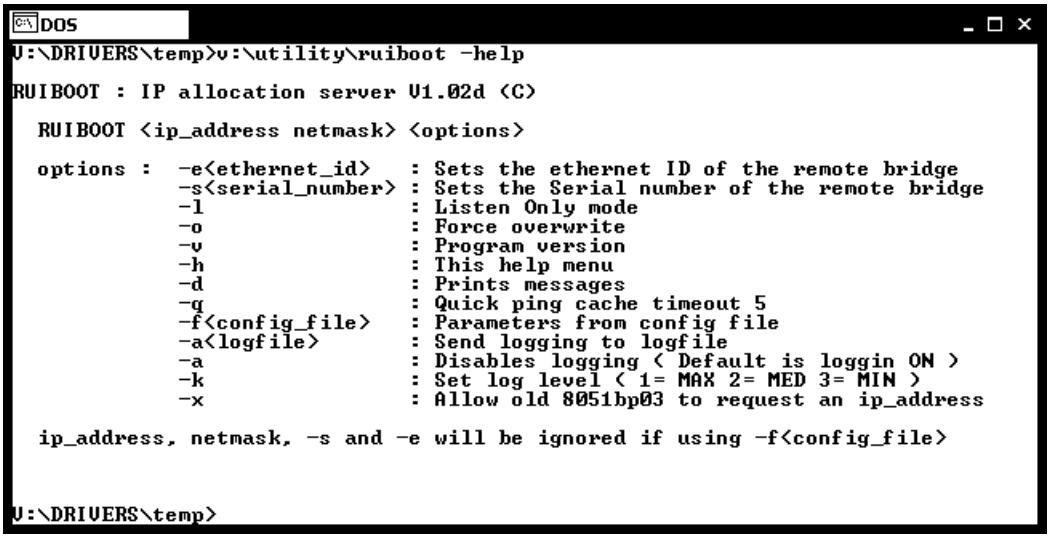


Figure 8.6

Appendix B.5.1. Single Shot vs. Continuous

The RuiBoot application can be used in single shot or continuous mode. Single and continuous modes cannot be used at the same time.

In single shot mode a single FieldServer is processed and allocated an IP Address/gateway Mask/MAC address/Serial number. The application is started, the FieldServer's power is cycled and the application sends data to a single FieldServer as it powers up.

In continuous mode the FieldServer monitors the network continuously, allocates IP addresses automatically, watches for duplicate nodes, listens for panics from FieldServers and pings all the FieldServer's continuously. A configuration file is used.

Appendix B.5.2. Set Ethernet ID (Mac Address) -e

Use this command with extreme caution. The Mac address of every network adapter manufactured in the world is meant to be unique. The Mac address is a 6 byte number specified in the hexadecimal dot format.

0.50.4e.nn.mm.kk

Where nn, mm, kk are hexadecimal values in the range 0-ff.

The first three bytes of the Mac address are reserved for FieldServer Technologies and may not be changed.

RuiBoot 192.168.1.10 255.255.255.0 –e0.54.4e.1.2.3

Appendix B.5.3. Serial Number -s

The serial number is stored on the FieldServer but is not visible. Since it is not used by any FieldServer software, it is never required to be set or changed. Every FieldServer is currently allocated the serial number SMC001. This command is used in single shot mode and is ignored in continuous mode even if specified.

```
RuiBoot 192.168.1.10 255.255.255.0 –s1234
```

Appendix B.5.4. Listen Only -l

Using the ***–l*** option allows you run another instance of RuiBoot (perhaps on another computer) that monitors the FieldServer network without changing the IP address. The switch is best used in continuous mode.

```
RuiBoot –fether.cfg -l
```

Appendix B.5.5. Force Overwrite -o

This function has been removed. The switch has no effect.

Appendix B.5.6. Version -v

Reports the version of the RuiBoot application. This is a single shot mode command - when the version number has been printed the application exits. This command can be run without the computer being connected to the network.

```
RuiBoot –v
```

Appendix B.5.7. Help -h

Prints a list of command line switches that may be used to control the application. This command can be run without the computer being connected to the network.

Appendix B.5.8. Cache Timeout -q

This function has been removed. The switch has no effect.

Appendix B.5.9. Use Config file.-f

The ***–f*** switch is used to trigger continuous mode and may be used in conjunction with the ***–l***, ***-k***, ***-d***, -a switches.

```
RuiBoot –fconfig.ini
```

Appendix B.5.10. Set Log File -a

This switch allows the logging to be directed to a file other than RUIBOOT.LOG . This switch may be used in single shot or continuous mode.

```
RuiBoot 192.168.1.10 255.255.255.0 –anewlog.txt
```

Appendix B.5.11. Log Level (Verbosity) -k

This switch is used to set the level of detail that is kept in the log. Very few log records are kept when the level is set to 3 and this setting is used to minimize the log file size when using continuous mode. The default log level is 2. The ***–k*** switch can be used with continuous and single shot modes.

```
RuiBoot 192.168.1.10 255.255.255.0 –k3
```

Keep tabs on the size of the log file, especially when in continuous mode. You can delete the file at any time. The application will create a new log file next time it needs to append a log record.

Appendix B.5.12. Allow old version of 8051BP03 to make boot request -x

Use this switch to force the RuiBoot application to respond to requests from an older version of 8051BP03. This option is only required if IP allocation fails because a FieldServer has an older version of 8051BP03. This command is used in single shot mode and is ignored in continuous mode even if specified.

```
RuiBoot 192.168.1.10 255.255.255.0 –x
```

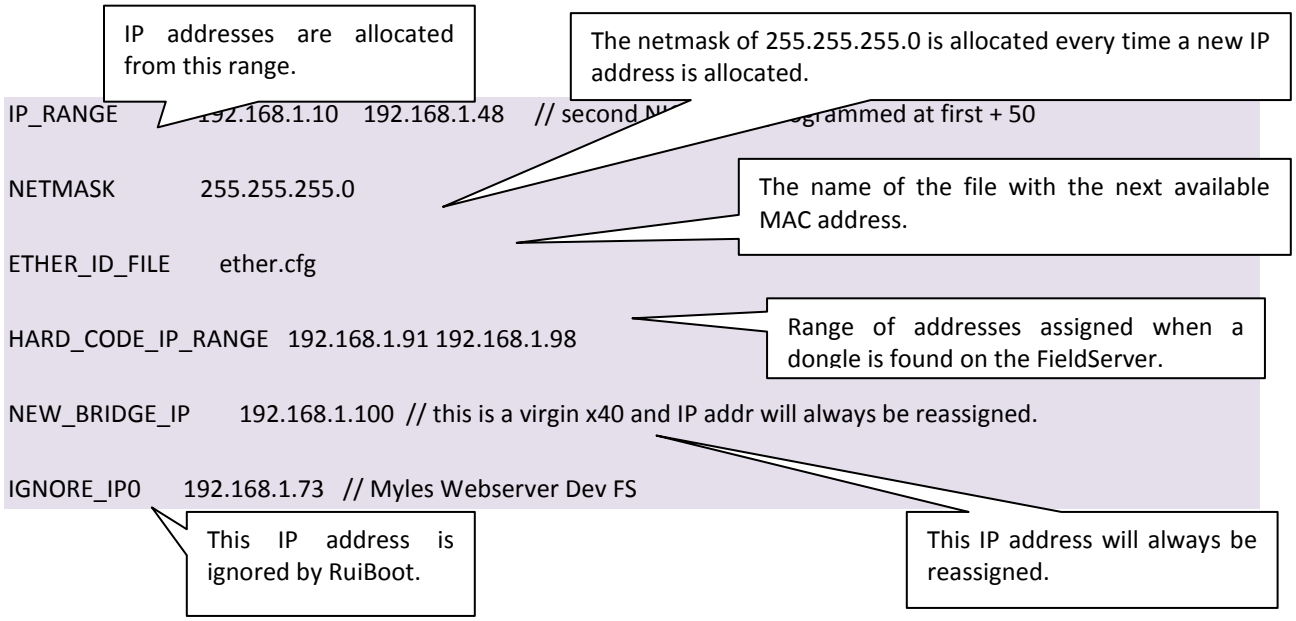
Appendix B.6. Continuous Mode - Configuration Files

The advanced use of RuiBoot requires two configuration files. One is specified in the command line and the other is specified in the configuration file.

When running in continuous mode RuiBoot continuously pings and builds up a list of in-use IP addresses. It takes a second or two for RuiBoot to ping all the FieldServers.

The configuration file is a text file. Each line begins with a keyword and each keyword is followed by a space and then one or more parameters.

Example.



Keyword	Description
IP_RANGE	This is the range of IP addresses that are used when a FieldServer needs to be allocated a new address. The FieldServer will be rebooted and allocated the next available IP address.
NETMASK	When the IP address is reassigned, the specified netmask is also assigned.
ETHER_ID_FILE	If the MAC address if the FieldServer is invalid, a new Mac address is automatically allocated by RuiBoot. The next available Mac address is obtained from the file specified by this keyword. The file is updated by RuiBoot every time a MAC address is allocated to prevent duplicate IP address allocation.
HARD_CODE_IP_RANGE[9]	If a FieldServer dongle is connected to the parallel port on the FieldServer then the FieldServer will be allocated the IP address corresponding to the dongle. There are 8 different dongles numbered 1 to 8. If dongle 3 is used then the 3rd address in the range is allocated to the FieldServer. This keyword specifies the range of IP addresses allocated to the dongles
NEW_BRIDGE_IP	A FieldServer with this IP address will automatically be assigned the next available IP address from the range specified in IP_RANGE.
IGNORE_IP0 to IGNORE_IP9	UP to 10 IP addresses can be specified as ignored. FieldServers with these IP addresses are not reassigned IP addresses or rebooted if their IP address is found to be duplicated.

Appendix B.6.1. Continuous Mode - Menu's

Push the 'C' key when RuiBoot is on continuous mode to display the following menu.

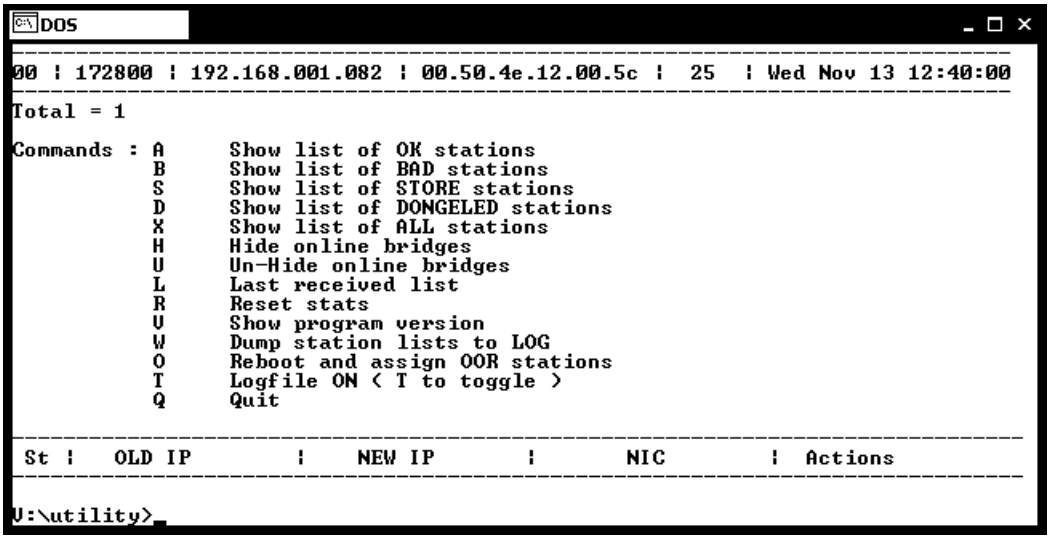


Figure 8.7

Command	Description
A – Show OK Stations	Lists all stations with no problems.
B – Show BAD Stations	Lists all stations that are bad. This does not include Out of Range Addresses.
S – Show List of Store Stations	Fit a dongle to a goof IP and RuiBoot will reboot the FieldServer and allocate the associated Dongle IP. The pre-dongle IP is stored so that when the dongle is removed then RuiBoot can reboot the FieldServer and allocate the pre-dongle IP address.

[9] This is only applicable to FieldServer models with multiple parallel ports. Refer to the Driver manual for further information.

Command	Description
D – Show List of Dongled Stations	List of FieldServers that currently have a dongle attached.
X – Show All	Lists all FieldServers being monitored.
H – Hide	Tells RuiBoot not to list the current set of FieldServers. This is used to remove messages which will clutter the diagnosis of a problem with one FieldServer.
U – Un-Hide	Tells RuiBoot to make all FieldServers available to the various listing, logging functions.
L – Last Received List	Lists all FieldServers whose *ping_cache_time* is greater than zero and whose last reboot time is greater than zero.
R – Reset Stats	Clears stats. When IP addresses are added to lists it appears that the addition is latched. Resetting unlatches the IP's and allows addresses to be removed from the lists,
V – Version	Prints the RuiBoot Utility version number.
W – Dump stations list to Log	Send the 'X' list to the log file.
O – Out of Range (OOR)	Reboots all the Out of Range IP addresses and allocates them In range addresses when they restart.
T – Log File Toggle	Turns the disk log file function on/off.
Q – Quit	Quits the application.

Appendix C. Reference

Appendix C.1. General Parameter Descriptions

Appendix C.1.1. Timeout Values

Timeout values may be defined for a Map Descriptor (lowest level), node or connection (highest level). If the timeout is not explicitly defined at a lower level then the driver uses the timeout defined at the higher level.

Appendix C.1.2. Probation delay

When a node is offline the driver tries to re-poll the device. Probation delay determines the length of time communication needs to be re-established for before the node is marked on-line again. The driver polls the device at the normal scan interval during the probation delay. The probation delay only affects the node status of a client.

Appendix C.1.3. Multidrop mode

If two FieldServers, each emulating a single node are on the same connection, one FieldServer emulates node 10 and the other node 11. If a remote device polls for data on node 11, the first FieldServer would normally respond with a no-node message (as it does not know that further down the line, a node 11 device exists). This behavior can be suppressed by defining the connection a multi-drop mode connection. RS-485 connections (R1-R2) force this parameter to “enabled” by default.

Appendix C.1.4. Inter character timeout (IC Timeout)

Used by Serial drivers to detect message failure once the first byte of a message has been received. When a driver receives a byte the IC_Timeout timer is reset. The next byte in the message must be received before this timeout completes or an IC Timeout is generated. Default value is 0.5seconds.

Appendix C.1.5. Data Caching

The first time an Client device polls a Server device for data via a FieldServer, the Client device is put on hold while the FieldServer polls the Server device. The FieldServer sends the reply to the Client device and creates a cache Map Descriptor where the data is stored. The FieldServer now automatically polls the Server device and keeps the buffer up to date. If the data is current on the next data poll, the FieldServer can respond immediately with data in the buffer. The ***Cache Age Limit*** determines the maximum age of the data that can be returned immediately.

- A *Cache hit* statistic is registered when the Client device polls for this information and the FieldServer can respond immediately with data in the buffer.
- A ***Cache miss*** statistic is registered if the data in the buffer is too old when the FieldServer is polled. The FieldServer triggers an immediate re-read of the data and increments the cache miss statistic.
- A ***Cache created*** statistic is registered if a new cache block is created for the data once it is updated. In this case both the cache miss and cache created statistics will be incremented.

It is important to optimize the configuration to minimize the number of cache blocks created and missed.

If the Client stops polling for this data, the FieldServer will continue to poll the Server for a further 5 minutes before removing the cache block as unused.

The ***Cache Size*** parameter determines the maximum number of cache Map Descriptors that can be created.

When a new cache Map Descriptor creation attempt is made but the maximum number already exists, the oldest cache Map Descriptor gets bumped to create room for the newest and the ***Cache Bumped*** statistic is incremented.

Appendix C.2. Default settings for parameters

Parameter	Default Setting
Default response timeouts	2000 ms = 2 sec
Inter character timeout	500 ms
SCADA hold	2000 ms = 2 sec
Data cache age limit for acceptable data	20000 ms = 20 sec
Cache	80
Retry Interval	10000 ms = 10 sec
Recovery Interval	30000 ms = 30 sec
Probation Delay	60000 ms = 1 min
Scan Interval	1 second
Poll Delay	50 ms
Retries	3
Activity Timer	120000 ms = 2 hour
Parity	None
Baud	9600
Data Bits	8
Stop Bits	1
Handshake Timeout	2000 ms = 2 sec