IOLINK-130
Ethernet Bridge/Router
Reference Manual

Issue 4

All Software Versions

 11/2000

Part # 168-335708

1 Introduction

The IOLINK-130 Ethernet Remote Bridge/Router

The IOLINK-130 Ethernet remote bridge/router provides IP and IPX routing combined with a protocol transparent bridge. This bridge/router combination is often the best solution to linking remotely located LANs where most of the traffic is IP or IPX with smaller amounts of traffic from other protocols such as NetBIOS or DEC LAT.

The IOLINK-130 Ethernet bridge/router supports the widely implemented Routing Information Protocol, otherwise known as RIP. RIP support allows the IOLINK-130 Ethernet to interoperate with other vendors' routers.

The IOLINK-130 Ethernet remote bridge/router will operate as delivered, providing increased LAN performance directly out of the box without the need for complex pre-configuration. However, in those situations where specific customization is required, an easy-to-use "hotkey" menuing Bridge/Router Manager console provides access to LAN and Link statistical information, and control of the network configuration.

With increased LAN and Link management capability, you will be able to detect LAN and Link problems, determine utilization patterns, and plan for future expansion that will optimize your existing data-communication resources.

The IOLINK-130 Ethernet bridge/router can be thought of as a group of discrete functions combined in a single box. The first functional module is the LAN interface, which receives all LAN traffic and then decides where individual frames should be sent: to the IP router, to the IPX router, to the bridge, to the management system, or discarded altogether. After the LAN interface there are several functional units including the IP router, the IPX router, the bridge, and the management system. Any traffic that these modules need sent across a link is then forwarded to the link module, which control data coming and going on the WAN ports. The following figure illustrates the relationships between the various component modules in a IOLINK-130 Ethernet bridge/router.

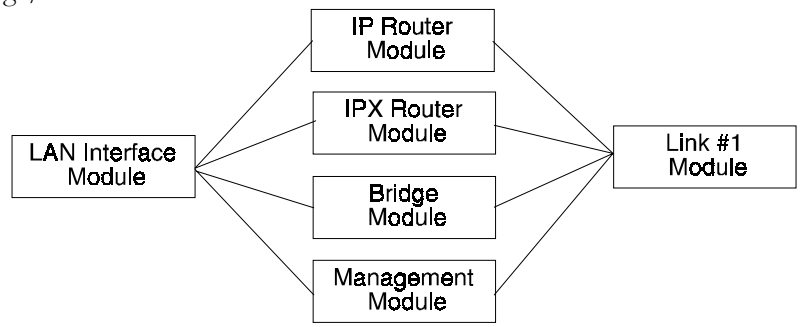


Figure 1 - 1 IOLINK-130 Ethernet Bridge/Router Block Diagram

The IOLINK-130 Ethernet menu system provides a method to control whether IP & IPX traffic is routed through the router modules, or bridged through the bridge module along with all other bridged data.

IP Routing and the IOLINK-130 Ethernet Remote Bridge/Router

The IOLINK-130 Ethernet bridge/router may be used to route only between subnets within the same network, or between different networks.

Network broadcasts sent within a subnet-routed environment will not be forwarded to the other subnets in the network.

The procedure for establishing an IP connection through an IP router is explained on the next few pages.

ARP—Address Resolution Protocol

A protocol called ARP (Address Resolution Protocol) is used to determine the MAC address of a particular IP address. The MAC (Medium Access Control) address is unique predefined number for each device on the LAN. The manufacturer of the device assigns MAC addresses. The IP address for each device is assigned by the network administrator according to the network structure.

If the originating station does not know the MAC address of the destination station, a MAC broadcast will be transmitted onto the LAN asking "Who has IP address 170.22.10.4?" This MAC broadcast is called an ARP request. Because the ARP request is a MAC broadcast, every device on the LAN will see the frame. The device that has the IP address 170.22.10.4 will respond with a frame to the originating station. The ARP reply frame will include the MAC address of the destination device.

Now when the two devices wish to send data across the LAN to each other, they will both use the MAC and IP address of the other device.

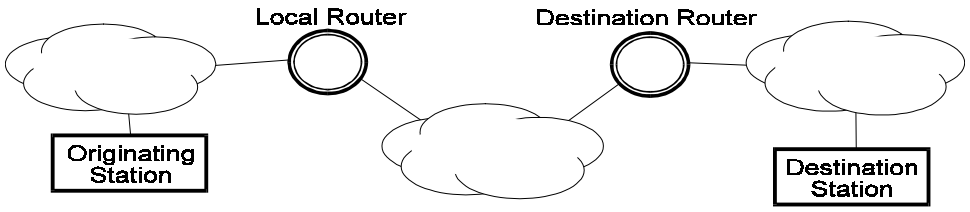
Each device on the LAN maintains a table for MAC addresses and IP addresses called the ARP cache. The ARP cache contains a list of IP addresses and their corresponding MAC addresses.

Proxy ARP

Each time an originating station does not know the MAC address of a destination station, the originating station sends out an ARP request. If the destination station is on a different network, the router connected to the originating network will see from the IP address that the frame is to be routed to another network. If the router has an entry for the destination address, the router will generate an ARP reply to send back to the originating station. The ARP reply will specify the MAC address of the router as the MAC address to send frames to for the IP address of the destination station.

The Complete IP Connection

The following are the steps that a frame of data will take when being transmitted from an originating station on an IP network to a destination station on a different IP network. In this example, the two networks are separated by a third network with two router hops between the originating network and the destination network.



- Originating station will send an ARP request if it does not have the MAC address of the destination station.
- Local router will see ARP request and send an ARP reply to the originating station with the MAC address of the local router port.
- Originating station will send the data frame addressed to the IP address of the destination station, and the MAC address of the local router port.
- Local router will receive the data frame and strip off the MAC portion. The resulting IP frame will be examined to determine the destination IP address.
- Local router will look in its routing table to find the IP address of the router to send the IP frame to next. The local router will see that the destination router is the next router.

- Local router will look in its ARP cache to find the MAC address of the destination router as determined by the IP address in the routing table.
- Local router will rebuild the complete frame with a new MAC header indicating the MAC address of the destination router. The local router does not alter the destination IP address, so the destination IP address will still be the IP address of the destination station.
- Destination router will receive the data frame and strip off the MAC portion. The resulting IP frame will be examined to determine the destination IP address.
- Destination router will look in its routing table to find the IP address of the router to send the IP frame to next. The destination router will see that the destination IP address is on a locally connected network.
- Destination router will look in its ARP cache to see if it has a MAC address for the destination IP address. If it does not have an entry, the destination router will generate an ARP request. The destination station will send an ARP reply.
- Destination router will rebuild the complete frame with a new MAC header indicating the MAC address of the destination station. The destination IP address once again will be unchanged and remain as the destination station IP address.
- Destination station will receive the data frame and process it.

If the destination station wishes to send a frame back to the originating station, the process will happen in the reverse direction.

If the path from the originating station to the destination station causes the frame to pass through more than two routers, the above process will simply be extended to include the interaction between the intermediate routers.

IP Header Details

Every IP header has common fields of information. The layout of the information is always the same. Refer to the following diagram for a representation of the IP header.

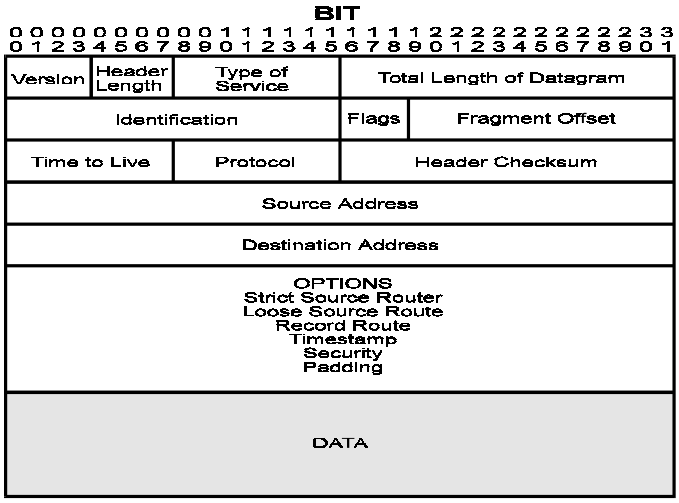


Figure 1 - 2 IP Header

Protocol

The protocol section is used to indicate the protocol being used by the transport layer. This could be TCP, UDP, or something else.

Time to live

The time to live section is used to prevent a frame from traversing the network forever. This field contains a number (maximum 255) that is set when the frame is originally generated. Each time the frame is passed through the bridge/router, the bridge/router will decrement the time to live by two. When the time to live reaches zero, the frame is discarded.

Header Checksum

The header checksum is used to verify the data in the IP header. The IP header is recalculated each time a frame is passed through a router. The recalculation is necessary because the time to live field is changed.

Fragmentation

Fragmentation occurs when an IP frame must be split up into smaller IP frames. When the originating device generates the IP frame, the device is not aware of all the paths the frame must traverse to get to the destination device. If the IP frame is to pass through a network that has small packet capabilities, the IP frame must be split up and reassembled at the destination device. Each of the fragments is assigned a fragment offset value, which determines where the fragment fits into the original IP frame.

The IOLINK-130 Ethernet bridge/router will accept fragmented frames directed to itself and reassemble them, but it will not fragment frames.

Options

There are various options that may be set for any IP frame.

Source Routing

Source routing is used to predetermine the path that the IP frame must travel through the network. There are two types of source routing: strict source routing and loose source routing.

Strict source routing will contain a list of IP addresses of routers that must be used when the IP frame is sent through the network. Strict source routing is used mainly to provide some type of data security. Once the IP frame has reached the destination station, the destination station will take the list of IP addresses from the options field, reverse them, and use them for a strict route back to the originating station.

Loose source routing will also contain a list of IP address of routers to be used on the path to the destination station. However, the IP frame may pass through other intermediate routers to get to the next IP address in the loose source routing list.

Route Recording

Route recording simply keeps a list of all the IP addresses of the routers that the IP frame has passed through on its way to the destination station.

Time Stamps

The time stamp option is used to record the time at which the IP frame passed through each router on its way to the destination station.

ICMP Messages

Internet Control Message Protocol (ICMP) messages are used to perform station and router protocol participation. ICMP messages are passed between routers, or between routers and stations. There are several different messages, as discussed below.

Unreachable

The "unreachable" message is sent back to the originating station when the path to the destination network has disappeared. A destination network may be unreachable because of a broken link, a downed router, a downed station, or other reasons.

Redirect

The "redirect" message is sent to the originating station when there is a better router to use to reach the destination network. Because the routers share routing tables, each router has the ability to determine whether it is the best router to use for network traffic. Once a station receives a redirect, all future IP frames destined for the particular destination network will be sent to the new router.

Quench

The "quench" message is sent to the originating station when the path to the destination network has become congested. The originating station will slow down the rate of transmission of frames for an internally (to the station) predetermined period of time upon receiving a quench message.

Ping

The "ping" message is actually a query status message that may be sent to devices on the LAN to query their operation status. The ping message is basically a message asking "Are you alive?" The LAN device will reply with a message if it is active.

Time and Mask server

Two other ICMP messages are used to query the time and/or subnet mask from a particular LAN device. A message is sent to a LAN device asking for the time or mask, and the device replies appropriately.

RIP—Routing Information Protocol

The most important function of the IP protocol is routing. IP routers constantly exchange information keeping their routing tables up to date. A method of communication is required to ensure compatibility between all IP routers in the network. RIP is the portion of the IP protocol that is used for router communication.

Route Tables

Each router will maintain a table of network addresses and the appropriate action to take with an IP frame it receives. A routing table entry will usually consist of the following items:

- Network or sub-network address
- IP address of the next hop router
- Network interface to use to get to the next hop router
- Subnet mask for this network interface
- Number of hops to reach the destination network
- Number of seconds since this route was updated

When a router receives an IP frame, the router will examine it to determine the destination network address. The router will then look in the routing table, determine the next router to send the IP frame to, and send the frame to that router.

The selection of the best route path is based solely on the number of hops to the destination network.

Update Mechanism

In order to ensure that the routing tables of all routers in the network are kept up to date, each router will broadcast its routing table onto each of its locally connected networks. The broadcast of the routing tables occurs every 30 seconds.

The process of updating a routing table with current information, and deciding which router to use to reach a destination network, creates a ripple effect of changes through the network. When a router goes down and an adjacent router determines that the path has disappeared, the remaining adjacent routers on that network must determine the next path to use to reach the destination network. Each router will now broadcast its new routing table with the updated information. The updated information will propagate through the network until all routing tables have been brought up to date. This process is called convergence.

The broadcast of the routing tables is also used as a method of determining whether a router is still alive or has been removed from the network. If a router has not heard from an adjacent router in 180 seconds, the local router will mark the adjacent router as unreachable and start to adjust the routing table, if necessary.

IPX Routing and The IOLINK-130 Ethernet Remote Bridge/Router

The IOLINK-130 Ethernet bridge/router may be used to route between IPX networks.

Novell Netware uses a suite of protocols for LAN communications. The Novell protocols include IPX, SPX, RIP, SAP, plus others, and operate at layers 3 and above. These protocols, their relationship with each other, and the general operation of a Novell network are discussed in this section.

The Netware Network Operating System implements the concept of "Client-Server" computing. In this system, there are various Servers, such as File Servers, Print Servers, and Fax Servers, to name a few. The Client stations, where the users work, connect to these servers to retrieve files, get application software, or submit print jobs. Most of the interaction between the Clients and Servers is invisible to the users. These operations rely on the transfer of packets between Clients and Servers using the IPX/SPX protocols.

IPX Addressing

The IPX protocol is based on the Xerox XNS protocol. The IPX header contains all the IPX addressing information, and not much else.

Network Layer Addressing vs. MAC Addressing

An Ethernet frame has at least two levels of addressing. The MAC addresses for both the source and destination are contained in the MAC header. The MAC addresses are essentially physical port addresses, and are globally unique. Hardware vendors encode the port MAC address as part of the manufacturing process. All Ethernet devices have the same MAC address format. The MAC address is used to communicate frames between LAN ports regardless of protocol.

The Network layer addressing is assigned by the network administrator, in a format prescribed by the layer 3 protocol, for example IPX. The network address is used to structure the network system and for communications between ports operating the same protocol.

Note that it is possible for a single network port to have several different network addresses, but it can have one and only one MAC address. An example of this is a computer acting as an IPX File Server, an IPX Router, and an IP Router. In this case the port would have a MAC address, an IPX address for its IPX functions, and an IP address for the IP Routing functions.

IPX Address Format

The IPX Address is made up of three components: the Network Number, the Node Number, and the Socket Number. These components are fixed length (unlike the IP addressing) and function.

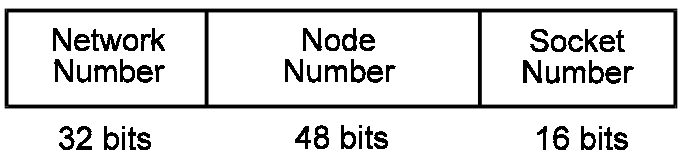


Figure 1 - 3 IPX Address Format

Network Addresses

The Network Number addresses the network. All stations on the same "network" will have the same Network Number. Note that a network could be a single segment, or multiple segments joined by either bridges or repeaters. In IPX internetworks, routers must be used to join different networks together.

Node Addresses

The Node Number identifies the individual stations in a Network. In IPX devices, this address is assigned automatically and is identical to the MAC address. This means that the Node Number is self-configuring, and will be unique within the Network because the MAC address that was copied is (supposed to be) unique.

The use of the MAC address as the Node Number allows IPX stations to be self-configuring. This makes the initial configuration of a station much simpler, but there are drawbacks. The Node Numbers cannot be structured as needed, with groups of stations having for example consecutive addresses. Instead, the network is forced to live with whatever MAC address is assigned to the LAN port.

Socket Addresses

The Socket Number identifies the process within the source/destination that is communicating. Common Sockets include File Servers (Socket Number 0451), SAP (Socket Number 0452), and RIP (Socket 0453). The Socket Number can be thought of as the address of the upper layer using the IPX communication.

The Socket Numbers are assigned by Novell and do not change from LAN to LAN. In other words, all communications with File Servers use Socket Number 0451. When a software vendor uses IPX to communicate across a Netware network, the vendor will apply to Novell to receive a Socket Number for the application. As an example, if Acme Schedule Company made a groupware scheduling program for Netware, they would get a Socket assigned for their use. No other communications on the LAN would use the Acme Scheduler Socket.

Other IPX Header Information

The IPX header contains some other information besides the source and destination addresses.

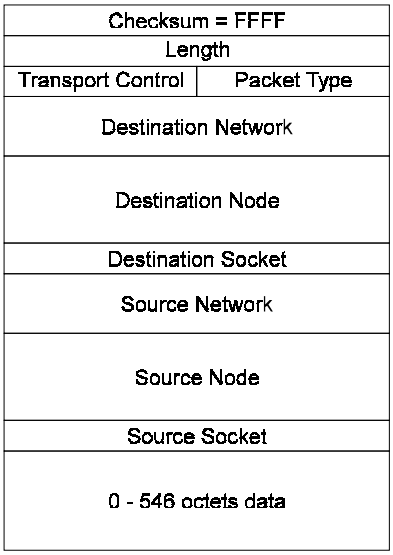


Figure 1 - 4 IPX Header

The checksum is a hold-over field from the XNS model used by Novell. In the original XNS header, the checksum was used; however, Novell decided that the MAC trailer CRC was enough protection and the IPX header checksum need not be used. Therefore the IPX checksum is permanently set to FFFF.

The length field indicates the total length of the IPX packet. Note that the data portion can be any length up to 546 bytes, so the length field is needed in the header.

The Transport Control field is used for counting the number of routers the frame has traversed. In other words, it is a hop count. This operation uses only 4 of the 8 bits; the remaining 4 bits are reserved (by Novell) for future use so we could see additional information contained in the Transport Control field if Novell decides to use the excess capacity.

The Packet Type indicates what type of service is using the packet. Some common packet types include type 1, RIP; type 2, Echo; type 4, IPX; and type 17, Netware Core Protocol.

Establishing an IPX Connection

The Netware model is Client/Server, where Clients initiate calls to Servers for various purposes. The Clients are made aware of the presence of Servers by listening for Service Advertisement Protocol (SAP) broadcasts. Servers send SAP broadcasts regularly to identify themselves, including their address and what type of service they offer (File Server, Print Server, Fax Server, etc.).

Services also are referred to by their name. Server names are assigned by the network administrator, and are usually representative of the server's function. As an example, a network might have three File Servers named "GeneralFS," "OrderProcessingFS," and "DevelopmentFS." Each of these servers would send out SAPs to inform the Clients of their presence. The Clients can display a list of Servers, and initiate a connection to the desired server using the servers name. Typically, Clients are pre-programmed with the name of the "Preferred Service," which allows the Client station to connect automatically (without human intervention) to the Preferred Server. When no Preferred Service is set, the Client automatically connects to the first Server it hears. This is because a Client without a Server is almost useless in most Novell applications.

Once an IPX connection has been established between a Client and the Server, there is often a security screen to manage access. File Servers are protected by a User ID/Password scheme to ensure that only authorized users are let into the server. Access privileges within the server are also assigned to the individual users. This prevents a Client logged into the "General" server from accessing files which are the private property of another user on the same "General" server.

Service Advertisement Protocol

The SAPs are broadcast by Servers at regular intervals, and collected by Clients so that they can keep track of what Servers are out there. Also, a Client may broadcast a Server Request ("Is there a Server named 'Whatever' out there?"), which would be heard by all Servers, and hopefully the Server which the Client is searching for would respond directly, telling the Client about itself (the Server).

SAP Broadcasts

The Service Advertisement Protocol broadcast is the standard mechanism that Servers use to announce their availability to the rest of the network. A server will broadcast a SAP containing from 1 to 15 different Services offered. Therefore if a single high-end PC is acting as a File Server, a Print Server, and a Fax Server, it would send out a single SAP that lists all three available Servers. Other servers that offer only a single Service would have only the one Server in the SAP.

SAP broadcasts are sent out every 30 seconds. They are received by all stations on the LAN (it's a broadcast after all), and the station decides what to do with it. Both Clients and Servers maintain a list of all Servers that are broadcasting availability. A Novell user can execute the SLIST.EXE program to display the current list of known servers.

When a Client or Server notices that a Server from its known Server list has missed a broadcast (it should get one about every 30 seconds), it starts up a counter, and when the Server has missed 3 broadcast intervals (about 180 seconds) that Service is removed from the known Server list. In this way Servers that crash or go off-line for any reason are aged out of the network.

SAP Requests

Sometimes Clients will need to find out if a specific Server is available. This may occur immediately after a Client is brought up, and before it has received any SAP broadcasts. The Client (or a new Server) sends out a SAP Request broadcast asking for a specific Server. That Server, or a router with the best route to that Server, will respond to the Client (Server) making the request.

Server Types

There are many different types of Servers. Each type is defined and given a type code by Novell. When new types of Servers are invented they will be assigned a new Server type. Some common Servers are:

Type	Description
0000	Unknown
0003	Print Queue
0004	File Server
0005	Job Server
0006	Gateway
0007	Print Server
0009	Archive Server
0024	Remote Bridge Server
0027	TCP/IP Gateway

Routing Information Protocol

The Novell Routing Information Protocol (RIP/X, where the X indicates IPX) is similar, but not identical, to the Routing Information Protocol used in IP routers. Novell RIP/X performs similar functions to IP RIP, in that RIP/X is used to communicate information about routes through routers to remote networks.

RIP/X Operation

The operation of RIP/X is, for all intents and purposes, identical to the operation of IP RIP. Routers send out broadcasts every 30 seconds containing the contents of that router's route table (the list of best routes to known remote networks). When a router comes on line, the extent of its route tables will be its explicit route. In the case of a local router, it will be a route between the two networks to which the router is connected. In a pair of remote routers linked via a WAN connection, the first RIP broadcasts will contain only the route to the remote network. As time goes on, and assuming there are more routers in the network (and correspondingly more remote networks), the various routers will by way of RIP broadcasts inform each other of the various routes.

RIP/X Broadcasts

A RIP broadcast is sent out by IPX routers every 30 seconds or so. Each broadcast may contain information on up to 15 different routes (to 15 different networks). If a router knows of more than 15 networks it will send out two (or more) broadcasts.

Note that to spread the network overhead a router will stagger the generation of RIP/X and SAP broadcasts. The router will send a RIP/X broadcast, followed 15 seconds later by a SAP broadcast, followed 15 seconds later by another RIP/X broadcast, etc., etc. The SAP and RIP/X broadcasts are sent every 30 seconds as required, but they are staggered by 15 seconds to spread the overhead.

RIP/X Requests

A Client may also request a route to a given network or server. To do so, the Client generates a Route Request broadcast that the routers hear, and routers that know of the route requested will respond to the originating station. In this way a new Client may find routes without waiting for the routers' broadcast, that could be up to 30 seconds away (if it just missed one). A new router on a network will also broadcast a general Route Request to fill its route tables quickly. Again, without this mechanism the router would have to wait for about 30 seconds until it heard from all other routers via their standard RIP/X broadcasts.

RIP/X Metrics

The RIP/X routing protocol measures routes based on two metrics, the hop count and the ticks delay. These metrics are used to compare different routes to the same network, with the goal of selecting the best (shortest) route.

The ticks delay is the primary metric used to determine the optimal route. The tick count is an indicator of how long a packet will take to get to the destination. Novell has defined 1 tick to be the length of time it takes a 512-byte frame to be transmitted on a 10-Mbps (Ethernet) LAN. This works out to about 18 ms. The real value of the tick delay is when evaluating routes across WAN connections. In these cases, the tick count is dependent on the link speed of the WAN connection(s), where a slower link will have a higher tick count.

The hop count is the secondary measure of the length of a route; it is exactly the same as the IP hop count. If a route goes through 1 router (the shortest route), it will have a hop count of 1. If a route goes through 6 routers, the hop count for that route will be 6. The maximum number of hops RIP/X supports is 15, but this is a very large number, considering the size of most internetworks. When two or more routes to the same network have the same tick count, the router will use the route with the smallest hop count.

Bridging and the IOLINK-130 Ethernet Remote Bridge/Router

The bridge portion of the IOLINK-130 Ethernet remote bridge/router is an Ethernet Media Access Control (MAC) level bridge providing an efficient means of interconnecting IEEE 802.3 Local Area Networks supporting a choice of standard Ethernet (10Base5), Thin Ethernet (10Base2) and Twisted Pair (10BaseT) interfaces. With the support of these industry-standard LAN interface technologies, the IOLINK-130 Ethernet remote bridge/router will resolve the media conflicts that might have otherwise prevented the consolidation of these resources.

The IOLINK-130 Ethernet remote bridge/router will also fit right into those environments that may require more than one bridge by using the IEEE 802.1D Spanning Tree Protocol. With this protocol, the IOLINK-130 Ethernet remote bridge/router will perform automatic network reconfiguration in the event of a link failure to one of the LAN segments. This provides maximum availability of the attached LAN services.

Immediately following are several short descriptions of LAN bridging operations specific to the IOLINK-130 Ethernet remote bridge/router. These descriptions will help you understand the concepts of bridging and how the IOLINK-130 Ethernet remote bridge/router performs these functions.

The remaining sections of this document describe how these functions are performed and configured. You are urged to spend the small amount of time necessary to familiarize yourself with the IOLINK-130 Ethernet remote bridge/router and the advanced functions it may perform for you.

The Initial Bridging Process

Each time a IOLINK-130 Ethernet bridge/router is powered up, it will perform extensive hardware and software tests to ensure the integrity of the unit and its attached LAN and Link interfaces. Upon successful completion of the power-up diagnostics, the IOLINK-130 Ethernet bridge/router will follow rules to "learn" several aspects of your LAN environment. These rules define what actions are taken under particular situations.

One of the more important rules employed by the IOLINK-130 Ethernet bridge/router is also a very fundamental part of the bridging process. This rule dictates how Ethernet Station Addresses are processed by the bridge. The process is outlined below:

Station Address Learning

The IOLINK-130 Ethernet bridge/router performs an important bandwidth-conserving function by a process termed Station Address Learning. This process determines the location of all active LAN Stations by monitoring the Ethernet frames being transmitted onto the LAN segments. Once it has learned the location of each station, the remote bridge/router will not forward those Ethernet frames destined for a station if the receiving station exists on the same LAN. Under these conditions, the bridge/router will only forward a frame if the location of the destination station has not yet been learned, or if the location has been determined to exist on the other LAN segment.

To perform this process, the IOLINK-130 Ethernet bridge/router follows the steps outlined below:

Learning Local Addresses

When the bridge/router is powered up, and after completing the power-up diagnostics, it will not immediately begin forwarding frames between LAN segments. Instead it will listen to local LAN activity in order to learn the location of each station address on each side of the bridge.

The bridge/router captures each frame and looks at the source address contained within the Ethernet frame. Since the bridge/router knows which LAN segment the frame was received from, it can determine that this station must be located on this segment. As a result, it has just learned the location of the station.

This process will continue for the period defined by the Forwarding Delay option, and in this fashion the first stage of the LAN address table is built.

Forwarding

Once the initial learning process is complete, the bridge/router enters a forwarding mode and examines frames that may need to be forwarded. The learning process does not stop at this time, however: The bridge/router will continue learning new stations as they become active on a LAN segment.

Local Destination Addresses

When a frame is received from a station on one segment, the frame is examined for the source address to ensure that this station has already been entered into the address table. If the source address exists, the Ethernet destination address is then viewed. The bridge searches the previously built address table for the location of the destination station. If it is determined that the location of the destination station exists on the same LAN segment (i.e. the destination address is local and the frame does not need to be forwarded across the bridge to the other LAN segment), then the bridge will "filter" and discard it.

Initially, the bridge will only recognize those addresses that are local to a specific LAN segment. The bridge will thereby filter (discard) all local packets and forward all unknown non-local packets to the second segment located on the outbound port across the bridge.

Forwarding Unknown Destination Addresses

When a frame is received from a LAN segment with an unknown destination address (an address that does not yet exist in the filter table), the bridge will forward the frame to the other segment, logging the address, and marking the location as "unknown."

Unknown Location Update

When the receiving station transmits a frame in the opposite direction, the bridge will now see the previously unknown destination address in the source address field. It will now process this source address as it did during the initial learning stage, adding the location to the address entry.

In this fashion (looking at source addresses of non-local packets), the bridge learns about non-local stations and their associated arrival ports. The bridge then updates the location of each address in its table. In the future the bridge will look up these stored non-local addresses to determine the bridge port on which to forward a packet destined for a known non-local station.

In summary, the IOLINK-130 Ethernet bridge/router will "learn" the location of a station by examining the source Ethernet address, and will "filter" frames based on destination address. A frame received from one segment that is of "unknown" location will be forwarded to the other segment. A frame that is received with a source address equal to a known address, but previously marked as an unknown location, will be updated in the filter table to add the location.

Aging Timer

During the bridging process, the filter table is built giving the location (bridge port or LAN segment) of known Ethernet addresses. The table would become quite large, eventually reducing performance, if stations were added, removed, or moved without the old information being purged periodically. Performance is affected since the larger the table, the more time it will take to process an incoming frame.

This purging process, called "aging," is an integral part of the learning function. It limits the size of the filter table and ensure that performance is not reduced unnecessarily.

Aging assumes that many of the addresses may not be active all of the time, and could be purged after a specified interval to keep the size of the filter table small. In general terms, the smaller the table, the higher the performance.

Address Purging

To achieve this routine housekeeping, each entry in the filter table contains the LAN addresses, the LAN port identifier, and a timer flag. Each time a particular address is looked up or added to the table, a timer flag is set for the "fresh" entry. When a time interval, defined by the Bridge/Router Manager expires, the address table is scanned and any "stale" entries that have not been used since the timer expired are removed. This timer is called the "aging timer" and may be controlled through the bridge options.

Purging the address does not prevent the station from using the bridging facilities, since the location of the station may be re-learned. However, since a small aging timer value will mean that the bridge must re-learn addresses more often, there must be a balance between table size and aging time to achieve optimal performance.

Aging Exception

"Permanent" address entries are an exception to the aging rule. A permanent address is one that is not subject to the aging timer and will remain in the filter table for an indefinite period of time.

A table is reserved for permanent address entries, separate from the table that is used for those non-permanent entries that are subject to aging. These tables may be displayed and modified with the bridge/router options discussed in this manual. Access is made locally from each Bridge/Router Console or one bridge/router can be made Master, able to control all functions of a partner IOLINK-130 Ethernet bridge/router.

Filled Address Table

Sometimes filter address table may become full. (The filter table can hold 2048 address entries.) If this occurs, an automatic procedure is followed.

This procedure defines that an address that is not in the table will not be added and will be treated as any other unknown address. The frame will be passed to the other segment. An alarm will also be generated with the message "Station Address Table Full," and from this point, another alarm will be generated only if in the meantime the table empties by 1/3 and then fills up again.

IOLINK-130 Ethernet Bridge/Router Feature Definitions

Telnet

A Telnet LAN station or another IOLINK-130 Ethernet bridge/router has the ability to connect to the Operator Interface of any IOLINK-130 Ethernet bridge/router supporting the Telnet feature. With the Telnet feature, all IOLINK-130 Ethernet bridge/routers on a network may be managed from a single point.

Once a connection is established, all of the menus of the other bridge/router are now available on the bridge/router that initiated the connection. All menu operation on the initiating bridge/router is suspended during the connection. Entering a control-C character <^C> at any time during the connection will cause a disconnection, and you will be back to the menu of the first bridge/router.

To implement the Telnet feature, each bridge/router requires an IP address (see the Internet Set-Up Menu). It is advisable to assign an IP address to each IOLINK-130 Ethernet bridge/router in your network.

The IP address of another bridge/router may be assigned a name to simplify the connection process. Telnet connection to the other IOLINK-130 bridge/router may be established by entering either the name or the IP address of that router. Refer to the Remote Site Set-Up Menu (under Configuration / WAN Set-UP) for more information on adding names to the bridge/router.

If a bridge/router does not have an IP address, Telnet connections cannot be initiated or received.

If a Telnet connected bridge/router receives a second connection attempt from another bridge/router the connection attempt will be ignored.

Connecting to a bridge/router while the remote bridge/router menu system is operating with a different terminal setting may cause unexpected screen errors. Once the connection to the bridge/router has been established, it is recommended that the operator change the terminal setting to be the same as the initiating device.

When a Telnet connection is made to a bridge/router, ensure that the Telnet session is in character mode, and carriage return padding (or translation) is set to NULL (or no translation). The extra character sent when carriage return padding is on will cause some displays to behave erratically.

Link Compression

The IOLINK-130 Ethernet Bridge/Router's compression option multiplies the effective data throughput across wide area links operating at speeds from 9600 bps through to 256 Kbps. The exact amount a given transmission can be compressed is dependent upon the type of data being transferred over the wide area network. As an example, because of their repetitive make-up, most graphics and database files can easily be compressed by a ratio of 6:1. In contrast, other types of files (such as binary files), that are not as repetitive, typically yield a compression ratio of 2:1. It should also be noted that compression ratios are entirely dependent upon the make-up of the specific file — while it may be possible to compress a given ASCII file far beyond the 6:1 ratio, a different ASCII file may only compress to a ratio of 4:1 or lower.

At link speeds above 256 Kbps, link compression is not advised as the processing time involved in compressing the data does not yield signifigant gains over the transmission of raw data.

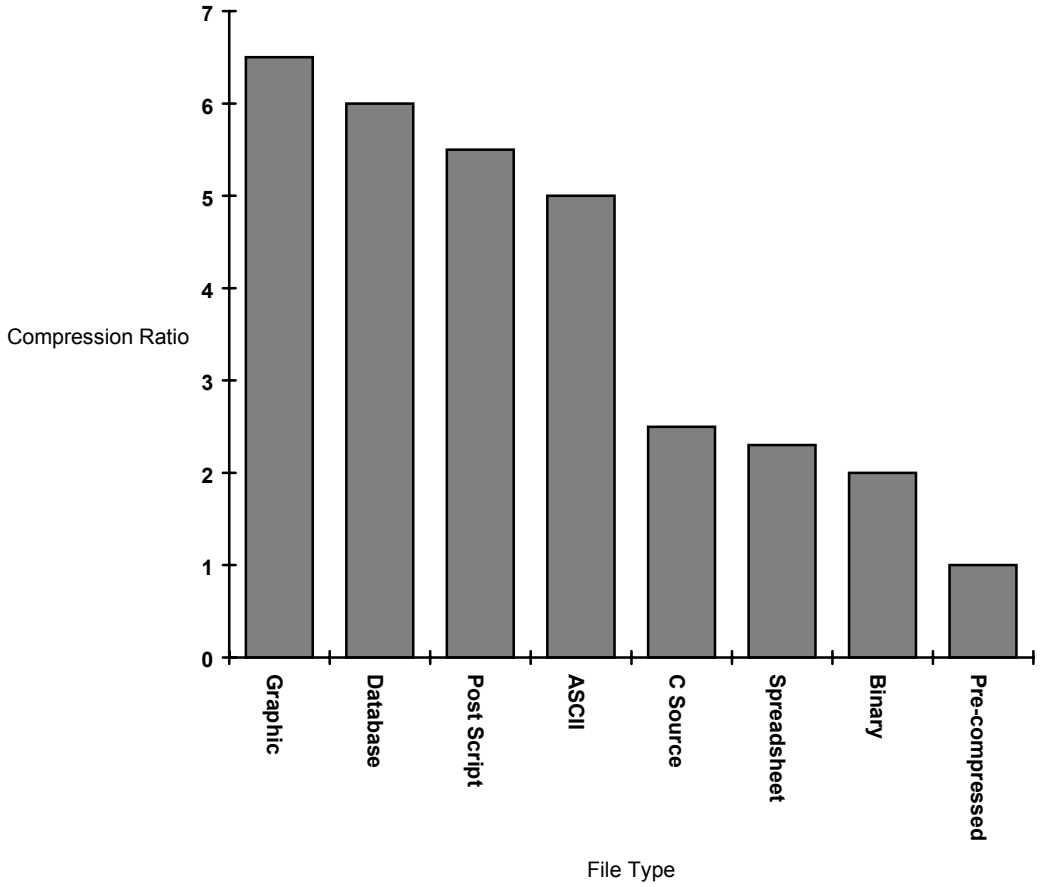


Figure 1 - 5 Typical Compression Ratios by File Type

Data compression will give a 56/64 Kbps link an effective throughput range from 112/128 Kbps when transferring binary files, to 364/384 Kbps when transferring graphic files. This increased throughput significantly reduces the bandwidth required between the LANs to achieve a given performance level, and also allows the use of lower-cost transmission facilities.

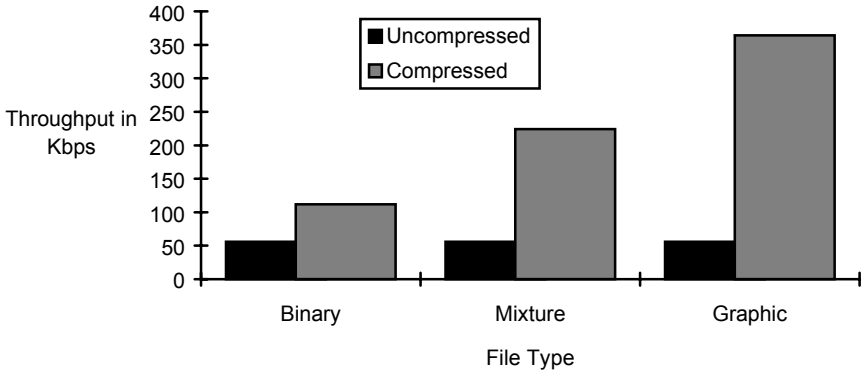


Figure 1 - 6 Typical Throughput Over 56 Kbps Link

Operating Software Upgrades

The IOLINK-130 Ethernet Bridge/Router includes flash memory, that allows new system code to be downloaded using the Trivial File Transfer Protocol (TFTP). This allows software updates to be performed quickly and painlessly from a host server (with TFTP capabilities) on the network.

The IOLINK-130 Bridge/Router also allows the downloading of software updates by using a direct management port connection and the ZMODEM transfer protocol.

For a detailed description of how to perform a software upgrade, please see the Load FLASH Set-Up section in the *PPP Menus Manual* or Appendix E in the *Installation and Applications Guide*.

2 Link Interface Reference

Pinout Information

The IOLINK-130 router is manufactured with four different WAN link modules: V.35, LXT411 CSU/DSU, Universal WAN or T1/E1. The type installed may be determined from the label on the WAN link output connector.

V.35 Module:

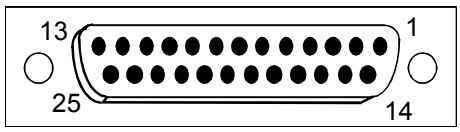
The V.35 link interface is provided as a DB25 connector on the back of the bridge/router, so an interface converter is needed to convert to the standard V.35 connectors.

When connecting two bridge/routers back-to-back without modems, a null-modem cable is required to crossover the pins on the links. Crossing over the pins allows two bridge/routers both configured as DTE interfaces to be connected together. With this configuration, both bridge/routers will provide clocking for the links, and each bridge/router must have a link speed defined.

UNIVERSAL WAN Module:

WARNING: ensure that the connector cable used with the Universal WAN interface module has the correct pinouts for the operational mode selected for the interface (V.11, V.35, RS232, or EIA530). Using the incorrect cable connector for the operational mode selected may cause permanent damage to the interface module.

The Universal WAN Interface module in this router may be configured to operate in one of four modes: V.11/X.21, V.35, RS232/V.24, or RS530/RS422. The interface connector for all types is a standard DB25 pin female connector.



WARNING: ensure that the connector cable used with the Universal WAN interface module has the correct pinouts for the operational mode selected for the interface (V.11X.21, V.35, RS232/V.24, or RS530/RS422). Using the incorrect cable connector for the operational mode selected may cause permanent damage to the interface module.

Pinouts for each mode of operation are listed on the pages following.

Link Clocking Information

The link interface on the IOLINK-130 acts as a DTE device, this means that it may be directly connected to DCE devices with the DCE devices providing the clocking for the link. The link speed is controlled by the DCE device. Setting the link speed on the IOLINK-130 will not result in a speed change on the link.

Some DCE devices allow the DTE devices connected to them to supply a clock signal which is then routed back to the transmit clock pins (external clock pins) on the DCE interface. This clock is then received by the IOLINK-130 link interface. By using this method, the IOLINK-130 may be in control of the link speed. The link speed may also be controlled by the IOLINK-130 when a null-modem cable is used to connect two IOLINK-130s in a back-to-back configuration.

Changing the link speed within the menu system of the IOLINK-130 changes the clock output speed that is generated on the DTE Terminal Timing pins on the link interfaces.

Console Connector

The console connector on the IOLINK-130 router is a DCE interface on a RJ45 pinout. The supplied DB9 to RJ45 converter should be used to connect to the DB9 connector of a DTE terminal. This connection will then provide access to the built-in menu system.
If the console interface is to be connected to a modem or other DCE device, a standard RS-232 crossover converter should be used.

The following table illustrates the console pinouts.

RJ45 connector on unit (DCE)	DB9 connector on converter (DCE)	RS-232 signal name
2	6	CTS
3	4	DTR
4	5	GND
5	2	RxD
6	3	TxD
7	8	DSR
8	1	CD

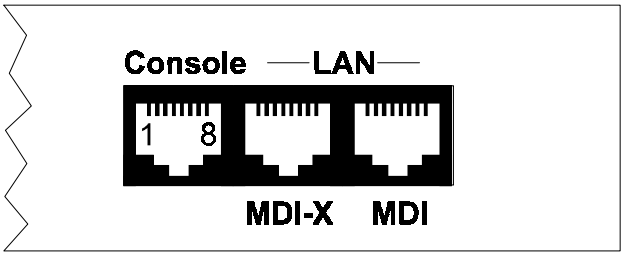


Figure 2-1 Rear View of the Console Connector

.

CSU/DSU Module:

IOLINK-130 routers with an LXT411 CSU/DSU interface module use a standard RJ45 service connector.

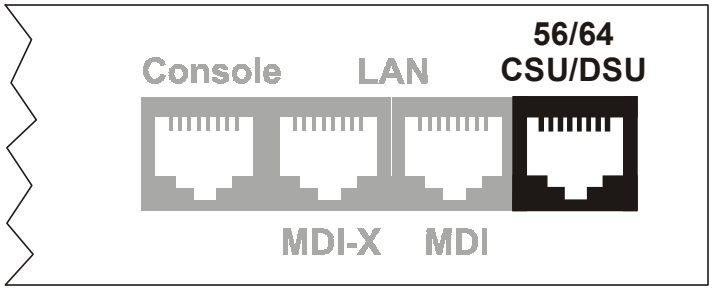


Figure 2-2 Rear View of the CSU-DSU Connector

The LXT411 CSU/DSU link connection is set to operate at 64 Kbps by default. The link may be set to 56 Kbps via the software menus if required.
When two CSU/DSU link routers are to be connected via a leased line in a back to back set-up, the unit must be set to 56 Kbps link speed and a null-modem crossover cable used for the connection.
A DSU/CSU crossover cable would be constructed as follows:

1 --> 7
2 --> 8
7 --> 1
8 --> 2

T1/E1 Module:

IOLINK-130 routers with a T1/E1 interface module use a standard RJ45 service connector.

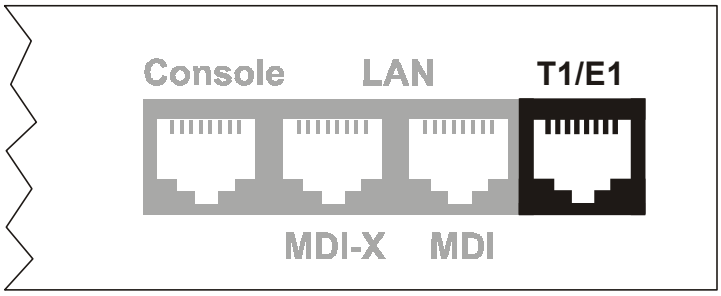


Figure 2-3 Rear View of the T1/E1 Connector

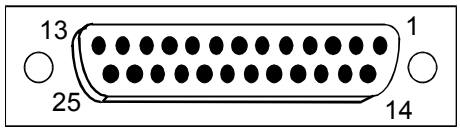
When two T1/E1 routers are to be connected in a back to back set-up, a null-modem crossover cable used for the connection.
A T1/E1 crossover cable would be constructed as follows:

1 --> 4
2 --> 5
5 --> 2
4 --> 1

Pins 1 and 2 are receive (1 = ring, 2= tip)
Pins 4 and 5 are transmit (4 = ring, 5= tip)

V.35 Link Pinouts

The connector shown here and pinouts described here correspond to the connector labeled "V.35" on the back of the IOLINK-130.



DB25 Female DTE

DB25 Contact Number	M.34 Contact Number	Circuit Name	Direction To DCE	Direction From DCE
1	A	Protective Ground	NA	
2	P	Transmitted Data (A)	X	
3	R	Received Data (A)		X
4	C	Request to Send	X	
5	D	Clear to Send		X
6	E	Data Set Ready		X
7	B	Signal Ground	NA	
8	F	Data Channel Received Line Signal Detector		X
9	X	Receiver Signal Element Timing (B)		X
10		----------		
11	W	Terminal Signal Element Timing (B) DTE	X	
12	AA	Send Signal Element Timing (B)		X
13		----------		
14	S	Transmit Data (B)	X	
15	Y	Send Signal Element Timing (A)		X
16	T	Received Data (B)		X
17	V	Receiver Signal Element Timing (A)		X
18	L	Local Loopback	X	
19		----------		
20	H	Data Terminal Ready	X	
21	N	Remote Loopback	X	
22		----------		
23		----------		
24	U	Terminal Signal Element Timing (A) DTE	X	
25	NN	Test Mode	X	

Figure 2 - 4 V.35 Link Pin Outs

The connecting cable must be a shielded cable.

Circuits which are paired (contain an (A) and (B) reference) should be connected to twisted pairs within the connecting cable.

NOTE For U.K. Approval:

The connecting cable should be manufactured from Belden Cable, or a cable with equivalent specifications. One end must be terminated in a male 34 pin X.21 bis connector as defined in ISO-2593 1984. The other end must be terminated in a male 25 pin X.21 bis connector as defined in ISO-2110 1989. The cable may be any length between 0 and 5M.

V.35 Null-Modem Cable Configuration

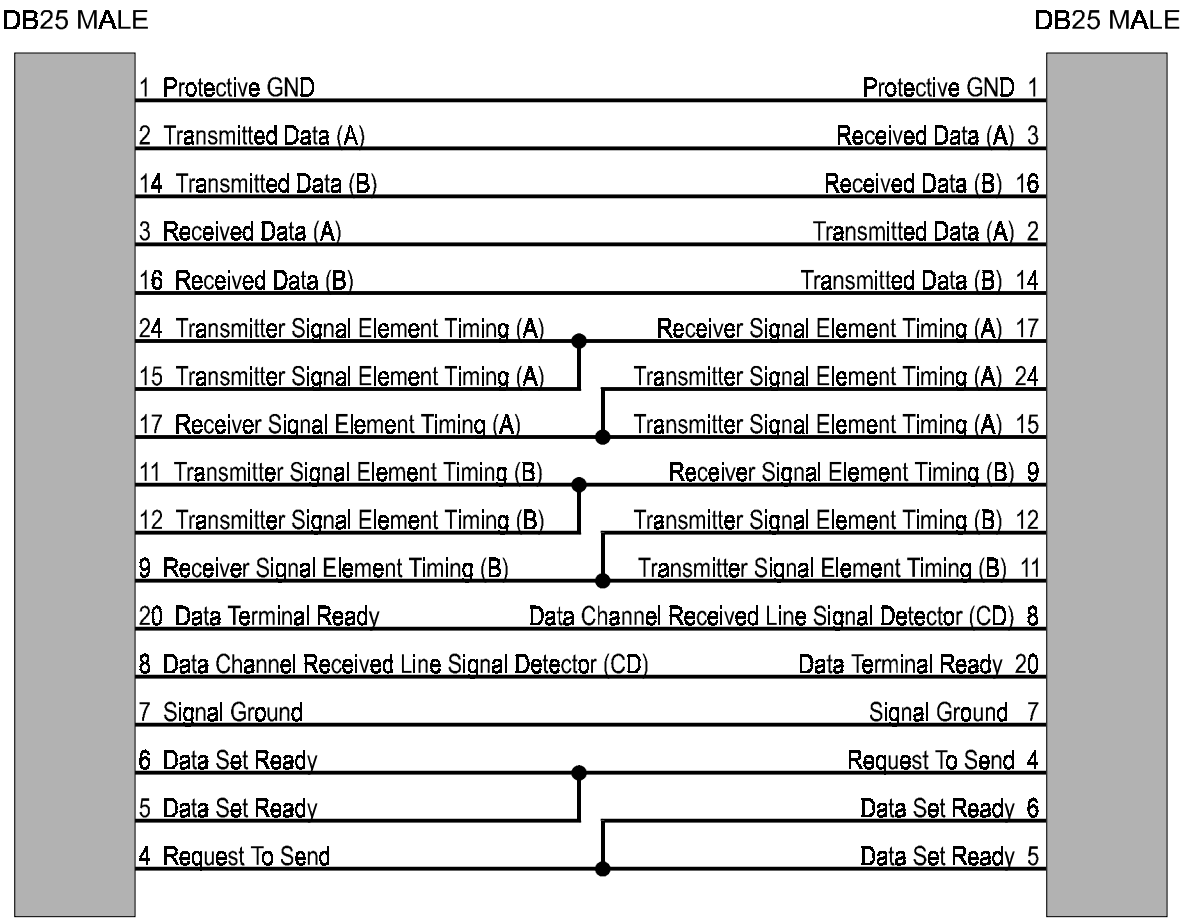


Figure 2 - 5 V.35 Null-Modem Cable

The connecting cable must be a shielded cable.

Circuits which are paired (contain an (A) and (B) reference) should be connected to twisted pairs within the connecting cable.

This cable is needed when it is necessary to connect two units back-to-back and a set of modems is not available. Note that this cable specifies DB25 connectors on each end to allow direct connection to the link interface connector on each unit.

The link speed must be defined for each of the two units. Link speed above 1.544 MBPS are not recommended with a null-modem connection

V.24 & RS232C Link Pinouts

The pinouts described here correspond to the RS232/ V.24 mode for a Universal WAN IOLINK-130.

DB25 Female DTE

Contact Number	CCITT Circuit Number	Circuit	Circuit Name	Direction To DCE	Direction From DCE
1	101	AA	Protective Ground	NA	
2	103	BA	Transmitted Data	X	
3	104	BB	Received Data		X
4	105	CA	Request to Send	X	
5			----------		
6	107	CC	Data Set Ready		X
7	102	AB	Signal Ground	NA	
8	109	CF	Received Line Signal Detector (CD)		X
9			----------		
10			----------		
11			----------		
12			----------		
13			----------		
14			----------		
15	114	DB	Transmit Signal Element Timing (DCE Source)		X
16			----------		
17	115	DD	Receive Signal Element Timing (DCE Source)		X
18	141		Local Loopback	X	
19			----------		
20	108.2	CD	Data Terminal Ready	X	
21			----------		
22	125	CE	Ring Indicator		X
23			----------		
24	113	DA	Transmit Signal Element Timing (DTE Source)	X	
25			----------		

Figure 2 - 6 RS232 Link Pinouts

The connecting cable must be a shielded cable.

NOTE For U.K. Approval:

The connecting cable should be manufactured from Belden Cable, or a cable with equivalent specifications. Each end must be terminated in a male 25 pin X.21 bis connector as defined in ISO-2110 1989. The cable may be any length between 0 and 5M.

V.11 & X.21 Link Pinouts

The pinouts described here correspond to the V.11/X.21 mode for a Universal WAN IOLINK-130.
Note: A DB25 to DB15 pin converter will be required to connect to V.11/X.21 service.

Contact Number	X.21 Circuits Reference	Circuit Name	Direction To DCE	Direction From DCE
1		Protective Ground	NA	
2	T (A)	Transmitted Data (A)	X	
3	C (A)	Control (A)	X	
4	R (A)	Received Data (A)		X
5	I (A)	Indication (A)		X
6	S (A)	Signal Element Timing (A)		X
7		----------		
8	Ground	Signal Ground	NA	
9	T (B)	Transmitted Data (B)	X	
10	C (B)	Control (B)	X	
11	R (B)	Received Data (B)		X
12	I (B)	Indication (B)		X
13	S (B)	Signal Element Timing (B)		X
14		----------		
15		----------		

Figure 2 - 7 V.11 Link Pinouts

The connecting cable must be a shielded cable.

Circuits which are paired (contain an (A) and (B) reference) should be connected to twisted pairs within the connecting cable.

NOTE For U.K. Approval:

The connecting cable should be manufactured from Belden Cable, or a cable with equivalent specifications. Each end must be terminated in a male 15 pin X.21 connector as defined in ISO-4903 1989, but one end of the cable must have UNC-4-40 screws and the other end must have M3 screws. The cable may be any length between 0 and 5M.

RS442 & RS530 Link Pinouts

The pinouts described here correspond to RS530 mode for a Universal WAN IOLINK-130.

Contact Number	Circuit	Circuit Name	Direction To DCE	Direction From DCE
1	Shield	Protective Ground	NA	
2	BA (A)	Transmitted Data	X	
3	BB (A)	Received Data		X
4	CA (A)	Request to Send	X	
5	CB (A)	Clear to Send		X
6	CC (A)	Data Set Ready		X
7	AB	Signal Ground	NA	
8	CF (A)	Received Line Signal Detector		X
9	DD (B)	Receive Signal Element Timing (DCE Source)		X
10	CF (B)	Received Line Signal Detector		X
11	DA (B)	Transmit Signal Element Timing (DTE Source)	X	
12	DB (B)	Transmit Signal Element Timing (DCE Source)		X
13	CB (B)	Clear to Send		X
14	BA (B)	Transmitted Data	X	
15	DB (A)	Transmit Signal Element Timing (DCE Source)		X
16	BB (B)	Received Data		X
17	DD (A)	Receive Signal Element Timing (DCE Source)		X
18	LL	Local Loopback	X	
19	CA (B)	Request to Send	X	
20	CD (A)	Data Terminal Ready	X	
21	RL	Remote Loopback	X	
22	CC (B)	Data Set Ready		X
23	CD (B)	Data Terminal Ready	X	
24	DA (A)	Transmit Signal Element Timing (DTE Source)	X	
25		----------		

Figure 2 – 8 RS530 Link Pinouts

The connecting cable must be a shielded cable.

Circuits which are paired (contain an (A) and (B) reference) should be connected to twisted pairs within the connecting cable.

RS232 Null-Modem Cable Configuration

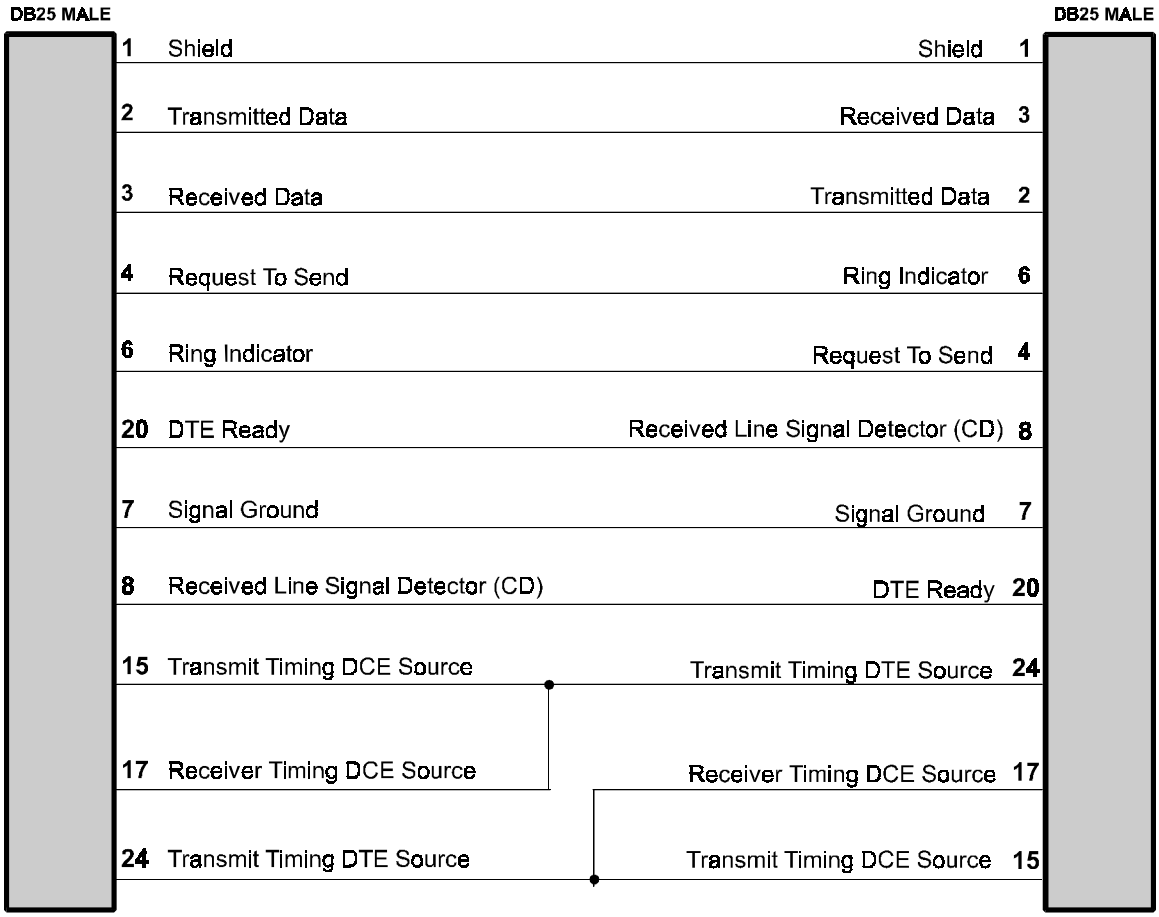


Figure 2 - 9 RS232 Null-Modem Cable

The connecting cable must be a shielded cable.

This cable is needed when it is necessary to connect two units back-to-back and a set of modems is not available. Note that this cable specifies DB25 connectors on each end to allow direct connection to the link interface connector on each unit.

The link speed must be defined for each of the two units.

RS530 Null-Modem Cable Configuration

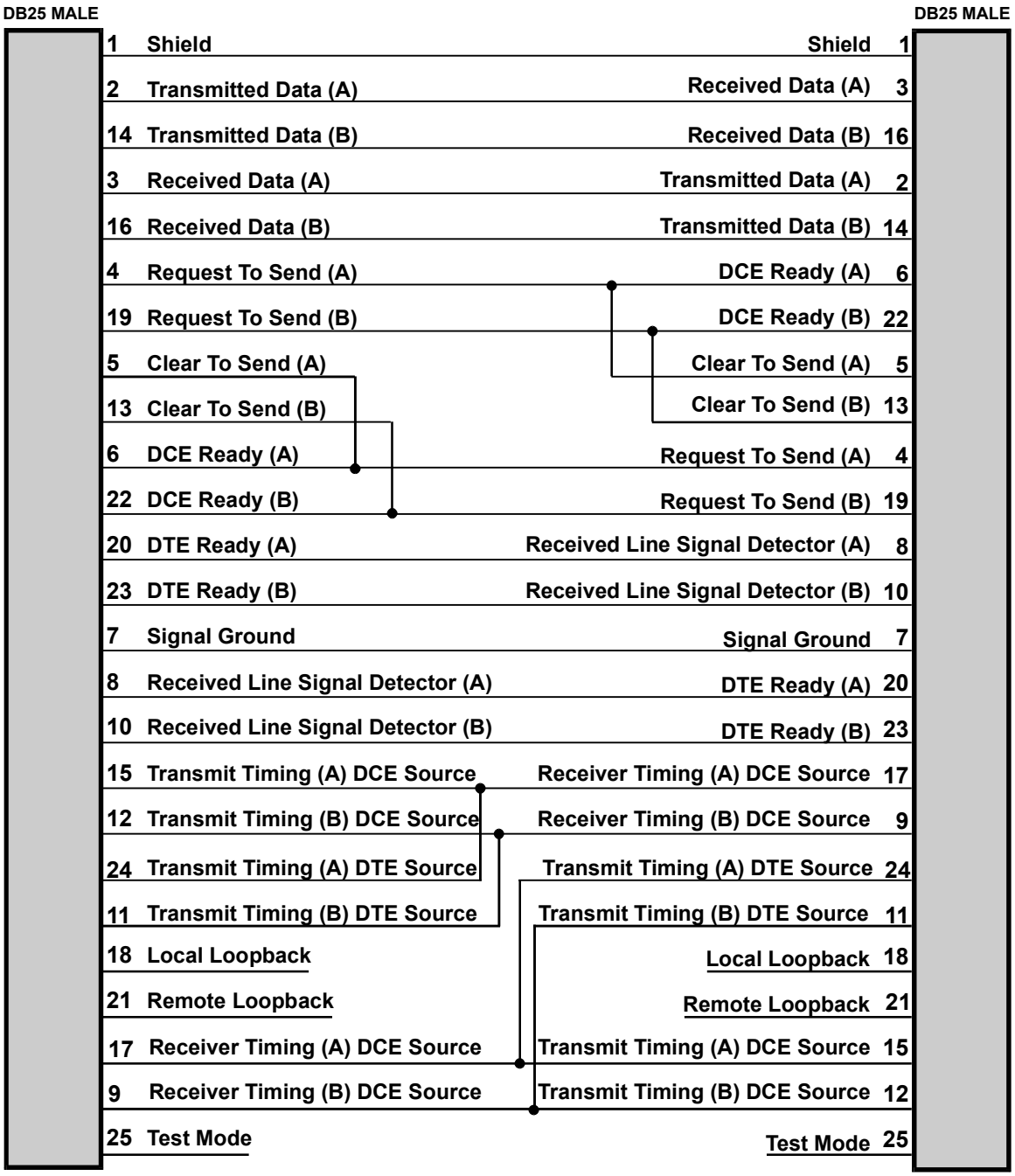


Figure 2 - 10 RS530 Null-Modem Cable

The connecting cable must be a shielded cable.

Circuits which are paired (contain an (A) and (B) reference) should be connected to twisted pairs within the connecting cable.

This cable is needed when it is necessary to connect two units back-to-back and a set of modems is not available. Note that this cable specifies DB25 connectors on each end to allow direct connection to the link interface connector on each unit.

The link speed must be defined for each of the two units.

V.11/X.21 Null-Modem Cable Configuration

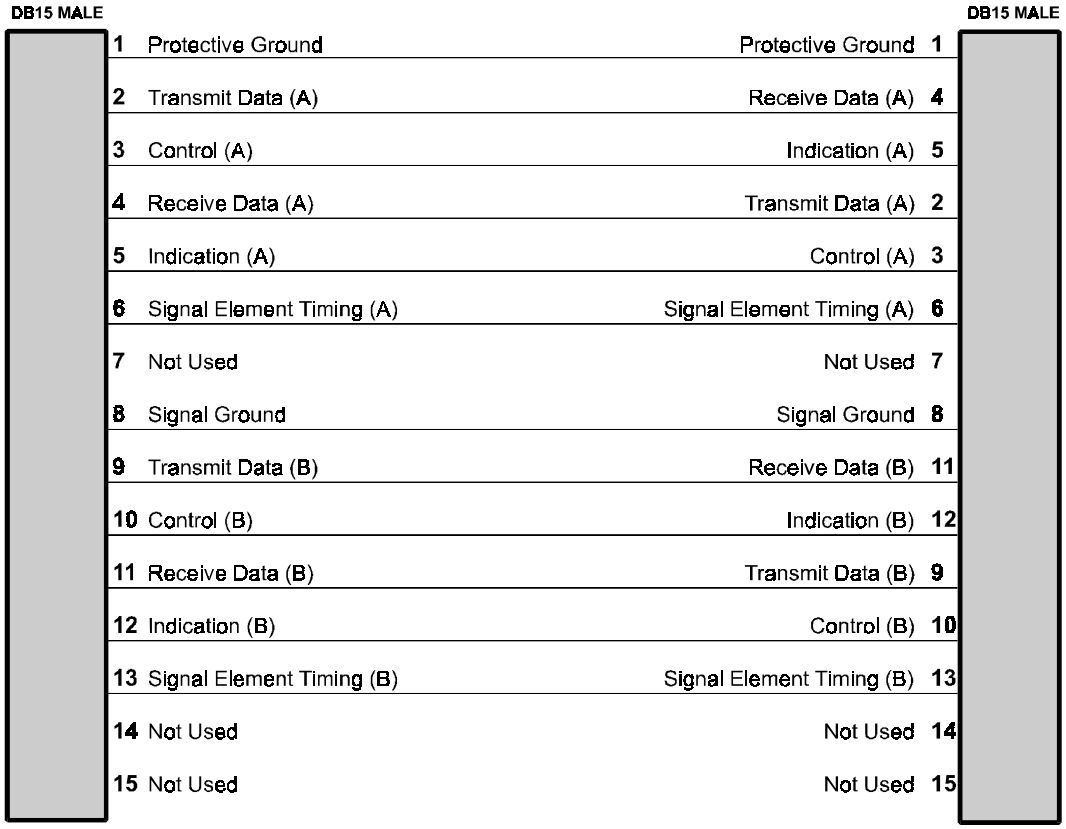


Figure 2 - 11 V.11/X.21 Null-Modem Cable

The connecting cable must be a shielded cable.
Circuits which are paired (contain an (A) and (B) reference) should be connected to twisted pairs within the connecting cable.
This cable is needed when it is necessary to connect two units back-to-back and a set of modems is not available. Note that this cable specifies DB15 connectors on each end to allow direct connection to the link interface connector on each unit. The link speed must be defined for each of the two units.
When using this cable to connect two units back-to-back, a jumper must be installed on pinheaders W8 and W9 on one of the V.11/X.27 interface modules. This allows that particular module to generate the required timing signals.

WAN Link Control-Signal Operation

CTS flow control is not supported.

1) When a call is made to the bridge/router, RI will be asserted by the modem. The bridge/router responds by driving DTR and RTS high to signal to the partner's modem that it is ready to establish communications. The bridge/router then waits (for the duration of the CD Wait Time, default 60 seconds) for the partner's modem to respond with incoming CD and DSR signals driven high. (With a dataset or DSU/CSU, RI is not applicable and the bridge/router will respond when CD is high).

2) After DTR goes high, if the incoming CD is not detected within the CD Wait Time then DTR and RTS are lowered for 5 seconds. This clears the link, hangs up the modem and causes the modem to redial in an attempt to establish the link. (This "DTR toggle" polling is continuous when the link is inactive. When DTR is low, and RI is subsequently asserted, then the bridge/router immediately drives DTR and RTS high and again waits for the duration of CD Wait Time for CD to be asserted). NOTE: If the level 2 process times out, DTR toggle will be started.

3) When incoming CD and DSR signals both go high, then it indicates that the local modem, etc. has established communications with the remote modem and that communications may begin.

4) If the link is up, and CD is subsequently interrupted for a period of 10 seconds, then the bridge/router assumes the signal is lost and will display an Alarm. As stated previously, if the level 2 protocols can continue to send, they will be allowed to do so. This prevents minor line glitches from taking the link down upon the loss of CD.

5) If the RI signal is asserted while DTR is already high, the CD Wait Timer is restarted. Once the CD signal is received by the called modem, the state of the RI signal is unimportant.

6) In the Conditional Link mode, the secondary link will be inactive by holding DTR and RTS low (the modem, etc., will normally be configured to drop the call when DTR and RTS are held low). When the secondary link needs to become active, DTR and RTS will be raised and the modem will make the call.

3 Event Logs

The IOLINK-130 Ethernet bridge/router generates event logs for various functions performed by the bridge/router. All of the event logs are stored in the internal event log file, which is accessible through the Network Events menu.

Certain event logs are classified as alarms because they are deemed to be of higher urgency. Alarm logs are indicated by an asterisk ("*") at the start of the alarm text and are printed on the ALARM line on the menu system as well as being stored in the event log. Alarms are listed in the second part of this appendix.

All WAN-link-related events include the link number in the event log. All remote site-related events include the remote site alias in the event log.

All LCP events are either link based or remote site based depending on whether frame relay is disabled or enabled.

Event logs:

Capture off

Generated when link trace capture is turned off.

Completed BCP negotiation with <remote site alias>

Generated when the Bridging Control Protocol negotiation has been completed with the remote site device associated with the stated remote site profile. Once BCP negotiations are complete, IP routing may take place between the two routers.

Completed CCP negotiation with <remote site alias>

Generated when the Compression Control Protocol negotiation has been completed with the remote site device associated with the stated remote site profile. Once CCP negotiations are complete, IP routing may take place between the two routers.

Completed IPCP negotiation with <remote site alias>

Generated when the Internet Protocol Control Protocol negotiation has been completed with the remote site device associated with the stated remote site profile. Once IPCP negotiations are complete, IP routing may take place between the two routers.

Completed IPXCP negotiation with <remote site alias>

Generated when the Internet Packet Exchange Control Protocol negotiation has been completed with the remote site device associated with the stated remote site profile. Once IPXCP negotiations are complete, IPX routing may take place between the two routers.

Configuration restored

Generated during a warm start when a configuration is successfully restored from non-volatile RAM.

Connection attempt to <IP address>

Generated when the bridge/router attempts a Telnet connection. The IP address of the target bridge/router is specified.

DHCP: pool deleted due to mismatch with new IP address

Generated when the address of this device is changed , rendering the block of addresses assigned to the DHCP pool invalid.

DHCP: Requested address unavailable <IP address>

Generated when the address requested by a client is unavailable

DHCP services – declined. Address <IP address> declined by client, mark it as unavailable.

Generated when a client declines an address offered by the DHCP service, usually because the client has found from some other source that the address is unavailable.

DHCP services – release. <IP address> released.

Generated when IP address displayed is released from its assignment to a device and put back in the IP pool for re-assignment.

Entering loopback. Initiated by remote device

T1/E1 signal loopback test started by the unit on the far end of this link.

Exiting loopback. Initiated by remote device

T1/E1 signal loopback test ended by the unit on the far end of this link.

Error executing: XXXXXX

Generated when an error is detected loading back a configuration. The invalid command is specified.

Incorrect password from <IP address>

Generated when an incorrect password is given for a Telnet connection. The connecting bridge/router's name or IP address is specified. After three incorrect login attempts within ten minutes, an alarm is generated (see Security alarms: "Possible intruder") and any further attempts from that IP address within the next ten minutes are rejected.

Invalid Relay Destination for subnetted network

Generated when a device attempts to do a network broadcast on a subnetted network.

IPX: Network configuration error from X, network Y.

Generated when the router and server do not agree on a network number.

IPX: routing table full

Generated when the IPX RIP table has been filled. Any new RIP frames received will be discarded.

IPX: Service table full

Generated when the IPX SAP table has been filled. Any new SAP frames received will be discarded.

IPX: X, unknown RIP packet type received

Generated when the device receives an invalid RIP packet.

IPX: X, unknown SAP packet type received

Generated when the device receives an invalid SAP packet.

IPX: X, demand RIP failed

Generated when this device reaches the end of the attempts to negotiate demand RIP for the IPXCP connection. After demand RIP has failed, the normal RIP and SAP updates will occur.

LCP X authenticating peer with CHAP

Generated when this device is using CHAP to authenticate the peer (remote) device.

LCP X authenticating peer with PAP

Generated when this device is using PAP to authenticate the peer (remote) device.

LCP X establishing

Generated when the Link Control Protocol of a PPP link or remote site is establishing between this device and the remote site PPP device.

LCP X no reply to Y Echo-Requests

Generated just prior to a link going down. The link or remote site has gone down due to no replies to the echo request messages sent.

LCP X operational

Generated when the Link Control Protocol of a PPP link or remote site has been negotiated and is now operational between this device and the remote site PPP device.

LCP X peer authenticating with CHAP

Generated when the peer (remote) device is using CHAP to authenticate this device.

LCP X peer authenticating with PAP

Generated when the peer (remote) device is using PAP to authenticate this device.

LCP X received Protocol-Reject for <protocol>

Generated when the peer (remote) device rejects one of the Network Control Protocols.

Link X - CSU/DSU Digital Loopback Results [Total: X] [Good: X] [Errors: X]

Generated after the CSU/DSU link module has completed a digital loopback test.

Link X - CSU/DSU Remote Loopback Results [Total: X] [Good: X] [Errors: X]

Generated after the CSU/DSU link module has completed a remote loopback test.

Link X - CSU/DSU Self Test Results [Total: X] [Good: X] [Errors: X]

Generated after the CSU/DSU link module has completed a self test.

Link X CSU/DSU: Connected

Generated when the CSU/DSU link module has established a connection to the remote partner CSU/DSU.

Link X CSU/DSU: Initialized

Generated when the CSU/DSU link module has completed initialization and has established communications with the IOLINK-130.

Link X CSU/DSU: Line ERROR

Generated when the CSU/DSU link module is in an unknown error state.

Link X CSU/DSU: Loopback Ended

Generated after the CSU/DSU link module completes a loopback test.

Link X CSU/DSU: Loopback Started

Generated when the CSU/DSU link module starts a new loopback test.

Link X - CSU/DSU out of service

Generated when the CSU/DSU is in communication with the local telco, but is not connected to the remote partner CSU/DSU.

Link X CSU/DSU: No Signal

Generated when the CSU/DSU link module cannot establish communications with the local telco.

Link X CSU/DSU: Reset

Generated when the IOLINK-130 has initiated a reset of the CSU/DSU link module.

Link X - external loopback results[Total: X] [Good: X] [Errors: X]

Generated after the CSU/DSU link module has completed an external loopback.

Link X - CSU/DSU CSU telco loopback started

Generated when the telco has initiated a CSU loopback to this CSU/DSU link module

Link X - CSU/DSU CSU telco loopback ended

Generated when the telco has terminated the CSU loopback to this CSU/DSU link module

Link X - CSU/DSU DSU telco loopback started

Generated when the telco has initiated a DSU loopback to this CSU/DSU link module

Link X - CSU/DSU DSU telco loopback ended

Generated when the telco has terminated the DSU loopback to this CSU/DSU link module

Link X – LMI Error Threshold exceeded.

Generated when the number of LMI enquiry errors exceeds the user-defined limit.

LMI discarding STATUS on link X – Enquiries not started

Generated when the bridge/router discards a Status message received from the frame relay network on a link that the bridge/router has not yet started the Local Management Interface.

No NCP's open, tearing link down

Generated when a PPP link does not have a Network Control Protocol operating. This may be due to BCP, IPCP, and IPXCP being disabled, or the NCP connection was not negotiated to completion.

Password accepted from <IP address>

Generated when a correct password is given for a Telnet connection. The connected Bridge/Router's name or IP address is specified.

Refused connection attempt from <remote site alias>

Generated when a connection attempt to a remote site is refused by that site

Remote Site <remote site alias> re-establishing due to multilink mismatch

Generated when multilink negotiated on by one end of the link and off by the other. The value for multilink will be toggled on this device and LCP will be re-negotiated.

Restoring boot DNLDSEG configuration

Generated upon entering Network Load Mode to initialize specific configuration information required for retrieving new code image. Generated upon entering operational after a successful code burn into flash.

Restoring boot EEPROM configuration

Generated when restoring values in EEPROM configuration, this occurs when entering a load or operational mode.

Running in System Load mode

Generated when the bridge/router is starting in System Load (Boot) mode. This is the mode for software upgrades. Once the software upgrade has been successfully completed, the bridge/router restarts in Operational mode.

Running in OPERATIONAL mode

Generated when the bridge/router is starting in Operational mode. This is the mode for normal operations of the bridge/router.

Starting BCP negotiation with <remote site alias>

Generated when the Bridging Control Protocol negotiation has been initiated with the remote site device associated with the stated remote site profile.

Starting CCP negotiation with <remote site alias>

Generated when the Compression Control Protocol negotiation has been initiated with the remote site device associated with the stated remote site profile.

Starting IPCP negotiation with <remote site alias>

Generated when the Internet Protocol Control Protocol negotiation has been initiated with the remote site device associated with the stated remote site profile.

Starting IPXCP negotiation with <remote site alias>

Generated when the Internet Packet Exchange Control Protocol negotiation has been initiated with the remote site device associated with the stated remote site profile.

Station address table has been filled

Generated when the station address table is filled. This event is not regenerated until the table size drops below 3/4 full and then fills again.

STP disabled

Generated when STP is disabled.

STP enabled

Generated when STP is enabled.

TFTP: stop putting filename to <IP address>

The bridge/router has sent the final data packet of a file (filename), but has timed out before receiving the final ACK. The session may or may not have succeeded in delivering the entire file.

TFTP: <IP address> finished getting filename

The bridge/router has sent the final packet of a file (filename) that a LAN device with IP address displayed was getting from the bridge/router.

TFTP: <IP address> finished putting filename

The bridge/router has ACK-ed the last packet of a file (filename) that a LAN device with IP address displayed was putting onto the bridge/router.

TFTP: <IP address> getting filename

A LAN device with IP address displayed is getting a file (filename) from the bridge/router.

TFTP: <IP address> putting filename

A LAN device with IP address displayed is putting a file (filename) onto the bridge/router.

Alarm logs:

* Auto-learning of LMI type on link X unsuccessful

  Generated when the LMI type on a link is not successfully auto-learned.

* Bad internal block checksum detected

  Generated when power up diagnostics finds a fault in the internal block of the EEPROM.

* Closing remote site X (call limit)

  Generated when the specified number of calls has been exceeded.

* Closing remote site X (frame relay closing)

  Generated due to the frame relay protocol being disabled on the bridge/router.

* Closing remote site X (force disconnect)

  Generated when the connection to a remote site is being closed due to a force disconnect.

* Closing remote site X (inactivity)

  Generated when the connection to a remote site is being closed due to an inactivity timeout.

* Closing remote site X (IP triggered RIP)

  Generated when the connection to a remote site is being closed due to failure of IP triggered RIP.

* Closing remote site X (IPX demand RIP)

  Generated due to the failure of IPX demand RIP.

* Closing remote site X (last session)

  Generated due to the termination of the last session.

* Closing remote site X (link disabled)

  Generated due to the link being disabled by the operator.

* Closing remote site X (no NCPs open)

  Generated when no Network Control Protocols operating.

* Closing remote site X (nonexistant)

  Generated when remote site being deleted by the operator.

* Closing remote site X (not enabled)

  Generated when remote site autocall being disabled by the operator.

* Closing remote site X (PVC change)

  Generated due to the enabling or disabling of PPP encapsulation over frame relay.

* Closing remote site X (resumption failure)

  Generated due to failure to resume a suspended connection.

* Closing remote site X (scheduled down)

  Generated due to time-of-day schedule deactivation of the connection.

* Closing remote site X (suspension timeout)

  Generated due to reaching maximum time that the connection may be suspended.

* Closing remote site X (usage limit)

  Generated due to reaching usage limit for this 24 hour period.

* Config. erase failed

  Generated when, during a software update, the device configuration is not erased from the non-volatile memory within the time limit. Possible hardware fault.

* Configuration saved

  Generated when the save configuration option has been activated.

* Configuration too large to be saved

  Generated when the bridge/router attempts to save a configuration that does not fit in the reserved area of non-volatile RAM.

* Connection to LAN X failed, trying ...

  Generated when failure of the LAN interface external loopback test is detected.

*Count overflow. Reset to history size.

  Generated when the number of events since the event log was cleared exceeds the counter capacity (32,768). Event numbers will start over again from 1.

* DHCP server – out of addresses in IP pool

  Generated when the last address from the DHCP IP Address pool has been assigned to a device.

* Download aborted – Incomplete file

  Generated when a TFTP download is aborted before the file transfer is complete

* Download aborted – Invalid FCS

  Generated when there is a checksum failure after a file download.

* Download aborted – Incompatable boot code

  Generated when the operating code file downloaded is incopatible with the boot code in this device. The updated boot code software must be loaded before the operating code can be updated.

* Download aborted – Incompatible software

  Generated when the software downloaded is incompatible with this device

* Download configuration too large

  The configuration file that is being downloaded will not fit in the memory of this router

* E-mail server added to firewall

  The IP address of the E-mail server added to the table of services available through the firewall.

* E-mail server removed from firewall

  The IP address of the E-mail server removed from the table of services available through the firewall.

* Erasing config. block: starting

  Generated as a notification that the configuration of this device is being erased from non-volatile memory prior to loading a software update.

* Error loading configuration

  Generated during a warm start when an error is detected while restoring a configuration from non-volatile RAM.

* Feature upgrade failure, try again

  Generated when the device detects a checksum error for the feature upgrade block.

* File copy failed: file crc: X, verify crc: X

  Generated when performing a code upgrade and the calculated CRC was found to be different from the transferred CRC (flash.fcs) value.

* FTP server added to firewall

  The IP address of the FTP server added to the table of services available through the firewall.

* FTP server removed from firewall

  The IP address of the FTP server removed from the table of services available through the firewall.

* IP protocol parameters initialized

  Generated when IP protocol communications to a remote site configured for frame relay are negotiated successfully.

* IP protocol parameters uninitialized

  Generated when IP protocol communications fail for a remote site configured for frame relay because an IP address does not exist.

* LAN connection established

  Generated on startup when integrity of the LAN interface has been successfully verified by the external loopback test.

* Link X attached to remote site <remote site alias>

  Generated when Link X has been identified as a connection to the specified remote site.

* Link X busy

  Generated when a call was attempted on the link while it already had a call in progress. This may also occur if the link was not activated at the time of the call.

* Link X control signals down

  Generated when a high-to-low transition is detected on the CD control signal. Note that there is no associated event for the low-to-high transition.

* Link X connection rejected

  Generated when Link X connection is being terminated as it could not be attached to a remote site. This may be due to usage limits or suspension resumptions.

* Link X, DLCI Y attached to remote site <remote site alias>

  Generated for frame relay applications when a connection has been made for the DLCI associated with the remote site alias.

* Link X down

  Generated when a WAN link goes down.

* Link X down to <remote site alias>

  Generated when a WAN link connection to the specified remote site goes down.

* Link X – LMI Error Threshold exceeded

  Generated when the defined error threshold has been exceeded on the specified link.

* Link not available for remote site <remote site alias>

  The link associated with this remote site is already in use.

*Link not configured for frame relay on remote site <remote site alias>

  The link that has been assigned to this remote site is not configured for frame relay.

*Link not configured for leased line on remote site <remote site alias>

  The link that has been assigned to this remote site is not configured for leased line operation.

* Link X Outgoing Data Call to [DN]

  Generated when a data call is outgoing to the dialing network.

* Link X up

  Generated when a WAN link comes up.

* Link X up at Y baud

  Generated when frame relay link is established.

* Link X up to <remote site alias>

  Generated when a WAN link connection to the specified remote site comes up.

* Local DNS server added to firewall

  The IP address of the Local DNS server added to the table of services available through the firewall.

* Local DNS server removed from firewall

  The IP address of the Local DNS server removed from the table of services available through the firewall.

* NAT UDP flooding – Possible security risk. Src is <IP address>

  Generated when more than the allowed maximum number of UDP entries has been attempted. This feature is in place to prevent denial of service attacks. The source IP address of the UDP datagrams is displayed.

* NAT table full

  Generated when no more ports are available for Network Address Translation.

* No available remote site for learned DLCI

  Generated when, during Frame Relay Auto-learning, the remote site table is filled, no space is available to create another entry. The user must manually edit the table to remove some remote site profiles before another entry can be made.

* No available remote site for leased line X

  Generated when attempting to set up a default leased line remote site and the remote site table is filled, no space is available to create another entry. The user must manually edit the table to remove some remote site profiles before another entry can be made.

* No saved configuration, using default

  Generated during a cold start when no saved configuration is available.

* No remote site available

  The remote site table is full, there is no space available to create a remote site profile for this IP address. The user must manually edit the table to remove some remote site profiles before another entry can be made.

* Old download method! Load in \"*.all\" file

  Generated when an attempt is made to load a *.fcs or *.lda format program file into hardware which will only accept *.all format code.

* Old format configuration, using default

  Generated when the saved configuration does not match the expected correct revision number. The old configuration formats will not be used.

* POP2/POP3 server added to firewall

  The IP address of the POP2/POP3 server added to the table of services available through the firewall.

* POP2/POP3 server removed from firewall

  The IP address of the POP2/POP3 server removed from the table of services available through the firewall.

* Remote Site <remote site alias> already active

  Generated when a connection is attempted to a remote site that is busy.

* Remote Site <remote site alias> already connected

  Generated when a connection is attempted to a site that is already connected to this router.

* Remote site <remote site alias> frame relay closing

Generated when frame relay is disabled on the link to the specified remote site.

* Remote site <remote site alias> resumed

  Generated when the connection to the specified remote site has been resumed.

* Remote Site <remote site alias> still closing

Generated when a connection is attempted to a remote site whose link is still in the process of being disconnected.

* Remote site <remote site alias> suspended

Generated when the connection to the specified remote site has been suspended.

* Remote site <remote site alias> terminated

Generated when the connection to the specified remote site has been terminated for connection management.

* Results of IPCP negotiation are incompatible

Generated when IPCP negotiations with a remote site PPP router result in a incompatible IP configuration. The remote site is then disconnected.

* Running in System Load mode

Generated when entering System Load Mode in preparation for a download of code to be burned into flash.

* SECURITY ALERT: SNMP community <X> has write access enabled to "ALL" hosts

The SNMP community displayed has had write access enabled to all hosts on the network; anyone may access any host to make changes.

* Service added to firewall

The IP address of the Service added to the table of services available through the firewall.

* Service removed from firewall

The IP address of the Service removed from the table of services available through the firewall.

* (T1/E1) Blue Alarm On

Alarm Indicator Signal of all 1s is being generated – triggered by DTE loss of signal.

* (T1/E1) Blue Alarm Off

Alarm Indicator Signal is being turned off – DTE signal restored.

* (T1/E1) Red Alarm On

Red Carrier Failure Alarm is being declared – caused by over 2 seconds of Loss Of Signal or Out Of Frame errors. Causes Yellow alarm indicator signal to be transmitted.

* (T1/E1) Red Alarm Off

Red Carrier Failure Alarm is being turned off – alarm cleared after 10 seconds of error free signal received.

* (T1/E1) Yellow Alarm On

Yellow Carrier Failure Alarm is being declared – generated upon receipt of Yellow Alarm indicator signal from far end unit.

* (T1/E1) Yellow Alarm Off

  Yellow Carrier Failure Alarm is being turned off – alarm cleared when Yellow Alarm indicator signal from far end stops.

* Telnet server removed from firewall

  The IP address of the Telnet server removed from the table of services available through the firewall.

* Telnet server added to firewall

  The IP address of the Telnet server added to the table of services available through the firewall.

* TFTP: Abort. ACK retry exceeded

  Aborted a TFTP session because the bridge/router did not receive a new data packet within the TFTP "T1" times "N2" interval.

* TFTP: Abort. ACK timeout

  Aborted a TFTP session because the bridge/router did not receive an ACK for the last data packet it sent within the TFTP "T1" times "N2" interval.

* TFTP: Abort. Error (#) received

  Aborted a TFTP session because of the reception of a TFTP error message from the connected device. The errors are: 0 - not defined, 1 - file not found, 2 - access violation, 3 - disk full or allocation exceeded, 4 - illegal TFTP operation, 5 - unknown transfer ID, 6 - file already exists, 7 - no such user.

* Unable to allocate memory for DHCP server save

  Generated when the memory on this device has become too fragmented to find a contiguous block of memory large enough for the DHCP server tables. Reset the device to defragment memory.

* Unable to bind UDP Boot P client port

  Generated as a result of an internal device error. Try resetting the device. If this is unsuccessful, contact a service representative.

* Unable to bind UDP Boot P server port

  Generated as a result of an internal device error. Try resetting the device. If this is unsuccessful, contact a service representative.

* Unable to bind UDP DHCP server port

  Generated as a result of an internal device error. Try resetting the device. If this is unsuccessful, contact a service representative.

* Unable to route!! UDP failure

  Generated when the device tried to open an already open UDP channel, causing IP routing to fail.

* Unknown call type on remote site <remote site alias>

  The attempted call is not a Frame Relay or PPP leased line call. Possible cause is a remote site profile being deleted while a connection attempt is being made.

* WWW (HTTP) server removed from firewall

  The IP address of the WWW (HTTP) server removed from the table of services available through the firewall.

* WWW (HTTP) server added to firewall

  The IP address of the Telnet WWW (HTTP) added to the table of services available through the firewall.

* X count overflow. Reset to history size

  Generated when the number of items logged exceeds the space available. X = "ALARM" or "EVENT"

PPP security logs:

CHAP authentication failure so terminate link.

Generated when the CHAP authentication sent by this router in response to a request from a remote site is rejected.

CHAP failed for <remote site alias>

Generated when the remote site router failed a CHAP authentication request from this IOLINK-130. The remote site name is displayed if known.

CHAP failed to complete

Generated when the remote site router sent a CHAP challenge and this IOLINK-130 sent a response, but no further information was received from the remote site router.

CHAP login refused by <remote site alias>

Generated when the remote site router sent a CHAP challenge and this IOLINK-130 sent a response, and the remote site router refused the connection. The remote site name is displayed if known.

Link X refused to authenticate

Generated when the remote site router refused to do authentication.

PAP authentication failure for user Y

Generated when the PAP password sent by this router in reply to the remote site router PAP password request is rejected.

PAP failed for <remote site alias>

Generated when the remote site router failed a PAP authentication request from this IOLINK-130. The remote site name is displayed if known.

PAP X failed to complete (Y)

Generated when the remote site router sent a PAP password request and this IOLINK-130 sent the PAP password in reply, but no further information was received from the remote site router.

PAP X peer failed to authenticate

Generated when the remote site router did not respond to a request to authenticate.

Possible Intruder <IP address> exceeded password attempts limit

A telnet connection attempt from the displayed IP address to gain access to the router management menus has tried to login over three times with incorrect passwords within the past ten minutes. This may be an attempt to gain unauthorized access to the management of this router. Any further attempts within the next ten minutes form this IP address to gain access will be rejected.

4 Programmable Filtering

Programmable filtering gives the network manager the ability to control under what conditions Ethernet frames are forwarded across bridge or bridge/router ports. There are many reasons why this might need to be accomplished, some of which are security, protocol discrimination, bandwidth conservation, and general restrictions.

To reach a specific filtering goal, there is usually more than one possible filter expression that may be used. This of course is dependent on the specific filtering requirement, and how flexible the filter should be.

The following pages describe how programmable filters may be used in typical applications. Although this is only a small sampling of the many possibilities, a cross-section of use of filters is presented.

MAC Address Filtering

Security

The need for security has become increasingly important in Local Area Networking, and with the use of programmable filters, security may be easily and effectively implemented across segment boundaries. By defining a programmable filter, the network manager may control what traffic is allowed between LAN segments, thereby controlling the security of resources by preventing unauthorized user access.

The IOLINK-130 Ethernet bridge/router provides three built-in functions – in addition to defined programmable masks – to control the access to resources. The first function is "Filter if Source"; the second is "Filter if Destination." The third function allows you to change the filter operation from "positive" to "negative". Positive filter operation causes the specified MAC addresses to be filtered according to the entered method. Negative filter operation causes the specified MAC addresses to be forwarded according to the entered method.

You may easily prevent any station on one segment from accessing a specific resource on the other segment; for this, "positive" filtering and the use of "Filter if Destination" would be appropriate. If you want to disallow a specific station from accessing any service, "Filter if Source" could be used.

You may easily prevent stations on one segment from accessing all but a specific resource on the other segment; for this, "negative" filtering and the use of "Forward if Destination" would be appropriate. If you want to disallow all but a specific station from accessing any service on the other segment, the use of "Forward if Source" could be used.

Example cases are found on the following pages.

TCP/IP, XNS, and Novell Netware frame formats, as well as some common Ethernet type codes, are found by the back cover.

Security—"Filter if Destination"

Filter if Destination is a function that allows you to filter an Ethernet frame based on the destination of its address. If the destination address equals the address that the Filter if Destination function has been applied to, the frame is filtered.

Example:
Assume that a host Computer is located on LAN segment 2 located on a partner bridge/router with an Ethernet address of:

00-00-01-02-03-04 (host Ethernet address)

Since each station on a LAN has a unique Ethernet address, this address uniquely identifies this host computer.

To prevent LAN users located on segment 1, located on the local bridge/router, from accessing this host system, follow the instructions below:

1 From the MAIN MENU of the console of the local bridge/router, enter a **1.**

(Enter an "=" from any menu to go back to the MAIN MENU.)
This will place you at the **CONFIGURATION MENU,** where access to the filtering menu is obtained.

2 From the CONFIGURATION MENU, enter an **8.**

This will place you at the **FILTER SET-UP MENU,** where access to the individual filtering menus is obtained.

3 From the FILTER SET-UP MENU, enter a **1.**

This will place you at the **MAC ADDRESS FILTERS MENU,** where access to the MAC Address filters is obtained.

4 From the MAC ADDRESS FILTERS MENU, make sure that Filter Operation is currently set to "positive".

This will cause the MAC Address Filters specified to be used for filtering frames with the specified MAC addresses.

5 From the MAC ADDRESS FILTERS MENU, enter a **1.**

This will place you at the first **EDIT MAC ADDRESS FILTER MENU** screen.

At the prompt enter the MAC address for which you want to specify the filter.

6 Enter the 12-digit Ethernet address of the host system in the following format:

000001020304 (enter a Return)

The edit screen will fill in the information that the table knows about this address. For this example, let us assume that it knows that the address is "present" and located on the LAN of the partner bridge/router.

7 Enter a **4** to Enable the "**Filter if Destination**" parameter. The screen will be updated with the new information.

At this point, the address is added to the permanent filter table of the local LAN. This entry, therefore, will not be subject to the aging timer, and will remain active until it is removed from the permanent entry table.

When a frame of information is seen on the local LAN that contains the address of the host system in the destination field of the frame, the bridge/router will not forward it, effectively preventing any access to this host from the local LAN.

Security—"Filter if Source"

Filter if Source is a function that allows you to filter an Ethernet frame if the source address of the frame equals the address that the Filter if Source function has been applied to.

Example:

Assume that a Personal Computer is located on segment 1 on the local bridge/router. This station is a community station that various departments may use for general processing. However, this station may only access those services that exist on its local segment, and it must be restricted from accessing any services on remote LANs. This can be easily accomplished with a "Filter if Source."

The Ethernet Address for this Personal Computer is: **01-02-03-04-05-06**

Again, this address uniquely identifies this computer station.

To configure the bridge/router to ensure that this station is unable to access facilities on a remote LAN segment, follow the instructions below:

1 From the MAIN MENU of the console of the local bridge/router, enter a **1.**

(Enter an "=" from any menu to go back to the MAIN MENU.)
This will place you at the **CONFIGURATION MENU,** where access to the filtering menu is obtained.

2 From the CONFIGURATION MENU, enter an **8.**

This will place you at the **FILTER SET-UP MENU,** where access to the individual filtering menus is obtained.

3 From the FILTER SET-UP MENU, enter a **1.**

This will place you at the **MAC ADDRESS FILTERS MENU,** where access to the MAC Address filters is obtained.

4 From the MAC ADDRESS FILTERS MENU, make sure that the Filter Operation is currently set to "positive".

This will cause the MAC Address Filters specified to be used for filtering frames with the specified MAC addresses.

5 From the MAC ADDRESS FILTERS MENU, enter a **1.**

This will place you at the first **EDIT MAC ADDRESS FILTER MENU** screen.
At the prompt enter the MAC address for which you want to specify the filter.

6 Enter the 12-digit Ethernet address of the Personal Computer system in the following format:

010203040506 (enter a Return)

The edit screen will fill in the information that the table knows about this address. For this example, let us assume that it knows that the address status is [not present] and is of [unknown] location.

In this example, the bridge/router is not aware of this station as of yet. The station has probably not been active for the bridge/router to "learn" any information about it.

Therefore, you will have to tell the bridge/router a little bit more about the station.

7 Enter a **2** to enter the location of the station.

8 The bridge/router will prompt you for the LAN that the station is located on; enter the name of the partner bridge/router LAN (LAN345678, for example).

Note that the Status of the address is marked as [present], the location is updated to LAN345678 and the Permanent entry is [enabled].

9 Enter a **3** to [enable] the "**Filter if Source**" parameter. The edit screen will be updated to show the new information.

At this point, the address is added to the permanent filter table of the local LAN. This entry, therefore, will not be subject to the aging timer, and will remain active until it is removed from the permanent entry table.

When a frame of information is seen on the local LAN that contains the address of the Personal Computer in the source field of the frame, the bridge/router will not forward it, effectively preventing any access from the PC to remote LANs.

Most programmable filtering options may be used for security purposes. The examples above are specific instances where the two "Filter if" functions may be used.

Security—"Forward if Destination"

Forward if Destination is a function that allows you to forward an Ethernet frame based on the destination of its address and filter all other frames. If the destination address equals the address that the Forward if Destination function has been applied to, the frame is forwarded.

Example:

Assume that a host Computer is located on LAN segment 2 located on a partner bridge/router with an Ethernet address of:

00-00-01-02-03-04 (host Ethernet address)

Since each station on a LAN has a unique Ethernet address, this address uniquely identifies this host computer.

To prevent LAN users located on segment 1, located on the local bridge/router, from accessing any only this host system and no other systems, follow the instructions below:

1 From the MAIN MENU of the console of the local bridge/router, enter a **1.**
(Enter an "=" from any menu to go back to the MAIN MENU.)

This will place you at the **CONFIGURATION MENU,** where access to the filtering menu is obtained.

2 From the CONFIGURATION MENU, enter an **8.**

This will place you at the **FILTER SET-UP MENU,** where access to the individual filtering menus is obtained.

3 From the FILTER SET-UP MENU, enter a **1.**

This will place you at the **MAC ADDRESS FILTERS MENU,** where access to the MAC Address filters is obtained.

4 From the MAC ADDRESS FILTERS MENU, make sure that the Filter Operation is currently set to "negative".

This will cause the MAC Address Filters specified to be used for forwarding frames with the specified MAC addresses.

5 From the MAC ADDRESS FILTERS MENU, enter a **1.**

This will place you at the first **EDIT MAC ADDRESS FILTER MENU** screen.

At the prompt enter the MAC address for which you want to specify the filter.

6 Enter the 12-digit Ethernet address of the host system in the following format:
000001020304 (enter a Return)

The edit screen will fill in the information that the table knows about this address. For this example, let us assume that it knows that the address is "present" and located on the LAN of the partner bridge/router.

7 Enter a **4** to Enable the "**Forward if Destination**" parameter. The edit screen will be updated to show the new information.

At this point, the address is added to the permanent filter table of the local LAN. This entry, therefore, will not be subject to the aging timer, and will remain active until it is removed from the permanent entry table.

When a frame of information is seen on the local LAN that contains the address of the host system in the destination field of the frame, the bridge/router will forward it. All other frames seen on the local LAN that are destined for the remote LAN will be filtered.

Security—"Forward if Source"

Forward if Source is a function that allows you to forward an Ethernet frame if the source address of the frame equals the address that the Forward if Source function has been applied to.

Example:

Assume that a Personal Computer is located on segment 1 on the local bridge/router. This station belongs to the head of Marketing. This station requires access to all the services that exist on the remote LAN but no other station on the local LAN is allowed to access the remote LAN. This can be easily accomplished with a "Forward if Source."

The Ethernet Address for this Personal Computer is: **01-02-03-04-05-06**

Again, this address uniquely identifies this computer station.

To configure the bridge/router to ensure that only this station is able to access facilities on a remote LAN segment, follow the instructions below:

1 From the MAIN MENU of the console of the local bridge/router, enter a **1.**
(Enter an "=" from any menu to go back to the MAIN MENU.)

This will place you at the **CONFIGURATION MENU,** where access to the filtering menu is obtained.

2 From the CONFIGURATION MENU, enter an **8.**

This will place you at the **FILTER SET-UP MENU,** where access to the individual filtering menus is obtained.

3 From the FILTER SET-UP MENU, enter a **1.**

This will place you at the **MAC ADDRESS FILTERS MENU,** where access to the MAC Address filters is obtained.

4 From the MAC ADDRESS FILTERS MENU, make sure that the Filter Operation is currently set to "negative".

This will cause the MAC Address Filters specified to be used for forwarding frames with the specified MAC addresses.

5 At this menu, enter a **1.**

This will place you at the first **EDIT MAC ADDRESS FILTER MENU** screen.

At the prompt enter the MAC address for which you want to specify the filter.

6 Enter the 12-digit Ethernet address of the Personal Computer system in the following format:
010203040506 (enter a Return)

The edit screen will fill in the information that the table knows about this address. For this example, let us assume that it knows that the address status is [not present] and is of [unknown] location.

In this example, the bridge/router is not aware of this station yet. The station has probably not been active for the bridge/router to "learn" any information about it.

Therefore, you will have to tell the bridge/router a little bit more about the station.

7 Enter a **2** to enter the location of the station.

8 The bridge/router will prompt you for the LAN that the station is located on; enter the name of this bridge/router's LAN (LAN456789 for example).

Note that the Status of the address is marked as [present], the location is updated to LAN456789 and the Permanent entry is [enabled].

9 Enter a **3** to [enable] the "**Forward if Source**" parameter. The edit screen will be updated to show the new information.

At this point, the address is added to the permanent filter table of the local LAN. This entry, therefore, will not be subject to the aging timer, and will remain active until it is removed from the permanent entry table.

When a frame of information is seen on the local LAN that contains the address of the Personal Computer in the source field of the frame, the bridge/router will forward it. All other frames seen on the local LAN that are destined for the remote LAN will be filtered.

Most programmable filtering options may be used for security purposes. The examples above are specific instances where the two "Forward if" functions may be used. Filter masks are presented in subsequent pages of this section.

Pattern Filter Operators

The following operators are used in creating Pattern filters and will be discussed further in the following pages. For additional information refer to the octet locations diagrams at the back of this manual. Each octet location may contain a HEX value.

- offset — Used in pattern filters to determine the starting position to start the pattern checking.

  Example: 12-80 — This filter pattern will match if the packet information starting at the 12th octet equals the 80 of the filter pattern.

| OR — Used in combination filters when one **or** the other conditions must be met.

  Example: 10-20|12-80 — This filter pattern will match if the packet information starting at the 10th octet equals the 20 of the filter pattern **or** if the packet information starting at the 12th octet equals the 80 of the filter pattern.

& AND — Used in combination filters when one **and** the other conditions must be met.

  Example: 10-20&12-80 — This filter pattern will match if the packet information starting at the 10th octet equals the 20 of the filter pattern **and** the packet information starting at the 12th octet equals the 80 of the filter pattern.

~ NOT — Used in pattern filters to indicate that all packets **not** matching the defined pattern will be filtered.

  Example: ~12-80 — This filter pattern will match if the packet information starting at the 12th octet does not equal the 80 of the filter pattern.

( ) brackets — Used in pattern filters to separate portions of filter patterns for specific operators.

  Example: 12-80&(14-24|14-32) — This filter pattern will be checked in two operations. First the section in brackets will be checked and then the results of the first check will be used in the second check using the first portion of the filter patter. If the packet information starting at the 14th octet equals 24 or 32, and the information at the 12th octet equals 80, the filter pattern will match.

Bridge Pattern Filtering

Protocol Discrimination

Protocol discrimination may be required to prevent or limit the protocols that may traverse a bridged Local Area Network.

In Local Area Networks there may be many different Network and Transport layer protocols that coexist on the same physical media. TCP/IP, DECNET, and XNS are just a few of the common protocols in use today. Each of these protocols is encapsulated within an Ethernet frame, and therefore is transparent to the normal bridging function. If you would like to discriminate against a particular protocol to prevent its use of the bridged LAN facilities, the IOLINK-130 Ethernet bridge/router provides programmable filter masks that may be defined to act on any part of the Ethernet frame.

In the examples below, several protocol types and combinations are presented to demonstrate the use of programmable filter masks to control the protocol traffic between Local Area Network segments. Since there are many possible combinations, these examples are only representative of some of them.

The Bridge Filter Patterns menu is located under the FILTER SET-UP MENU. Within the Bridge Filter Patterns Menu there exists a Help function that can be used as a reference during Bridge Filter Pattern creation. This Help function includes all of the logical operators that may be applied to the mask expression.

Protocol Type Field

Within an Ethernet frame, a protocol field exists at octet 12 and 13. These two octets, or 8-bit bytes, will represent the type of higher level protocol that exists in the Ethernet frame. There are more than 100 different protocol types that are defined for use within an Ethernet frame. In many networks there will be fewer than 10 that are in use, but in many larger networks there may be upwards of 30 or more. This, of course, will depend on the type of equipment and the applications that are being used within the Local Area Network.

Internet Protocol (IP)

The Internet Protocol (IP) is the most widely used protocol within an Ethernet environment. As a result there may be a need to restrict in one form or another this protocol traffic.

Filter all IP Packets

To prevent IP traffic from being passed across the bridged network, a mask must be created that represents this protocol type. The IP protocol type is 0800H.

Since the protocol field starts at octet location 12, the necessary filter mask to prevent IP traffic from traversing the bridged network is as follows: **12-0800**

The 12 is the offset into the Ethernet frame, the "-" is the argument separator, and the 0800 represents the protocol type of IP.

In this example, whenever a frame is seen on the LAN port, for which this filter mask has been specified, with a protocol of type equal to IP, the frame will be filtered.

Note that when you filter on IP frames, all frames using the IP protocol will also be filtered. This includes TCP, UDP, SNMP, etc.

IP, and no more

This example performs just the opposite function to the above example. Only IP packets will be allowed to be passed across the bridged network.

For this function there must be a method to prevent all but IP packets from being filtered. For this the NOT ("~") logical operator is used. The NOT operator specifies that the expression has to be FALSE before the frame is filtered. In other words, only frames that are NOT equal to the expression will be filtered and discarded.

To create this mask, the following expression is entered: **~(12-0800)**

The parenthesis simply ensures that the NOT operator will apply to the entire expression.

In this case, whenever a frame is received, the frame will be filtered if the protocol type is NOT equal to 0800 (IP).

Only one filter pattern may be used that contains the NOT operator.

Transport Control Protocol / Internet Protocol (TCP/IP)

The previous example showed how to filter all Ethernet frames that contained an IP protocol packet. However, IP is used as the Network-layer protocol for more than 40 different Transport-layer protocols, TCP being only one of them. Therefore, with the mask that was used as noted in the previous IP example, all Transport layer protocols that used IP would also be filtered. This may not be desirable in all cases.

For this example, the discrimination of the Transport Layer used within an IP packet will be demonstrated. This requires an AND function, since we want to filter data that both is IP and contains TCP information.

Within the IP frame, there is a single octet field that may be used to indicate the protocol of the Transport layer, or the protocol of the data in the IP packet. If TCP were the protocol within the IP packet, this octet, or 8-bit byte, would be equal to 6.

The location of this field, remembering that the start of the Ethernet frame is always the base reference, is octet 23.

Filter only TCP/IP

To filter only those packets that are TCP/IP, the mask would therefore be: **12-0800&23-06**

The 12-0800 is the IP expression and the 23-06 will represent TCP in an IP frame. The "&" is the logical AND operator, so the expression requires that the frame be both an IP and TCP.

Filter all IP without TCP traffic

To filter all IP packets that do not contain TCP traffic, the mask would be: **12-0800&~(23-06)**

Filter all except TCP/IP

To filter all other packets except TCP/IP packets, the mask would be: **~(12-0800&23-06)**

Local Area Transport (LAT)

The Local Area Transport (LAT) protocol is used exclusively by DEC for terminal access between DEC hosts and terminal servers located on an Ethernet network.

This example is similar to the Internet Protocol example described previously.

The protocol type field value that is used for LAT frames is equal to 6004.

Filter all LAT

Therefore, to filter all LAT frames, the filter mask would be: **12-6004**

Filter all but LAT

To filter all frames but LAT frames, the filter mask would be: **~(12-6004)**

DEC

DEC uses protocol types 6000 to 600F, and although some are undefined, a simple filter mask can be created to filter all DEC traffic.

Filter all DEC

The mask to filter all DEC traffic would be: **12-600X**

The **X** is a variable representing the last four bits (a nibble) of the type. This will effectively filter all Ethernet frames that contain a protocol type of 6000 through to 600F. All 16 possible combinations are covered.

Bandwidth Conservation

Reducing traffic on each LAN segment is one benefit of the bridging functions of a IOLINK-130 Ethernet bridge/router. There are several simple methods that may be used to provide a further reduction of inter-LAN traffic. The examples that follow present a few very simple methods to reduce inter-LAN traffic, without necessarily reducing resource capability.

Ethernet Broadcasting

On an Ethernet LAN, any station may broadcast information to all other stations by setting the Ethernet Destination address to FF-FF-FF-FF-FF-FF. By configuring the destination address to this setting, it is telling all other stations that this is a broadcast message.

In many situations, stations will abuse this broadcasting capability and send useless information to other stations in the network. To prevent this information from being seen across the link on the other LAN segment, a filter mask can be used.

To prevent broadcast information from being passed across the link, use the following filter mask:

0-FFFFFFFFFFFF

This prevents any frame with a destination address field set to the broadcast address from being passed to the second LAN segment across the link.

Ethernet Multicasting

An Ethernet multicast is a frame of data where the destination address has the high-order bit set to a "one" condition. It is similar to a broadcast, but is to be received by a "group" of stations that meet the remainder of the address. In this manner, a broadcast is focused to a specific group of stations.

To filter multicast frames, the following mask could be used: **0-'1XXX'X**

In this example the high-order bit by multi-cast definition must be set to a "one". The single quotes around the first four positions instructs that the four positions constitute 4 bits, or a nibble, of the entire expression; each position representing a single bit. The "1" indicates that that bit position must be equal to a "1" before the expression is true. The X's that are included within the single quotes represent a single don't care for those bit positions in the first nibble. The X that is located outside of the single quotes represents a don't care condition for the later nibble. NOTE: With this mask, both broadcast frames and multicast frames will be filtered.

General Restrictions

Bridge Filter Masks may be created to generally restrict access for various purposes. Some of these purposes may be to filter specific combinations of information. This section will generally depict masks that may be created to control traffic across the bridged LAN network.

Internet Addresses

Within the Internet Protocol, there exist two address fields that are designated the Source and Destination Internet Addresses. It is these addresses that the IP uses for routing purposes.

To filter Internet Addresses, a mask must be created to look at the Source or Destination address field within the IP header.

As an example, assume a station's Internet address is equal to 128.001.002.003, and a restriction is desired to prevent any other station from across the link on the opposite LAN from gaining access to it. In this case, the mask must filter any IP packet that is destined for this Internet address. The Destination address field within the IP header is at an offset of 30 octets into the Ethernet frame. This address is four octets long.

(Note: Although an Internet address is written in decimal notation, the address within the IP header is always in hexadecimal.)

To accomplish this, the mask would look like this: **12-0800&30-80010203**

This will filter IP packets that contain the Internet address of 128.001.002.003.

As another example, assume that this Internet address should also be filtered if it originates any data. In addition to the mask above, an OR condition will have to be added to look at the IP source address. The new mask would be as follows: **12-0800&(26-80010203|30-80010203)**

This would filter any frame that is both an IP packet destined for or originating from Internet address 128.001.002.003. The parenthesis must be added around the Internet portion to ensure that the proper logical ordering is retained.

Ethernet Station Addresses

Ethernet addresses are assigned to LAN users in blocks. These blocks are normally assigned to manufacturers of Ethernet LAN hardware, and the blocks are sufficiently large to provide unique addresses for a given manufacturer for many years.

Thus, a manufacturer will have a block of addresses, and filtering may be performed to prevent a particular manufacturer's LAN hardware from using the bridge facilities.

As an example, Xerox has a block of addresses that cover the range from 0000AA000000 to 0000AAFFFFFF. To prevent this equipment from accessing facilities on another LAN segment, a generic filter may be created. A mask that looked at the Source Ethernet address field would be required. The mask would be as follows: **6-0000AA**

The remainder of the address is considered a "don't care" condition. This mask results in the entire address block from using the segment LAN facilities.

Mask Combinations

Mask combinations may be required to ensure that a frame is sufficiently qualified before the decision to filter is made. The qualification a frame must go through before a filter decision is made depends on the reason for the filter. Nonetheless, a few examples below have been provided that should aid in the creation of a mask that may require that extra little bit of qualification.

Example

To prevent a specific Ethernet station from accessing any TCP/IP host on the other segment. Assume the Ethernet address is 01-02-03-04-05-06.

The mask would be: **6-010203040506&12-0800&23-06**

Example

To prevent a specific protocol type from accessing a specific Ethernet Address. Assume the Ethernet address is 01-02-03-04-05-06, and the protocol type is Appletalk®. The filter mask would be: **0-010203040506&12-809B**

Example

To prevent any Ethernet address with the 10th bit set to a 0 from accessing a LAT host or an IP host with an Internet address of 128.001.001.128.

This particular mask, although not particularly useful, might be best served by creating two masks instead of one long mask. The decision is up to the Bridge Manager, but a longer mask is always more difficult to understand later. Both methods are presented below:

Combined Filters **4-X'XX0X'&(12-6004|(12-0800&30-80010180))**

Separate Filters **4-X'XX0X'&12-6004**
4-X'XX0X'&12-0800&30-80010180

IP Router Pattern Filtering

Pattern filtering may be used on any portion of the IP frame. IP pattern filtering behaves the same as bridge pattern filtering, except the start of the IP frame is offset 0, because the IP router function of the bridge/router handles only the IP frame itself.

IP pattern filtering may use any combination of filtering operators as described in the bridge pattern filters.

Protocol Discrimination

Protocol discrimination may be required to prevent or limit the protocols within an IP frame that may traverse a routed Local Area Network.

In Local Area Networks, there may be many different Transport layer protocols that coexist within the IP Network layer. TCP, UDP, and ICMP are just a few of the common protocols in use today. Each of these protocols is encapsulated within an IP frame, and therefore is subject to the IP routing function. If you would like to discriminate against a particular protocol to prevent its usage of the routed LAN facilities the IOLINK-130 Ethernet remote bridge/router provides programmable filter masks that may be defined to act on any part of the IP frame.

The IP Router Filter Patterns menu is located under the Filter Set-Up Menu. Within the IP Router Filter Patterns Menu there exists a Help function that can be used as a reference during IP Router Filter Pattern creation. This Help function includes all of the logical operators that may be applied to the mask expression.

IPX Router Pattern Filtering

Pattern filtering may be used on any portion of the IPX frame. IPX pattern filtering behaves the same as bridge pattern filtering, except the start of the IPX frame is offset 0, because the IPX router function of the bridge/router handles only the IPX frame itself.

IPX pattern filtering may use any combination of filtering operators as described in the bridge pattern filters.

The IPX Router Filter Patterns menu is located under the Filter Set-Up Menu. Within the IPX Router Filter Patterns Menu, there exists a Help function that can be used as a reference during IPX Router Filter Pattern creation. This Help function includes all of the logical operators that may be applied to the mask expression.

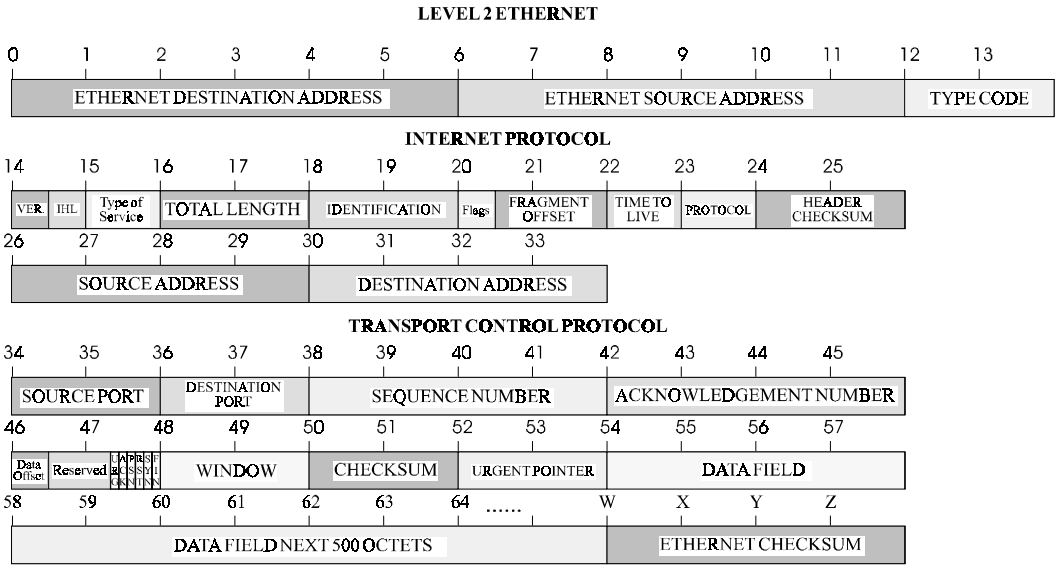
5 Frame Formats

This appendix provides octet locations for the various portions of three of the common Ethernet frames. When creating pattern filters these diagrams will assist in the correct definition of the patterns. The offset numbers are indicated by the numbers above the frame representations.

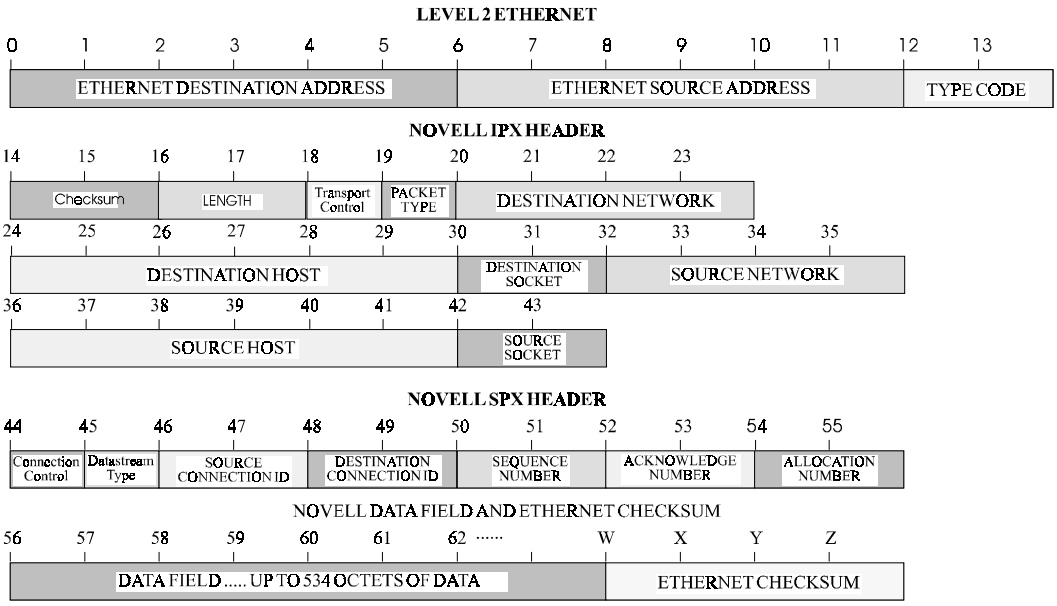
Note the differences in the TCP/IP and Novell frames when bridging and when routing. When routing, the TCP/IP and Novell frames are examined after the Level 2 Ethernet portion of the frame has been stripped from the whole data frame. This means that the offset numbers now start from 0 at the beginning of the routed frame and not the bridged frame.

Some of the common Ethernet type codes are also shown here. The Ethernet type codes are located at offset 12 of the bridged Ethernet frame.

Octet Locations on a Bridged TCP/IP Frame



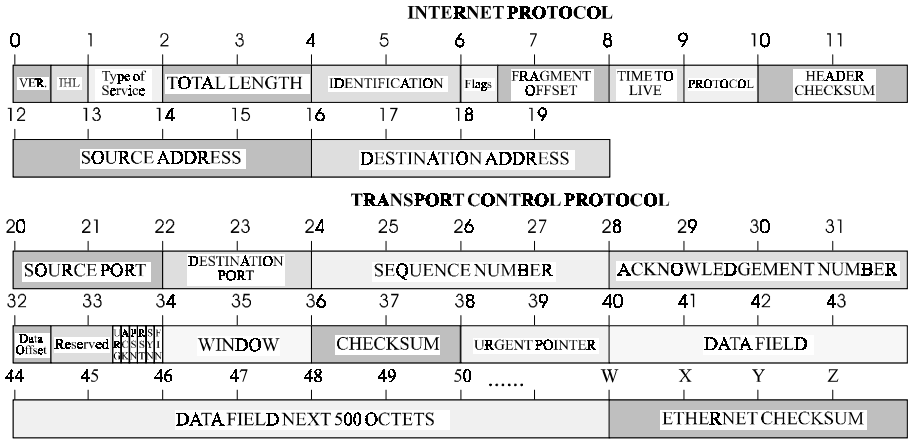
Octet Locations on a Bridged Novell Netware Frame



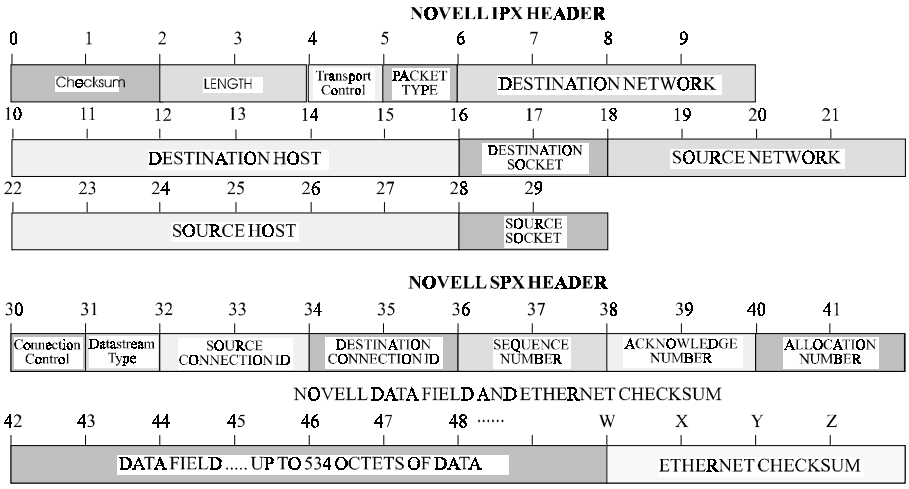
ETHERNET TYPE CODES

Type Code	Description
0800	DOD IP
0801	X.75 Internet
0804	Chaosnet
0805	X.25 Level 3
0806	ARP
0807	XNS Compatibility
6001	DEC MOP Dump/Load
6002	DEC MOP Remote Console
6003	DEC DECNET Phase IV Route
6004	DEC LAT
6005	DEC Diagnostic Protocol
6006	DEC Customer Protocol
6007	DEC LAVC, SCA
8035	Reverse ARP
803D	DEC Ethernet Encryption
803F	DEC LAN Traffic Monitor
809B	Appletalk
80D5	IBM SNA Service on Ether
80F3	AppleTalk AARP (Kinetics)
8137-8138	Novell, Inc.
814C	SNMP

Octet Locations on an IP Routed TCP/IP Frame



Octet Locations on an IPX Routed Novell Netware Frame



Octet Locations on a Bridged XNS Frame