



# TCP/IP User's Guide Vx.4

This manual provides information on how to setup, program, and interface IBC readers and controllers with internal tcp/ip communication chips. This is a supplemental manual which should be used along with the manual for your specific reader.

Additional copies of this manual can be obtained by contacting IBC or an authorized distributor. This manual may not be copied or duplicated in any way without the express written consent of IBC.

Any errors or omissions from this manual should be noted and sent to the Technical Services dept. of IBC for correction.

If you need any additional information concerning the tcp/ip readers, contact IBC's Technical Support department from 08:00 to 17:00 Eastern Standard Time, at 860-659-9660, or e-mail us ([support@interbar.com](mailto:support@interbar.com)). Technical information and update information is also available on the internet at our home page <http://interbar.com>.

IBC Document ID: UG-TCP  
Version: x.4  
Revision: 0.0  
Date: 07 MAR 11

## **Revisions**

The following firmware versions contain new features which are supported by Version 4.4 of the utility software:

V 3.4	J, SA/STA, DC readers without POE
V 4.4	7000 and 8000 series
V 5.4	J, SA/STA, DC POE readers
V 6.4	Jx, SAx, STAx, DCx series
V 7.4	901 Fingerprint

New enhancements include:

- An Inactivity Timer which will automatically shut down a server connection due to inactivity
- An option to allow multiple server connections. A second connection automatically shuts down an earlier connection
- An option to turn Keep Alives on or off
- An option to restart the system if a physical link disconnect occurs
- An option for the client system to switch to a second server if the main server is unreachable

Version 4.4 of the IBC TCP/IP Utility program is required for programming readers and controllers with version x.4 firmware. Earlier versions of the utility will not allow programming of these new options.

## Introduction

IBC readers and controllers with tcp/ip connectivity contain an additional processor which handles tcp/ip, udp, and other traffic over a network.

This processor "chip" is independent from any processor chips which are running the reader or controller operations, and it is programmed independently using the IBC Tcp Utility software.

Changes made to the operation of this chip do not affect the operation of the reader or controller in any way, other than to change the tcp/ip (or udp) communications layer.

Once a connection is made through a tcp/ip socket or udp connection to an IBC device containing this chip, normal serial communications to and from the reader or controller can occur. Basically, think of the tcp/ip processor chip as providing a "bridge" for tcp/ip or udp communications to the reader or controller.

The TCP/IP firmware in readers and controllers is constantly being updated with new features. This firmware can be loaded into the tcp embedded chip by using the IBC TCP utility.

Note that within this manual, the term "reader" refers to any IBC device which a TCP interface, including readers, controllers, fingerprint readers, and other devices.

## Operating Modes

The reader can operate in one of three modes:

- Tcp/ip server
- Tcp/ip client
- Udp (peer-to-peer)

In the TCP/IP modes, a *logical connection* between the reader and another device must be established before communications can occur. This is done by having one device operate as a *server*, and the other device operate as a *client*. If you are unfamiliar with these terms, the concept is very simple. One device is passive, and the other is active. The server is the passive device. It simply waits for other devices (clients) to connect to it. Once it sees a client wants to connect to it, it accepts the connection, and data traffic can now flow between the two devices.

The client on the other hand is active. It starts the connection by contacting a listening server with a message to open a connection and start data communications. The server then either accepts or denies the request.

IBC readers which operate in the **server** mode are passive. They wait for a pc (or device) to connect to it before any data transfer can occur. The pc connects to the reader by sending out a connection request to the ip address of the reader. The ip address can either be a static address programmed into the *Reader IP Address* parameter of the reader, or an address assigned through DHCP.

IBC readers which operate in the **client** mode attempt to connect to a server which is "listening" and waiting for clients to connect. The server that the reader attempts to connect to is designated by the ip address stored in the *Server Address* field of the reader parameters.

If the reader is in the **server** mode, you have the option of allowing *multiple connections*. Prior firmware versions allowed only one connection (see the Multiple Connect parameter later in this manual).

If the reader is in the **client** mode, you have the option of connecting to one of 2 servers, depending on which one is available (see the Multiple Servers information later in this manual).

Any TCP connection between the reader and host (whether the reader is a server or client) remains exclusive until the connection is closed either explicitly or implicitly.

The Udp mode operates differently than the tcp modes. In the TCP modes, one device needed to be a server, and the other device a client. Once one device signaled to the other that it wanted to establish a communications connection, the other device responded and a connection (logical link) between the two was created.

UDP does not use the concept of connections. It is a *connectionless* protocol. Basically, both devices operate as peers, and send messages to each other using the UDP protocol. It is similar to "texting" on a phone. The major difference between UDP and TCP is the fact that TCP using an acknowledgement scheme for data packets. If a data packet is not acknowledged by the receiver within a certain period of time, the TCP sender will resend the data packet. This is built into the protocol. UDP has no such automatic acknowledgement built into the protocol.

On a healthy network, however, UDP is just as reliable as TCP and provides faster throughput.

IBC readers which operate in the **udp** mode send their messages to the device programmed into the *Server Address* field of the reader parameters, and will receive messages only from that address.

We recommend udp over tcp/ip when used on internal networks that have little congestion. For devices that will be communicating over multiple internal subnets, or directly over the internet, tcp/ip is the recommended approach.

The question of whether readers should operate as clients or servers is highly dependant on network topology and software design. Each one has it's own distinct operational advantages.

## Ports

Connections to and from devices on a network are achieved by using what are know as *virtual* ports. To talk with a de-

vice, you need to know not only the IP address that you want to talk with, but also the port number that you want to talk with. This is the same for both tcp/ip devices, and udp devices.

Ports on a device are identified by a unique number. For example, on the internet, most browser traffic (http) uses port 80. Telnet uses port 23.

There are 3 of these virtual ports in the IBC tcp/ip readers which are readily accesible from the network. Two of these ports are what we call **data ports**, and 1 is referred to as a **control port**.

The data ports are the ports where normal traffic (communications to and from the reader) will occur. By default, we use port 57 for tcp server and client connections and port number 53002 for Udp connections. When you want to connect to a reader which is operating as a server, you will connect to port number 57 (by default).

When a reader operating as a client connects to a pc, it will request a connection on port 57 (by default). To talk with a reader operating as a Udp peer device, you will connect to port number 53002 (by default). These ports are programmable options in the reader and you can change them easily usng commands or using the IBC utility.

The third port in the reader is called the *control port*. The control port is used for programming the reader, and also optionally for controlling data connections. More about this will be discussed later.

Note that the tcp server functions and tcp client functions share the same data port. This means that the reader cannot operate as both a client and a server at the same time. The reader must be specifically programmed to operate either as a client or a server if you are using tcp.

Although Udp uses it's own data port, the reader cannot accept udp messages and tcp/ip messages at the same time. In order to accept and respond to Udp messages, the reader must be programmed to operate in the Udp mode.. The 2nd data port used for Udp communications is set up as a 2nd data port only so that it is not

confused with the tcp data ports.

Remember that the embeded tcp/ip chip in the reader operates independently from the reader itself. When you are communicating using the data ports, you are talking directly to the reader, and not to the embedded tcp/ip chip. When you use the control port, you are talking directly to the embedded tcp/ip chip.

Note that in certain installations where firewalls are running, it may not be possible to connect to specific ports or possibly even to the default ports in the reader. You should check with your system administrator prior to installing the reader to ensure that you can connect to the programmed reader ports.

Do **not** use port 53001 for any traffic. Port 53001 is a reserved port which is used by IBC readers for multicast traffic. This is the method by which the utility program "discovers" IBC readers and controllers over the network.

## Tcp/Udp Protocols

Tcp and Udp are completely different protocols as described earlier. The tcp protocol requires a connection between devices, which normally infers a one-to-one relationship. The protocol itself maintains control over the connection by making sure that each side is alive and each side acknowledges transmissions. The Udp protocol, on the other hand, does not maintain a connection and does not verify receipt of transmitted data.

Note that although tcp/ip is an ack/nak protocol based transmission, it is still possible to loose data with tcp/ip connections. Physically connecting a tcp/ip reader to a network does not imply that reader data is automatically transmitted to a "receiver" on the network. For tcp/ip communications to operate properly, there needs to be a "socket" connection between the reader (server) and a client.

For example, if you were to take an rs232 output reader, power it, and scan barcodes without connecting the reader to the serial port the data would be lost. The

same thing will happen if you connect a tcp/ip reader to a network and you do not have a server-client connection established.

For "J" series readers, the server-client connection needs to be active at all times. For stand-alone readers or controllers, the connection needs to be active only when uploading and downloading occur.

With the Udp protocol, there is no acknowledgement within the protocol layer and therefore Udp is potentially less reliable than tcp/ip, however on well desinged networks udp is just as robust as tcp. UDP is also recommended when multiple readers need to talk wit the same pc. The development and implementtion of software using UDP for this purpose is much quicker.

## Reader Parameters

The following parameters can be reprogrammed in the tcp/ip reader:

- IP Address
- Subnet
- Data Port Number
- Control Port Number
- Connection Speed
- UDP Port Number
- Password
- Gateway Address
- Server Address (1&2)
- Server Port Number (1&2)
- DHCP on/off
- Reader Identifier
- Data Communication Password Control
- Inactivity Timer
- Keep Alives on/off
- Physical Link monitor on/off
- Automatic Ident transmission
- Multiple connections on/off
- Multiple Clients on/off

These items are defined in detail below:

## **IP Address**

This is the unique IP address given to the reader. The IP address must be unique on your local network. The default IP address in the reader is 192.168.1.55.

## **Subnet**

This is the subnet value used for your network. Most people use a Class C subnet which is 255.255.255.0. This is also the default subnet programmed in the reader.

## **Data Port Number (tcp)**

This is the port you will use for communications to and from the reader when the reader is operating as a Tcp Server. The default Data Port number is 57.

## **Control Port Number**

This is the port used for programming and sending control commands to the reader. The default Control Port number is 87.

## **Connection Speed**

The connection speed can be set to:

- 10BaseT Half Duplex
- 10BaseT Full Duplex
- 100BaseT Half Duplex
- 100BaseT Full Duplex
- Auto Negotiate

Note that Auto Negotiate is different than Auto Sense. An Auto Sense system senses the speed only. Auto Negotiation is a system where both devices negotiate to come up with an acceptable speed. Some older switches or Access Point Routers use the Auto Sense system. If you program the reader to Auto Negotiate and connect it to an Auto Sense system, the

chances are pretty good that you will not be able to establish a connection.

The default speed setting for the reader is AUTO.

## **Data Port Number (udp)**

This is the port for incoming UDP traffic to the reader. This is not the outgoing traffic port. The default Udp incoming port is 53002. This means that if you want to send a Udp message to the reader, you will send it to port 53002, or whatever port you reprogram the reader to for Udp incoming traffic.

## **Password**

There is a 5-character password programmed into the reader. Before you can program the reader parameters, you must send the correct password to the reader. The password can be reprogrammed and *we encourage you to do so*.

There is also a password protection option in the reader, which uses this same password. If the password protection option is turned on - data cannot be received by the reader on the tcp data ports until the correct password is sent over the control port. The Udp data port is not affected by the password protection.

The default password is 11111.

## **Gateway Address**

The Gateway Address is the address of the router in your system which directs traffic. The default setting is 192.168.1.1 . You only need to change this address if you have a router which uses a different address. If you have multiple routers in the system and you are attempting to connect to readers across multiple networks, then the gateway address should be set to the router that is acting as a bridge or router to the other network.

## **Server Addresses 1 and 2**

The Server Address parameter serves multiple purposes. When the reader is configured as a *tcp client*, the reader will attempt to establish a connection with this address. The Server Address will have to be programmed to the IP address of the pc which will be listening for the connection.

In Udp mode, the Server Address parameter represents the IP address of the reader's peer, or the device it will be talking to. When you set this parameter and the reader needs to transmit a Udp message, that message will be sent to the IP address programmed into the Server Address parameter. In addition, because it is assumed that the reader's udp transmissions are meant to be received by only one receiver, *all udp messages* received by the reader are checked to be sure that they originated from the same IP address programmed into this Server Address parameter. In other words - when you program the reader to operate in the Udp mode - even though it is connectionless, it is assumed that the reader is sending and receiving data from only one IP address - the IP address programmed into the Server Address parameter.

The default Server Address is set to 192.168.1.102

Version x.4 of the firmware allows a 2nd server address to be entered in. If the option is set to use 2 servers, then if the first server is not available (in client mode), the reader then attempts to connect to the ip address programmed into the Server Address 2 field.

Note that in UDP mode, the reader can transit to only one server. Server Address 2 is ignored in UDP mode.

## **Server Port Number**

In the Tcp Client mode, the reader attempts to establish a connection with the pc that has the same IP address as that which is programmed into the Server Address parameter. It will attempt to connect to that pc using this port number.

In the Udp mode - when the reader sends a message to the pc using the Server Address, will also send the data to this port.

The default Server Port number is 57.

Server Port 2 is the port number used for the 2nd server address, when the Use 2 servers option is selected.

## **Dhcp**

Dhcp is a protocol which is used for obtaining dynamic IP addresses (from a router or a dhcp server). In some systems, people use a static IP address (one which never changes), or a dynamic IP address which can change every time the device is turned on. Depending on the type of network that you are running, you may wish to turn Dhcp on. By default, Dhcp is turned off in the reader, which means that the reader is always addressed with the same IP address.

Note that if you turn DHCP on and will be programming the reader to operate as a Tcp Server, the IP address may change each time the reader is turned on. For this reason, when the reader is run as a server, we recommend not using DHCP.

If DHCP is turned on, but there is no DHCP server in the network, the reader will default it's ip address to the ip address programmed into the "Reader ip address" field.

## **Reader Identifier**

There is a 20-character reader identifier stored in the reader, which can be used to "name" or identify a reader. Although IP addresses or Macs can be used to uniquely identify a device - the IP or Mac address does not give a descriptive identification of the device. If you program the Reader Identification parameter, you can use it to identify the reader in a descriptive way. By default, the Reader Identification is set to "ibc reader".

## **Data Communication Password**

There is an option to turn on data password protection in the reader. In this mode, the reader will not accept any incoming traffic unless the proper password has been sent to the Control Port. If you want to make sure that no one can communicate with the reader in any way, unless they have the proper password, then you will want to turn this option on.

## **Multiple Connect option**

If the reader is operating as a server, you can optionally program the reader to accept multiple incoming connections from a client.

This option is not meant to allow multiple clients to talk to the reader at one time or to allow one client to have two connections with the reader at the same time.

The purpose of this option is to allow a client to reconnect to the reader when an already open connection exists. In rare occasions, tcp/ip connections can get into a half-open or half-closed state. This happens when one side believes the connection is open and established, and the other side believes it is closed.

Here's a simple example of how this can happen. A reader is set up as a server, and a pc (client) connects to it. The pc at some time decides to close the connection but when the close command is sent to the reader, the reader is not accessible because one of the switches has lost power. The pc may software may sense this, because an acknowledgement did not come back from the reader. The reader, however, never received the close message and therefore believes that the connection is still open. This is called a half-closed connection.

If the option for multiple connects is off (default), then if the client pc believes the connection is closed and attempts a reconnect, the reader will not accept the connection because it believes there already is a connection open.

If the option for multiple connects is on, then the reader will accept the connection, and it will close the earlier connection. Communication can then continue as if nothing happened.

It is important to note that this option has been added as one way of dealing with half closed connections. It really should not be used unless you are experiencing connection issues, and the other methods of detecting connection failures (keep-alives and inactivity timer) are not working.

## **Keep-Alives option**

Keep-alive refers to a tcp packet of information that has no data. When a keep-alive packet is sent to the other device, the other device will acknowledgement receipt of the packet.

In a system where the reader may have long periods of inactivity, the keep-alive may be a useful tool. Although tcp connections are supposed to logically remain connected indefinitely, in some cases they may remain logically connected but physically disconnected.

If the keep-alive option is on, then the reader will transmit a keep-alive message for every minute of inactivity. If the other end does not respond, the reader will close the connection and wait for another connection (if it is a server), or attempt another connection (if it is a client).

The keep-alive is the preferred method of detecting half-closed or severed connections, but it may not work in all cases.

Some programmable switches respond to keep-alive packets directly. If a switch responds to the reader keep-alive packets directly, but there is no path to the reader because another switch in the path is down, then the reader believes that the other device is still there (because the keep-alive is being acknowledged) . In this case, the reader keeps the connection open, but the physical connection has been severed.

If you are using keep-alives as a method of maintaining a connection, then it is impor-

tant to evaluate the switches in your system to ensure they are programmed to process keep-alives in the proper way to achieve your desired results.

### **Inactivity Timer**

The inactivity timer is a method of detecting half-closed connections. If the inactivity timer is set to a non-zero value, then the reader will reset a timer after every character that is transmitted or received. If the timer period expires, then it is assumed that there is a problem with the connection, and the reader will close the connection.

The Inactivity Timer parameter is programmable in seconds, from 01 thru 99.

If your software is running a "heartbeat", then the Inactivity Timer is the preferred method of detecting severed connections, as both sides will detect the connection issue at the same time.

### **Use 2 Servers option**

Applies to client mode only. If this option is on, then if the reader is unable to connect to the first server (designated in *Server Address*), then it will attempt to connect to the server designated in *Server Address 2*.

If *Server Address 2* is not available, it will go back and attempt a connect to *Server Address*.

### **Phy Monitor option**

If this option is on, the reader will monitor for a physical link on the tcp line. If a link goes away, the reader will restart. Note that the restart is not a graceful restart, since any connections which were open cannot be notified of the closure because the link was lost. The default for this option is off.

### **Client Send ID option**

Applies to client mode only. When a reader connects to a server, if this option is on, it will send a message immediately identifying itself to the server. The message consists of the 20 character reader *identifier*.

### **UDP Send ID option**

Applies to udp mode only. If this option is on, every udp packet sent by the reader will be preceded with the *identifier* field.

## **Password Protection**

There are two levels of password protection in the reader. The first level is the default level which requires that you always send a password to the reader in order to program the reader.

The second level of password protection involves the *data connections* themselves. If you wish, you can also password protect these data connections. If you do this, you cannot talk to the reader through the tcp data ports without first sending the password to the control port. This protects the reader from being connected to by unauthorized sources.

Note that if you turn the data password protection on, you must connect to the control port and send the password *prior* to any data transmission on the data port, and you must keep the control port connection active while communications are in progress.

The Data Port password protection applies to reader receive transmissions only, and not to reader transmissions. The reader can still transmit data to the programmed receiver even if the password has not been entered. The Data Port password protection is designed to keep unauthorized parties out of the reader, not

to keep the reader from transmitting it's data to a legitimate source.

The Data Port Password Protection applies only to the tcp ports, and not the Udp ports. It is assumed that if you are using Udp then the Server Address parameter is protection enough to ensure that data traffic is legitimate.

Use 2 Servers	OFF
Client Send ID	OFF
UDP Send ID	OFF
Phy Monitor	OFF
Keep Alives	OFF
Mult. Connect	NO

## Programming

Programming the reader is done by connecting to the Control Port (tcp port 87 by default), and sending the proper programming commands. You can also use the IBC Tcp utility program to program the reader.

Prior to sending any programming commands, you must send the reader the PASS command followed by the 5 character password for the reader. The reader will not accept any programming commands unless the proper password is sent.

Once you send the PASS command with the proper password, you can then use the programming commands to program the reader.

The defaults for each of the programmable parameters in the reader is as follows:

IP Address	192.168.1.55
Subnet	255.255.255.0
Data Port	57
Control Port	87
UDP Data Port	53002
Password	11111
Gateway Address	192.168.1.1
Server Address	192.168.1.102
Server Port	57
Server Address 2	192.168.1.103
Server Port 2	53050
DHCP	OFF
Speed	Auto
Reader Identifier	ibc reader
Mode	Server
Password Req'd	OFF
Inactivity Timer	00 (OFF)

Prior to changing any of these parameters, it is suggested that you first connect to the reader using the default settings, and verify reader operation. If you are passing through a *router or firewall* to get to the reader, you may have to set up the firewall or router to allow communications with the default port 87 in order to do this.

The programming commands are listed later in the manual.

Note that each of the programming commands sent to the reader must be followed by a c/r (hex 0d). The reader will respond with OK if the command is accepted.

## How to Use the Reader

Depending on whether you are running tcp or udp, the methods of connecting and communicating with the reader will vary. The following are some examples of how to communicate with the reader in the different modes. These examples assume that you are using the defaults which are set in the reader.

### Reader as a Tcp Server

The reader is set up as a Tcp server which means that you must establish a connection to the reader before any data can be transferred back and forth. The steps to do this are as shown below. Those items marked with an \* are done only if the data port password protection is turned on.

1) Open a tcp socket in your software for

- the data communications.
- 2) Open a tcp socket to the control port\*
  - 3) Set the remoteport property for the data connection to 57.
  - 4) Set the remoteport property of the control connection to 87\*
  - 5) Set the IP address for the Data connection
  - 6) Set the IP address for the Control connection\*
  - 7) Invoke a connect request on port 87 wait for the connect and send the password\*
  - 8) Invoke a connect request on port 57 wait for the connect and start data communications.
  - 9) When completed, close both ports.

### **Reader as a Tcp Client**

The reader is set up as a Tcp client which means that the reader must establish a connection to the pc before any data can be transferred back and forth. In order to do this, you must *listen* for the data connection and then accept the data connection when it arrives. The steps to do this are as shown below. Those items marked with an \* are done only if the data port password protection is turned on.

- 1) Open a tcp socket in your software for the data communications.
- 2) Open a tcp socket to the control port\*
- 3) Set the remoteport property of the control connection to 87\*
- 4) Set the IP address for the Control connection\*
- 5) Invoke a connect request on port 87 wait for the connect and send the password\*
- 6) Set the remoteport property for the data connection to 0.
- 7) Invoke a listen request on port 57 wait for the reader to connect and start data communications.
- 8) When completed, close both ports.

### **Reader as a Udp Peer**

The reader is set up as a Udp peer which means that the reader will accept and send Udp transmissions over the network from and to the specified programmed

host. The steps to do this are as shown below. Note that the Udp port cannot be protected by the Data Port protection password.

- 1) Open a tcp socket in your software for the Udp transmit.
- 2) Open a tcp socket in your software for the Udp receive.
- 3) Set the remoteport property for the receive socket to 0.
- 4) Set the localport property for the receive socket to 57.
- 5) Set the remoteport property for the transmit socket to 53002.
- 6) Set the IP address property for the transmit socket to 192.168.1.55.
- 7) Invoke a bind request on port 57
- 8) Transmit Udp data over the transmit socket.
- 9) Receive Udp data over the receive socket.

You do not need to maintain a "connection".

## **Maintaining Persistent TCP Connections**

TCP connections can become half-closed or severed for a number of reasons.

Usually, this occurs due to a power failure or link failure somewhere along the path between the reader and pc. Other causes could be intelligent switches dropping the connection due to inactivity.

If you are familiar with network topology and also familiar with the programming of all switches in the system, you can tune the TCP parameters in the reader to provide a most robust system. If you're not sure what type of switches or routers are in the system, it's best to rely on the most trusted way of keeping a connection open - which is keeping traffic on the line. This is referred to as a "heartbeat".

IBC tcp readers and controllers do not implement a heartbeat directly, but there are techniques that you can use, and

options that you can set, to aid in this process.

The TCP firmware has four methods of directly controlling and monitoring connections:

- Inactivity Timer
- Multiple Connections
- Keep-alives
- Physical Link Monitor

The inactivity timer can be used for both server and client tcp modes. If the inactivity timer is on (set to any value other than 00), then the timer runs in the background whenever any connection is made.

The timer monitors traffic and resets itself every time a character is received or transmitted. If there is no traffic in either direction for a period equal to the inactivity timer setting, the reader will automatically close the port, and wait for another incoming connection (server) or attempt another outgoing connection (client).

Note that when the reader closes a connection, a message is sent to the other end (pc) notifying it that the reader wants to close the connection. The receiver (pc) must also close the connection at it's end as well, otherwise a half-closed connection occurs. It is easy to detect this at the pc end because any query to the reader will be unanswered.

To effectively operate a heartbeat from the pc end, you can send commands to the reader or controller (such as a V version command) every few seconds. The reader or controller will respond back and you will know that the connection is still alive.

If you do not receive data back after a couple of attempts, then it means the reader is unreachable for some reason, and you can close the connection. Since the inactivity monitor on the reader will also detect the loss of incoming data, it will also close the port, and a re-connect can be initiated from either end.

The inactivity timer is the preferred way of handling connection issues.

You can also utilize the multiple connections feature, if the reader is running as a server. When this option is on, a request to connect to the reader is honored, even if there is already a connection open. It is assumed that the controlling process (pc) believes that the connection has been closed, but the reader does not. This can happen with a severed connection.

When the 2nd request to connect arrives, the reader will close the first request, and respond to the new open request and open the connection. The reader will do the same if it sees another open request. It will close the currently running connection, and start a new one.

Keep-alives are another way of maintaining an open connection. When this option is on, the reader will transmit a zero-length tcp packet every 1 minute of inactivity. The receiving pc will automatically acknowledge the zero-length packet, keeping the line alive.

Problems arise, however, with keep-alives when intelligent switches are involved that respond to the keep-alives themselves. This has the effect of keeping a connection open even if the pc believes it is closed. For this reason, keep-alives are recommended only when using standard non-intelligent switches. In that configuration, the keep-alive mechanism should be sufficient to maintain open idle connections.

The Physical link monitor runs in the background and monitors the link status of the tcp connection. It does not monitor logical connections, but rather the physical link. Depending on the type of switches used, the loss of a physical link does not necessarily mean the loss of a logical link or connection. If there is no heartbeat running, or no keep-alives running, then the line can remain physically disconnected without disturbing a logical connection, as long as no traffic is occurring.

In some network configurations, however, a link may be lost due to the dropping of the connection by a switch for a number of reasons. If you prefer to have all connections closed immediately by the reader when this happens, turn the physical link monitor on. The reader tcp firmware will

then restart whenever a link loss is detected. Note that the granularity of the physical link monitor is 10 seconds, therefore a link can remain out for 10 seconds before the monitor will detect it.

Because you have programmed the router to direct port 57 traffic to 192.168.1.55, the connection can be made.

If you want to allow incoming connections to more than 1 device, then each device will need to be set up with a different data port, so that the router can be programmed to redirect traffic for each of the ports.

## Communicating Across Networks

In order to communicate with a reader across networks, the correct *gateway address* needs to be set in the reader. The Gateway Address is the address of the router in your system which connects to the other network (internal or external). Traffic meant to flow to the other network first travels to the device at the gateway address which then redirects the traffic using the proper routing.

In you are operating the reader as a server, and want to accept incoming connections from another network, any router between you and the other network will have to be configured to allow the incoming traffic to get to the reader. This may involve programming the router to allow port forwarding. Port forwarding instructs the router to pass through all data meant for a specific port to a specific device (in this case, the reader).

For example, if your reader address is 192.168.1.55, on a network with a router address of 192.168.1.1, the gateway address should be set to 192.168.1.1. If you leave the data port at 57, you must program the router to allow all port 57 traffic to travel to the reader at 192.168.1.55.

From the outside world on the other side of the router (internet or another network), the incoming connection request cannot be made to 192.168.1.55, as that address is a local address. The incoming request must be made to the router address.

### Command Set for Versions 3.4, 4.4, 5.4, 6.4, 7.4

All commands sent to the reader must be followed by a carriage return (hex 0d).

When you send a command you will receive a response back from the reader. If the command asks the reader for information, such as the IBCIDENTIFY command, the reader data will be returned followed by a carriage return (hex 0d). If the command does not require the reader to return data, but simply to execute a command, the reader will respond with "OK" if the command was accepted.

Note that in all cases where you must send data, such as an IP address or a Port Number, do not blank fill the number. The only exception to this rule is the password which must be 5 characters long. For example, to change the Tcp Data Port to 100, send the command DPT100 and not DPT100 followed by 2 spaces.

If you reprogram any of the dynamic operating parameters, such as the IP address or a port number - the change is made permanent in the reader but the changes will not take effect until you either repower the reader, or issue the REBOOT command.

The following commands do not require that the password be sent:

IBCIDENTIFY Returns the reader's current IP Address followed by ":" followed by the reader's MAC address followed by ":" followed by the Reader Identification.  
For example, with the default settings, this command will return 192.168.1.55:mmmmmmmmmmmm:ibc reader where mmm... is the MAC address for the reader

PASSxxxxx This command sends the password xxxxx to the reader, so that the reader can be reprogrammed or other commands can be sent..

The following commands require that the password has already been sent:

CON Returns the reader connection speed the reader is connected at. This is not the programmed connection speed, but rather the actual speed that the reader connected at. The response can be any of the following: 10HD, 10FD, 100HD, or 100FD,

MAC Returns the reader's MAC address as 12 hex characters.

SPEEDx Changes the reader connection request speed to x, which can be:

- 1 - 10 half duplex
- 2 - 10 full duplex
- 3 - 100 half duplex
- 4 - 100 full duplex
- 5 - auto negotiate

NEWPASSxxxxx Changes the password to xxxxx

CIPxxx.xxx.xxx.xxx	Changes the reader IP address
DPTxxxxx	Changes the reader tcp data port number
CPTxxxxx	Changes the reader's tcp control port number
DHCPX	Turns DHCP on or off (0=off, 1=on)
DPASSx	Turns the Data Password Protection on or off (0=off 1-on)
CSNxxx.xxx.xxx.xxx	Changes the subnet value
GTWxxx.xxx.xxx.xxx	Changes the Gateway address
SIPxxx.xxx.xxx.xxx	Changes the Server IP address which is used for tcp client connections and Udp receive and transmit connections. This address is for the first server, if multiple servers are used.
UDPP XXXXX	Changes the Udp receive port number.
IDENTxxxxx.....	Changes the Reader Identifier (up to 20 characters max)
SPRTxxxxx	Changes the Server Port which is used for Tcp Client connections and Udp connections. For client connections this port refers to the port used with the first server address.
MODEX	Changes the reader operating mode to one of the following: 0 - Tcp Server 1 - Tcp Client 2 - Udp Peer
REBOOT	Reboots (restarts) the reader.
SWTxx	Sets the inactivity timer to xx seconds. Use 00 for no timer.
KEEPx	Turns keep-alives on (1) or off (0)
USE2x	Use 2 servers (1) or use only 1 server address (0).
PHYx	Turn physical link monitor on (1) or off (0).
MULTx	Allow multiple connects in server mode (1) or off (0).
SP2xxxxx.....	Set the 2nd server ip address.
SPR2xxxxx	Set the 2nd server port number.
STAT	This command returns all of the settings in the reader. Each value is sent without space padding, and each value is delimited with a ":". To parse the string, simply look for the colons and parse the data up until the point of each colon. The returned parameters (in order) are:  Current IP address setting MAC address

Reader Identifier  
Version Identifier (format xx.xx)  
Reader Programmed IP address  
Tcp Data Port Number  
Tcp Control Port Number  
Gateway address  
Server IP address  
Server Port Number  
Operating mode  
Speed setting  
DHCP setting  
reserved (up to 5 characters - do not use)  
Subnet mask  
Udp Port Number  
reserved (up to 5 characters - do not use)  
Data Password required setting  
Client send id setting  
UDP send id setting  
Inactivity Timer # or seconds  
Keep alive setting  
Physical monitor setting  
Multiple connect setting  
Server address 2  
Server port 2  
Use 2 servers settings  
Connected speed

When you update a parameter, the parameter is written immediately to the flash storage. If you intend to update multiple parameters, you can use the WAIT command prior to sending parameter update commands. The WAIT command causes the reader to update the data internally, but not write to flash. When you are finished updating, you can send the UPDATE command, which will then make the changes permanent. Note that you do not need to use the WAIT and UPDATE commands. They are there to facilitate speedier updating if you intend to update multiple parameters.

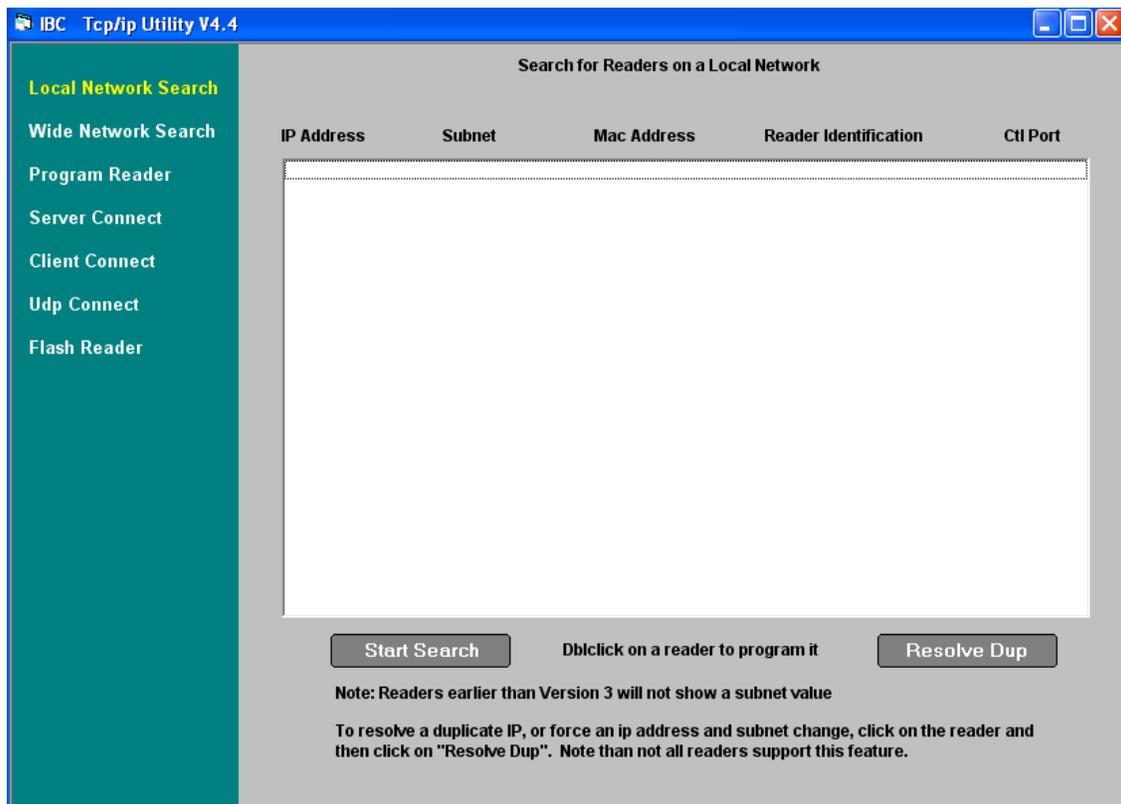
**NOTE** once you have changed parameters, you must unpower the reader and then power up again for the new settings to take effect, or you must issue the REBOOT command. Note that when you issue the REBOOT command you will lose your current connection because the reader will automatically close the connection.

## Using the IBC TCP/IP V4.4 Utility Software

The IBC TCP/IP Utility software can be found on IBC's internet software page. This software can be used for reprogramming the parameters in the reader as well as testing readers and searching for readers on the network.

The software has 7 screens for performing certain functions. Each screen is accessed by clicking on the appropriate screen function on the left side of the main screen. When you click on these items, the appropriate sub screen will be displayed.

### Search Readers on a Local Network



This screen allows you to search for tcp/ip enabled IBC readers on a local network. To search for readers click on the *Start Search* button. Any readers which are found on the local network are shown, with their IP address, MAC address, Subnet value, Control port, and the Reader Identification.

If you wish to reprogram one of the readers displayed, simply double click on that reader and the programming screen will appear with the IP address and default control port already typed in.

Although technically not allowed, it is possible that multiple readers connected to the

network have the same IP address. This will occur, for example, if you connect two new readers and you have not changed their IP addresses yet. On a local physical network only, the IBC utility software can resolve this issue. To change the IP address for one specific reader, click on the Resolve Dup button and follow the instructions.

**NOTE** this screen does not search IP addresses one by one. It uses an IGMP multicast command to identify the IBC readers. This IGMP command will not pass through routers which are not multicast enabled and therefore use of this screen is restricted to local physical networks only. The IGMP command is sent on Port 53001 and therefore this screen will not work if Port 53001 is blocked by a router or firewall.

The local network search can also identify readers which are not programmed with the same network id or subnet, as long as they are physically on the same network.

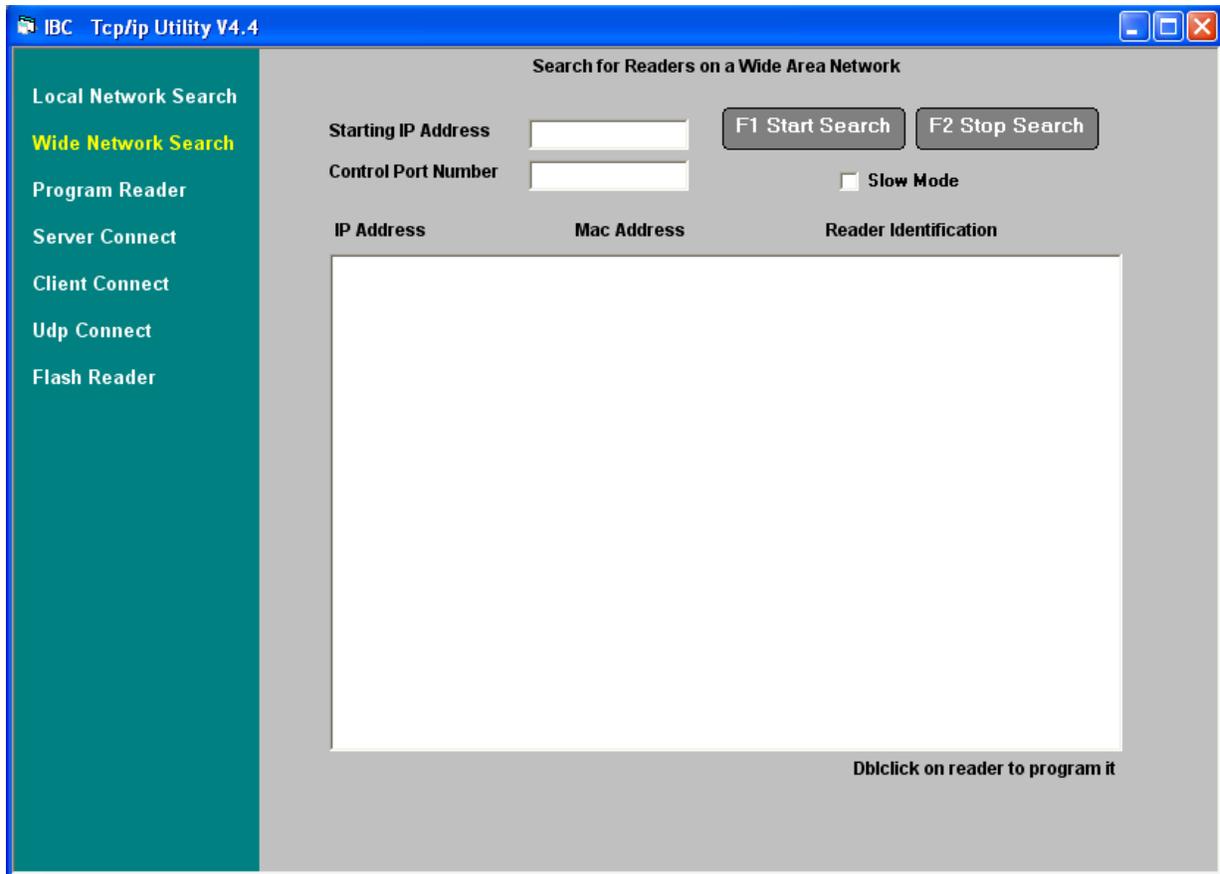
For example, you can have one reader configured as 192.168.1.55, subnet 255.255.255.0, and another reader configured as 10.19.20.20, subnet 255.255.0.0. The local search will find both of these readers, provided that the firmware in the readers is at least V2.2. Prior versions of reader firmware are not capable of responding to the search request if the pc is not on the same logical net.

It is important to note that in order for the search utility to work, the readers need to be physically connected to the same network and reachable without going through a router. If you need to go through a router, then multicast traffic must be enabled on the router in order to pass the traffic.

If you have a situation where a reader is programmed to a different net and you want to program it to the net you are currently connected to, follow this sequence:

- 1) Find the reader using the local search. Make sure the reader is physically connected to the local network.
- 2) Change the ip address of the reader to an address which is valid for your current local network. Do not worry about the submask because you can change this later. You do this by clicking on the Resolve Dup button and changing the ip address to a local address.
- 3) Search again to ensure that the ip address has been changed. After you know it has been changed, go to the Program screen and attempt to get the parameters and then change them.
- 4) Note that in some cases, depending on the subnet mask, it may still not be possible to directly communicate with the reader. If this is still the case, you can temporarily change the subnet mask on your pc to a class B subnet (255.255.0.0) and you should then be able to communicate in order to change the subnet mask in the reader.

### Search Readers on a Wide Area Network



This screen allows you to search for tcp/ip enabled IBC readers on a wide area network. To search for readers set the starting IP address to search, and set the Control Port number. Then click on the *Start Search* button. Any readers which are found on the network are shown, with their IP address, MAC address, and also Reader Identification.

This screen searches for the readers one-by-one, incrementing the IP address each time. By default, the screen will wait about 1 second to see if a reader responds to its request. On networks which are congested, it may take longer for a reader to respond. If you are not getting a response from a reader or have a congested network, click on the *slow mode* box, which will increase the wait time to about 3 seconds for each reader.

If you wish to reprogram one of the readers displayed, simply double click on that reader and the programming screen will appear with the IP address and default control port already typed in.

## Programming a Reader



To program a reader, enter in the IP address for the reader and also the Control Port Number. You will also need to enter in the 5-character password. Click *Get Parameters* and the current settings in the reader will be displayed. Note that the password field will be blank and the Mac Address, Firmware, and Connected Speed fields will not be enabled. Two other fields (UDP Log address and UDPO Log port) also will not be enabled. These are for future use.

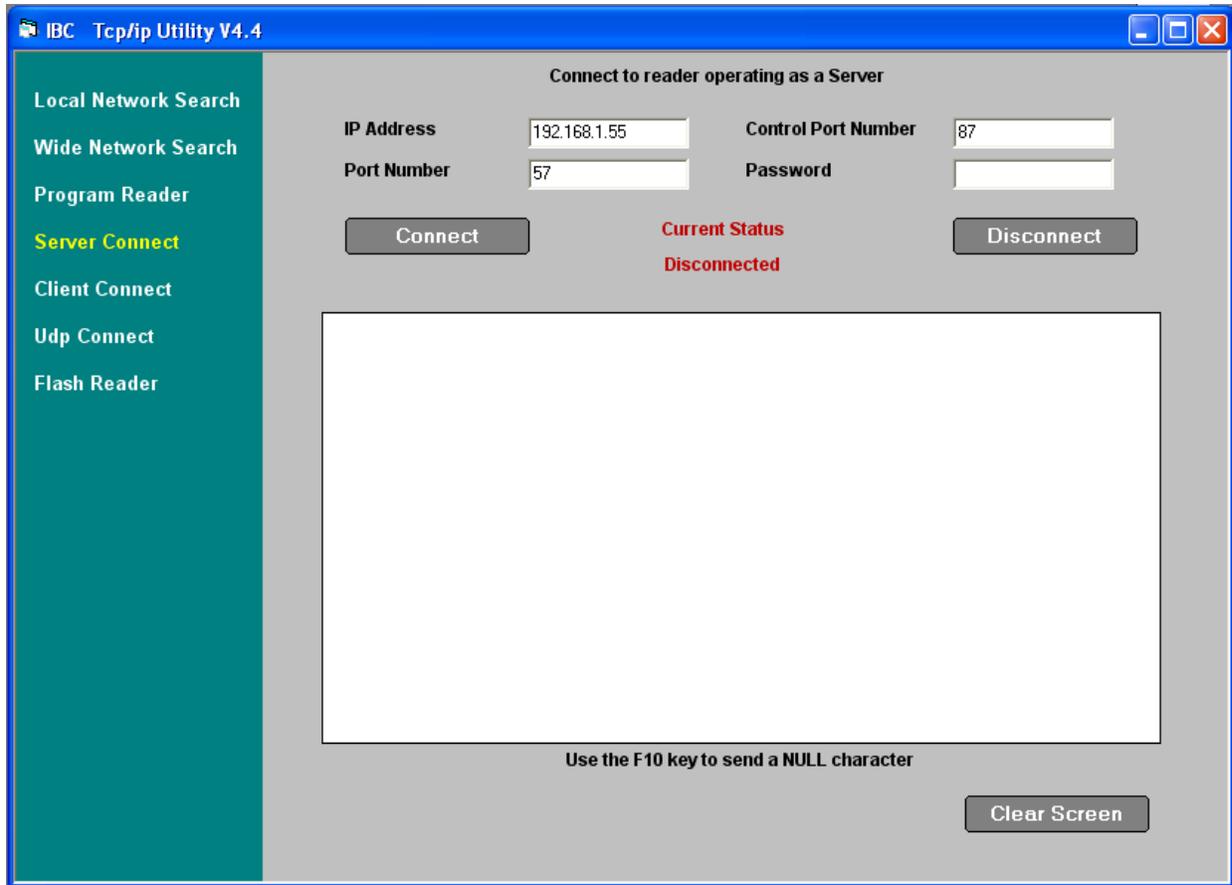
You can change any of the parameters on the screen and when you are finished - click on *Set Parameters* to start the reader programming.

If you want to change the password in the reader, enter a new 5-character password. If you do not want to change the password - then leave that field blank.

Note that if you forget your password - there is no way to connect to the reader or reset the password. If you change the password make sure that you write it down and store it in a safe place.

Once you have reprogrammed the reader - you will need to click on the *Reboot* button in order to restart the reader with the new parameter settings.

**Testing a reader which is set up as a Server**



This screen allows you to connect to a reader which is programmed to operate as a server. You must enter in the IP address for the reader and also the Data Port number. If you have the Data Port Password protection ON, then you must also enter in the number for the Control Port and the password, otherwise the screen will not be able to send and receive data.

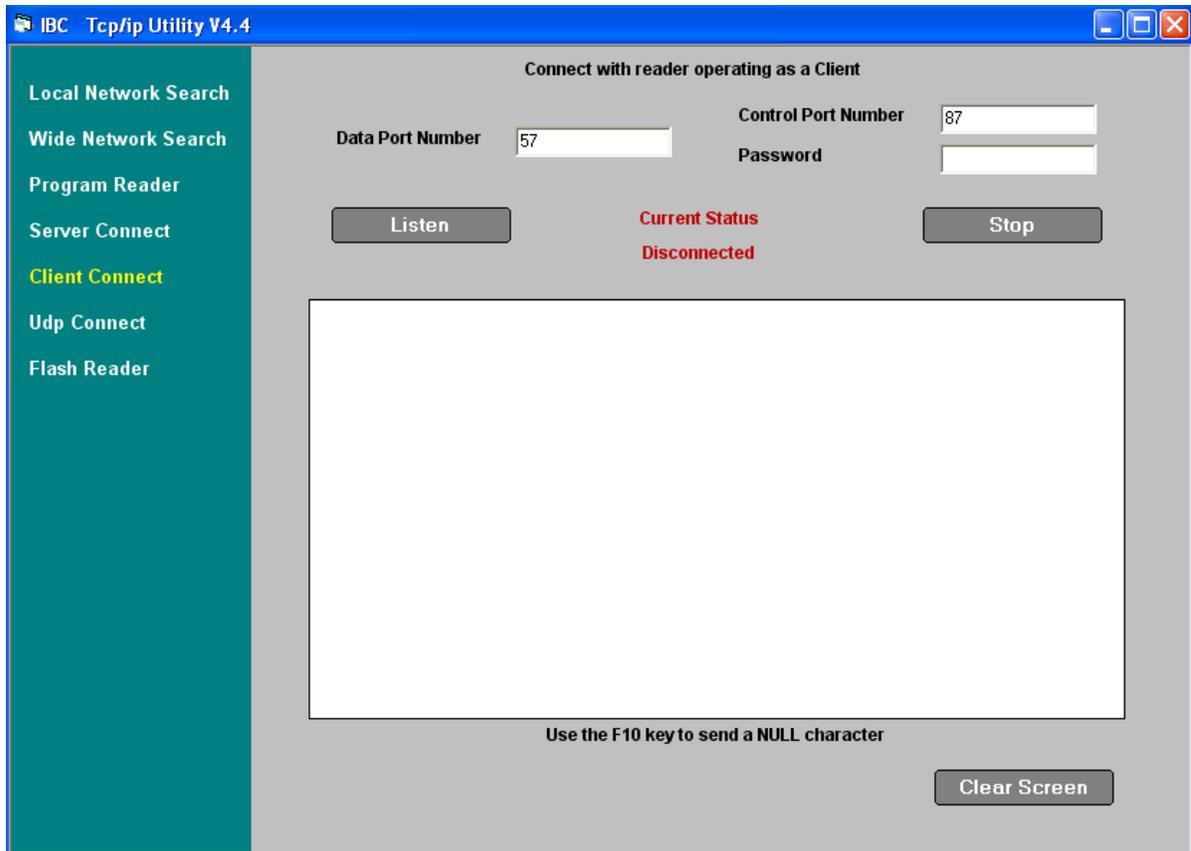
To connect, click on the *Connect* button. To disconnect, click on the *Disconnect* button.

Once you have connected, you can type data which will be sent to the reader, and any data received from the reader will be displayed on the screen.

NOTE that if you connect to the reader and then change screens without disconnecting - the reader will still remain connected and other screens potentially may not work.

If the connection is disconnected by the reader or the link is lost, the Current Status indicator will change from Connected to Disconnected.

**Testing a reader which is set up as a Client**



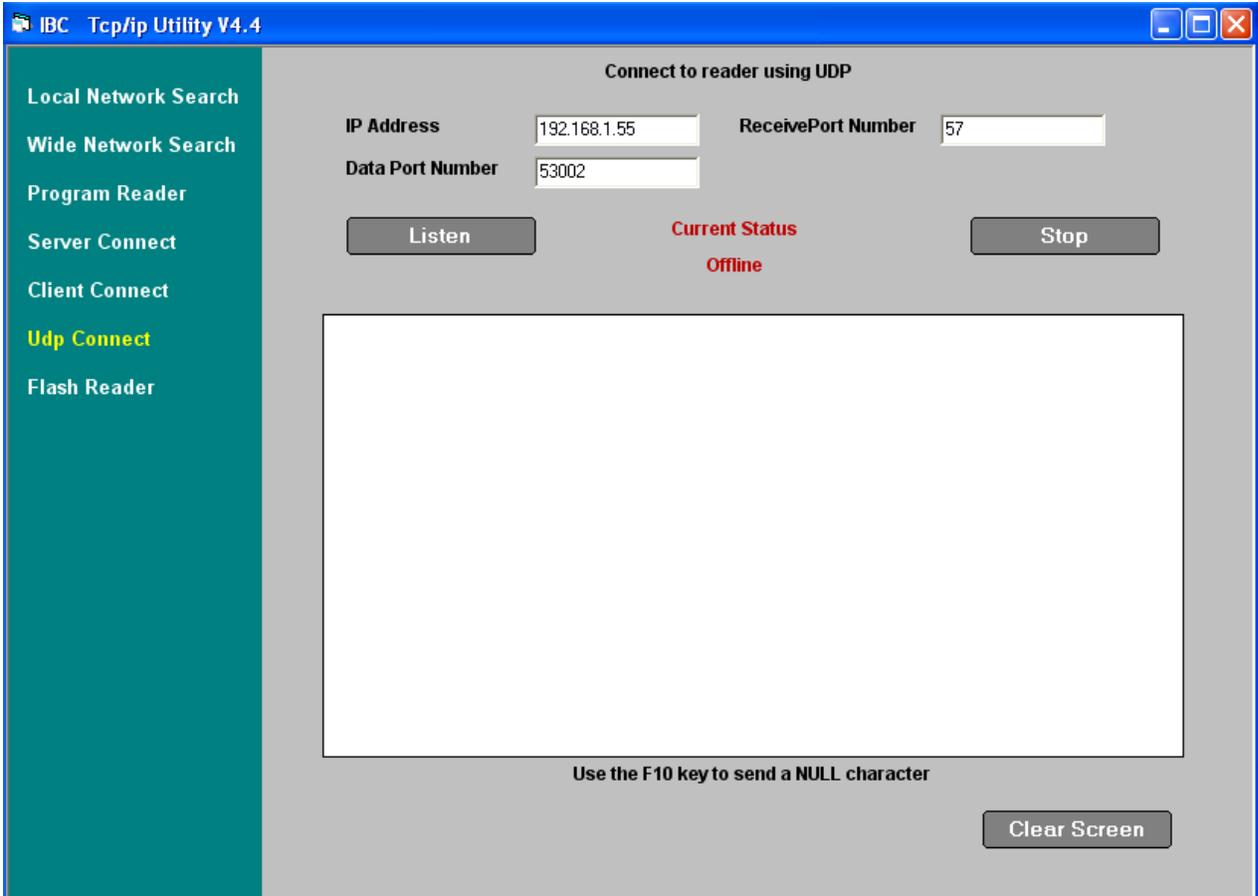
This screen allows you to connect to a reader which is programmed to operate as a client. The reader must have the pc's IP address programmed into the Server Address field in the reader. Enter in the port number (Listen Port) that is the port number programmed into the reader's Server Port field. If you have the Data Port Password protection ON in the reader, then you must also enter in the number for the Control Port and the password, otherwise the screen will not be able to send and receive data.

To connect, click on the *Listen* button. The IBC tcp/ip readers attempt to connect to their host device every couple of seconds. If the reader is present at the entered IP address, then it should momentarily connect.

Once you have connected, you can type data which will be sent to the reader, and any data received from the reader will be displayed on the screen.

NOTE that if the reader has connected and then you change screens without clicking the *Stop* button - the reader will still remain connected and other screens potentially may not work.

## Testing a reader which is set up as a Udp Peer



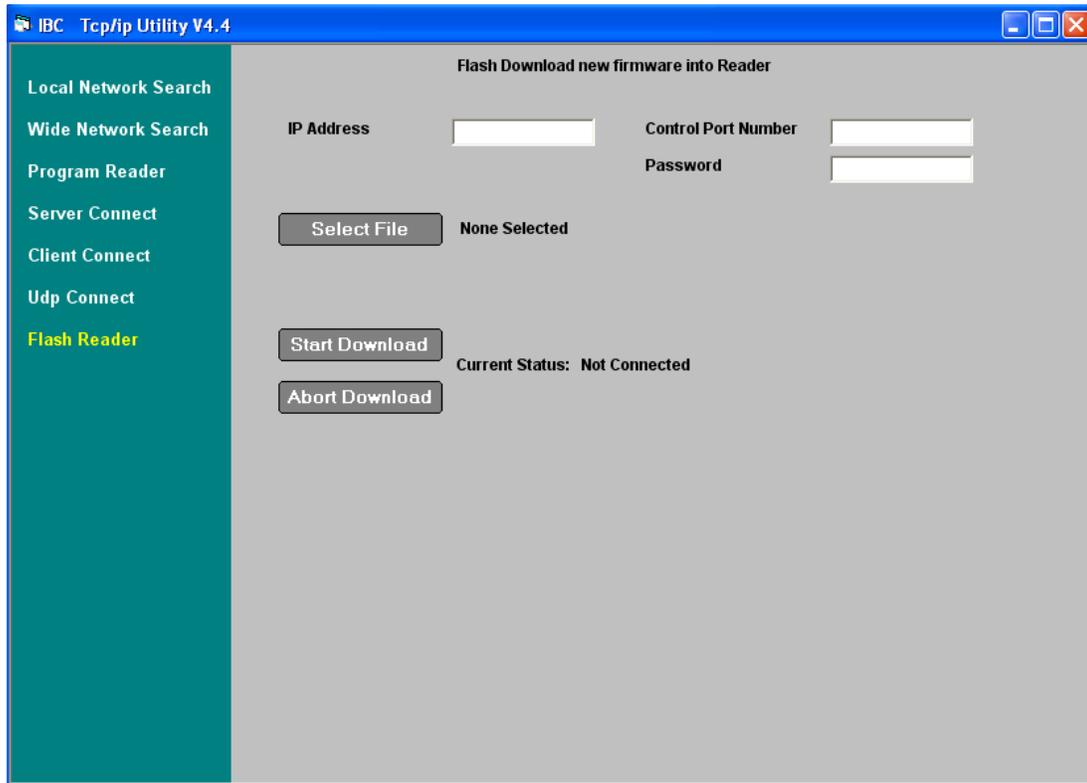
This screen allows you to connect to a reader which is programmed to operate as a Udp peer. You must enter in the IP address for the reader and also the Data Port number. You must also enter in the Receive Port Number, which will be the *Server Port* which is programmed into the reader.

To connect, click on the *Listen* button.

Once you have connected, you can type data which will be sent to the reader, and any data received from the reader will be displayed on the screen.

NOTE that if the reader has connected and then you change screens without clicking the *Stop* button - the reader will still remain connected and other screens potentially may not work.

## Firmware Updating



All IBC Tcp/ip readers are capable of receiving new updated firmware by utilizing this Flash Download screen.

To send the new firmware to the reader, you must enter in the IP address for the reader, and also the Control Port number and the current Password.

Click on the *Select File* button to select the file which is to be downloaded (supplied by IBC) and then click on *Start Download*.

The new firmware will start to be downloaded and the status will be displayed on the screen. You can abort the download at any time by clicking on the *Abort Download* button. Aborting the download does not harm the reader in any way. The reader will not load the new firmware unless a full download completes.

When the download is complete, you will be asked if you want to restart (reboot) the reader. If you respond yes then the reader will be restarted, and will load the new firmware into it's memory. This process can take up to 3 minutes.

NOTE that during the download process the reader internally stores the downloaded information but does not reflash itself until a reboot or restart occurs. This process can take up to a few minutes. Even if you respond No to the request to restart the reader - the next time the reader is restarted or powered up - it will load in the new firmware automatically so upon a restart of a freshly downloaded reader, you can expect up to a few minutes to occur before you are able to connect to the reader.