US 20050025129A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2005/0025129 A1**

Meier (43) **Pub. Date: Feb. 3, 2005**

(54) **ENHANCED MOBILITY AND ADDRESS RESOLUTION IN A WIRELESS PREMISES BASED NETWORK**

(76) Inventor: **Robert C. Meier**, Cedar Rapids, IA (US)

Correspondence Address:
JOHN H. SHERMAN, LEGAL DEPT.
INTERMEC TECHNOLOGIES CORPORATION
550 2ND STREET SE
CEDAR RAPIDS, IA 52401 (US)

(21) Appl. No.: **10/790,631**

(22) Filed: **Mar. 1, 2004**

Related U.S. Application Data

(63) Continuation of application No. 09/183,767, filed on Oct. 30, 1998, now Pat. No. 6,701,361, which is a continuation-in-part of application No. 08/916,601, filed on Aug. 22, 1997, now abandoned.

(60) Provisional application No. 60/024,648, filed on Aug. 22, 1996. Provisional application No. 60/043,395, filed on Apr. 2, 1997.

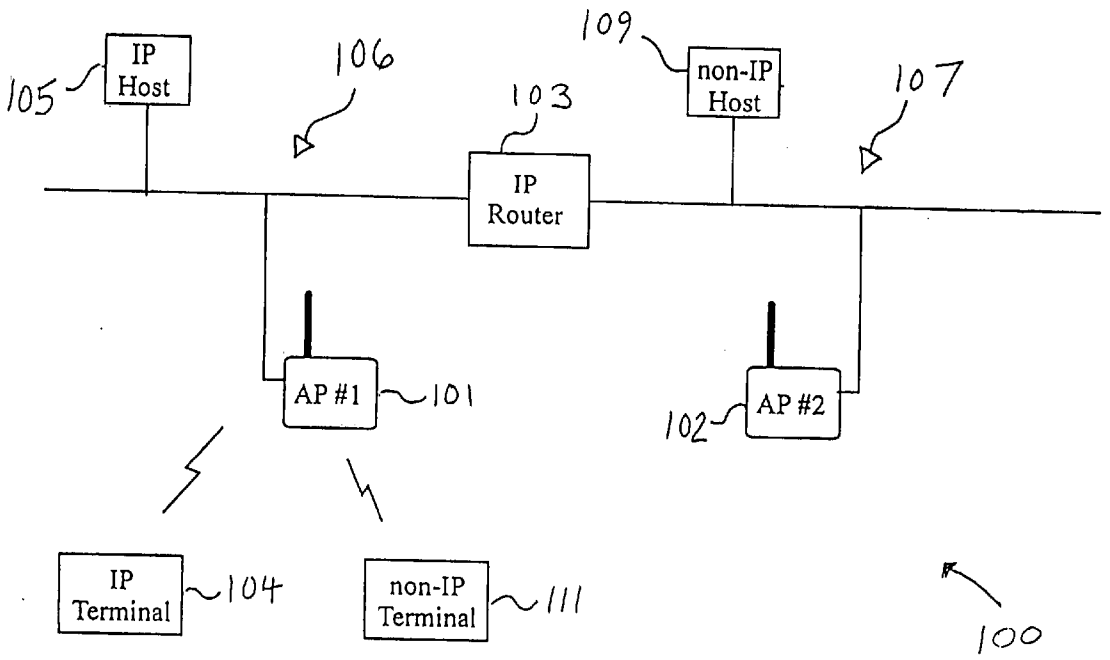
Publication Classification

(51) **Int. Cl.**[7] **H04L 12/66**

(52) **U.S. Cl.** **370/352**

(57) **ABSTRACT**

A premises based wireless network having a multi-segment wired network and a plurality of wireless access points connected to the wired network. The wired network operates according to a wired network protocol which may be the Internet Protocol. Wireless terminals communicate with the wireless access points according to a wireless network protocol, inconsistent with the wired network protocol. Each of the wireless terminals has a wired network address corresponding to one of the wireless access points. Each wireless terminal also has an address according to the wired network protocol. As the wireless terminals roam throughout the premises, protocol tunnels route communications between wireless terminals, thereby preserving communications while roaming by allowing the wireless terminals to retain their wired network addresses during the ongoing communications. The wireless terminals are connected to wireless access points. These wireless access points are in turn linked by data link tunnels to a root access point for a subnet. The data link tunnels enable the root access point for a subnet to forward data to the wireless access points. The forwarded data is not bridged onto the particular subnet that connects the wireless access point and the root access point for that subnet.



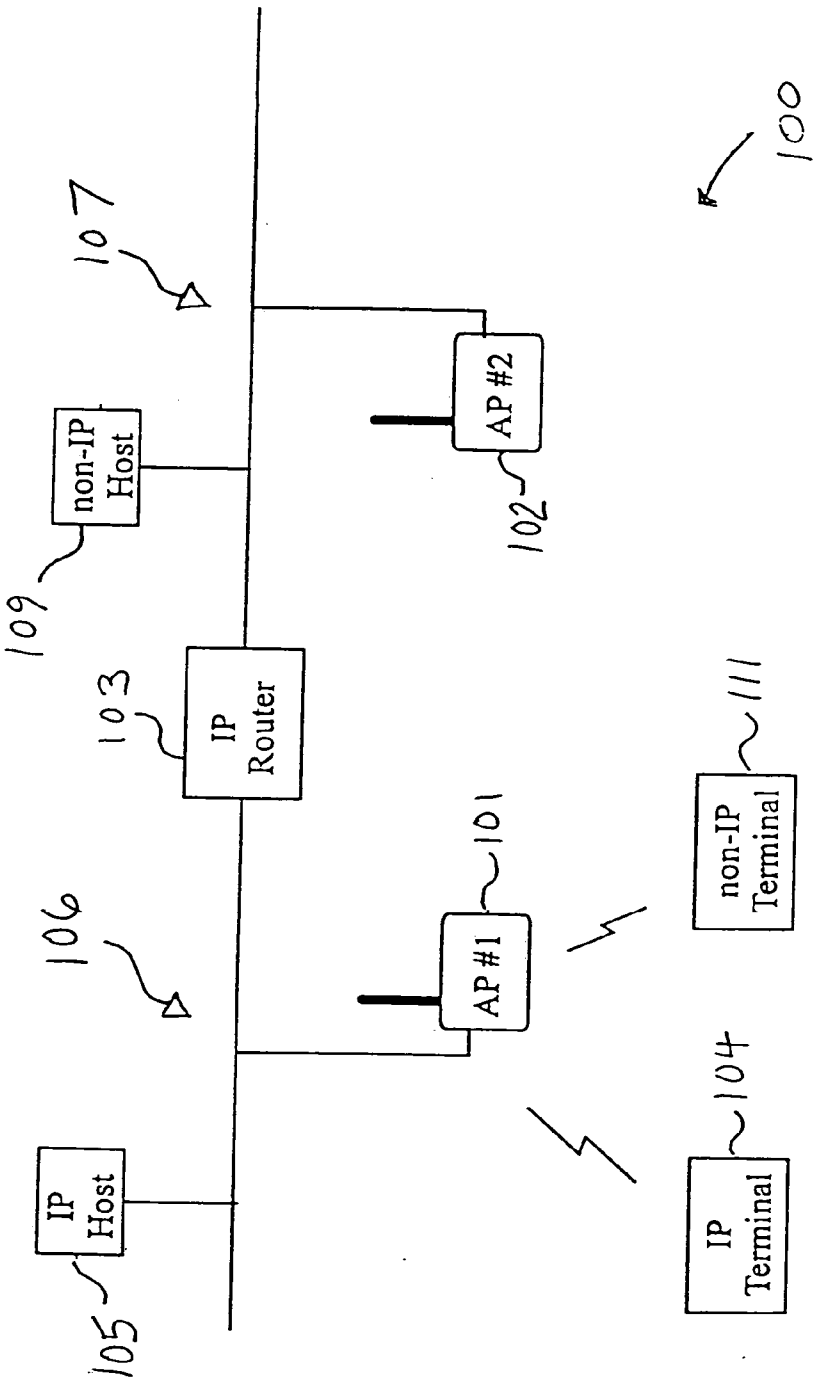

100
107
non-IP Host
109
AP #2
102
IP Router
103
106
AP #1
101
non-IP Terminal
111
IP Host
105
IP Terminal
104

Fig. 1

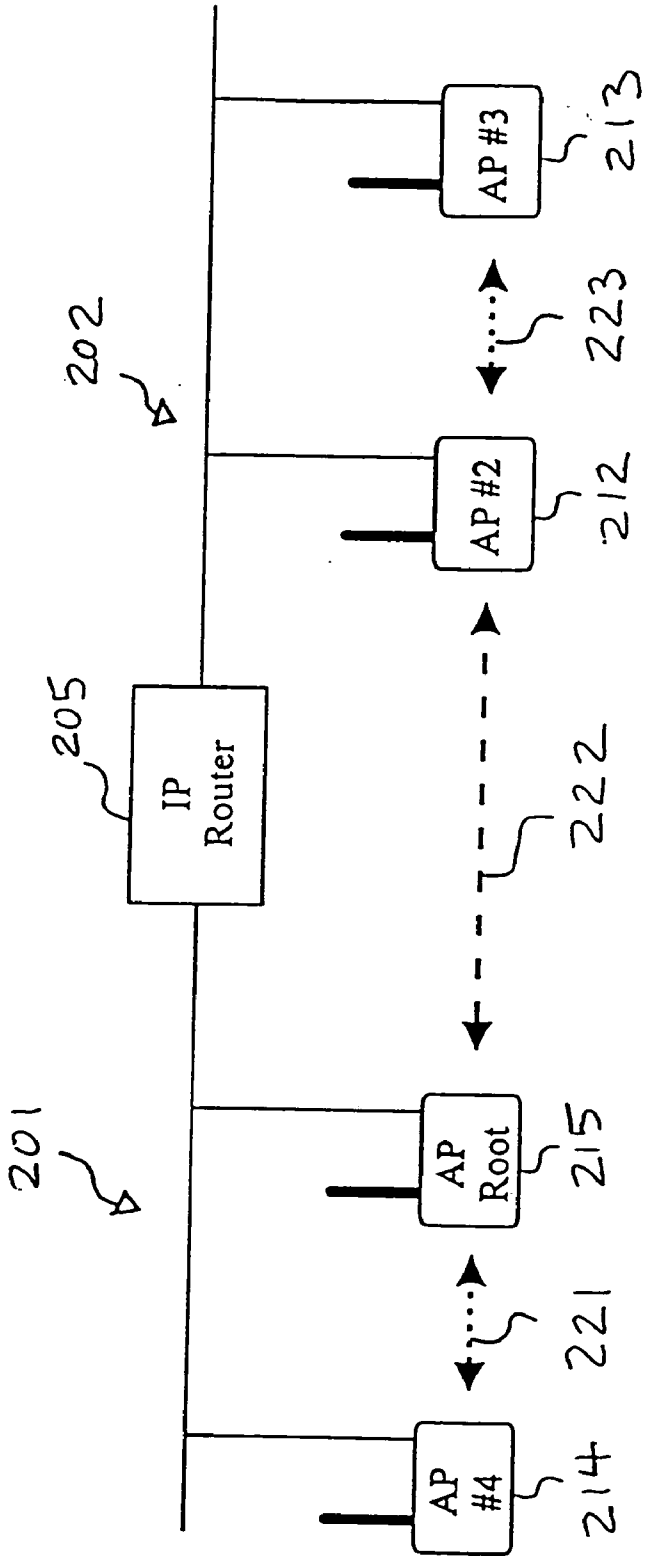

201
202
205
IP Router
AP Root
215
AP #2
212
AP #3
213
AP #4
214
221
222
223

Fig. 2

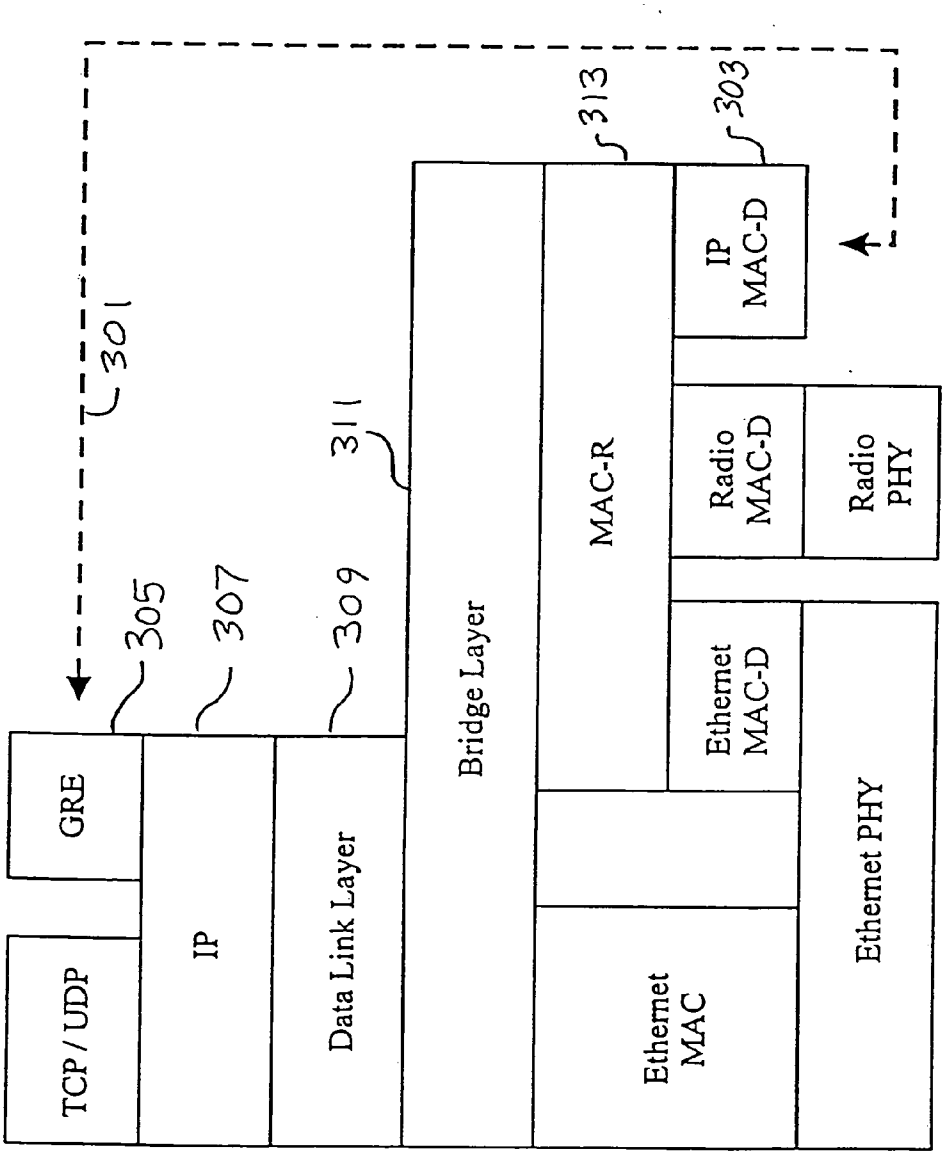

301
305
307
309
311
313
303
TCP / UDP
GRE
IP
Data Link Layer
Bridge Layer
Ethernet MAC
MAC-R
Ethernet MAC-D
Ethernet PHY
Radio MAC-D
Radio PHY
IP MAC-D

Fig. 3

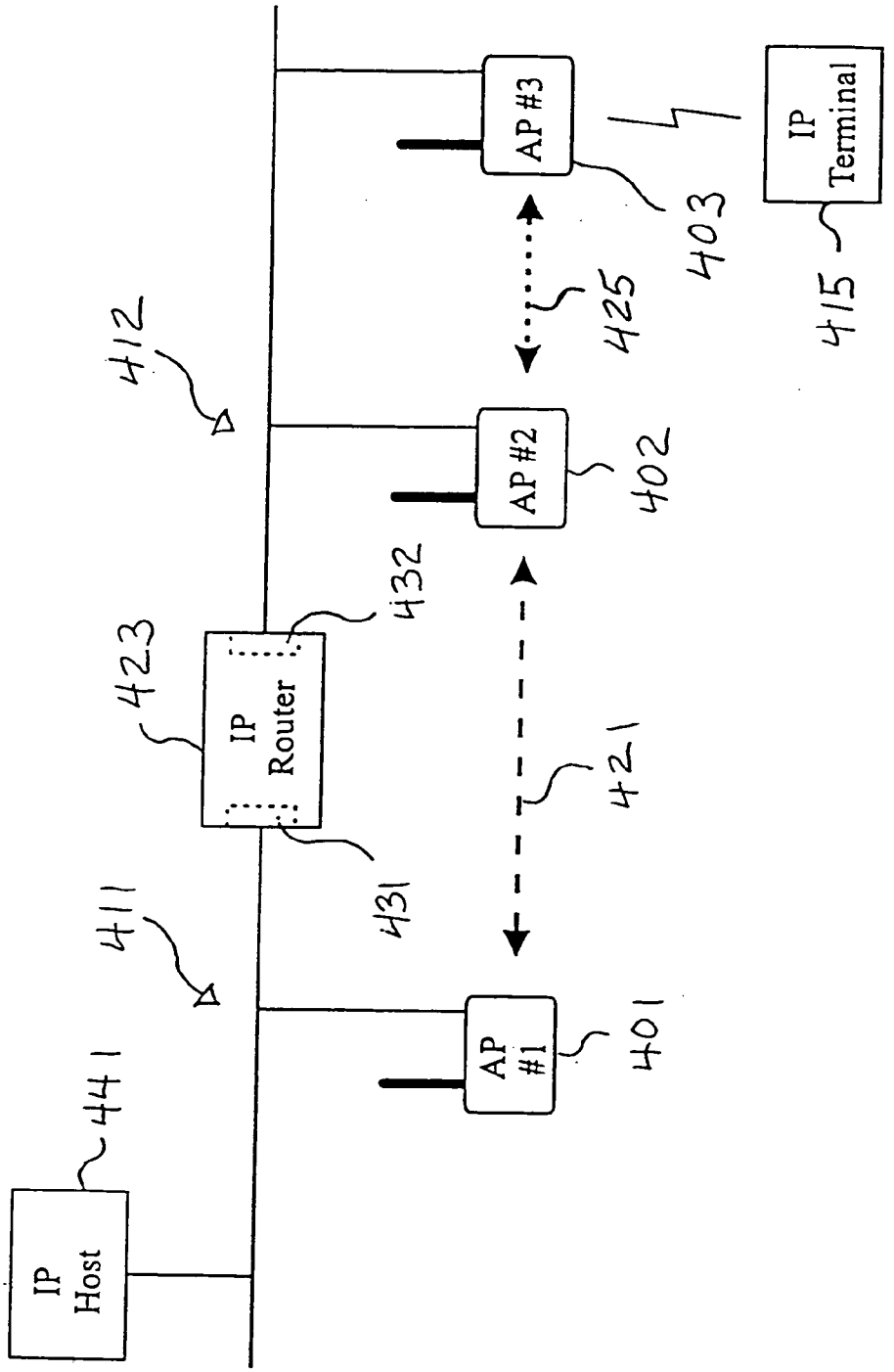

IP Host
441
411
IP Router
423
431
432
412
AP #1
401
AP #2
402
AP #3
403
421
425
IP Terminal
415

Fig. 4

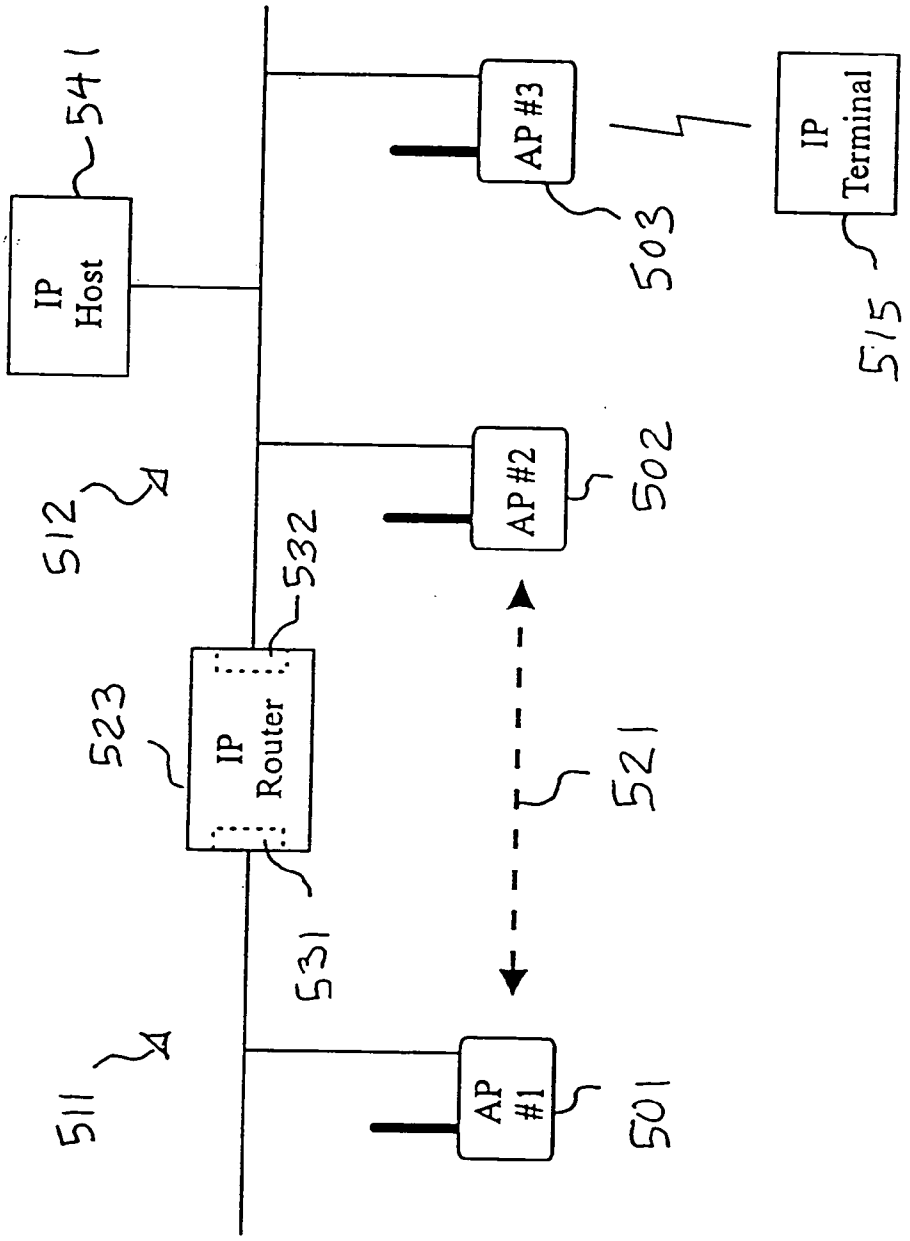

IP Host
541
AP #3
503
IP Terminal
515
512
AP #2
502
532
523
IP Router
521
531
511
AP #1
501

Fig. 5

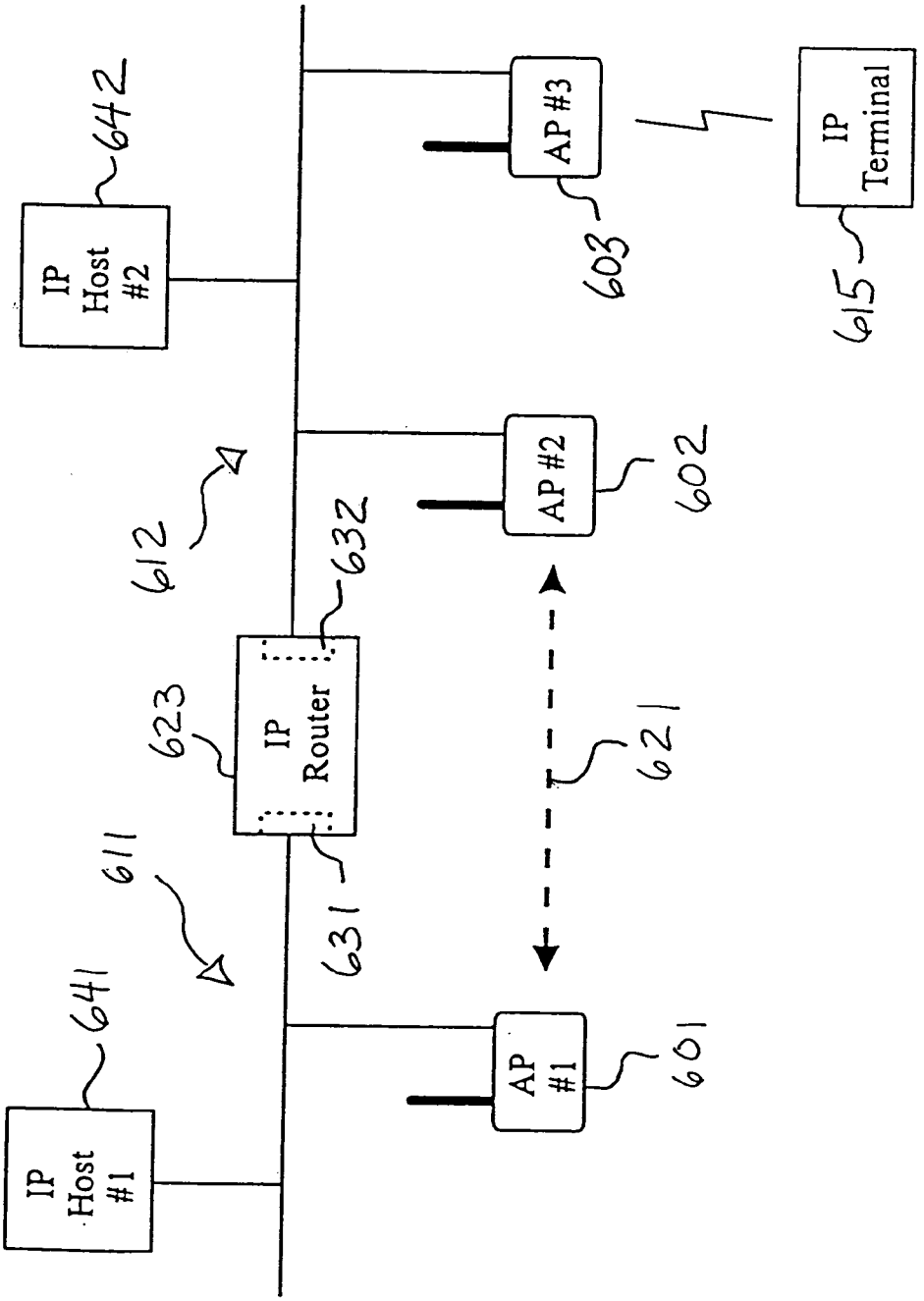

IP Host #2
642
AP #3
603
IP Terminal
615
612
632
AP #2
602
623
IP Router
621
611
631
641
AP #1
601
IP Host #1

Fig. 6

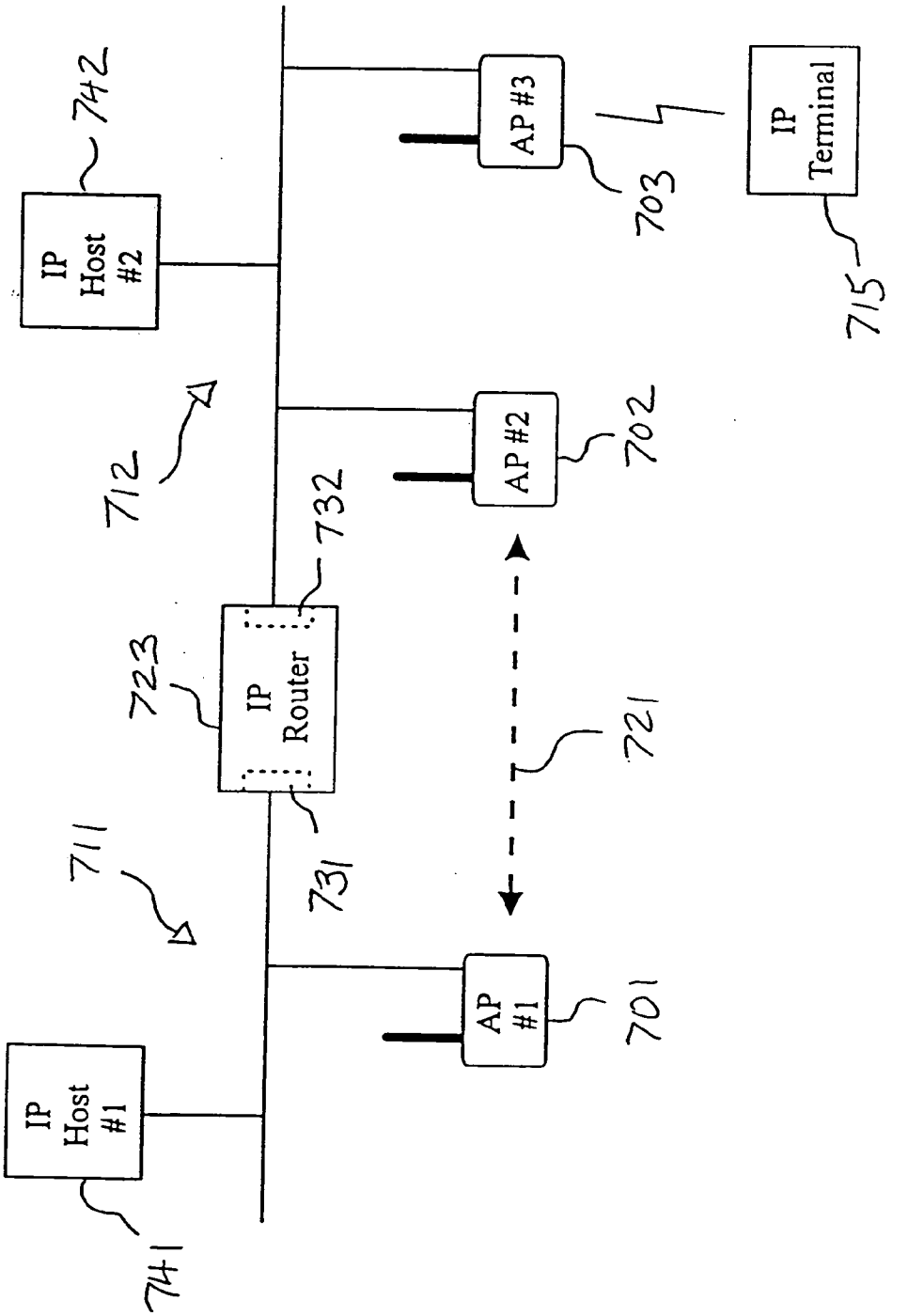

IP Host #1
741
711
IP Host #2
742
712
IP Router
723
731
732
AP #1
701
AP #2
702
AP #3
703
721
IP Terminal
715

Fig. 7

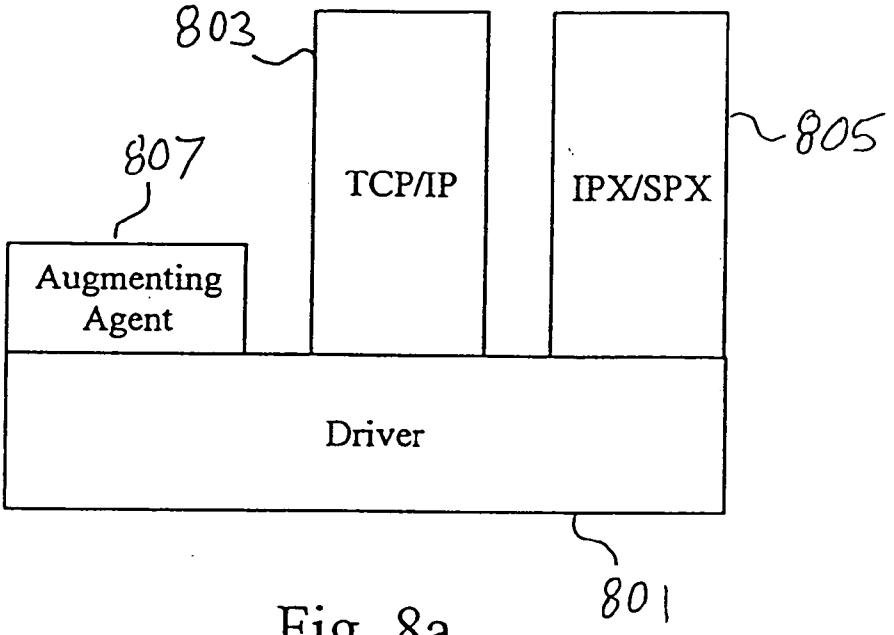

803
807
TCP/IP
IPX/SPX
805
Augmenting Agent
Driver
801

Fig. 8a

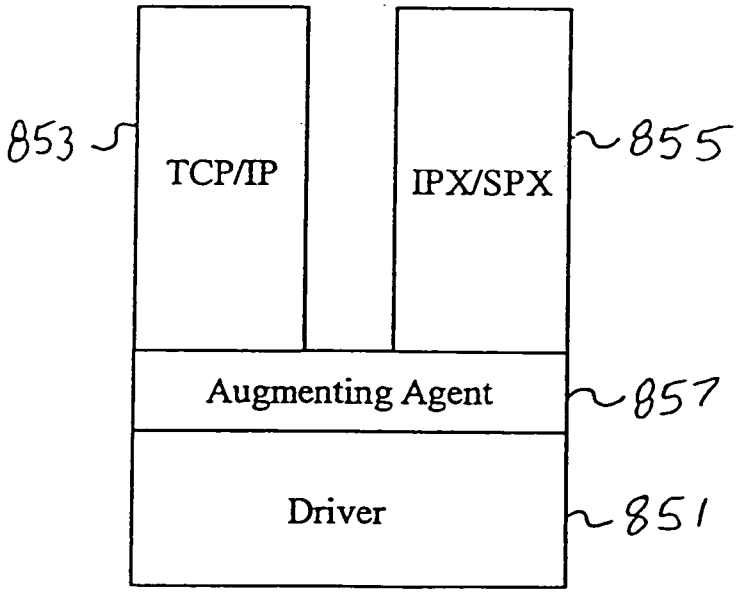

853
TCP/IP
IPX/SPX
855
Augmenting Agent
857
Driver
851

Fig. 8b

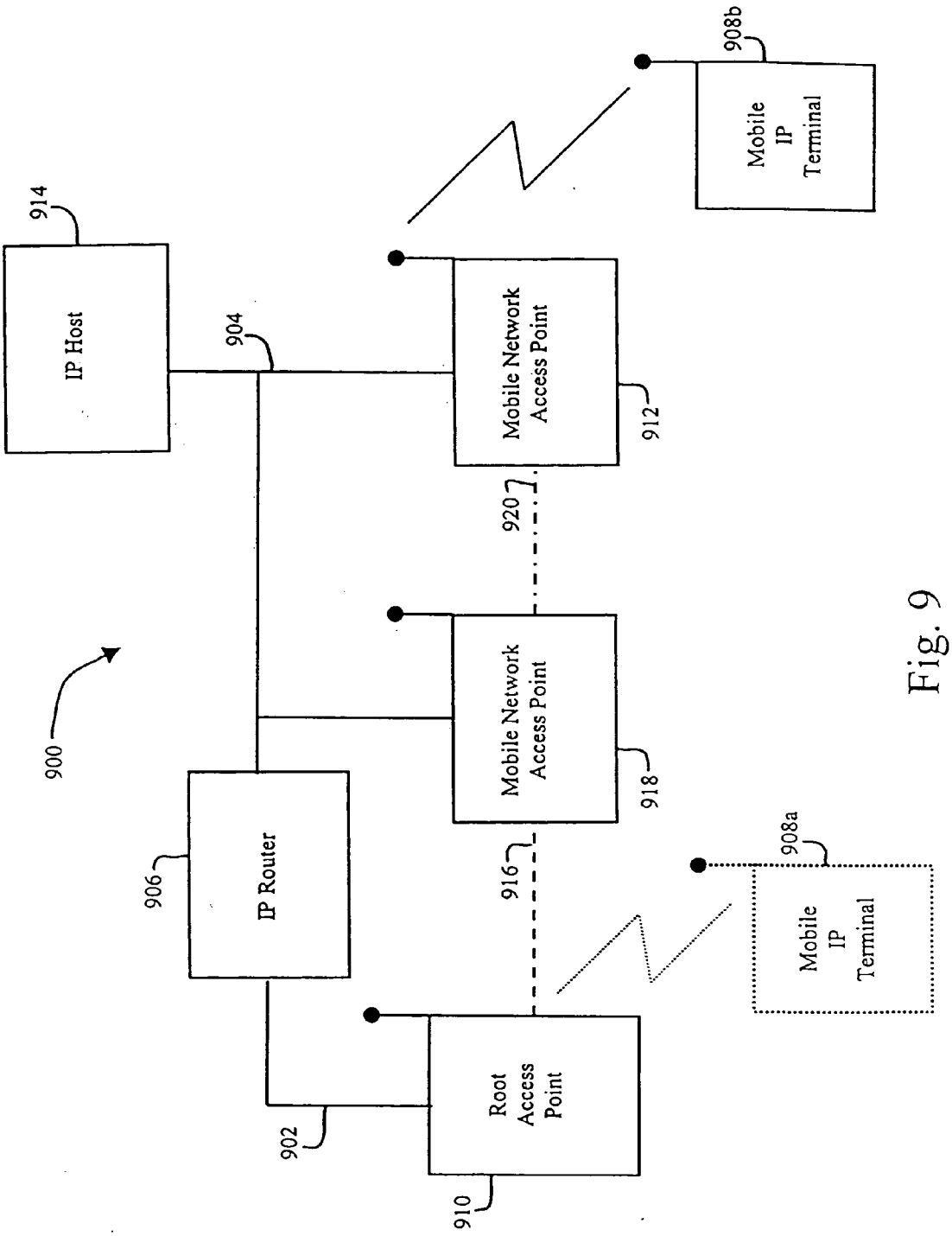

900
IP Host
914
IP Router
906
Root Access Point
902
910
Mobile Network Access Point
904
912
Mobile Network Access Point
918
920
916
Mobile IP Terminal
908b
Mobile IP Terminal
908a

Fig. 9

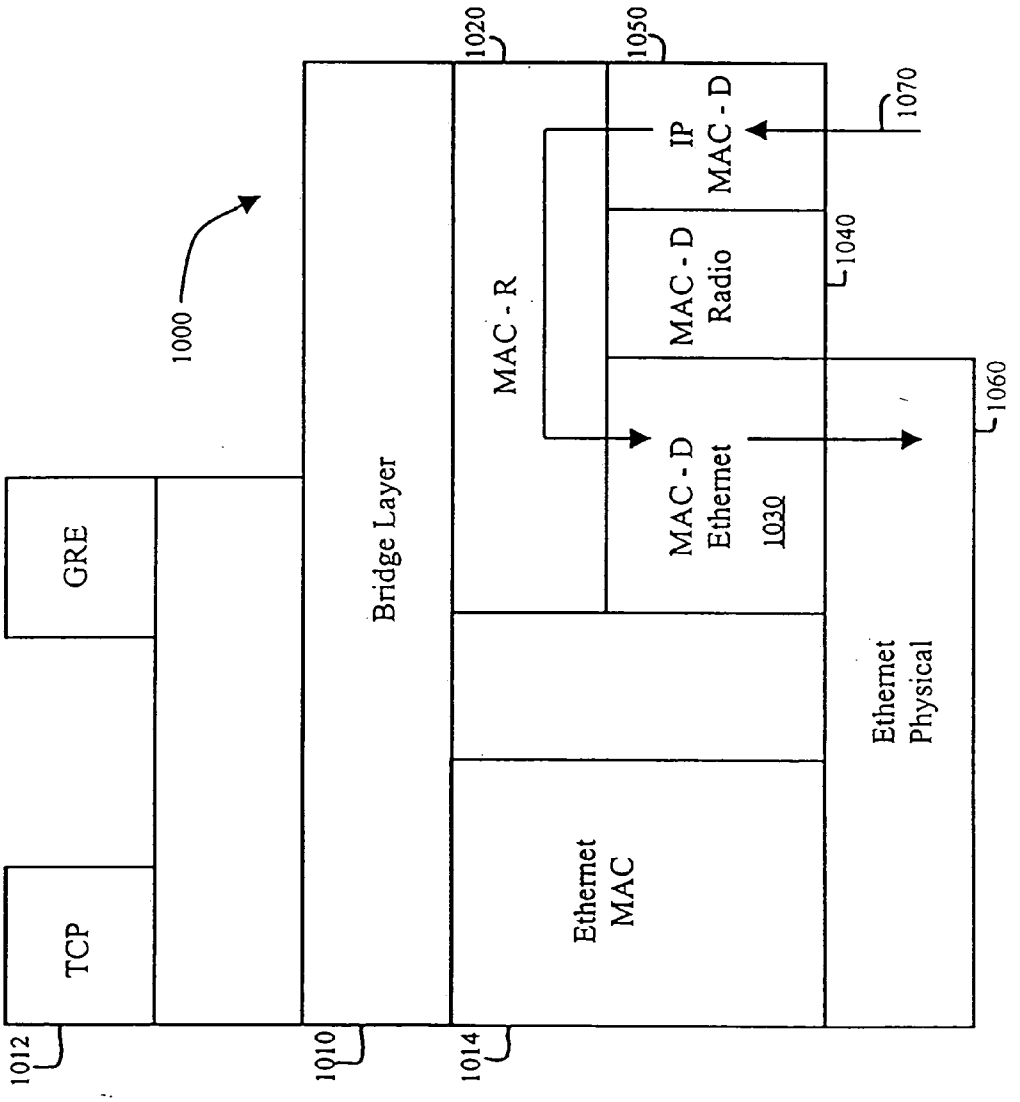

1000
1020
1050
1070
1040
1060
1012
1010
1014
GRE
TCP
Bridge Layer
MAC - R
IP
MAC - D
MAC - D
Radio
MAC - D
Ethernet
1030
Ethernet
MAC
Ethernet
Physical

Fig. 10

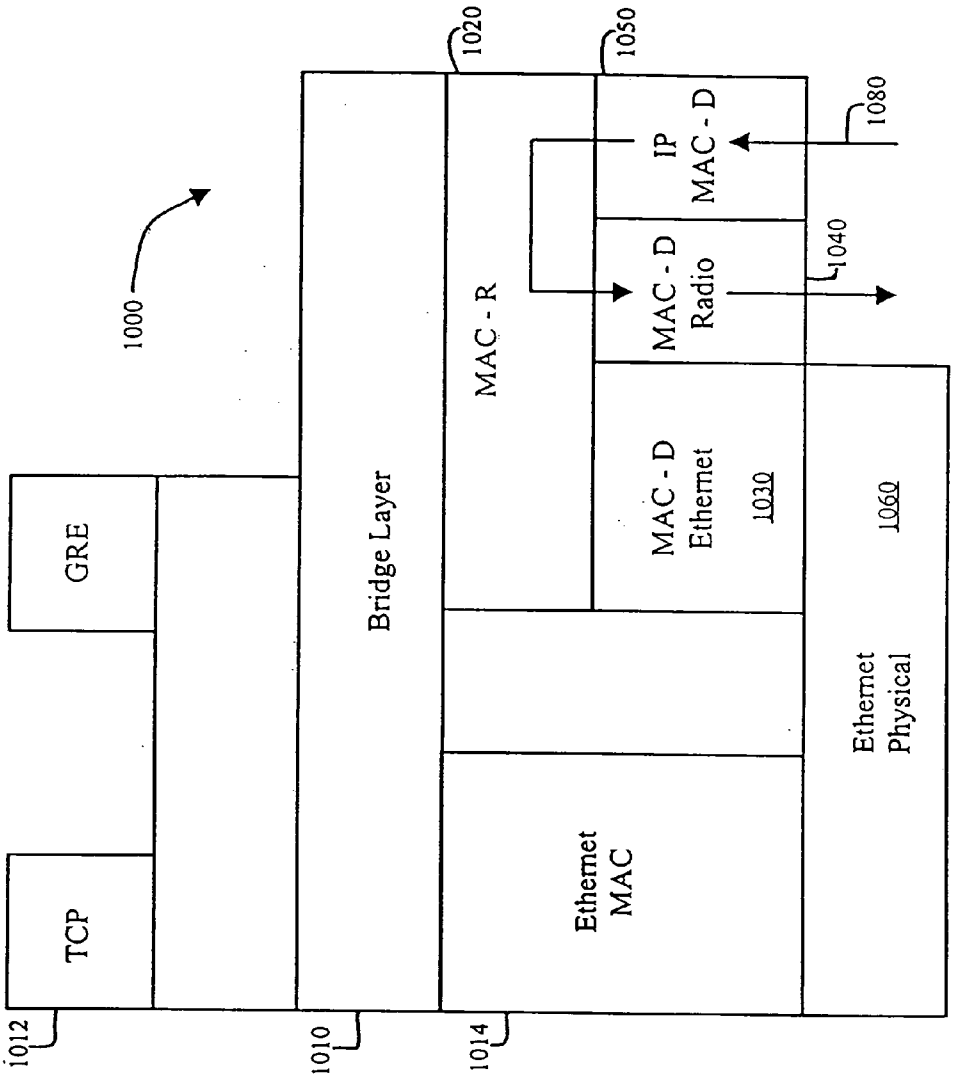

1000
1020
1050
1080
1040
GRE
TCP
Bridge Layer
MAC - R
IP
MAC - D
MAC - D
Radio
MAC - D
Ethernet
1030
Ethernet
MAC
Ethernet
Physical
1060
1012
1010
1014

Fig. 11

ENHANCED MOBILITY AND ADDRESS RESOLUTION IN A WIRELESS PREMISES BASED NETWORK

CROSS-REFERENCES To RELATED APPLICATIONS

[0001] The present case is a continuation of U.S. application Ser. No. 09/183,767 filed Oct. 30, 1998, (which is to issue as U.S. Pat. No. 6,701,361 on Mar. 2, 2004), which is a continuation-in-part of U.S. application Ser. No. 08/916,601 filed Aug. 22, 1997, which claims the benefit of U.S. Provisional Application No. 60/024,648 filed Aug. 22, 1996, and U.S. Provisional Application No. 60/043,395 filed Apr. 2, 1997, all of which are hereby incorporated herein by reference in their entirety.

BACKGROUND

[0002] 1. Technical Field

[0003] The present invention relates generally to premises based wireless networks wherein wireless terminals roam between network segments and utilize address resolution techniques for data packet routing purposes; and, more particularly, it relates to techniques for enhancing the mobility of such wireless terminals within the wireless networks while minimizing wireless traffic for address resolution.

[0004] 2. Related Art

[0005] Communication systems often include interconnected wired and wireless networks that together support communication within an enterprise. These communication systems typically include one or more wired networks that connect network elements such as workstations, servers and access points. Communication cells established by wireless access points (APs) provide links between network elements connected to the wired backbone and mobile terminals. Such communications often pass through both the wireless and wired networks.

[0006] Wired networks typically operate according to one or more communication protocols, or protocol stacks that were specifically designed with strategies to maintain and manage wired networks. Similarly, wireless networks have evolved with protocols and associated maintenance strategies to support mobile network nodes and other unique characteristics associated with wireless network. Thus, it is often difficult to merge wired and wireless networks together without degrading performance on either the wired or wireless network.

[0007] For example, in conventional installations, APs are used to bridge between the wired and wireless networks. However, higher level protocols operating in the wired networks often create problems for the wireless networks, especially in those wireless networks where terminals frequently roam. Specifically, when terminals that communicate with a first AP on one IP (internet protocol) segment of a wired LAN (local area network) roam to communicate with a second AP attached to a second IP segment of the wired LAN, ongoing communication may be lost due to the a need to reregister the roaming device on the second IP segment and unregister that device from the first IP segment. Thus, IP nodes cannot transparently roam to another IP subnet. Further, because the APs in different IP segments often reside adjacent one another, the roaming terminals frequently move back and forth between the cells, creating significant problems in the network.

SUMMARY OF THE INVENTION

[0008] In order to overcome the shortcomings described above and additional shortcomings, a wireless network according to the present invention includes a multi-segment wired network and a plurality of wireless access points connected to the wired network. The wired network operates according to a wired network protocol which may be the Internet Protocol. Wireless terminals communicate with the wireless access points according to the wired network protocol, inconsistent with the wireless network protocol. Each of the wireless terminals has a wired network address corresponding to one of the wireless access points. As the wireless terminals roam throughout the premises, protocol tunnels route communications between wireless terminals via the wired network, thereby preserving communications while roaming by allowing the wireless terminals to retain their wired network addresses during the ongoing communications. Such protocol tunnels are transparent to the wired network.

[0009] Additional functionality is provided through the use of data link tunnels that connect access points within a wireless network. The data link tunnels allow passage of data under the wired network protocol to wireless terminals operating under the wired network without extraneous overhead in a communications protocol.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] **FIG. 1** is a drawing of an exemplary enterprise network built in accordance with the present invention utilizing tunneling to accommodate migration between IP network segments.

[0011] **FIG. 2** is a drawing providing an exemplary illustration of access point interaction via an IP router to carry out IP tunneling in accordance with the present invention.

[0012] **FIG. 3** is a drawing of an exemplary protocol stack used in an access point of the present invention such as one of those shown in **FIGS. 1 and 2** which has an IP port.

[0013] **FIG. 4** is a drawing illustrating the operation of the present invention with a roaming IP terminal in an enterprise network built in accordance with the present invention.

[0014] **FIG. 5** is a drawing illustrating a variation from that of **FIG. 4** used to illustrate further aspects in the enterprise network built in accordance with the present invention relating to roaming.

[0015] **FIG. 6** is a drawing of an exemplary enterprise network used to illustrate the functionality of address resolution using ARP proxy servers in accordance with the present invention.

[0016] **FIG. 7** is a drawing of an exemplary enterprise network used to illustrate the functionality of address resolution using ARP translation servers in accordance with the present invention.

[0017] **FIG. 8*a*** is a drawing illustrating operation of an augmenting agent built in accordance with the present invention which supplements off-the-shelf protocol stacks to

support various enhanced features that may prove desirable in specific enterprise network configurations.

[0018] **FIG. 8***b* is a drawing illustrating an alternate implementation of the augmenting agent of **FIG. 8***a* wherein, instead of operation as an independent, monitoring application, the augmenting agent operates as a shim between the proprietary or defacto industry standard drivers and the higher level protocols.

[0019] **FIG. 9** is a block diagram of a communication system illustrating the use of an IP tunnel and a data link tunnel to access a roaming terminal in accordance with the invention.

[0020] **FIG. 10** is a drawing illustrating a protocol stack associated with the access point at the endpoints of the IP tunnel and the data link tunnel in **FIG. 9**.

[0021] **FIG. 11** is another drawing illustrating a protocol stack associated with the access point at the endpoints of the IP tunnel and the data link tunnel in **FIG. 9**.

DETAILED DESCRIPTION

[0022] **FIG. 1** is a drawing of an exemplary enterprise network **100** built in accordance with the present invention utilizing tunneling to accommodate migration between IP network segments. An enterprise as used herein refers to a business operation which may be self contained within a single premises or within multiple premises. For example, the enterprise network may be a wired and wireless network used within a single warehouse to support inventory control. It may also include support for mobile, vehicle based communication with such warehouse via a wide area network ("WAN"). Likewise, the enterprise might also include a second warehouse or manufacturing facility located near or remote to the warehouse with wired, satellite or WAN connectivity.

[0023] In particular, within the enterprise network **100** of **FIG. 1**, the protocols of the present invention, hereinafter referred to as OWL (open wireless local area network) protocols, support a variety of features which enhance mobile or portable terminal mobility while minimizing transmissions within the wireless networks. The OWL protocols function at the MAC (media access control) sub layer of the ISO (industry standards organization) protocol stack and allow the mobile network nodes (e.g., wireless terminals, printers, code readers, etc.) to roam from one wireless access point (OWL AP) to another in a manner which is transparent to higher layer protocols. The features of the present invention may be viewed as extensions to wireless network architectures such as those found in Appendix A entitled "OWL Network Architecture", Appendix B entitled "Open Wireless LAN Theory of Operation," Appendix C entitled "OWL Network Frame Formats," and Appendix D entitled "UHF/Direct Sequence MAC-D Protocol Specification."

[0024] The protocols of the present invention enable mobility across IP subnets for both IP and non-IP nodes, and enables non-IP nodes, on two or more IP subnets, to communicate as if connected by a single (possibly bridged) local area network. These protocols do not require any changes to an existing TCP/IP protocol stack in IP routers or mobile IP stations.

[0025] Without the protocols of the present invention an AP (access point) **101** and an AP **102** cannot belong to the same OWL network unless an IP router **103** is configured to bridge OWL frames (i.e. DIX type hex. 875C). Assume that an IP terminal **104** attached to the AP **101** is communicating with an IP host **105**. The IP host **105** and the IP terminal **104** each have IP addresses for a subnet **106**. If the IP terminal **104** attaches to the AP **102** (i.e. with a different LAN ID), then the IP host **105** cannot send packets to the IP terminal **104** because the IP router **103** would not forward packets within the subnet **106** to a subnet **107**. A non-IP terminal **108** on the subnet **106** cannot communicate with a non-IP host **109** on subnet **107** unless the IP router **103** is configured to forward non-IP packets. However, with the protocols of the present invention, such and other problems are overcome.

[0026] **FIG. 2** is a drawing providing an exemplary illustration of access point interaction via an IP router to carry out IP tunneling. Features of the protocols of the present invention may be implemented by adding a logical port to an OWL access point (AP) which is, essentially, a port to an "IP tunnel". OWL packets and layer **2** data frames which are sent on the logical "IP port" are encapsulated inside of IP packets and sent through the tunnel. An IP tunnel exists between the IP port on an AP which "originates" the tunnel and an IP port on an AP which attaches to the OWL spanning tree through the "remote" end of the tunnel. The IP tunnel functions as a branch in the OWL spanning tree.

[0027] The user configures the IP tunnel port (i.e. with the bridge port menu) on an OWL AP. By default, the IP port is enabled so that an AP can attach to an OWL network through the remote end of an IP tunnel; the user can explicitly disable the IP port to prevent the AP from attaching through the tunnel. If the IP port is enabled, then the user can configure the port so that the AP will originate an IP tunnel. Typically only a small number of APs should be configured to originate an IP tunnel. If an IP port is configured to originate a tunnel, then a list of 1 or more IP addresses must be defined for the port. A type is entered for each address in the list. The type can be UNICAST, BROADCAST, or MULTICAST. The AP software places no restrictions on addresses in the list (other than the size of the list). The address list is selected so that IP packets destined to addresses in the list will be heard by APs which should attach to the OWL network through an IP tunnel. For example, in **FIG. 1**, an IP tunnel can be established between the AP **101** and the AP **102** by enabling the AP **101** to originate an IP tunnel and adding the IP address of the AP **102** to the address list associated with the IP port in the AP **101**. The AP **101** and AP **102** are configured with the same OWL LAN ID.

[0028] An IP port can be configured so that it can only originate a tunnel if it assumes the root node status or if it becomes the designated AP for a secondary LAN.

[0029] A set of permanent filters and a set of user-defined filters are used to restrict flooding through an IP tunnel. The filters can be used, for example, to limit traffic through an IP tunnel to OWL frames and Norand Network Layer (NNL) frames. The permanent filters are used to prevent IP routing information packets and broadcast/multicast IP packets from passing through an IP tunnel. By default, only NNL packets, OWL packets, ARP packets, and unicast IP packets with a protocol type of UDP, TCP, or ICMP can pass through an IP tunnel. Some ICMP types and UDP/TCP protocol ports are

also filtered, by default, to prevent IP routing information from passing through the tunnel. A "subnet filter" can be enabled if all mobile IP nodes belong to the same "root" subnet. Filters are discussed in more detail below.

[0030] The user can enable/disable a "proxy ARP server" or an "ARP translation server" (discussed below) and, optionally, create permanent ARP server entries. The user can also set a network wide parameter which prevents broadcast ARP requests from being forwarded to radio terminals and through IP tunnels. The parameter can be set so that no ARP requests are forwarded or only those which cannot be "resolved" by the particular ARP server.

[0031] Although the higher level protocols (e.g., such as that set forth in IEEE 802 standards) may prohibit a bridge from reordering (i.e. forwarded) frames, it is possible that frames forwarded through an IP tunnel may be reordered by the underlying network. The user can configure an IP port so that strict frame sequencing is enforced. If strict frame sequencing is enabled, then the IP port will insert a sequence number in outbound frames and cache address/sequence number pairs for inbound frames. Delayed frames which arrive out-of-order are simply discarded.

[0032] An IP port can be enabled on an AP configured with an IP address. If IP subnet addressing is used, then the AP should also be configured with an IP subnet mask.

[0033] An OWL IP tunnel is logically equivalent to any other physical link (i.e. radio link) in the OWL spanning tree. An OWL AP forwards a packet along a branch in the spanning tree by sending the packet to the MAC-D destination address of the next hop. The MAC-D addresses used on an IP port are IP addresses which identify the AP at each end of the tunnel. Note that the TCP/IP software in an AP is responsible for binding the IP address to the correct **802** LAN address (i.e. with ARP).

[0034] The root node and other attached OWL APs broadcast HELLO packets or "beacons" on each IP port and radio port once per HELLO period. The root node and designated APs also broadcast HELLO packets on ethernet links. If the port is an IP port, then a copy of the HELLO packet is created for each IP address in the user-defined list for the port. The MAC-D destination address, of the HELLO packet, is an IP address from the list, and the MAC-D source address is the IP address of the AP. If the destination IP MAC-D address in a HELLO packet is a multicast address, then the HELLO packet may be received by more than one AP. For example, an IP port on the root AP can be configured with the "all-subnets" address. In this case, no other configuration may be required, since all APs in an enterprise IP network, potentially, can receive HELLO packets addressed to the all-subnets address. (Note that IP routers must be enabled to forward packets addressed to the all-subnets address or a group address, if such an address is used.) As a second example, an IP port on the root AP can be configured with a list of unicast addresses, to limit HELLO propagation and/or to explicitly control which APs attach to the remote end of a tunnel.

[0035] The IP software in the AP binds the destination IP address in a HELLO packet to an ethernet address. If the IP address type is UNICAST, then the first hop on the path to the IP destination is derived from the IP route table in the AP. Note that the user can configure a default route and can also configure special routes for a specific IP address or group of addresses. If the type is BROADCAST, then the destination ethernet address is the ethernet broadcast address, hexadecimal FFFFFFFFFFFF. If the type is MULTICAST, then the HELLO packet is sent to a multicast ethernet destination address which is formed from the IP address according to RFC 1112. The first 3 bytes of the ethernet address are hex. E and the last 23 bits are taken from the last 23 bits of the IP address.

[0036] OWL APs which are on an IP subnet which is different than the IP subnet of the OWL root node, can attach to the OWL spanning tree through an OWL IP port. The "cost" associated with an IP port is greater than the cost of an ethernet port, but less than the cost of a radio port. An unattached AP may receive HELLO packets on one or more ports. If the lowest cost path to the root node is through an IP port, then an AP will send an ATTACH request to the root node through the IP port. The MAC-D destination address of the ATTACH request is equal to IP address of the tunnel originator and the MAC-D source address is the IP address of the attaching AP. Note that the IP destination address is obtained from the MAC-D source address of a HELLO packet. The tunnel link is complete as soon as the attaching AP receives an ATTACH response on the IP tunnel port.

[0037] An AP which attaches through an IP tunnel link (or OWL radio link) can be the designated AP for a secondary OWL ethernet LAN. An AP can be the designated AP for a secondary LAN at a given time. More than one AP, attached to the same secondary LAN segment, may receive HELLO packets through an IP port (or radio port) if a multicast IP address is used or if two or more unicast addresses are defined (i.e. for redundancy). The protocol to elect the designated AP operates consistently whether the path to the parent AP is through an IP tunnel or radio link. The designated AP, for a secondary LAN, is always the parent of any other AP which can bridge frames to the secondary LAN segment.

[0038] More particularly, in **FIG. 2**, a subnet **201** is the OWL primary LAN. Further a subnet **202** is an OWL secondary LAN, and an AP **212** is the designated bridge for the secondary LAN **202**. OWL spanning tree branches **221**, **222** and **223** are denoted by dashed lines. The branch **222** from AP **212** to a root AP **215** is through an IP tunnel via an IP router **205**, which was originated by the root AP **215**. By default, an AP **213** can bridge frames onto subnet **202**. Therefore, the AP **213** must attach to the OWL network through the designated AP for the subnet **202**, i.e., the AP **212**. An AP **214** is attached to the root AP **215** through an ethernet branch **221**, rather than an IP tunnel branch, because the cost of an ethernet hop is lower. Similarly, ethernet branch **223** exists between the AP **212** and the AP **213**.

[0039] A node in an OWL network is identified by its MAC-R address which is a 6-byte **802** (i.e. ethernet) address. A port on an OWL device is identified by a MAC-D address. The path to an OWL node is defined by the OWL spanning tree, which can be derived from routing tables stored in APs. The key to a routing table entry is a MAC-R **802** address. An AP forwards an outbound ethernet frame, for example, by looking up the destination ethernet address in a routing table. A MAC-D port address and local port ID, stored in the route table entry for the destination, define the first hop on the path to the destination. If the first hop is

through an IP tunnel, then the MAC-D address is an IP address which identifies an IP port at the remote end of the tunnel. The IP MAC-D layer encapsulates the frame inside of an IP packet and forwards it to the remote IP port. The IP MAC-D layer in AP at the remote end of the tunnel removes the IP encapsulation and posts the frame to the MAC-R layer, which forwards the frame to its final destination.

[0040] The size of an encapsulated frame may exceed the maximum frame size for an ethernet link. The IP software in the AP is responsible for fragmenting and re-assembling packets which exceed the maximum ethernet frame size.

[0041] The MAC-D entity associated with an IP port on an AP passes a frame to the local IP stack for transmission. The IP stack formats the IP packets, binds the destination IP address to an ethernet address, and passes the frame to its data link layer interface for transmission. In an OWL AP, the data link layer interface for the IP stack exists on top of the OWL bridging layer. Therefore, the IP-encapsulated frame passes through the bridging layer and, possibly, through the MAC-R layer and a second MAC-D layer before it is transmitted on a physical port. The destination ethernet address of the IP-encapsulated frame should be the ethernet address of an IP router port attached to the local subnet. If the destination ethernet address is unknown, then the frame would normally be flooded. However, encapsulated frames, identified by the IP protocol type, are always passed to the ethernet MAC for transmission. Received encapsulated frames are discarded by the bridging layer, if the input source is not the ethernet MAC. This restriction prevents internal routing loops in the AP and prevents tunnels from existing on top of radio links. Note that the path cost would be distorted if an IP tunnel existed over a radio link.

[0042] **FIG. 3** is a drawing of an exemplary protocol stack used in a access point of the present invention such as those shown in **FIGS. 1 and 2** which has an IP port. A dashed line **301** between an IP MAC-D entity **303** and a GRE transport entity **305** logically represents a path through the protocol stack for IP-encapsulated frames. More particularly, this path flows between the GRE transport entity **305** the IP MAC-D entity **303** via an IP layer **307**, a data link layer **309**, a bridge layer **311** and a MAC-R entity **313**. Descriptions regarding other pathways through the protocol stack may be found, for example, in Appendix B.

[0043] If the AP receives a frame and the destination is unknown, the frame may be flooded, depending on the configured flooding levels. Note that the destination of a multicast frame is never known. Frame flooding through an IP tunnel is consistent with flooding on any other link type. If multicast hierarchical flooding is enabled, for example, then multicast frames which originate in the radio network are forwarded inbound to the primary LAN. Multicast frames which originate on the primary LAN are flooded throughout the OWL network. The path to the primary LAN may include an IP tunnel.

[0044] Flooding through an IP tunnel can be reduced with a number of configuration options. As noted above, filters can be defined to prevent some types of frames from being forwarded.

[0045] Ethernet bridging can be disabled on selected OWL APs to prevent flooding across subnet boundaries. In **FIG. 2**, for example, if bridging is disabled on AP 2 and AP 3, then frames transmitted on subnet 2 will not be bridged into the OWL network, and, therefore, will not be flooded to subnet 1. Only frames received on radio ports will be forwarded inbound by AP 2 and AP **3**.

[0046] If unicast hierarchical flooding (see OWL theory of operation) is enabled, then unicast frames transmitted on subnet 1, the primary LAN, will not be flooded to subnet 2, if the destination is unknown; however, unicast frames will be forwarded from subnet 1 to subnet 2 if the root AP has a route table entry for the destination and the first hop is through the IP tunnel link.

[0047] An AP will not forward a frame through an IP tunnel if the destination ethernet address identifies the default IP router port. An AP can determine the ethernet address of its default IP router port from its local ARP cache.

[0048] As used herein, a "mobile IP node" is any IP node that can roam across IP subnet boundaries. In an OWL network, each mobile IP node is configured with a single IP address, which defines its "home" IP subnet. In theory, any IP subnet(s) can be a home subnet for mobile nodes. In practice, the IP subnet which is attached to the OWL root node is the preferred home subnet for mobile IP nodes. In this case, the home subnet is equivalent to the OWL primary LAN. If the primary LAN is the same as the home subnet and mobile nodes communicate exclusively with stations on the primary LAN, then MAC-level flooding and triangular routing can be reduced or eliminated.

[0049] In an IP/ethernet network which uses subnet routing, a first IP node sends an IP packet to a second node on the same subnet by sending the IP packet to the ethernet address of the second node. If the second node is on another subnet, the first node sends the packet to the ethernet address of an IP router. The ethernet address is typically discovered with the ARP protocol. Since the destination MAC address of the IP packet is an ethernet address, the packet will be forwarded correctly in an OWL network.

[0050] If a mobile IP node (or mobile non-IP node) roams away from its home subnet and attaches to an AP on a "foreign" subnet, it must send an ATTACH request to the OWL root node before it can send or receive data frames. The ATTACH request fully establishes the path to the mobile node. For example, the AP at the home end of the IP tunnel, which links the home and foreign subnets, will create a route entry for the mobile node, which points to the tunnel as the first hop on the path to mobile node, when it receives the ATTACH request from the terminal. The key to the route entry is the ethernet address of the mobile node. If the AP receives an ethernet packet, with the destination ethernet address of the mobile node, then it will forward the encapsulated ethernet frame through the IP tunnel.

[0051] If a mobile IP node is attached to an AP on a foreign subnet, then it still responds to ARP requests which are transmitted on its home subnet. If multicast flooding is enabled, then broadcast ARP requests are flooded throughout the OWL network, including through OWL tunnel links. Therefore, the mobile node can receive the broadcast ARP request on the foreign subnet, and respond with a unicast ARP response, containing its ethernet address. Likewise, an ARP request from the mobile node will be flooded to the home subnet. Note that the target IP address, in an ARP request from the terminal, may designate either a target host

or a router port on the node's home subnet. In either case, IP packets are forwarded through the OWL network to the node identified by the destination ethernet address.

[0052] **FIG. 4** is a drawing illustrating the operation of the present invention with a roaming IP terminal in an enterprise network built in accordance with the present invention. As shown, a mobile IP terminal **415** has roamed from its home subnet, subnet **411**, to an AP **403** on a subnet **412**. The mobile IP terminal **401** may be any device which contains a radio transceiver such as a portable computing device, a code reader, a printer, digital camera, RF TAG, etc. An AP **401** serves as the OWL root node. An AP **402** is the designated AP for the secondary LAN which is the subnet **412**. The AP **402** is attached to the AP **401** through an IP tunnel **421**. The AP **403** is attached to the AP **402** through an ethernet link **425**. Note that the physical path for the IP tunnel **421** between the AP **401** and the AP **402** is through an IP router **423**. The IP router **423** has two ports, port **431** attaches to the subnet **411** while port **432** attaches to the subnet **412**. The IP address for port **431** identifies subnet **411**, while the IP address for port **432** identifies the subnet **412**. The subnet **411** is the primary OWL LAN.

[0053] As a first example, assume that the terminal **415** has been actively communicating with an IP host **441** when it roams from the AP **401** to the AP **403**. When the terminal **415** roams, it must send an ATTACH request to the root, and wait for a matching ATTACH response, before it can send or receive data frames. The ATTACH request causes the root to update its route table entry for the terminal so that the first hop port and MAC-D address are its IP port and the IP address of the AP **402**, respectively. The AP **402** and the AP **403** also update their routing tables to reflect the new path. If the host **441** sends a packet to the terminal **415**, the destination ethernet address is the ethernet address of the terminal **415**. The packet will be routed to the terminal **415** via the tunnel **421**. If the terminal **415** sends a packet to the host **441**, the destination ethernet address will be the address of the host **441**. The packet will be forwarded inbound until it reaches the primary LAN (the subnet **411**), where it will be bridged and received by the host **441**.

[0054] If the terminal **415** roams before it begins communicating with the host **441**, it does not know the ethernet address of the host **441**. Thus, the terminal **415** sends a broadcast ARP request which contains the IP address of the host **441** to determine the ethernet address of the host **441**. The AP **403** bridges the ARP request onto the subnet **412**. No IP node on the subnet **412** will respond to the ARP request because the target IP address does match any of the subnet **412** IP addresses. The AP **402** receives and forwards the ARP request inbound through the IP tunnel **421** to the AP **401**. The AP **401** bridges the request onto the subnet **411**, where it is received by the host **441**. The ARP response is sent to the unicast address of the terminal **415**. If the host **441** sends an ARP **441** request which contains the IP address of the terminal **415**, then the ARP request can either be serviced by a proxy ARP server (i.e. in the AP **401**) or flooded outbound through the IP tunnel **421** and to the terminal **415**.

[0055] **FIG. 5** is a drawing illustrating a variation from that of **FIG. 4** used to illustrate further aspects in the enterprise network built in accordance with the present invention relating to roaming. The home subnet of an IP terminal **515** is a subnet **511**. An IP router **523** has a port **531** which is the default router port associated with the subnet **511** and a port **532** associated with the subnet **512**. The port **531** is the default router port for the terminal **515**; and the port **532** is the default router port for an IP host **541**.

[0056] Assume the terminal **515** was actively communicating with the host **541** when it roamed from an AP **501** to an AP **503**. The host **541** is sending IP packets to the terminal **515** which have a destination ethernet address for the port **532** on the IP router **523**. The terminal **515** is sending IP packets to the host **541** which contain the ethernet address of port **531** on the router **523**. After the terminal **515** roams, it will continue to send packets with the ethernet address of the port **531**. A packet from the terminal **515** will be bridged onto the subnet **512** by the AP **503**. An AP **502** will receive and forward the packet inbound to the primary LAN. The AP **501** bridges the packet onto subnet **511**, where it will be received by the router **523** on the port **531**. The router **523** will forward the IP packet to the host **541** on subnet **512**. A packet transmitted by the host **541** will be forwarded from the subnet **512** to the subnet **511** by the router **523**. The AP **502** will not forward the packet, transmitted by the host **541**, inbound to the AP **501** if it has learned that the port **532** on the router **523** is on the subnet **512**. Otherwise, it will flood the (i.e. duplicate packet) packet to the subnet **511**. Note that no ethernet adapter on the subnet **511** should receive the duplicate packet.

[0057] As before, ARP requests will be generated if the terminal **515** roams before communicating with the host **541** (or if ARP caches are aged). The terminal **515** will send an ARP request with the IP address of the port **531** as the target IP address. The ARP request will be forwarded inbound through the IP tunnel **521** and bridged onto subnet **511** by the AP **501**, where it will be received by the router **523**. The router **523** will send a unicast ARP response packet to the terminal **515** which contains the ethernet address of the port **531**. The host **541** will send an ARP request with the IP address of the port **532** as the target IP address. The router **523** will send a unicast ARP response packet to the host **541** which contains the ethernet address of the port **532**. Note that the router **523** will receive both ARP requests on both ports; however, it will (correctly) respond only to those ARP requests which match the port IP address. Also note that the AP **502** will learn that the ethernet address of the port **532** is on the local subnet when it sends an ARP response.

[0058] The OWL/IP protocols run on top of an IP "transport-layer" protocol defined in RFC 1701 entitled "Generic Routing Encapsulation (GRE) protocol." The IP protocol type for GRE is decimal **47**. GRE is used to encapsulate a variety of non-IP network layer protocols (i.e. to route non-IP packets through an IP network). The GRE header is contained in 4 or more bytes. Two of the bytes contained in the GRE header contain the DIX type of the encapsulated protocol, which is hexadecimal 875C for OWL/IP. The general format of an OWL/IP frame transmitted as a DIX ethernet frame is shown below:

Field	Size
Ethernet Destination Address	6 bytes
Ethernet Source Address	6 bytes
Ethernet Version 2 Type (hex. 800)	2 bytes

-continued

Field	Size
IP Header (protocol = 47)	20 bytes
GRE Flags	2 bytes
GRE Type (hex. 875 c)	2 bytes
MAC-D Protocol ID	1 byte
MAC-D Control	1 byte
MAC-D OWL LAN ID	1 byte
MAC-D Fragment ID	1 byte
MAC-D Length	2 bytes
MAC-R Control	2 bytes
MAC-R 802 Destination Address	6 bytes
MAC-R 802 Source Address	6 bytes
MAC-R Parameters	M bytes
802.3 Length or DIX Type	2 bytes
LLC Header/Data	N bytes

[0059] The first two bytes in the GRE header contain a flag which indicates if the GRE header contains an optional 4-byte sequence number. The sequence number can, optionally, be included if strict frame sequencing, through an IP tunnel, must be enforced.

[0060] Filters may be used to prevent unwanted frame forwarding through an OWL/IP tunnel. For example, such filters may operate to prevent forwarding of: (1) 802.1d bridge PDUs any OWL AP port; (2) IP packets with a broadcast or multicast ethernet address (preventing nodes on a remote IP subnet from receiving "bridged" IP packets, for example); (3) IP packets with protocol types such as EGP, IGP, IDPR, IDRP, MHRP, DGP, IGRP, and OSPFIGP; (4) IP ICMP packets except types such as Echo Request, Echo Reply, Destination Unreachable, Source Quench, Redirect, Alternate Host Address, Time Exceeded, Parameter Problem, Time Stamp, Time Stamp Reply, Address Mask Request, Address Mask Reply, and Trace Route; (ICMP types which include Router Advertisement, Router Selection, Mobile EP types, and IPv6 types may not be forwarded); and (5) IP/UDP or IP/TCP packets with source or destination protocol port numbers such as RIP, RAP, and BGP.

[0061] Similarly, a user can explicitly filter DIX types, however, as a default, only the following DIX types are forwarded: OWL (hex. 875C), NNL (hex. 875B), ARP (hex. 0806), and IP (hex. 0800). Further, IP protocols can also be filtered. But, as a default, the IP protocols ICMP(1), UDP(17), and TCP(6) are not filtered. All such filters may be modified or extended as proves desirable for a given enterprise network installation.

[0062] The user can also enable subnet filtering for IP networks which use subnet routing. Subnet filtering can be used if: a) all mobile nodes belong to the same subnet as the root AP —the "root subnet;" and b) the root AP initiates all IP tunnels. Servers/hosts can be on any subnet. If subnet filtering is enabled, an AP forwards IP packets inbound through an IP tunnel if the source IP address belongs to the remote subnet and the source ethernet address identifies a mobile node in the sub tree rooted at the AP. An AP forwards broadcast ARP packets (with an IP protocol type) inbound through an IP tunnel if the source IP address, in the ARP packet, belongs to the remote subnet and the source ethernet address identifies a mobile node in the sub tree rooted at the AP. This option can be used in conjunction with hierarchical unicast flooding to eliminate unnecessary IP packet forwarding and inbound ARP flooding. If the unicast hierarchical flooding option is used, then IP packets are not forwarded from the root subnet unless the destination is in the subtree below the root subnet. Note that multicast and broadcast IP packets are not forwarded. In addition, a proxy ARP server or an ARP translation server can be used to prevent ARP flooding.

[0063] An OWL AP functions as a transparent MAC layer bridge. A transparent bridge may flood a frame, received on one port, to all other ports, if the destination is unknown. In an OWL network, unicast frames may be flooded through an IP tunnel if flooding is enabled. As noted above, broadcast and multicast IP packets are not forwarded through an IP tunnel. In many cases, flooding through an IP port can be eliminated with the "subnet filter" option and the hierarchical unicast flooding option.

[0064] Occasionally, flooding through an IP tunnel may cause a duplicate IP packet to be delivered to another "remote" subnet. This can happen, for example, if an AP with an active IP port has not yet "learned" the ethernet address of a router port which is on the same "local" subnet as the AP. In this case, an IP packet addressed to the ethernet address of the router port may be flooded through the IP tunnel, by the AP, and also forwarded by the IP router. However, the packet flooded through the tunnel should not be received by any ethernet adapter attached to the remote subnet because the destination ethernet address designates the router port attached to the local subnet. It should be noted that IP does not provide "reliable" network layer services. Packets may be lost, duplicated, delayed, or delivered out-of-order.

[0065] An AP with an IP port may also occasionally flood IP packets to the wrong subnet(s), if the AP has not learned the destination address of a local host. Again, such packets should not be received by any ethernet adapter on the remote subnet(s).

[0066] In general, an AP should not forward a frame through an IP tunnel, if the destination ethernet address of the frame identifies a node on the local subnet. An AP uses "backward learning" to discover which ethernet addresses belong to nodes on the local segment. Learned addresses are stored in a "filtering database." Filtering database entries are aged and discarded if the node associated with an entry is not active for some period of time. An AP will not forward an ethernet frame, if it has learned that the destination is on the segment on which the frame was received. In an IP environment, packets destined for another subnet are always addressed to the ethernet address of a router port on the local subnet. Therefore, such packets are usually not forwarded (i.e. through an IP tunnel) by an AP.

[0067] In practice, IP nodes do not transmit IP packets, without first transmitting an ARP request and receiving an ARP response. ARP caches are typically aged, so ARP requests and responses are generated periodically for active nodes. Also, routers usually broadcast routing information packets periodically. In general, any periodic message will cause any AP on the local subnet to refresh its filtering database. Therefore, each AP on a subnet should have a fresh filtering database entry for each router port or host port attached to the subnet.

[0068] The following rules apply to typical OWL/IP protocol installations: (1) OWL/IP does not bridge across an IP router if the router is configured to bridge OWL frames (i.e. DIX type hex. 875C); (2) OWL/IP does not bridge frames across an IP router, for some network protocol type, if the router is also configured to bridge the network protocol type. For example, NNL frames should not be bridged through an IP tunnel, if any intermediate IP routers are configured to bridge NNL frames. Note that some routers (i.e. brouters) can be configured to bridge any frame type which cannot be routed; (3) OWL/IP should not be used to bridge frames with routable non-IP network layer types (e.g. OWL/IP should not be used to bridge Novell IPX frames in an environment which includes combined IP/IPX routers.); (4) As a rule, OWL/IP can be used to bridge frames with non-routable network layer types, where a "non-routable" type is any type which will not be forwarded by a router (e.g. NNL, for example, is a non-routable type); and (5) An OWL network should not be installed so that two IP subnets are bridged by a radio link. For example, in **FIG. 1**, the spanning tree link between the AP **101** and the AP **102** should not be a radio link. Note that the AP **102** will attach to the AP **101** through its OWL/IP port, even if it has a physical radio link to the AP **101**, because the cost of an IP tunnel hop is lower. In general, a path that can be bridged by single radio hop cannot include more than two IP tunnel hops and should include at least one IP tunnel hop. If IP roaming or NNL communications to a remote NNL host are not required, then each set of OWL nodes contained within an IP subnet should be configured as an independent OWL network with a unique LAN ID.

[0069] In a typical IP/ethernet environment, the ARP protocol is used to bind an ethernet address to an IP address. An ARP request packet, which contains a target IP address, is sent to the ethernet broadcast address. Each IP node on the LAN receives and examines the request. The node designated by the target IP address will return the ARP response packet, which contains its unicast ethernet address. If the target IP node is mobile, then the request must be flooded over a radio link(s) and, possibly, through an IP tunnel to reach the mobile node.

[0070] However, in many enterprise network installations, it may prove undesirable to flood ARP requests over radio links and tunnel links for several reasons. The most obvious reason is that it adds broadcast traffic, which has added overhead on radio links. In addition, in a typical mobile node, the radio module interrupts its host processor when a frame is received with the unicast destination address of the mobile node or a broadcast destination address. If the mobile node contains power-management logic, then the host processor may be "sleeping" when a received frame arrives. If the radio module is enabled to receive broadcast ARP requests, for example, then the host processor will constantly be interrupted and awakened. On a busy IP LAN, the mobile node would almost never sleep. Among other reasons, flooding through a tunnel link also circumvents the ability of routers to contain traffic within LAN segments.

[0071] In some cases, a proxy ARP server can be used to reduce or eliminate the need to flood ARP requests to mobile nodes through an IP tunnel or radio port. (Note that filters can be used to reduce non-ARP broadcast traffic.) The proxy ARP server exists on each AP which can bridge to an ethernet port. If the server is enabled, it maintains an ARP database, where each entry in the database contains a status, an age, and an IP address/ethernet address pair. Each address pair designates an 'P node which is on the server's IP subnet. The status value can be "PROXY", "LOCAL", or "PENDING". If the status is PROXY, then the server is servicing ARP requests for the associated IP node, which is in the OWL sub tree rooted at the AP. If the status is LOCAL, then the server has learned that the target IP node is on the local ethernet link. A PENDING entry is created when an ARP request is received and the server does not have an entry for the target node. The age in an entry is set to 0 when the entry is created or updated, and is incremented once a minute. Entries in the database are indexed by the IP address and by the ethernet address.

[0072] The AP bridging module calls the ARP server each time an ARP request is received, and passes a pointer to the ARP packet. The ARP server returns a value to the bridging module which indicates if the request should be forwarded or discarded. There are two general cases —the request frame can either be received on an "inbound" link or an "outbound" link. A link is inbound if the AP is attached to the link through its root port; otherwise, it is outbound. In the special case of the root AP, the primary LAN is considered an inbound link. If an ARP request is received on an inbound link and the server has a PENDING entry, for the target IP address, then it indicates that the request should be flooded (i.e. outbound); otherwise, it indicates that it should be discarded. If the server does not have an entry, a PENDING entry is created. Note that if the server receives another ARP request with the same target IP address, it will indicate that the request should be forwarded. If an ARP request is received on an outbound link and the server does not have an entry or has a LOCAL, then it indicates that the request should be forwarded inbound only, and a PENDING entry is created. If the server has a PENDING entry, then it indicates that the request should be flooded (i.e. forwarded inbound and, possibly, to other outbound ports). In either case, if the server has a PROXY entry for the target IP address, then the server will transmit a "proxy" ARP response, which contains the ethernet address of the associated IP node, and indicate that the frame should be discarded.

[0073] In an exemplary embodiment, the server follows the rules listed below to maintain its ARP database and forward ARP request packets. Note that the database can contain only one entry per IP address; therefore, before an entry is "created" any existing entry must be deleted. In this discussion, a "route" can be a route table entry or a "secondary" entry in the AP bridge table. If the server indicates that an ARP request should be forwarded, then it is flooded according to ARP and multicast flooding configuration parameters.

[0074] (1) The ARP database is tightly coupled with routing tables in the AP. The ARP database cannot contain a PROXY entry for a node, unless the node is in the spanning tree rooted at the AP. Therefore, a PROXY entry cannot be created unless the AP has a route to the node. A PROXY entry is deleted if the route to a node is deleted.

[0075] (2) The server in the root AP or in the designated AP for a secondary ethernet LAN, cannot create a PROXY entry for a node if the route to the node is "distributed". (A route is "distributed" if the first hop to the node is through an AP on the same ethernet link, which is responsible for bridging frames to/from the ethernet link from/to the node.)

[0076] (3) The ARP database is never updated with an IP address which belongs to another subnet. The ARP server always indicates that an ARP request should be discarded if either the target or source IP address belongs to a subnet which is not the same as the subnet of the AP.

[0077] (4) If the server receives an ARP response packet on a non-ethernet port, it creates a PROXY entry for the target IP node (i.e. the node which generated the response), if the AP has a consistent non-distributed route to the node. If the route is distributed, a LOCAL entry is created.

[0078] (5) If the server receives an ARP request packet on a non-ethernet port, it creates a PROXY entry for the source IP node (i.e. the node which generated the request), if the AP has a consistent non-distributed route to the node. If the route is distributed, a LOCAL entry is created.

[0079] (6) An IP node in the OWL network can explicitly register its IP address with the ARP server each time it sends an OWL ATTACH request packet. An AP creates a PROXY entry for the source node if it is responsible for bridging frames to/from the source node on its ethernet port; otherwise, if the route is distributed, it creates a LOCAL entry. The ethernet address stored in the PROXY entry is the MAC-R source address of the ATTACH request packet. The ARP database is not updated if the ATTACH request is invalid (i.e. out-of-sequence).

[0080] (7) If the server receives an ARP response packet on an ethernet port, it creates a LOCAL entry for the target IP node if it does not have an entry or if it has a LOCAL or PENDING entry. If it has a PROXY entry and the AP is not the root AP, then an ALERT request is sent to the root AP. If the path to the node has changed, the root AP will return an ALERT response to delete the old path fragment.

[0081] (8) If the server receives an ARP request packet on an ethernet port, it creates a LOCAL entry for the source IP node, if it does not have an entry or if it has a LOCAL or PENDING entry. If it has a PROXY entry and the AP is not the root AP, then an ALERT request is sent to the root AP. If the path to the node has changed, the root AP will return an ALERT response to delete the old path fragment.

[0082] (9) LOCAL entries are aged and discarded after 30 minutes. PENDING entries are aged and discarded after 2 minutes. PROXY entries are deleted if the route to the associated node changes.

[0083] **FIG. 6** is a drawing of an exemplary enterprise network used to illustrate the functionality of address resolution using ARP proxy servers in accordance with the present invention. A terminal **615** has an IP address for a subnet **612**. Assume that the terminal **615** has either sent an inbound ARP frame or registered its IP address within an ATTACH request packet. The ARP server in an AP **603** has a PROXY entry for the terminal (assuming the AP **603** has bridging enabled). A server in an AP **602** has a LOCAL entry for the terminal **615** because the route for the terminal **615** is distributed, i.e., the AP **603** is responsible for bridging frames from ethernet to the terminal **615**. A root AP **601** cannot have an entry for the terminal **615** because it is on another subnet **611**. If an IP Host **642** sends a broadcast ARP request frame with the target IP address of the terminal **615**, then the server in the AP **603** will generate an ARP response frame which contains the ethernet address of the terminal **615**. The AP **602** will ignore the request. The path between the AP **602** and the AP **603** could contain an off-the-shelf transparent bridge. If the request is flooded inbound, any AP on the subnet **611** will also ignore the request because the target IP address is on another subnet. An IP Host **641** will initiate a conversation with the terminal **615** by sending an ARP request with a target IP address that designates port **631** on the IP router **623**.

[0084] The proxy ARP server can be configured so that ARP requests are never forwarded outbound from an ethernet segment into the radio network. In this case, the server needs to have perfect knowledge of any IP nodes contained within the sub tree rooted at the AP, so that it can generate proxy ARP responses. Normally, this mode is used if all nodes in the radio network explicitly register their IP addresses.

[0085] By default, a broadcast ARP request packet, or any other broadcast packet, which originates in the radio network is forwarded inbound until it reaches the primary LAN. The multicast flooding level can be set so that broadcast frames are always flooded throughout the OWL network.

[0086] Two or more APs may generate ARP response packets for a single node, if an old path is not successfully deleted when the node roams. In this case, the forwarding database in an off-the-shelf bridge may be updated incorrectly. An equivalent problem in an OWL AP has been corrected by not submitting ARP response frames to the backward learning process. Previously, the backward learning logic in the AP assumed that a frame could not be delayed for more than 5 seconds. If an AP received a frame on the primary LAN, for example, and it had an outbound route for the source address, then it deleted the route, if the route was more than 5 seconds old. This logic fails if an AP continues to generate ARP response frames for a terminal, for some time after the terminal has roamed to another AP. To avoid incorrect updates, the filtering database and route tables in an OWL AP are not updated when a received ARP response indicates that the path to the source node may have changed. Instead, an ALERT request is generated to determine if the node has, in fact, roamed. If an ALERT response indicates that the node has roamed, then the AP will delete its PROXY server entry for the node and will no longer generate incorrect ARP responses for the node.

[0087] **FIG. 7** is a drawing of an exemplary enterprise network used to illustrate the functionality of address resolution using ARP translation servers in accordance with the present invention. In particular, another approach involving the use of ARP translation servers often proves to be a more desirable solution to that provided by the proxy ARP server approach of **FIG. 6**. The ARP translation prevents undesirable flooding of ARP requests through radio and tunnel links.

[0088] An ARP translation server operates nearly identically to the proxy ARP server discussed with reference to **FIG. 6**. Instead of acting as a proxy, the ARP translation server unicasts ARP requests through the wireless network. Thus, whether or not an ARP request is received on an inbound or an outbound link, the ARP translation server will translate the broadcast destination address, in the ethernet header, to the unicast ethernet address of the target node, if the ARP translation server has PROXY entry for the target IP address. The unicast frame is then routed through the

OWL network to the target node so that the target node can return an ARP response packet.

[0089] In the exemplary enterprise network of **FIG. 7**, a terminal **715** has an IP address for a subnet **712**. Assume that the terminal **715** has either sent an inbound ARP frame or registered its IP address within an ATTACH request packet. The ARP server in an AP **703** has a PROXY entry for the terminal (assuming the AP **703** has bridging enabled). A server in an AP **702** has a LOCAL entry for the terminal **715** because the route for the terminal **715** is distributed, i.e., the AP **703** is responsible for bridging frames from ethernet to the terminal **715**. A root AP **701** cannot have an entry for the terminal **715** because it is on another subnet **711**. If an IP Host **742** sends a broadcast ARP request frame with the target IP address of the terminal **715**, then the server in the AP **703** will translate the broadcast destination address, in the ethernet header, to the unicast ethernet address of the target node, the IP terminal **715**. The unicast frame is then transmitted to the IP terminal **715**. The IP terminal **715** responds with an ARP response packet which is a unicast packet directed to the IP host **742** via the AP **703**.

[0090] Thus, unlike the proxy ARP server approach, the ARP translation server approach does not require the server to have perfect knowledge of the IP nodes contained within the sub-tree at the corresponding AP. Instead, the ARP translation server merely directing (unicasting) the ARP request when it believes an IP node is contained within its subtree. Whether or not this is true does not matter because the IP node will only respond with an ARP response if it is present and has not roamed.

[0091] Although **FIGS. 1-2** and **4-7** are diagrams with simplistic network configurations with a single wireless hop to a terminal, the aforementioned features and functionality can also be applied to more complex configurations including enterprise networks with multiple wireless hopping pathways to such terminals.

[0092] **FIG. 8***a* is a drawing illustrating operation of an augmenting which supplements off-the-shelf protocol stacks to support various enhanced features. A typical off-the-shelf protocol stack would include a proprietary or defacto industry standard driver **801**, which provides a MAC layer interface to higher level protocol layers such as TCP/IP **803** or IPX/SPX **805**. Exemplary MAC layer interfaces are defined by industry standards such as ODI (open data link interface) or NDIS (network device interface specification) among others.

[0093] Using a conventional approach to enhance functionality, higher level layers of the protocol stack such as the TCP/IP **803** or the IPX/SPX **805** would be modified creating potential incompatibility and duplicity in efforts. Instead, an augmenting agent **807** has been added to interface with the off-the-shelf protocol stacks to provide the enhanced features of an enterprise network built in accordance with the present invention, without requiring modification to the off-the-shelf protocol stacks. The augmenting agent **807** is placed as an independent application to monitor the interface between the driver **801** and the higher layer protocols, e.g. TCP/IP **803** and the IPX/SPX **805**.

[0094] **FIG. 8***b* is a drawing illustrating an alternate implementation of the augmenting agent of **FIG. 8***a* wherein, instead of operation as an independent, monitoring application, the augmenting agent operates as a shim between the driver and the higher level protocols. Specifically, a proprietary or defacto industry standard driver **851** interfaces with protocols TCP/IP **853** and IPX/SPX **855** via the augmenting agent **857**. Although the augmenting agent may intercept all intended exchanges between the driver **851** and the protocols **853** and **855**, the augmenting agent **857** need only intercept those exchanges necessary to provide the desired enhanced functionality. The driver **851** is unaware of the existence of the augmenting agent **857** as are the protocol layers **853** and **855**. Such is the case in **FIG. 8***a* as well.

[0095] The functionality described above regarding ARP registration is carried out by an augmenting agent. Other functionality that might be added through the augmenting agent includes, for example: (1) encypherment/encryption; (2) device authentication; (3) global network configuration; (4) diagnostics such as loop-back testing, signal strength feedback, wireless retry counts, network route tracing, network management via SNMP agent functionality; and (5) filtering and flooding restrictions. Thus, using the augmenting agent, these and other enhanced functions can be added transparent to a given proprietary protocol stack.

[0096] **FIG. 9** is a block diagram of a communication system illustrating the use of an IP tunnel and a data link tunnel to access a roaming terminal in accordance with the invention. A network **900** comprises two subnets **902** and **904**. A router **906** connects the subnets **902** and **904**. A mobile IP terminal **908***a* originally contacts the network **900** through a root access point (AP1) **910** of a wireless network, such as an OWL network. As shown, the root access point **910** is a part of the subnet **902**. The mobile IP terminal has a wired network address respective to the root access point **910**.

[0097] The mobile IP terminal **908***a* subsequently roams, shown as the mobile IP terminal **908***b*. has moved. The mobile IP terminal **908***b* now contacts the network **900** through another access point **912** of the OWL network. The access point **912** is part of the subnet **904**.

[0098] An IP host **914** communicates with the IP terminal **908** through the root access point **910** by the methods described previously in this specification. To access the mobile IP terminal **908**, the IP host **914** directs data to the root access point **910** through the router **906**. An IP tunnel **916** is created between the root access point **910** and another access point **918** for the subnet **904**. The access point **918** serves as a root access point for other wireless access points in the subnet **904**.

[0099] The access points **918** and **912** are nodes in an OWL network. Thus, the root access point **910** may transmit OWL packets through the IP tunnel **916** to the mobile IP terminal **908**.

[0100] In order to facilitate faster access to the mobile IP terminal **908** through the access point **912**, a data link tunnel **920** is created between access point **912** and the access point **918**. This data link tunnel enables data to flow from the access point **918** to the access point **912**, where it is forwarded to the mobile IP terminal **908**. The data link tunnel allows the data to be passed between the access point

918 and the access point **912** without the necessity of bridging the data onto the subnet **904**. Thus, the IP host **914** can communicate with the mobile IP terminal **908** without having to bridge the data onto the subnet **904** after the data reaches the access point **918**.

[**0101**] To communicate with the mobile IP terminal **908**, the IP host **914** sends a packet addressed to the mobile IP terminal **908**. The IP host **914** forwards the packet to the router **906**, where the router **906** forwards the packet from the IP host **914** to the subnet **902**.

[**0102**] There, the root access point **910** forwards the data packet to the access point **918** via the IP tunnel **916**. The access point **918** then forwards the packet to access point **912** via the data link tunnel **920**. The data link tunnel enables the access point **918** to pass data to the access point without bridging the data packet onto the subnet **904**. Thereafter, the access point **912** directs the data packet to the mobile IP terminal **908**.

[**0103**] More specifically, the mobile IP terminal **908** is connected to an OWL network, in this case consisting of the root access point **910** and the access point **918** that serves as a root for the subnet **904**. The subnet **904** connects the third access point **912** to the access point **912** in the OWL network. The connection between the access point **918** and the access point **912** may be via a wired or wireless connection.

[**0104**] The mobile IP terminal **908***b* has a wired network address relative to the root access point **910**. The mobile IP terminal **908***b* wanders from position **908***a* and connects to the the network **900** via the access point **912**. Assume that the IP host **914** sends a packet of data directed to the mobile IP terminal **908***b*. Initially, the IP host **914** sends an IP packet to the root access point **910** via the router **906** using the IP address of the IP terminal **908***b*. Upon receipt, the root access point **910** encapsulates the IP packet in an OWL packet. The root access point **910** encapsulates the resulting OWL packet within another IP packet addressed to the access point **918**. The IP packet is then sent to the access point **918** via the IP tunnel **916**.

[**0105**] The access point **918** de-encapsulates the original IP data and re-encapsulates the IP data in an OWL data frame. The access point **918** then forwards the re-encapsulated packet to the access point **912** through data link tunnel **920**. The access point **912** de-encapsulates the original IP data and forwards it to the mobile IP terminal **908**.

[**0106**] The data link tunnel **920** is established by disabling bridging on the access point **918**. When bridging is disabled, no data is bridged onto the subnet **904**.

[**0107**] **FIG. 10** is a drawing illustrating an exemplary protocol stack associated with the access point at the end-points of the IP tunnel and the data link tunnel illustrated in **FIG. 9**. When a bridging protocol **1010** is disabled, any incoming packets of data cannot flow through the bridging layer **1010** to a TCP layer **1012** or to an ethernet MAC layer **1014**.

[**0108**] When the bridging protocol **1010** is disabled, the data packets must flow through the MAC-R protocol layer **1020**. The packets then flow to the various MAC-D protocols, such as a MAC-D ethernet layer **1030**, a MAC-D radio layer **1040**, and an IP MAC-D protocol layer **1050**.

[**0109**] A line **1070** in **FIG. 10** illustrate the paths an IP data packet destined to the mobile IP terminal **908** can take through the protocol stack **1000** in access point **918** when the data link tunnel **920** is activated. Line **1070** illustrates the path the IP data packet takes when access point **918** connects to access point **912** via an ethernet link. In this case, the packet enters the protocol stack **1000** through the IP MAC-D protocol layer **1050**. This corresponds to the data packet entering the access point **918** via the IP tunnel **916**. The MAC-R protocol layer **1020** then processes the data packet. Since bridging on the access point **918** has been disabled, the data packet need not enter the bridging protocol layer **1010**. The MAC-R protocol layer **1020** turns the processing of the data over to the MAC-D ethernet protocol layer **1030** for processing. The MAC-D ethernet protocol layer **1030** then formats the data packet for transmission to the access point **912** via an ethernet physical protocol layer **1060**.

[**0110**] **FIG. 11** is a drawing illustrating an exemplary protocol stack associated with the access point at the end-points of the IP tunnel and the data link tunnel illustrated in **FIG. 9**. A line **1080** illustrates the path of the packet through the protocol stack **1000** when access point **912** connects to access point **918** via a radio link. As before, the IP data packet enters the protocol stack through the IP MAC-D protocol layer. The data packet is then processed by the MAC-R protocol layer **1020**. The data packet is then turned over to the MAC-D radio protocol layer for transmission to access point **912**.

[**0111**] Note that in both cases, the encapsulated IP packet did not cause access point **918** to bridge the data onto the subnet **904** during the transfer of the IP packet to the access point **912**.

[**0112**] Thus, the access points **912** and **918** serve as logical extensions of the subnet **902**. A home subnet, such as the subnet **902**, can be transparently extended into an enterprise IP network, such as network **900**, by concatenating data link tunnels, such as the data link tunnel **920**, to an existing IP tunnel, such as IP tunnel **916**. The data link tunnels prevent frames passing through an IP tunnel from one IP subnet to another from being bridged onto the second subnet.

[**0113**] In view of the above detailed description of the present invention and associated drawings, other modifications and variations will now become apparent to those skilled in the art. It should also be apparent that such other modifications and variations may be effected without departing from the spirit and scope of the present invention as set forth in the claims which follow.

[**0114**] Incorporation by Reference

[**0115**] Pages 35-308 comprises the following Appendices which are hereby incorporated herein by reference in their entirety:

APPENDIX A	OWL NETWORK ARCHITECTURE	Pages 35–61
APPENDIX B	OPEN WIRELSS LAN THEORY OF OPERATION	Pages 62–250
APPENDIX C	OWL NETWORK FRAME FORMATS	Pages 251–264
APPENDIX D	UHF/DIRECT SEQUENCE MAC-D PROTOCOL SPECIFICATION	Pages 265–308

[**0116**]

APPENDIX A

OWL NETWORK ARCHITECTURE

Overview.

Norand's open wireless LAN (OWL) architecture is designed to facilitate wireless communications at the MAC sub layer of the ISO protocol stack. An OWL radio network can function as a stand-alone LAN or it can function as a subnet in an 802 LAN to provide wireless access to wired 802 subnets. An 802 LAN may include multiple wired 802 subnets and OWL subnets. Figure 1 shows an example 802 LAN which includes an OWL subnet. The OWL subnet (i.e. subnet 4) includes the OWL radio network (i.e. subnet 2) and a "secondary" 802.3 subnet (i.e. subnet 3).

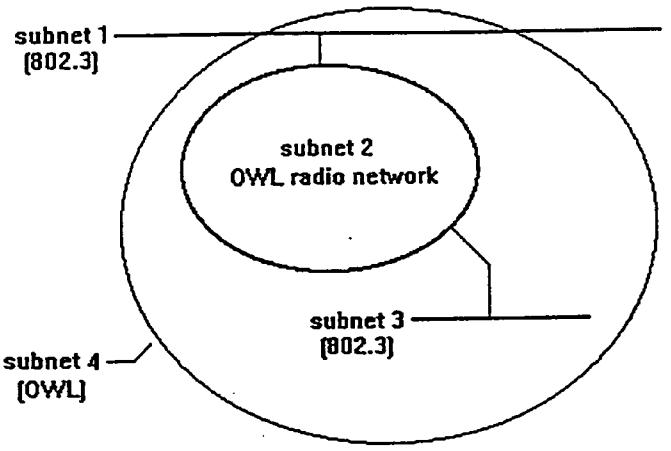


figure 1.

Figure 2 shows an example 802 LAN, similar to the LAN in figure 1, with an expanded view of the OWL radio network. Subnet 1 is not part of the OWL subnet, however it provides a distribution LAN for the OWL subnet. An OWL radio network provides wireless access to the 802 LAN for mobile radio-equipped computers (MRCs). An OWL radio network can also provide a wireless transparent bridge between wired 802 subnets (i.e. an OWL subnet can include a wired 802 subnet). Any node in an 802 LAN, which includes an OWL subnet, can communicate with any other node, at the logical link control (LLC) sub layer of the data link layer. In figure 2, remote station 1 can communicate with either MRC or remote station 9. MRC 6 can communicate with MRC 8 or either remote station.

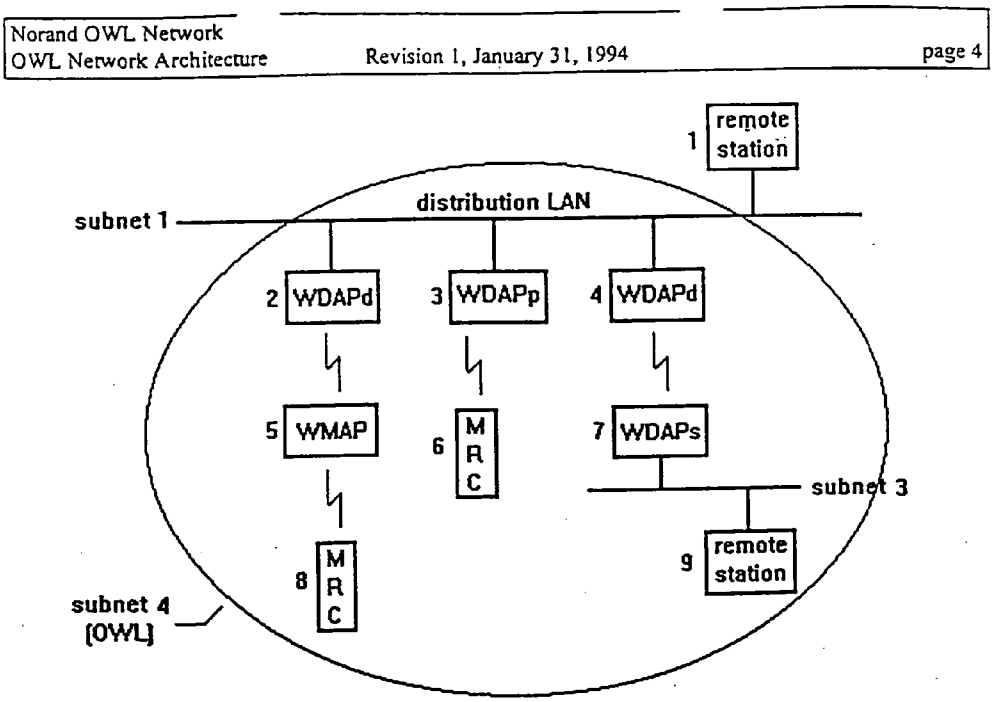


Figure 2.

The IEEE 802.11 committee has defined two basic types of wireless networks - hierarchical and ad hoc. Hierarchical networks contain radio-equipped access points which provide a centralized control function for a given radio coverage area. All communications pass through the access point. The access point also provides access to a wired LAN. A hierarchical network may contain multiple access points which provide an extended seamless radio coverage area. Mobile computers can roam from one access point coverage area to another. Ad hoc networks facilitate peer-to-peer communications in the absence of a central control point. This document is primarily directed toward hierarchical networks.

The OWL protocol stack is contained in the MAC sub layer of the ISO protocol stack. An OWL MAC (i.e. in a terminal node) provides MAC sub layer services to the LLC sub layer of the ISO data link layer. The OWL MAC is subdivided into 4 sub layers: MAC-D, MAC-R, MAC-Q, and MAC-S.

MAC-D - The MAC-D sub layer is analogous to the data link layer in the ISO protocol stack. The MAC-D layer provides data link services to the MAC-R layer. It is responsible for channel access control and the reliable transmission of MAC-R PDUs across a single link in the OWL network. The MAC-D sub layer is specific to the link type (i.e. radio, ethernet, etc.).

MAC-R - The MAC-R sub layer is analogous to the network layer in the ISO protocol stack. The MAC-R layer provides routing services to the MAC-Q layer. It is responsible for correctly routing MAC-R PDUs through the OWL subnet, which may include multiple hops and circular physical paths.

MAC-Q - The (optional) MAC-Q sub layer adds reliability to the radio network by retransmitting lost PDUs. The MAC-Q layer is responsible for discarding out-of-sequence and duplicate PDUs. The MAC-Q sub layer can be implemented as an entity in the MAC-R sub layer. MAC-Q entites exist at entry points to the radio network.

MAC-S - The (optional) MAC-S sub layer is responsible for providing services for security, compression, etc. MAC-S entities exist at entry points to the OWL radio network.

A logical **OWL node** is a MAC-R addressable entity in an OWL radio network. An OWL node can be one of two types: 1) a **terminal node** or 2) a **relay node.** Terminal nodes are end points in the network; relay nodes forward PDUs at the MAC-R sub layer. Figure 3 shows MAC protocol stacks for both node types. The arrows represent the flow of data between MAC sub layers in each node type. (The upper layers in the relay stack are used to process PDUs addressed to the relay node.)

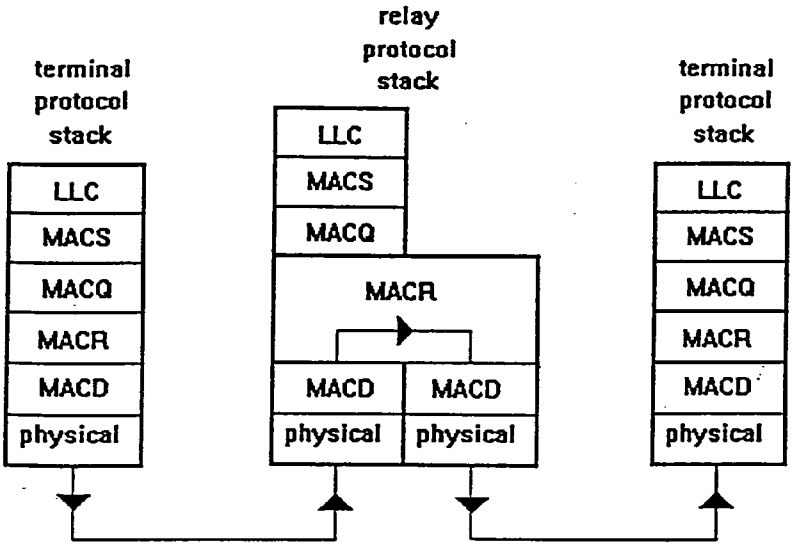


figure 3.

A wireless domain access point (WDAP) is an OWL bridge which is used to bridge a radio subnet to a wired 802 subnet. A WDAP contains a **bridge protocol stack.** Figure 4 shows the MAC protocol stack for a WDAP. Note that the bridge protocol stack contains a relay protocol stack. The 802.3 MAC-D sub layer is used to send OWL PDUs over an 802.3 link that is part of the OWL radio network. The MAC-Q and MAC-S sub layers serve as proxy MAC-Q and MAC-S entities for stations on the 802.3 sub net. The MAC-Q and MAC-S sub layers also service PDUs for the local WDAP 802 address.

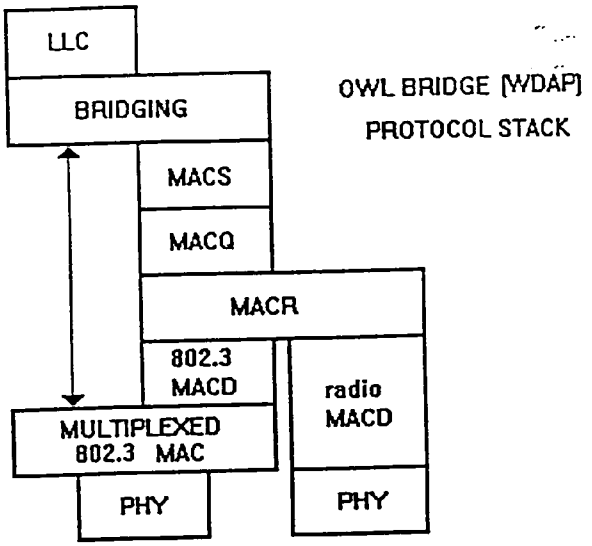


figure 4.

Figure 5 illustrates how data flows through a bridge protocol stack. The dotted line represents the path a PDU takes as it travels from a station on an 802.3 LAN to terminal 2 in an OWL radio network. The WDAP "bridges" the PDU from the 802.3 subnet to the radio subnet. The solid line represents the path a PDU takes as it travels from terminal 1 in the radio network to terminal 2 in the radio network. Since the path is contained in the radio network, the PDU does not have to be bridged.

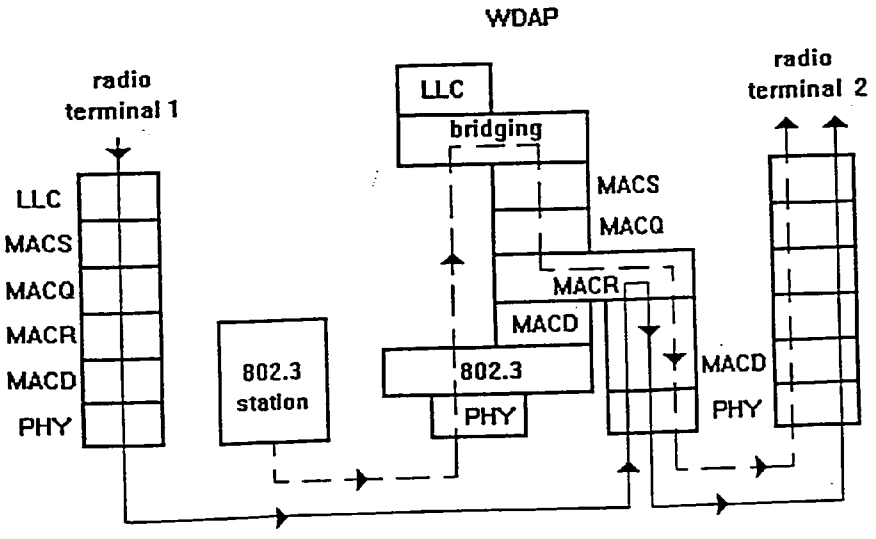


figure 5.

In general, PDUs are bridged across subnet boundaries; PDUs are routed within the radio network. A bridging entity in a WDAP uses a forwarding database to determine if a PDU should be bridged from one subnet to another subnet. A forwarding database contains a list of 802 address associated with each subnet to which the WDAP is attached. A MAC-R entity uses a routing table to determine how a PDU should be routed within an OWL subnet.

Network components and definitions.

802 LAN - a (possibly bridged) local area network which conforms to the IEEE 802 standards. For the purpose of this discussion, it is assumed that "802 LAN" refers to a LAN which contains wired 802.3 (ethernet) subnets and 1 or more OWL subnets.

802 subnet - a subnet in an 802 LAN which is not an OWL subnet.

OWL subnet - a subnet in an 802 LAN which includes an OWL radio network and 0 or more 802 subnets.

OWL Radio Network - An OWL subnet minus its wired subnets (see figure 1). An OWL radio network may include wired (i.e. 802.3) communications links. The OWL radio network consists of MAC-R addressable nodes and communications paths.

Mobile Radio-equipped Computer (MRC) - A mobile radio-equipped computer which contains an OWL terminal node.

Wireless Media Access Point (WMAP) - a radio-equipped base station which allows physical access to a wireless link in an OWL LAN. A WMAP may be connected to the radio network through a wired link or a radio link. A typical OWL radio network has multiple WMAPs with overlapping coverage areas. MRCs can roam between coverage areas. Except for possible timing issues, roaming has no effect on protocol layers above the MAC sub layer.

Wireless Domain Access Point (WDAP) - a logical access point to an OWL radio network. There are several types of WDAPs which are defined below. A WDAP is typically contained in a WMAP which is directly connected to a wired 802 subnet. The WDAP provides a bridge between the radio network and the wired subnet. A WDAP has a MAC-S and MAC-Q sub layer since it provides an entry point to the radio network. At any given time, one, and only one, WDAP provides access to a distribution LAN for a node in the OWL subnet.

OWL Node - A MAC-R addressable entity in an OWL radio network.

OWL Terminal Node - A MAC-R addressable OWL node which is an end point in an OWL radio network. A terminal OWL node is simply referred to as a terminal when the meaning is not ambiguous. A terminal has a MAC-S and MAC-Q sub layer since it provides an entry point to the radio network.

OWL Relay Node - A MAC-R addressable OWL node which is an interior node in an OWL radio network. MAC-R frames are routed through OWL relay nodes.

OWL Spanning Tree - An OWL spanning tree consists of a single root node, OWL relay nodes, terminal nodes, and edges, where a single edge logically connects two nodes for routing purposes. A branch is a logical path which contains 1 or more edges and the associated nodes. MAC-R frames are routed along branches of a spanning tree.

OWL Network Spanning Tree - All nodes in a hierarchical OWL subnet are organized into a network spanning tree for control purposes. A single network spanning tree constitutes an OWL domain. The root

of the network spanning tree contains a primary WDAP. Note that an 802 LAN may be attached to multiple OWL network spanning trees (domains).

OWL Access Spanning Tree - An access spanning tree is a sub tree in a network spanning tree. The root of an access spanning contains a distributed or primary WDAP and provides direct access to a distribution LAN.

Net ID - The Net ID identifies the set of nodes which belong to a single OWL domain - a network spanning tree or an instance of an ad hoc network. A hierarchical bit specifies whether the Net ID is for a hierarchical network or an ad hoc network. All nodes in an OWL domain share a common Net ID.

Super Root - the root of a network spanning tree. Multiple access points, attached to a distribution LAN, can negotiate to determine which node should function as the super root of a network. The super root is the node with the highest super root priority. The super root must have direct access to a distribution LAN. The super root is the primary WDAP.

Access Root - the root of an access spanning tree. An access root is a primary or distributed WDAP.

Distribution LAN - An 802 LAN segment which connects a wired subnet to the OWL subnet through the primary WDAP and 0 or more distributed WDAPs.

Distributed Root - the set of nodes which consists of the super root and all access roots. For a single OWL node, the distributed root can be viewed as the super root and the distributed WDAP which is providing access for the node to the distribution LAN.

Primary WDAP (WDAPp) - A single primary WDAP serves as the super root and provides a single control point for an OWL subnet. The primary WDAP has direct access to the distribution LAN. The primary WDAP "bridges" 802 frames from the distribution LAN to the OWL subnet and from the OWL subnet to the distribution LAN.

Distributed WDAP (WDAPd) - A distributed WDAP provides direct physical access to the distribution LAN. Distributed WDAPs exist within the domain of the primary WDAP. A distributed WDAP "bridges" 802 frames from the distribution LAN to the OWL subnet and from the OWL subnet to the distribution LAN.

Secondary WDAP (WDAPs) - An OWL subnet may include remote 802 subnets other than the distribution LAN. A single secondary WDAP serves as the designated bridge between the remote wired subnet and the OWL subnet. 802 frames are bridged from the remote wired subnet to the radio subnet and from the radio subnet to the remote wired subnet through the secondary WDAP. If more than one secondary WDAP is attached to the remote LAN, then the AP with WDAP with the highest bridge priority is "elected" as the designated bridge.

Secondary LAN - an 802 subnet in an 802 LAN which is attached to the OWL network through a secondary WDAP.

Access Point (AP) - a primary, distributed, or secondary access point.

Station - an entity in the 802 LAN which has a unicast 802 address.

OWL Station - a station in an OWL radio network.

Remote Station - a station which is not in an OWL radio network (i.e. a station on the distribution LAN or a secondary LAN).

Node Address - the 48-bit 802 address which uniquely identifies a node in an OWL network.

Node ID - A 16-bit node identifier which can be used to replace the 48-bit node address. In a hierarchical network, each OWL node must obtain a node ID from the super root. The concatenated Net ID and node ID uniquely identify the node within the radio network.

Port Address - the 48-bit 802 address or 16-bit address which uniquely identifies a port in an OWL network. A port address can also be used as the node address.

Port ID - A 16-bit port address, which is used to replace the associated 48-bit port address. In a hierarchical network, each OWL node can obtain a network unique port ID from the super root, for each of its ports. The concatenated Net ID and port ID uniquely identify the port within the radio network.

originator - the node which originates a unicast or multicast transmission.

sink - the target node of a unicast transmission.

conversation - a series of transmissions which are used to forward a frame from an originator to a sink. The frame may be divided into multiple fragments.

MDPDU - a MAC-D sub layer protocol data unit.

MRPDU - a MAC-R sub layer protocol data unit.

MQPDU - a MAC-Q sub layer protocol data unit.

MSPDU - a MAC-S sub layer protocol data unit.

MQPDUID - The concatenation of the MQPDUID and 802 source and destination addresses uniquely identifies an MQPDU in an OWL radio network.

inbound - Nodes which are logically closer to the root node of a spanning tree are considered "inbound" from nodes which are further from the root. A inbound PDU is any PDU which is traveling toward the root.

outbound - Nodes which are logically further from the root node of a spanning tree are considered "outbound" from nodes which are closer to the root. An outbound PDU is any PDU which is traveling away from the root. An OUTBOUND bit in a MAC-R control field is set ON to indicate that the source of a MRPDU is inbound from the destination of the PDU. Note that terminal nodes never set the OUTBOUND bit ON.

OWL Addressing.

Each physical port in an OWL network has a unique 48-bit 802 address. An OWL access point (AP) has an ethernet port and 1 or more radio ports, each with its own unique 802 address. Each radio port requires a unique address, because it is possible for more than one radio port to be active on the same radio channel at the same time. The 802 address for one of the ports is also used as the MAC-R node address. A terminal node has a single radio port. The 802 radio port address is also used as the terminal's node address. A 16-bit radio port ID can be used as a 16-bit port address in lieu of a 48-bit 802 port address, for either a terminal or AP. A 16-bit port ID is obtained from an address server in the root node of an OWL LAN. A 16-bit port ID is unique within the domain of an OWL LAN ID. The port address size used on each port (i.e. 2 or 6 bytes) is dependent on the underlying MAC-D type. The address size is assigned when the port is initialized and is fixed for all packets transmitted or received on the port.

The 16-bit port ID for one of the AP ports is also used as the node ID. The node ID for a terminal must be the same as its 16-bit radio port ID. The 16-bit node ID uniquely identifies a node in the OWL radio network. A child's node ID will appear in a pending message list (i.e. in a HELLO packet) when the parent AP is storing messages for the child.

OWL packets carry data and network management information within the OWL radio network. For OWL packets, the DIX ethernet type is hex. 875C or the SNAP SAP is hex. 00C0B2875C. OWL packets contain a MAC-D and a MAC-R header. The destination and source addresses in the MAC-D header are required to uniquely identify the hop destination port and source port, respectively. The destination and source addresses in the MAC-R header identify the end-to-end destination and source nodes, respectively. In addition, inbound unicast MAC-R PDUs which are relayed by an AP, contain the node address of the AP. When an ethernet frame is bridged into the radio network, the detination and source 802 addresses in the ethernet header are copied into the MAC-R destination and source address fields.

An AP bridges packets across subnet boundaries (i.e. radio network to ethernet) and bridges packets from AP ports to internal applications. Since an AP port address may be different than the AP node address, an AP must internally "bridge" packets from an AP port to/from higher layer service access points (SAPs) contained in the AP. The source 802 address for packets generated by a higher layer SAP is always the AP node address. Likewise, the destination 802 address for packets destined for a higher layer SAP in the AP, is the AP node address. For example, the 802 address returned in an (i.e. TCP/IP suite) ARP packet is the AP node address. IP packets destined to the node address can be received on any AP port (i.e. with a different ethernet address) because each port is operating in promiscuous mode. Packets destined to the AP node address are forwarded to the higher layer SAP identified in the DIX ethernet or LLC header. As a second example, assume that an AP has two 2.4 GHz radio ports, each with a port address which is different than the AP node address. A remote RF device can send a packet to the AP node address by using the AP port address (i.e. the MAC-D destination address) to relay the packet. The port address uniquely identifies one of the radio ports. The remote RF device does not need to know that the MAC-R destination exists on the AP that owns the port. Once the packet arrives at the AP, the MAC-R destination address is used to determine that the packet belongs to a local SAP.

MAC-D Sub Layer.

The MAC-D sub layer controls access to the channel and is responsible for providing reliable transmission between any two devices in the radio network. A radio network may include both wired and radio links. The MAC-D sub layer is specific to the physical link type. An 802.3 MAC-D sub layer is used on 802.3 links and a radio MAC-D sub layer is used on radio links.

MAC-D Sub Layer for radio links.

The radio MAC-D sub layer provides "acknowledged connectionless" services to the MAC-R sub layer. A "connection" is not required to transmit an MRPDU; however, each PDU is acknowledged at the MAC-D sub layer and errors are reported to the MAC-R sub layer. For a terminal node, a MAC-D link error provides an indication that the terminal has roamed.

Radio MAC-D Protocol Data Units.

An MDPDU is classified as either a control frame or a data frame. Control frames facilitate network access and error recovery for unicast conversations. Data frames contain an MRPDU. A common header format is used for both control and data frames.

MAC-D header format.

protocol ID
network ID
destination node ID
source node ID
control
reservation

Control frames.

Control frame format.

preamble
SFD (start frame delimiter)
<physical layer header>
MAC-D header
CRC

Note that control frames have a fixed length.

Control frame types.

A control frame is classified as either a request frame or a response frame. A single bit in the type field indicates if a control frame is a request or a response.

Control request frame types.

RTS - an RTS frame is used to reserve the network for a unicast conversation.

ENQ - an ENQ frame is used by an originator to determine the status of a previous fragment transmission. The sink responds by re-transmitting its last ACK or CLEAR. If the sink node does not have ACK state information, it responds to an ENQ by transmitting a REJECT. Note that an ENQ/ACK pair correspond to an RTS/CTS pair with respect to channel access.

ABORT - an ABORT can be used by an originator to abort an active conversation. Note that a conversation can be restarted at any time.

Control response frame types.

CTS - a CTS frame is used to acknowledge an RTS frame and grant access to the network.

ACK - an ACK frame is used to acknowledge the reception of a unicast data frame fragment. The control byte in an ACK frame contains the 1-bit sequence number of the next data frame fragment expected.

CLEAR - a CLEAR frame is used to acknowledge the reception of the last unicast data frame fragment in a conversation. A last-in-chain (LIC) bit distinguishes a CLEAR frame from an ACK frame.

REJECT - a REJECT frame is used by a sink to notify an originator that a unicast conversation has been aborted by the sink or that the sink does not have ACK state information for the originator. The originator must restart the conversation.

Data frames are used to send MAC-R data. The control field in a data frame contains a 1-bit sequence number used to facilitate fragmentation and re-assembly of large unicast frames. All broadcast and multicast transmissions consist of a single DATA frame. Unicast frames may be broken into multiple DATA fragments for transmission. A first-in-chain (FIC) bit is set ON in the first DATA fragment of a frame. A last-in-chain (LIC) bit is set ON in the last DATA fragment of a frame. Note that both FIC and LIC are set ON in single-fragment frames. An EOD (end-of-data) fragment is a data fragment with the LIC bit set ON. Fragmentation and re-assembly at the MAC-D sub layer is transparent to the MAC-R sub layer.

Data frame format.

preamble
SFD
MAC-D header
MRPDU fragment
CRC

Frame transmission.

Multicast frames are sent as a single EOD frame.

Example multicast transmission:

EOD ——————————>

Example unicast transmission with no errors:

RTS ——————————>
<—————————— CTS
DATA 0 ——————————>
<—————————— ACK 1
DATA 1 ——————————>
<—————————— ACK 0
EOD 0 ——————————>
<—————————— CLEAR 1

If a sink receives an RTS frame and the channel is reserved, then the sink must withhold the CTS frame. The originator must calculate a random back off time and retry later.

Example transmission with errors:

```
RTS ------------------>
  <------------------ CTS
DATA 0 ---------------->
  <------------------ ACK 1
    { ACK lost}
ENQ ------------------->
  <------------------ ACK 1
DATA 1 ---------------->
    { DATA lost }
ENQ ------------------->
  <------------------ ACK 1
 DATA 1 ---------------->
  <------------------ ACK 0
EOD 0 ----------------->
  <------------------ CLEAR 1
    { CLEAR lost }
ENQ ------------------->
  <------------------ CLEAR 1
```

Radio Channel Access.

Channel access in an OWL radio network is complicated by the presence of multiple overlapping radio coverage areas and hidden nodes. A given first radio transceiver is said to be hidden from a second transceiver, if the second transceiver is not in range of the first, but transceivers exist which are in range of both. In figure 6, the large circles represent the radio coverage area of nodes A, B, C, and D. C, for example, is considered to be hidden from A since it is not in A's coverage area, but a node, B, is in the coverage area of both.

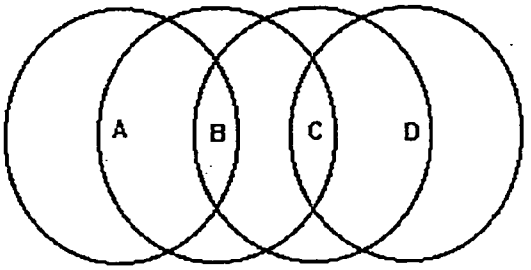


figure 6.

The hidden node problem can severely limit bandwidth utilization in a simple carrier sense radio network if the percentage of hidden nodes is significant. As an example, assume that node A, in figure 6, is transmitting a frame to node B. If, at the same time, C senses the channel it will appear idle, since C can not hear A. If C begins transmitting to D, the transmission from A will collide with the transmission from C at B and will likely be lost. (The transmissions from A and C will not collide at D.)

The OWL MAC-D sub layer uses a listen-before-talk (LBT) collision avoidance protocol to reduce the number of collisions caused by hidden nodes. Nodes reserve the channel for unicast conversations. The reservation in request frames reserves the channel for succeeding data frames. Response frames echo the reservation in the previous corresponding request frame. The reservation in a request frame does not have to span an entire conversation since the reservation can be extended in succeeding data frames. (Shorter reservations reduce dead times when frames are lost.) The reservation in a request frame includes an implicit reservation for the required response (including turnaround time).

The channel reservation technique generally restricts channel access contention to RTS frames. In the absence of lost frames, an LBT algorithm is executed only once per MAC-D conversation. An originator executes the LBT algorithm and transmits an RTS frame if the channel is free. The originator owns the channel for the duration of a conversation as soon as it receives a CTS from the sink. Subsequent DATA fragments can be sent without additional channel access logic. If the channel is not free, a random back off algorithm, chooses a back off delay as a function of the LBT slot time and the number of retries. An LBT slot is defined as a function of the best case and worst case busy-sense time. The best case busy sense time is equal to the amount of time from the point at which a node detected the channel idle, before transmitting, until another node can detect the transmission in progress. The worst case busy-sense time is equal to the time required by the originator to sense the channel idle and send an RTS frame plus the time required by a sink to start sending a CTS frame. Figure 7 shows a time line for a unicast conversation between two nodes, A and B. If the originator, A, senses the channel idle at time 0, then the worst-case busy sense time is t_{ws}.

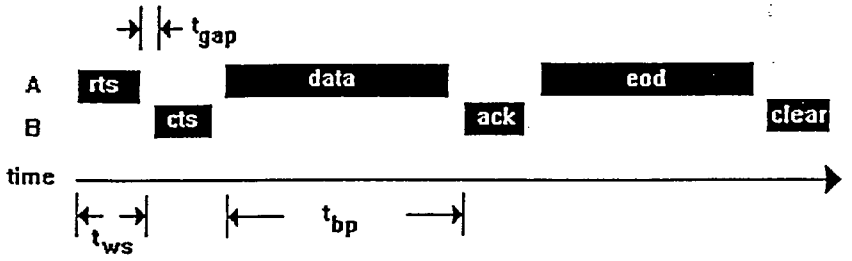


figure 7.

Each node in the network must maintain RESERVE_TIME and RESERVE_NODE channel reservation variables and a local clock. The channel is reserved if the RESERVE_TIME value is greater than the current time. The RESERVE_TIME variable is updated if a reservation is received and 1) the channel is currently not reserved, or 2) the transmitter of a request frame is the RESERVE_NODE node, or 3) the destination of a response frame is the RESERVE_NODE node, or 4) the reservation field in a unicast frame reserves the channel for a time greater than the current RESERVE_TIME period. The RESERVE_TIME is set to 0 whenever a reservation of 0 is observed and the RESERVE_NODE node is the destination of a response frame. The RESERVE_TIME is set to 0 whenever the local node is the target of a unicast transmission from the RESERVE_NODE.

The RESERVE_NODE is set to the concatenated Net ID and node ID of the node which is reserving the channel (i.e. the source node ID and Net ID in a request frame or the destination node ID and Net ID in a response frame) whenever the RESERVE_TIME is updated.

The channel is considered busy if it is sensed busy or if it is reserved. If the channel is reserved, the random delay, chosen by the random back off algorithm, is added to the reserve time. When the delay time expires, the originator repeats the LBT algorithm.

A basic service set (BSS) includes a WMAP and its children. In a frequency hopping network, each BSS is, for the most part, isolated from its neighbors by channel frequency separation, but BSS frequencies may occasionally overlap. Reservations may be missed if BSS frequencies overlap for part of a conversation. If a frequency hop time begins with a synchronization frame, then the synchronization frame can include an indication that the channel is busy.

A sleeping node is any node which has not been actively listening to network traffic. A sleeping node may miss an RTS/CTS sequence. The OWL radio MAC-D protocol uses a busy-pulse technique to support sleeping terminals. CTS and ACK frames provide periodic pulses to indicate that the source node is busy. A sleeping terminal is required to monitor the channel for a busy-pulse period before accessing the channel. If a conversation is in progress, the terminal is guaranteed to hear either the originator or the sink within the busy-pulse period. In figure 7, the busy-pulse period is t_{bp}. The busy-pulse period is well-defined if the maximum fragment and turn-around times are fixed. The combined OWL reservation and busy-pulse protocols provide a channel access solution which is analogous to a busy-tone channel access protocol.

Terminal nodes should limit the total retry time at the MAC-D sub layer, so that roaming can be quickly detected, and a new path in the spanning tree quickly re-established. Relay nodes should lower the number of retries, due to lost frames, when the sink is a terminal node, since the lost frames may be due to roaming. The retry limit should be much higher when both the originator and sink are relay nodes.

802.3 MAC-D Sub Layer.

The 802.3 MAC-D sub layer is used to forward MAC-R PDUs across 802.3 links. All 802.3 MAC-D frames use a common reserved 802 multicast address and LLC SNAP access point identifier in the 802.3 and LLC header, respectively. The OWL MAC-D PDU is contained within the LLC PDU. The 802.3 MAC-D sub layer is used when two (or more) nodes in the OWL network spanning tree are physically connected by an 802.3 link. Note that the same physical link can function both as a distribution LAN and as the physical link associated with a path in the network spanning tree. It is important to understand the following distinction. If a WDAP bridges a frame (i.e. from the radio network) onto a distribution LAN, then the frame is no longer on a branch in the OWL network spanning tree, even if the destination 802 address belongs to a node in the OWL subnet; however, if a WMAP routes an MRPDU to another WMAP then the PDU is forwarded on a branch in the spanning tree, even if the physical link used to forward the PDU also serves as the distribution LAN.

The 802.3 MAC-D PDU fields are shown below. All 802.3 MAC-D transmissions consist of a single data PDU. No control frames are defined. An 802.3 MAC-D sub layer does not fragment MAC-R PDUs.

802.3 MAC-D header format.

protocol ID
network ID
destination node ID
source node ID
control
reservation

802.3 MAC-D data frame format.

802.3 header
LLC header with SNAP access points
MAC-D header
MRPDU
CRC

MAC-R Sub Layer.

The MAC-R sub layer is responsible for correctly routing higher layer PDUs through the OWL subnet. OWL nodes are organized into a network spanning tree and PDUs are routed along branches of the spanning tree. The MAC-R sub layer also provides support for sleeping terminals and distributes network node IDs. The MAC-R sub layer provides unacknowledged connectionless services.

MAC-R Protocol Data Units

MAC-R Header Format

control
destination 802 address
source 802 address
<type specific fields and optional parameters>

MRPDU types.

REGISTRATION - A node sends a REGISTRATION request to the super root to obtain an OWL network node ID, and, optionally, a port ID for each of its physical ports. The registration PDU contains the 802 node address and, optionally, a list of 802 port addresses. The super root records the 802 addresses and returns a node ID, and a port ID for each port address, in a REGISTRATION response PDU. A REGISTRATION request may contain a node alias. The alias is the permanent name of a node in the OWL radio network.

ATTACH - A node sends an ATTACH request to a parent node to attach to the OWL subnet. The ATTACH request is forwarded to the distributed root to establish full connectivity in the OWL subnet. The distributed root returns an ATTACH response packet to acknowledge the ATTACH request. An attach indication (ATTI) bit in the control field of the ATTACH request indicates if the path to the node which generated the ATTACH request has changed. The MAC-R entity in a distributed WDAP (i.e. access root) sets a DISTRIBUTED bit ON in the control field of an ATTACH request before forwarding the request to the super root. The super root records the DISTRIBUTED bit in its routing table and does not bridge frames from the distribution LAN to the attaching node if the DISTRIBUTED bit is ON (i.e. because the distributed WDAP is responsible for bridge frames to the attaching node).

HELLO - Each relay node in a hierarchical OWL radio network periodically broadcasts HELLO response PDUs to advertise its presence. Pending messages for sleeping terminals and broadcast messages can be associated with HELLO PDUs. A node can send a HELLO request PDU to solicit (unscheduled) HELLO response PDUs from attached relay nodes. Each HELLO response PDU contains the 802 address of the super root and a super root sequence number. The super root address and sequence number are used to uniquely identify an occurrence of an OWL network. In addition, each node in the network can learn the 802 address of the super root. Optionally, HELLO response PDUs can contain an encrypted security ID, which uniquely indentifies the OWL domain. HELLO PDUs are ignored if the security ID does not match the OWL domain security ID.

DATA - DATA request MRPDUs are used to transport higher layer data.

R-DATA - DATA response MRPDUs are used to reroute undelivered DATA request MRPDUs after a route has changed.

ALERT - A relay node sends an inbound ALERT request when it is unable to deliver a PDU to a child. The ALERT request is used to determine if the path to the child is still valid and is optionally used to alert the child that it has missed a PDU and should re-attach.

DETACH - A relay node sends a DETACH response node to delete a path to an outbound node.

OWL Network Spanning Tree.

Nodes in an OWL radio network are organized into a network spanning tree. A primary WDAP serves as the (super) root of the spanning tree. PDUs are routed along branches of the spanning tree. Figure 8 shows physical devices and links in an example OWL network. Figure 9 shows the same network organized as a logical network spanning tree.

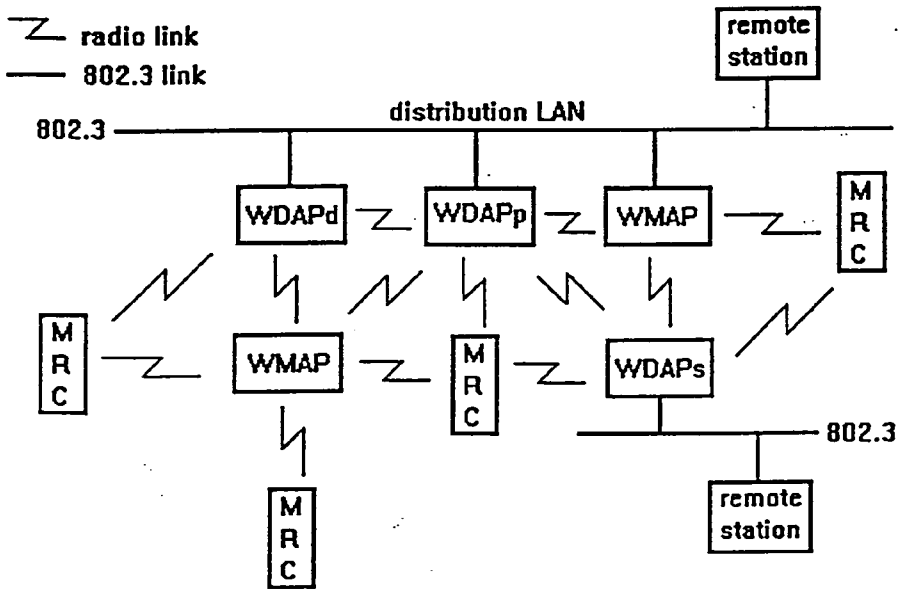


figure 8.

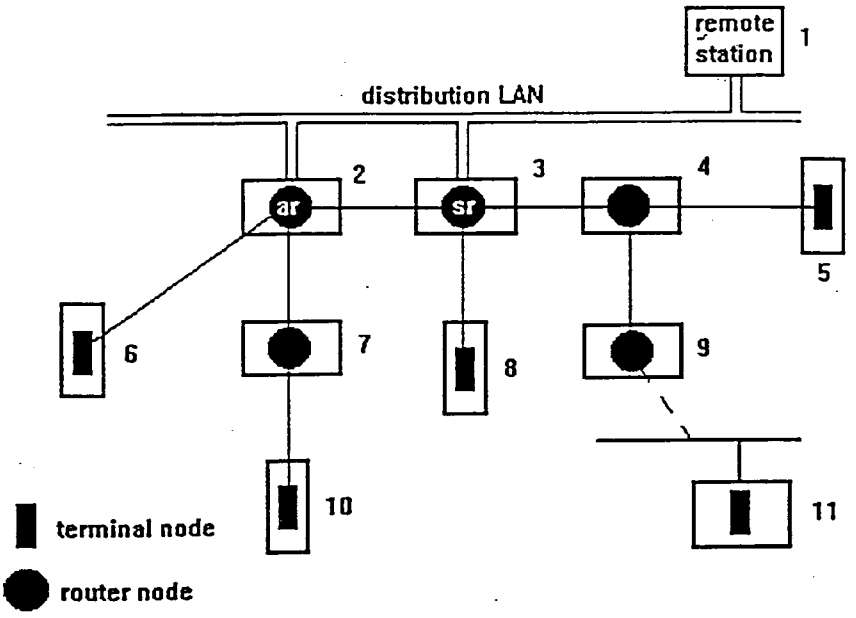


figure 9.

The spanning tree eliminates loops in the physical topology. The node labeled "sr", in figure 9, is the super root and the node labeled "ar" is an access root. The parallel lines represent the distribution LAN, which is not part of the spanning tree. The super root and access root both have access to the distribution LAN; the WMAP labeled 4 can not directly access the distribution LAN. WMAP 4 forwards PDUs destined for the distribution LAN through the super root (i.e. with an 802.3 MAC-D sub layer). The remote station, labeled 1, on the distribution LAN is not part of the network spanning tree; however, the secondary 802 LAN and the remote station, labeled 11, can be viewed as part of the spanning tree (as indicated by the dotted edge).

Building the Spanning Tree.

Nodes in the radio network are generally categorized as attached or unattached (i.e. to the network spanning tree). Initially, only the super root is attached. A single WDAP can be designated to contain the root node, or multiple root candidates can negotiate to determine which node assumes the super root status. The root and other attached relay nodes broadcast HELLO response PDUs at calculated intervals. The HELLO response PDUs enable unattached nodes to learn the optimum path to the super root before attaching to the network. The HELLO response PDUs include: 1) the source node ID and 802 address; 2) a broadcast destination node ID and 802 address; 3) the "cost" to the super root; 4) a "seed" value used to calculate the time of the next HELLO response PDU; 5) a hello displacement time; 6) the priority of the super root node (or root candidate); 7) the 802 address of the super root (or root candidate); 8) a super root sequence number, used to distinguish between multiple occurrences of the network spanning tree with the same super root; and 9) an optional security ID, which uniquely indentifes the OWL domain.

The HELLO "cost" field indicates the total "distance" to the super root, and is equal to the sum of the costs of each hop on the path to the root. (Note that the super root broadcasts HELLO PDUs with the cost field set to zero.) The incremental cost of the hop between a node and its parent is primarily a function of the physical link type (i.e. ethernet or radio). The cost component is intended to bias path selection toward high-speed (i.e. wired) connections. On radio links, spanning tree attachment is biased toward the link

with the best signal strength. Signal strength is not a factor in the cumulative path cost. The HELLO "displacement" field specifies the displacement of the actual hello time from the calculated hello time or indicates that the hello time was unscheduled. A well-known randomization algorithm is used to calculate the next hello time. The HELLO "seed" field is used as a seed for the calculation. The "root 802 address" and "root sequence" fields are used to define a single instance of the radio network. Attached nodes must forget their node ID and return to the unattached state whenever a HELLO response PDU is received with a new root 802 address or root sequence number. HELLO response packets can contain other optional parameters such as a load indication, a distributed clock or a pending message list. A load indication parameter can be used to balance the number of terminal nodes between access points.

Nodes without a parent in the spanning tree are in an **unattached** state. In the unattached state, a node learns which attached relay node is closest to the super root by listening to HELLO response PDUs. (If no HELLO response PDUs are received, the node can wait (i.e. sleep) and retry later.) After the learning period expires an unattached node sends an ATTACH request packet to the attached relay node with the lowest cost to the super root. The ATTACH request contains an ATTACH ID, which is a sequence number that is incremented whenever an ATTACH request is generated. (Nodes without a node ID must first send a REGISTRATION request packet to the root to obtain an OWL node ID.) The attached relay node forwards the ATTACH request packet to the super root. The end-to-end ATTACH request functions as a discovery packet and enables relay nodes along the path to the super root to quickly learn the path to the source node. The super root returns the request as an end-to-end ATTACH response PDU. The node which originates an ATTACH request is responsible for retrying the request until a matching response is received, to insure that it is fully attached. When the unattached node receives the ATTACH response PDU it goes to an **attached** state and sets internal **root port** and **parent** variables. The root port is the physical port on which the response PDU arrived and the parent variable contains the node ID and 802 address of the parent node. A child node will only accept outbound unicast MRPDUs from its parent. If the newly attached node is a relay node, it calculates its cost to the super root, by adding its root port link cost to the HELLO cost of its new parent, and begins to broadcast HELLO response PDUs.

ATTACH requests are always forwarded to the super root. Inbound ATTACH requests establish a new path from the super root to the source node. The super root will convert an ATTACH request to either an ATTACH response or to a DETACH response (i.e. if an old path exists). In general, if an ATTACH response is at an AP, and a AP exists which is on the old path to the attaching node, but is not on the new path, then the AP must convert the ATTACH PDU to an outbound DETACH request PDU and forward it on the old path. When the "old parent" AP receives the DETACH request, it will read filter and forward sequence numbers from the request and will enter its filter and forward sequence numbers for the source node into the request. The old parent will then delete its routing table entry for the node which originated the ATTACH request and will return the DETACH request as an inbound DETACH response. The AP which originated the DETACH request will convert the DETACH response to an outbound ATTACH response and will forward it to the attaching node. The old parent AP can (optionally) re-route any undelivered PDUs, destined to the attaching node, as MAC-R R-DATA PDUs. An R-DATA PDU is routed inbound until the new outbound branch is reached. If the new branch has been used, the R-DATA PDU is discarded (i.e. to avoid out-of-sequence delivery); otherwise, the PDU is routed outbound along the new path.

Unattached terminal nodes can optionally broadcast a global HELLO request PDU with a multicast relay node ID and broadcast 802 destination address to solicit unscheduled HELLO response PDUs from attached relay nodes. The net effect is that the unattached state can (optionally) be shortened. (Note that only attached relay nodes respond to request PDUs.) The HELLO request facility is intended for unattached terminals with higher-layer transmit requests in progress.

Each attached node must transmit an ATTACH request PDU at least once per ATTACH_TIMEOUT time period to maintain its path in the radio network. An attached node must also transmit an ATTACH request PDU to its parent whenever it misses MAX_HELLO_LOST consecutive scheduled HELLO response PDUs from its parent and whenever it receives an alert. An alert can occur in an alert PDU or in an

optional alert list in a HELLO response PDU. If a relay node is unable to deliver a PDU to a child node, then the relay node adds the node ID of a child node to its alert node list and, optionally, generates an alert PDU which is sent down all branches of the spanning tree.

Each node (except the super root) should maintain an **in-range list** which contains the node ID and 802 address of potential alternate parent nodes. If a child loses its parent (i.e. due to a MAC-D link error) or detects a better path, then the child can change its path in the spanning tree by selecting the best candidate from the in-range list and attaching to the new parent. Relay nodes must avoid sporadic path changes. If a child loses its parent and the in-range list is empty, it must remain in a quiet learning state until a potential parent is discovered.

MAC-R Routing.

All PDUs are routed along branches of the spanning tree. Relay nodes "learn" the path to outbound nodes by monitoring inbound traffic (i.e. traffic directed toward the root). Whenever a relay node receives an inbound REGISTRATION or ATTACH request PDU from an outbound node, it creates or updates an outbound entry for the source node in its **routing table**. The entry includes the source node's 802 address and the port address of the port which sent the PDU (i.e. the hop address). When a relay node receives a PDU from an inbound node the PDU is forwarded to the outbound hop address which is specified in the routing entry for the 802 destination. The PDU is discarded if a routing entry does not exist. (Note that an AP may also maintain a filtering database which contains entries for inbound nodes and nodes on either the distribution LAN or a secondary LAN. The filtering database is used to facilitate bridge layer flooding.)

As an example, the routing table for relay node 4, in figure 9, is shown in figure 10 below. The **destination** field contains the 802 address of a node in the sub tree rooted at 4. The **first hop** field contains the MAC-D (i.e. port) address of the first hop on the path to the destination. (The node labels from figure 9 are used in lieu of node and port addresses, in this example.) The **child** field indicates if the destination is a child. The **attach ID** field is used to associate ATTACH and DETACH requests and responses. The **port** field specifies the physical port used to communicate with the first hop. The **type** field can be RELAY or TERMINAL. The **status** field is used to mark each entry as BOUND or UNBOUND. UNBOUND entries can be used to route outbound REGISTRATION response PDUs. An UNBOUND entry becomes BOUND when the AP receives an ATTACH response PDU with a matching Attach ID. The super root must also mark each entry which specifies a path through a distributed WDAP as DISTRIBUTED. The **age** field indicates the last time the destination was active and is used to "age" away old table entries. Assume that relay 4 has received an ATTACH request from node 11 through relay 9. Relay 4 adds an entry for destination 11 with the first hop set to 9, the age set to 0.

Destination	Type	Child	First Hop	Attach Time	Attach ID	Port	Status	Age
11	TERMINAL	No	9	1223	4	1	ATTACHED	0
5	TERMINAL	Yes	5	802	2	1	ATTACHED	2
9	RELAY	Yes	9	907	5	1	ATTACHED	1

figure 10.

PDUs from outbound nodes are forwarded to the next inbound node (i.e. the parent) in the branch of the spanning tree. No explicit routing is required for inbound traffic because the route is defined by the structure of the spanning tree. A PDU travels inbound until a node is reached which has an entry in its routing table for the destination 802 address. The PDU is then explicitly routed outbound until it reaches

its destination. Thus, communications between any two nodes is accomplished by routing all traffic through the nearest common ancestor of both the source and destination node. If a PDU reaches a primary or distributed WDAP and an entry for the 802 destination does not exist in the routing table of the WDAP, then the PDU can not be routed outbound (i.e. a common ancestor does not exist). In this case, the WDAP can "bridge" the PDU, as an 802 frame, onto the distribution LAN. Note that a PDU which is bridged onto the distribution LAN by a distributed WDAP, will be bridged back into the OWL subnet (i.e. by another WDAP) if the 802 destination is in the OWL subnet.

As an example, in figure 9, if a PDU is sent from terminal 10 to terminal 5 it will be routed as follows: Terminal 10 will send the PDU to its parent, WMAP 7. Since WMAP 7 does not have an entry in its routing table for terminal 5, it will forward the PDU inbound to its parent, WDAP 2. The MAC-R entity in WDAP 2 does not have an entry in its routing table, so it will forward the PDU to its bridging entity and the PDU will be bridged onto the distribution LAN as an 802 frame. The bridging entity in WDAP 3, the super root, , will forward the frame to its MAC-R entity because it has an entry in its forwarding data base, which specifies the radio network as the subnet for terminal 5. The MAC-R entity in WDAP 3 has an entry in its routing table for terminal 5 and will forward the PDU to the first outbound hop, WDAP 4, over the wired link (i.e. with an 802.3 MAC-D sub layer). WDAP 4 will then deliver the PDU to terminal 5.

As a second example, if remote station 11, in figure 9, sends a PDU to remote station 1 it will be routed as follows: The bridging entity in the secondary WDAP, 9, will determine that station 1 is not on its local 802.3 subnet (i.e. by querying its forwarding database) and will bridge the PDU into the radio network (i.e. by passing the frame to its MAC-R entity). The MAC-R entity in WDAP 9 will forward the PDU inbound to WMAP 4, since it does not have an entry for station 1 in its routing table. WMAP 4 will forward the PDU to WDAP 3. The MAC-R entity, in WDAP 3, does not have an entry for station 1 and will pass the PDU to its bridging entity. The bridging entity will forward the PDU onto the distribution LAN as an 802 frame addressed to station 1.

Dynamic routing changes and PDU retransmission.

Paths in the spanning tree change often as terminals roam. PDU transmission errors due to roaming fit into one of two possible cases: 1) a terminal node is unable to deliver a PDU to its parent AP, or 2) an AP is unable to deliver a PDU to a child terminal.

In the first case, the terminal can simply select a new parent and re-attach to the network by sending an ATTACH request. An attach indication is generated whenever the path to a terminal node changes. The MAC-R entity in a relay node updates its routing table entry for an outbound source node if an inbound ATTACH (or REGISTRATION) request PDU is received from the node and the hop source is not the same as the first hop in the table entry for the node. The first hop field, in the routing table entry, is overlaid by the hop source of the PDU and outbound PDUs are now routed along the new path. (Note that an old disconnected path fragment may still exist in the spanning tree after a new path has been established.) ATTACH requests are always forwarded to the super root.

An ATTACH PDU may be converted to an outbound DETACH request to deleted an old path fragment. The ATTACH ID in a DETACH request is the same as the ID in the associated ATTACH PDU and the destination is the 802 address of the attaching node. If a relay node on the old path has a routing table entry for the destination, with an ATTACH ID that matches the ID in the DETACH request, then the relay node will forward the DETACH request outbound and will delete the entry. The DETACH request is forwarded outbound until it reaches the relay node which was the old parent of the attaching node. It is then converted into a DETACH response and is forwarded inbound until it reaches an AP which is on the new path to the attaching node. The AP on the new path will convert the DETACH response to an ATTACH response and route the ATTACH response outbound to the attaching node. Note that if the ATTACH response is lost, then a DETACH PDU will not be generated on the next ATTACH retry (i.e. because the old path was deleted).

A relay node may not be able to deliver a DATA PDU to a child, for several reasons: 1) the child may be asleep; 2) the channel may be reserved in the child's coverage area; 3) the PDU may be lost due to excessive errors; or 4) the child may have selected a new parent (i.e. due to roaming). It is assumed that most undelivered PDUs are lost because child nodes roam. If a parent relay node can not deliver a PDU to a child node, then (if the routing table entry for the child has not been updated) the parent node will add an alert record for the child node to its internal alert list and send an ALERT request to the super root. The ALERT PDU contains the ATTACH ID from the routing table entry for the child node. When a relay node, on the path to the super root, receives an inbound ALERT request it determines a) if the alert ATTACH ID matches the ATTACH ID in its routing table and b) if the hop source in the ALERT request is the same as the first hop field in the routing table entry for the alert destination. If both conditions are satisfied then the relay node will 1) optionally add the associated alert record to its internal alert list, 2) forward the ALERT request to the next hop on the path to the super root, and 3) optionally forward the ALERT request down each of its outbound branches, other than the one on which it arrived. If either condition is not satisfied then the relay node will, instead, send an outbound ALERT response on the path on which the ALERT request arrived. The ATTACH ID in the ALERT response is the same as the ID in the ALERT request and the destination is the 802 address of the lost child. If a relay node on the old path has a routing table entry for the destination, with an ATTACH ID that matches the ID in the ALERT response, then the relay node will forward the ALERT response outbound and will delete the entry. The ALERT response is forwarded until it reaches the relay node which was the old parent of the lost child.

An ALERT request may reach the super root before the associated child node re-attaches. In this case, the ALERT request is simply discarded.

Outbound ALERT requests are used to quickly notify a lost child that it should re-attach to the network. If a relay node receives an outbound ALERT (i.e. from its parent) request, it first checks to see if it has a routing table entry for the lost child with a "newer" ATTACH ID. If it does, then the ALERT request is simply discarded. Otherwise, a relay node which receives an outbound ALERT request will forward the ALERT request to each child node which is a relay node and will multicast the ALERT request (i.e. with a multicast MAC-D destination address) once on each of its radio ports. Each relay node adds the ALERT ID in the request to its internal alert list

Records in a relay node's internal alert list in each relay node are copied into HELLO response PDUs for MAX_HELLO_LOST + 1 scheduled hello times to notify nodes to re-attach, where MAX_HELLO_LOST is the maximum number of HELLO PDUs that can be missed by a child before the child re-attaches. An alert record contains a target node ID, a source node ID, and an ALERT ID (which equates to an ATTACH ID). The concatenated source node ID and ALERT ID are used to uniquely identify each alert occurence. A target node can ignore any any duplicate alert record which is received within MAX_HELLO_LOST+5 HELLO periods.

Note that old path fragments are simply aged away if no outbound PDUs are sent along the path fragment.

A terminal node must set the attach indication (ATTI) bit ON in the MAC-R header of an ATTACH request when it first attaches to a new parent. The ATTI bit indicates that the path to the ATTACH request source node has changed. The MAC-R entity in the AP which was the previous parent of the source node posts an attach indication error to the MAC-Q sub layer when it receives the ATTACH request PDU with the ATTI bit set ON. The MAC-R sub layer in a terminal node posts an attach indication error to the MAC-Q sub layer when it receives the associated ATTACH response with the ATTI bit set ON. An attach indication is a positive indication that a node has just attached to the network and can be used to trigger an immediate (re)transmission. The attach indication includes the 802 source address and receive sequence number for the source node of the ATTACH request. If the MAC-Q entity is holding any undelivered DATA PDUS for the node, it can respond by re-transmitting the undelivered PDUs as R-DATA PDUs. The R-DATA PDUs will be discarded if they are duplicates or arrive out-of-sequence. The R-DATA PDUs are automatically routed along the new path.

The MAC-R layer in a terminal node is responsible for retrying a DATA PDU transmission, if the MAC-D layer is unable to deliver the DATA PDU to its parent. The MAC-D layer indicates the success or failure of a transmission. Occasionally, the MAC-D entity will not be able to positively determine success or failure (i.e. if CLEAR frames are missed in a MAC-D conversation). If the MAC-D layer indicates positive failure, then the MAC-R layer can choose a (possibly new) parent, re-attach, and retransmit the DATA PDU; otherwise, the MAC-R layer must discard the PDU. The MAC-Q may retransmit the DATA PDU as an R-DATA PDU when an attach indication is received (i.e. when an ATTACH response is received with the ATTI bit set ON).

Registration.

A node is initially in an unregistered state and returns to the unregistered state under certain error conditions. Each unregistered node in the network must send a REGISTRATION request to the super root before it attaches. The REGISTRATION request is used to obtain a network node ID and is used to validate access to the network. The REGISTRATION request is returned by the super root as a REGISTRATION response. The node which originated the request is responsible for retrying the request until a matching response is received.

Registration logic is similar to attach logic with some key differeneces. REGISTRATION requests can only be sent to the super root when no other inbound PDU for the source node exists in the network. No other PDU types may be sent in the unregistered state. A node goes to the registered state when a matching registration response is received from its parent.

A node's registration is valid as long as it is actively attached to the network. A node returns to the unregistered state if it does not receive an ATTACH response within a MAX_ADDRESS_LIFETIME time period or if it detects that the super root has changed.

Broadcast routing.

PDUs with broadcast (or multicast) 802 destination addresses are (optionally) routed along all branches of the network spanning tree. Broadcast messages are associated with HELLO response PDUs on radio links. A broadcast parameter in a HELLO response PDU indicates that terminals should stay awake for broadcast messages which will immediately follow the HELLO PDU. A secondary WDAP forwards broadcast messages onto its attached wired subnets. If a broadcast message orginates on the distribution LAN, then each primary or distributed WDAP is responsible for bridging it to the OWL sub tree for which it is the access root. Broadcast messages which originate within an OWL subnet are forwarded on each branch of the network spanning tree, except the branch on which the message arrived. The access root of the sub tree in which the broadcast message originated is responsible for bridging the message onto the distribution LAN. The message is (optionally) bridged back into the radio network by each other access root.

As an option, broadcast (or multicast) messages which originate in the radio network are only forwarded to the distribution LAN. A broadcast message which originates in the radio network is forwarded inbound until it is bridged onto the distribution LAN by a primary or distributed WDAP. The message is not relayed back into the radio network by each other primary or distributed WDAP (i.e. as above). To facilitate this option, each distributed WDAP must keep a table of inbound entries. An inbound entry is defined relative to a WDAP and is any OWL node which is not in the subtree rooted at the WDAP. A distributed WDAP adds an entry to its inbound table when it receives a DIST_ATTACH packet. A DIST_ATTACH PDU is generated by the access root, which is responsible for bridging to a node, when the node attaches to the network. It is used to notify transparent bridges and other distributed WDAPs that the node has roamed. A distributed WDAP will not bridge a broadcast packet from the distribution LAN into the radio network if the source address belongs to a node in its inbound table. The primary WDAP will not bridge a broadcast packet from the distribution LAN into the radio network if the source node belongs to a node in its route table.

Sleeping Terminal Support.

The MAC-R sub layer provides several facilities to support sleeping terminals. A sleeping node initially "synchronizes" on a HELLO response PDU from its parent. The node can calculate the time of the next expected HELLO response PDU from its parent and can power-down with an active timer interrupt set to wake it just before the next HELLO response PDU is transmitted. The MAC-R entity in a parent node can store a message for a sleeping node until the node "requests" the message by notifying its parent that it is awake. A terminal learns that it must request unsolicited saved messages by examining a pending message list in the HELLO response PDU. This implementation enables sleeping terminals to receive unsolicited messages and relaxes the timing constraints for transaction oriented messages. ATTACH and DATA request PDUs can contain several MAC-R parameters which are used to enable pending messages. A "delivery service type" parameter, indicates that a terminal (i.e. which sent the request) is sleeping. An "awake time window" parameter is used to specify an awake time period. An "awake time offset" parameter is used to specify the start of the awake time window. (The awake time window is effective immediately if an awake time offset is not specified.) An "auto awake" delivery service type can be used to implicitly set an awake time window each time the parent node receives a message from the sleeping terminal. A "maximum stored message count" field specifies the maximum number of HELLO times that a message should be stored in the parent relay node. The MAC-R entity in a parent node will store pending messages until 1) the message is delivered, or 2) "maximum stored message count" hello times have expired.

Broadcast messages are associated with HELLO PDUs so that sleeping terminals will be awake when the broadcast message is transmitted.

WDAP bridging.

A WDAP maintains a forwarding data base with an entry for each known network node. Each entry contains an 802 destination address and an associated subnet identifier. When a PDU arrives at the bridging entity in a WDAP, the forwarding database is searched to determine the subnet of the 802 destination. If the destination is found and the destination is on another subnet (i.e. other than the one on which the PDU arrived) then the PDU is bridged to the subnet of the destination. If the destination is not found, then the action taken by the bridging entity is dependent on the configuration of the WDAP. 1) The PDU can be forwarded to every subnet except the subnet on which it arrived (i.e. flooding), or 2) the PDU can be discarded.

Typically a primary or distributed WDAP is configured to only forward unicast frames from the distribution LAN to the OWL subnet if an entry exists in its MAC-R routing table for the 802 destination. This implies that the MAC-R entity must notify the bridging entity that a destination exists in the radio subnet, when a MAC-R routing table entry is created, so that the bridging entity can update its forwarding database. Likewise, the bridging entity must be notified when a routing table entry is deleted. The forwarding database in a distributed WDAP contains entries for each node in its access spanning tree. The forwarding database in the primary WDAP contains entries for all nodes in the OWL subnet which are not in an access sub tree rooted by a distributed WDAP.

Flooding options.

The user can set unicast and/or multicast flooding options for the distribution system and for each secondary LAN. Distribution flooding options are configured on the primary WDAP and are distributed in REGISTRATION response packets. Therefore, distribution flooding options should be configured for each AP with a non-zero root priority. Multicast and/or unicast flooding should be enabled for each secondary LAN which requires multicast or unicast flooding, respectively. Therefore, multicast and

unicast flooding should be configured on each AP which can be the designated secondary WDAP for a secondary LAN.

If distribution unicast flooding is set to level 1 (the default) then unicast frames which originate on the distribution LAN are discarded if the destination is unknown. Unicast frames which originate in the radio network are forwarded inbound, until the packet arrives at an AP which a) has a route entry for the destination or b) is a primary or distributed WDAP. A primary or distributed WDAP will relay an inbound unicast frame onto the distribution LAN, if the destination is unknown.

If distribution unicast flooding is set to level 2, then MAC-R will flood unicast frames, for which the destination is unkown, to the distribution LAN and to each secondary LAN which has unicast flooding enabled. For example, a unicast frame which originates in the radio network is forwarded to the distribution LAN and to each secondary LAN which has unicast flooding enabled.

If distribution multicast flooding is set to level 1, then multicast frames which originate on the distribution LAN are forwarded to secondary LANs which have multicast flooding enabled. Multicast frames which originate in the radio network or on a secondary LAN are forwarded to the distribution LAN.

If distribution multicast flooding is set to level 2 (the default) then multicast frames which originate on the distribution LAN are flooded throughout the OWL network. Multicast frames which originate in the radio network or on a secondary LAN are forwarded to the distribution LAN.

If distribution multicast flooding is set to level 3 then all multicast frames are flooded throughout the OWL network. For example, if a multicast frame originates in the radio network then it will be forwarded to the distribution LAN and each AP in the OWL network. An AP will broadcast the message on each of its radio ports and will relay the message to any attached secondary LAN.

Frame filters.

A frame filter can be used to filter multicast frames and/or specified frame types. The user can enter a unicast and/or multicast "frame filter expression" for each AP port which is attached to a remote subnet. Each received frame and an expression pointer are passed to fltr_is_enabled. TRUE is returned if the frame is enabled; otherwise, FALSE is returned. If the frame is enabled, it will be bridged into the radio network; otherwise, it will be discarded.

Bridging to a remote subnet.

A remote 802 subnet is bridged to the OWL radio network through a secondary WDAP. The secondary WDAP is responsible for attaching the remote subnet and its remote stations to the radio network. Remote stations can be attached in two ways: 1) The AP which is the designated bridge for the secondary subnet (i.e. the secondary WDAP) can include a remote attach list in its ATTACH request packets. 2) The path to a remote station is automatically established whenever the station sends a DATA packet inbound. A remote attach request is generated whenever an AP receives an inbound DATA packet from a remote station and the source is not in the AP's MAC-R routing table. Data can be piggybacked on a remote attach request. A remote attach request is converted to a remote detach request, if the path to remote node changes.

The bridging layer in a secondary WDAP maintains a list of remote stations which exist on the remote subnet (i.e. a forwarding database). The WDAP port attached to the remote subnet always operates in promiscuous mode. If a frame is received and the destination is not in the port's station list, then the frame is forwarded to the MAC-R layer. The MAC-R layer will either forward the frame outbound, if an entry for the destination exists in its routing table, or will forward the frame inbound to its parent.

A secondary WDAP negotiation protocol is used to select a single designated bridge, if more than one secondary WDAP is connected to a wired secondary LAN. The designated bridge is solely responsible for bridging between its secondary LAN and the radio network.

Optimization considerations.

If a primary or distributed WDAP has two subnets - a distribution LAN and the OWL subnet - and the WDAP is configured to allow flooding onto the distribution LAN and to not allow flooding onto the OWL subnet, then each entry in its forwarding database corresponds to an entry in its MAC-R routing table. All frames which are passed to the bridging entity from the MAC-R entity (i.e. from the OWL subnet), are forwarded to the distribution LAN. Frames will only be forwarded from the distribution LAN to the OWL subnet if an entry exists in the MAC-R routing table. For any configuration, entries in the forwarding database which are associated with the OWL subnet correspond to entries in the MAC-R routing table. A shared forwarding database/MAC-R routing table data structure could be used to optimize the learning process required for bridging and to avoid two lookups (i.e. a forwarding database lookup and a MAC-R routing table lookup) each time a PDU is forwarded from the distribution LAN into the OWL subnet.

MAC-Q Sub layer.

The (optional) MAC-Q can be viewed as an end-to-end reliability layer between entry points to the radio network. The MAC-Q sub layer is responsible for delivering received PDUs to the next higher layer in the order in which the PDUs entered the radio network. The MAC-Q sub layer also retransmits lost MQPDUs, and filters any resulting duplicate or out-of-sequence MQPDUs. The MAC-Q sub layer is intended to significantly reduce the number of lost PDUs due to "roaming" terminals, without introducing duplicate or out-of-sequence PDUs. It does not guarantee that PDUs will never be lost. MAC-Q entities exist at entry points to the radio network. The MAC-Q entity in an AP provides a proxy MAC-Q layer for nodes in the OWL network which are not in the radio network.

MQPDUs contain a MQPDUID in the MQPDU header. The concatenation of the MQPDUID and 802 source and destination addresses uniquely identifies an MQPDU in an OWL radio network. The MQPDUID is generated by the MAC-Q entity in a WDAP or terminal when a frame first enters the OWL radio network.

A primary or distributed WDAP maintains an "MQPDU table" with entries for each outbound node. Each entry contains the 802 address of an outbound node and an associated forward MQPDUID and filter MQPDUID. Forward MQPDUIDs are generated to uniquely identify an MQPDU for its lifetime in the OWL network. Filter MQPDUIDs are used to detect duplicate and out-of-sequence PDUs. Before a primary or distributed WDAP forwards an 802 frame from a wired backbone into the OWL radio network, it increments the forward MQPDUID, associated with the destination 802 address, and enters the it into the MQPDU header. The MQPDU is then passed to the MAC-R sub layer for transmission. Note that this approach assumes that remote stations do not move quickly from subnet to subnet. If a node is physically attached to two subnets, then a unique 802 address should be used for each subnet.

Terminal nodes maintain an MQPDU table with an entry for each active remote MAC-Q network entry point. Each entry contains a filter MQPDUID, a subnet identifier, and an 802 address. Subnet 0 is always the radio network and subnet 1 is the distribution LAN. Other subnet identifiers can be assigned to a secondary WDAP. The 802 address is blank for subnets 1 and higher. Note that there can be multiple entries for subnet 0, but only 1 entry for each other subnet. A terminal also maintains a single forward MQPDUID variable and stores up to one MQPDU for possible retransmission. The value of the forward variable is incremented and entered into the MQPDU header whenever a terminal prepares a new PDU for transmission. The terminal MAC-Q entity retransmits an MQPDU whenever the MAC-R layer returns a transmit error (until a maximum retry count is exceeded).

The filter MQPDUID, in an MQPDU table, is the ID of the last MQPDU received from the associated 802 address. Duplicate MQPDUs are discarded. An MQPDU is accepted by a sink if 1) a retry bit in the MAC-Q header is set OFF or if 2) the MQPDUID in the PDU is not in a "duplicate range" defined by the filter MQPDUID in the table. If an MQPDU table filter entry does not exist for an 802 source address, then data PDUs from the source should be discarded if the retry bit is set ON. The entries in the MQPDU table must be aged so that a filter MQPDUID (and stored MQPDU) is never older than the "roll over" time of an MQPDUID.

An entry in an MQPDU table in a distributed WDAP may be transferred to another primary or distributed WDAP if a terminal "roams". If a terminal moves and its new path to the super root is through another WDAP, then the forward and filter MQPDUIDs for the terminal must be transferred from the old WDAP to the new WDAP. The super root obtains the information (if it exists) from the old WDAP and forwards it to the new WDAP. Note that the new WDAP can accept MQPDUs with the retry bit set OFF while waiting for an MQPDU table entry to be transferred.

Ideally, each MAC-Q entity in the radio network should be notified when the terminal node associated with an entry in its forward list has roamed and re-attached. If a MAC-Q entity holds an undelivered PDU, destined for the re-attached terminal, then the PDU can be retransmitted along the new path to the terminal. A more practical approach would be to notify each MAC-Q entity which has recently transmitted a PDU to the terminal. If it is assumed that most traffic is not contained in the radio network, but rather is directed to or from the distribution LAN, then it may be practical to simply notify the MAC-Q entities in primary or distributed WDAPs on the old path to the terminal.

MAC-S Sub Layer.

The (optional) MAC-S sub layer provides data compression and security services.

Network management tools can be used to create security associations between any two stations in an 802 LAN which contains an owl subnet. MAC-S entities exist in WDAP's. A MAC-S entity can encipher a frame when it enters the radio network if a security association exists between the source and destination stations at the entry WDAP. A MAC-S entity, in an exit WDAP, can correctly decipher a frame as it exits the radio network if it contains a corresponding security association. Network management access to a MAC-S entity in a distributed WDAP is always through a primary WDAP. The primary WDAP (i.e. the super root) "knows" the path to all outbound nodes. A MAC-S entity in a primary or secondary WDAP provides a "proxy" MAC-S layer for security associations involving remote stations on wired subnets.

A global security association can be used to consistently encipher and decipher each frame as it, respectively, enters and exits the radio network. Global association must be enabled at the MAC-S entity in each primary, secondary, and terminal node in the OWL subnet.

Simple compression (i.e. independent of any security encryption) is enabled by a single compression bit in the MAC-S header.

APPENDIX B

OPEN WIRELESS LAN THEORY OF OPERATION

PREFACE

Purpose of This Book

This book describes the open wireless local area network (LAN) by Norand and the LAN's wireless infrastructure. This book is function-oriented; that is, it describes the network, its components, and how it operates. User guides and other publications for specific open wireless LAN products contain step-by-step design and installation procedures. The end of this section lists related publications.

Intended Audience

The main audience for this book is the system administrator responsible for implementing the site's NORAND® open wireless LAN. This book is intended for sites with established NORAND systems and for sites installing a NORAND system for the first time.

Anyone considering the purchase or installation of the NORAND open wireless LAN will benefit from the information in this book because it describes how the open system provides wireless connectivity solutions. This book will help the network hardware installer because it shows how NORAND open wireless LAN products connect to Ethernet media.

Organization

This book has the following sections and appendixes. The appendixes contain supplemental information about the open system.

Section 1, "Network Terminology"

Section 1 defines some network terms.

Section 2, "The Wireless Infrastructure"

Section 2 provides an overview of the NORAND open wireless LAN's wireless infrastructure by describing how the infrastructure is the mechanism for data transfer between the Ethernet medium and wireless end-user stations.

Section 3, "Wireless Infrastructure Operation"

Section 3 describes how the wireless infrastructure configures itself and operates through its network spanning tree. The section also covers related operations such as frame forwarding, flooding, filtering, and roaming.

Section 4, "Network Configurations"

Section 4 shows examples of network configurations with open wireless LAN network devices.

Section 5, "Network Connectivity"

Wireless network interface cards (NICs) and PEN*KEY® computers by Norand provide a range of network connectivity solutions. Section 5 describes these network products and the solutions they provide.

Section 6, "Host Connectivity"

Section 6 provides an overview of NORAND host connectivity devices and terminal emulation stations. It also describes the terminal emulation protocol stack.

Section 7, "Wireless Access Points"

Section 7 covers wireless access points and the solutions they provide.

Section 8, "Installation"

Section 8 provides useful overviews of design and installation strategies for the open wireless LAN. This section also shows how wireless infrastructure and host connectivity devices connect to Ethernet media.

Section 9, "System Management"

Section 9 covers these system software management tasks: configuring system software through local and remote sessions, downloading the latest version of system software, and querying devices for status information through Simple Network Management Protocol (SNMP).

Appendix A, "Radio Options"

Appendix A describes radio options for the wireless infrastructure and includes radio specifications, international frequencies, and data rates.

Appendix B, "Recommended Network Products"

Appendix B lists network communication products Norand recommends for use with the open system.

Appendix C, "ODI and NDIS Driver Configurations"

Appendix C has examples of Open Data-Link Interface (ODI) and Network Device Interface Specification (NDIS) driver configurations for the PEN*KEY 6100 Computer.

Appendix D, "6710 Access Point Specifications"

Appendix D covers electrical, mechanical, and physical specifications for the NORAND 6710 Access Point.

Appendix E, "Host Connectivity Device Specifications"

Appendix E covers electrical, mechanical, and physical specifications for the NORAND RC4030E Gateway and 6950 Enterprise Gateway Server.

Related NORAND Publications

For more information about specific NORAND open wireless LAN products, refer to the following publications. Numbers in parentheses indicate the Norand part number (NPN) of the publication.

▶ NOTE: *We welcome your comments about this Open Wireless LAN Theory of Operation and our other publications. Please write your comments on the Reader's Comments card included with the publication and then drop the card in the mail.*

Access Point User Guide

6710 Access Point User's Guide (NPN: 961-047-081)

The user guide for the 6710 Access Point describes how to install, configure, and troubleshoot the access point.

PEN*KEY Computer User Guides

PEN*KEY computer user guides describe how to set up, operate, and maintain PEN*KEY computers. Specific user guides are:

*PEN*KEY 6100 Computer User's Guide (NPN: 961-028-085)*

*PEN*KEY 6400 Computer User's Guide (NPN: 961-028-093)*

*PEN*KEY 6600 Computer User's Guide (NPN: 961-028-084)*

PEN*KEY Computer Programmer Guides

PEN*KEY computer programmer guides contain information about windows applications, power management, system and device support, and system messages for PEN*KEY computers. Programmer guides also cover tool kits. Specific programmer guides are:

*PEN*KEY Model 6100 Computer Programmer's Reference Guide (NPN: 977-054-001)*

*PEN*KEY Model 6200/6300 Computer Programmer's Reference Guide (NPN: 977-054-003)*

*PEN*KEY Model 6600 Computer Programmer's Reference Guide (NPN: 977-054-002)*

Host Connectivity Device User Guides

6910 Integrated Gateway/Access Point User's Guide (NPN: 961-047-095)

The 6910 Integrated Gateway/Access Point combines the host functionality of the RC4030E Gateway and access point functionality of the 6710 Access Point. The user guide describes how to install, configure, and troubleshoot the gateway/access point.

6950 Enterprise Gateway Server User's Guide (NPN: 961-047-091)

The user guide for the 6950 Enterprise Gateway Server describes how to install and configure the gateway server.

RC4030E Gateway User's Guide (NPN: 961-047-087)

The user guide for the RC4030E Gateway describes how to install, configure, and troubleshoot the gateway.

Wireless Network Access Server User's Guide (NPN: 961-051-006)

This user guide describes how to configure the Wireless Network Access Server software, which runs on a host platform.

Terminal Emulation Station User Guides

Terminal emulation station user guides describe how to set up, operate, and maintain radio terminals in each series of terminal. Specific user guides are:

1100 Series User's Guide (NPN: 961-047-069)

*PEN*KEY 6400 User's Guide (NPN: 961-028-093)*

RT1700 Radio Terminal User's Guide (NPN: 961-047-068)

RT5980 Radio Terminal User's Guide (NPN: 961-047-092)

Development Kit Manuals

▶ NOTE: *See also PEN*KEY Computer Programmer Guides.*

Application Developer's Kit Reference Manual Volume A (NPN: 977-051-004) and Volume B (NPN: 977-051-005)

These two volumes cover the commands programmers can use to write various applications for NORAND terminal emulation stations.

Terminal Emulation Programmer Guides

3270 Terminal Emulation Programmer's Reference Guide (NPN: 977-047-040)

This guide describes how terminal emulation stations emulate IBM 3278 Model 2 terminal operation through the 3270 data stream. This guide also covers asynchronous controller commands, and terminal emulation station commands and orders.

5250 Terminal Emulation Programmer's Reference Guide (NPN: 977-047-039)

This guide describes how terminal emulation stations emulate IBM 5291 Display Station operation through the 5250 data stream. This guide also covers 5250 display data stream commands.

Native Terminal Emulation Asynchronous Programmer's Reference Guide (NPN: 977-047-038)

This guide describes components in the radio network using asynchronous NORAND Native communications. This guide also contains commands and orders terminal emulation stations can accept from a host.

VT220/ANSI Terminal Emulation Programmer's Reference Guide (NPN: 977-047-037)

This guide describes how terminal emulation stations emulate VT220 terminal operation. This guide also describes VT220 received codes, transmitted codes, and character sets.

System Management Publications

NORAND Management Information Base Reference Manual (NPN: 977-051-002)

This manual describes the private NORAND Management Information Base (MIB) for the 6710 Access Point, RC4030E Gateway, and 6910 Integrated Gateway/Access Point.

NORAND Open Wireless LAN with HP OpenView for Windows User's Guide (NPN: 961-051-009)

This guide describes how to install and use the OpenView for Windows network management platform by Hewlett-Packard (HP).

OWLView for HP OpenView for Windows User's Guide (NPN: 961-051-010)

This guide describes how to install and use the OWLView for HP OpenView for Windows network management platform.

Related Standards

Related standards include ANSI/IEEE standards and Request For Comments (RFC) documents.

ANSI/IEEE Std 802.3 (ISO/IEC 8802-3)

The Local and Metropolitan Area Network standard specifies the media access control characteristics for the Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access method. The standard also specifies the media, Medium Attachment Unit (MAU), and physical layer repeater unit for 10 MB/s baseband and broadband systems. Specifications for 10BASE2, 10BASE5, and 10BASE-T are included.

ANSI/IEEE Std 802.11 [DRAFT]

This draft standard specifies the rules of interoperability for wireless network components. The 802.11 standard was still in development when this book was printed. The standard is scheduled for completion in 1996 and submission to ANSI in 1997.

RFC1155

RFC1155 provides the common definitions for the structure and identification of management information for Transmission Control Protocol/Internet Protocol–based (TCP/IP) internets.

RFC1157

RFC1157 defines a simple protocol (SNMP) by which management information for a network element may be inspected or altered by logically remote users.

RFC1213

RFC1213 describes the management information base (MIB) for network management of TCP/IP–based internets (MIB-II).

Customer Support

The goal of Norand Corporation is 100 percent customer satisfaction. If you need assistance with the open wireless LAN, contact Norand through the Customer Response Center.

In the United States, call:	800-221-9236 *or* 319-369-3533
In Canada, call:	800-633-6149
FAX:	319-369-3453 (ATTN: Customer Response Center)
Mailing address:	Norand Corporation ATTN: Customer Response Center 550 Second Street SE Cedar Rapids, IA 52401

Section 1

Network Terminology

Access Point

Access points provide the following functions:

- A *wired bridge* is an access point that attaches to the network through an Ethernet link and has bridging enabled (through access point configuration menus). A wired bridge converts wireless LAN frames to Ethernet frames, and Ethernet frames to wireless LAN frames. A wired bridge also forwards wireless LAN frames to wireless LAN nodes.
- A *designated bridge* is an access point that bridges frames to and from a secondary Ethernet LAN or secondary Proxim LAN. A designated bridge for a secondary Ethernet LAN attaches to the network through a radio port.
- A *wired access point* is an access point that attaches to the network through an Ethernet link and has bridging disabled (through access point configuration menus).
- A *wireless access point* is an access point that attaches to the network through a radio port. A wireless access point provides a wireless store-and-forward operation with frames transmitted over the wireless media to reach their destination. Note that a wireless access point forwards frames; a *wired bridge* forwards *and* bridges frames.

NOTE: *Section 3, "Wireless Infrastructure Operation," contains detailed examples of each type of access point.*

Backbone

A *backbone* is a main cable running vertically or horizontally in a building to provide wired connectivity to different areas in the building. Lower tiers of subnetworks attach to the backbone through bridges, routers, or other internetwork devices. A backbone is not designed for direct system access. Examples of backbone media include Fiber Distributed Data Interface (FDDI) and Token Ring.

Bridging

In this document, *bridging* refers to the translational bridging process of converting open wireless LAN frames to Ethernet frames, and Ethernet frames to open wireless LAN frames.

Direct Sequence

Direct sequence is a spread spectrum technique by which the transmitted signal is spread over a wide frequency range. In a direct sequence system, the bandwidth is large relative to the data rate.

Distribution LAN

The *distribution LAN* is the Ethernet segment to which the access point super root connects. Distribution LAN is also called *primary LAN*.

Ethernet

In this book, *Ethernet* is a general term indicating both 802.3 and DIX Ethernet (also called Ethernet 2.0).

Forwarding

A frame is forwarded by sending it to the next hop on the path to the final destination. All access points (including wireless access points) forward frames.

Frequency Hopping

Frequency hopping is a spread spectrum technique by which the band is divided into a number of channels and the transmissions hop from channel to channel is in a specified sequence.

Infrastructure

The *infrastructure* is the permanently-installed elements of the open wireless LAN. It provides coverage and connectivity between wireless devices throughout the service area.

LAN (Local Area Network)

A *LAN* is a group of network devices in which each device can communicate through a wired or wireless link. The wired link may be composed of several segments joined by repeaters and bridges. The LAN is characterized by the relatively short distance it is designed to cover, a high speed of operation, and relatively low error rates. The geographic scope of LANs is limited to thousands of feet or closely-spaced building complexes.

Media Access Control (MAC) Sublayer

The *MAC sublayer* is the lower portion of the Data Link layer of the Open Systems Interconnection (OSI) model. Norand has divided the MAC sublayer into MACR and MACD.

Open System

An *open system* comprises protocols and components that meet standards set by industry-accepted governing bodies. The standards ensure that when new protocols and components are introduced into an existing system, the protocols and components will meet the standards and be able to communicate with the existing system. The OSI model is the basis for a system to communicate with any other system. The model is a framework of standards Norand uses to create protocol stacks and applications for their network products.

Primary LAN

The *primary LAN* is the Ethernet segment to which the access point super root connects. Primary LAN is also called *distribution LAN*.

Radio Network

The *radio network* consists of radio-enabled network devices and communication paths. It is a group of fixed-end devices and wireless stations in which each can communicate with at least one other device through either a radio or wired Ethernet link. Secondary Ethernet LANs are part of the radio network; the distribution LAN is not part of the radio network.

Secondary Ethernet LAN

A *secondary Ethernet LAN* is an Ethernet segment that connects to the distribution LAN through a wireless link. A single access point functions as the *designated bridge* for the secondary LAN.

Secondary Proxim LAN

The radio coverage area of the Proxim 2.4 GHz radio option is a *secondary Proxim LAN*. The designated bridge for the secondary Proxim LAN is the access point with the radio.

Segment

In LANs, a *segment* is a length of cable from termination to termination. For example, a 10BASE2 cable segment is the length of cable between the 50-Ohm terminators that attach to each end of the cable. For proper network communications, cable segments must meet ANSI/IEEE standard specifications.

Subnet

A *subnet* is a subset of a network that shares a network address with other subnets but is distinguished by a unique subnet number.

Super Root

The *super root* is a wired bridge that operates as the central control point for network-wide parameters, network registration, and other operations. The super root connects to the distribution LAN through an Ethernet link, and is the root node in the open wireless LAN spanning tree.

Terminal Emulation

Terminal emulation enables a wireless station to communicate with a host system that is set up to communicate only with a specific type of terminal (such as used for the 3270 and 5250 data streams). Terminal emulation causes the terminal emulation station to operate almost exactly as the type of terminal the host is programmed to expect.

Terminal Emulation Stations

Terminal emulation stations are PEN*KEY® 6400 Computers and radio terminals set up for 3270, 5250, NORAND® Native, or VT220 terminal emulation. Radio terminals include models in the RT1100, RT1700, and RT5900 Series.

Wireless Network Interface Card (NIC)

A *wireless NIC* is a connectivity device with an internal antenna or with an attached antenna unit.

Wireless Stations

Wireless stations is an inclusive term that refers to the following:

- PC-compatible computing stations equipped with Type III, Type II, mini-ISA, or ISA NICs. PC-compatible computers are PEN*KEY 6100, 6400, and 6600 Computers by Norand, and third-party laptop, notebook, and desktop computers.
- NORAND terminal emulation stations equipped with internal radios or field-replaceable radio modules. Terminal emulation stations are PEN*KEY 6400 Computers and radio terminal models in the RT1100, RT1700, and RT5900 Series.

Section 2

The Wireless Infrastructure

About This Section

The open wireless LAN is a general purpose wireless infrastructure that conforms to the OSI model. This section provides an overview of the wireless infrastructure, its main components, and data flow through its protocol stack.

What is the Wireless Infrastructure?

The wireless infrastructure provides data transfer between the wired physical medium and wireless computing stations, and may provide a wireless link between wired Ethernet segments. Because the infrastructure operates at the MAC sublayer of the OSI Data Link layer, the infrastructure is transparent to most industry-standard communication protocols, and supports arbitrary protocol stacks above the MAC sublayer. The result is wireless connectivity support for applications such as transaction-oriented client-server computing and file transfers.

In addition to providing wireless connectivity to computing stations, the wireless infrastructure supports portable operation within the wireless environment. This support includes power conservation for battery-operated devices and seamless roaming of wireless stations between access point coverage areas.

Benefits

Following is a summary of wireless infrastructure benefits.

Best-path frame forwarding

Media (radio) independence

MAC-layer bridging

Protocol independence

Management through SNMP

Management through RS-232 serial diagnostic port or remote TELNET session to view and change the system configuration

Electronic software distribution through Trivial File Transport Protocol (TFTP) over the network backbone

Security

Automatic or manual network configuration (redundancy) through a spanning tree

Power management for battery operated stations

Unicast and multicast flooding and filtering options

Roaming

Wireless access points

Wireless point-to-point and multidrop bridging

Service for NORAND® gateways and terminal emulation stations

Direct connection to 10BASE2, 10BASE5, or 10BASE-T Ethernet

Components

The wireless infrastructure consists of access points wired to a physical medium (in most cases Ethernet), wireless access points (optional), and a family of wireless NICs that conform to PC card and ISA bus standards.

Wireless NICs install into these PC-compatible computing stations:

- Portable, hand-held PEN*KEY® computers by Norand.
- Third-party computers equipped with a PC card slot or ISA bus slot. Computers equipped with these slots include portable notebooks and laptops, and desktop stations.

Section 5, "Network Connectivity," describes the NICs and PC-compatible computers for the open wireless LAN.

Norand also offers high-performance network gateways and terminal emulation stations, which interconnect with VT220, 5250, 3270, and NORAND Native host applications. Emulation products use the services of the wireless infrastructure, which lets multiple terminal emulation applications and standard applications share a common infrastructure. Section 6, "Host Connectivity," describes host connectivity solutions, NORAND terminal emulation stations, and the terminal emulation protocol stack.

Integrated support for wireless communications is a feature of many wireless computing stations. Integration provides improved ruggedness and low profile antennas. These products provide the same open systems device driver support as the general purpose wireless NICs.

Ethernet Physical Medium

You can install a separate segment of the Ethernet physical medium for the open wireless LAN to specifically support the installation. Or, you can connect access points to the site's existing Ethernet medium to provide a transparent extension to an enterprise network. Section 8, "Installation," shows how NORAND access points and gateways connect to 10BASE2, 10BASE-T, and 10BASE5 Ethernet media.

The wired LAN can comprise several cable segments joined by repeaters and off-the-shelf transparent bridges. Usually, the LAN is in one building or several buildings near each other at the same site. Section 4, "Network Configurations," shows examples of network configurations with LANs in the same building, and in separate buildings connected by a wireless link. Section 8 contains general information about Ethernet segments, repeaters, and off-the-shelf bridges.

Multiple Ethernet LANs can connect to the enterprise network through routers. Section 8 contains general information about routers.

Access Point

The access point is the core of the wireless infrastructure. Figure 2-1 shows current designs of the NORAND 6710 Access Point. Information in this book applies to both designs.

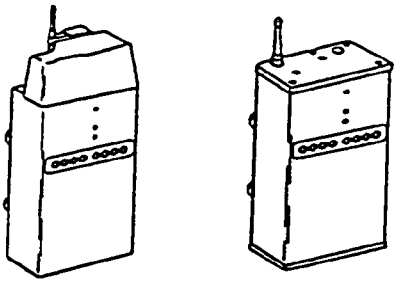


Figure 2-1
6710 Access Points

The access point operates as a protocol-independent bridge by providing transparent, wireless communications for the enterprise LAN. In general, the access point forwards frames from any network node to any other network node on the path through the access point. For example, the access point does the following:

- Forwards frames generated by a wireless station and destined for a host or server on the wired LAN.
- Forwards frames appearing on the wired LAN and destined for a wireless station within the access point's coverage area.
- Forwards frames generated by a wireless station and destined for another wireless station within the same basic service area.
- Forwards frames between wired Ethernet segments across a radio link or links.

Media Independence

The access point accepts a variety of field-replaceable, modular radios through two Type II or Type III PC card-compatible slots for the installation of wireless NICs (Figure 2-2).

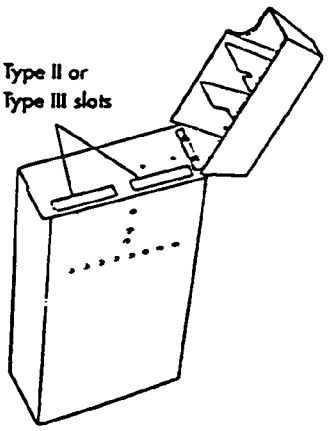


Figure 2-2
Wireless NIC Slots

The NICs enable you to adapt the system to provide different data throughput and coverage tradeoffs depending on required usage. See Appendix A, "Radio Options," for wireless NIC options and specifications.

▶ NOTE: *The access point does not support arbitrary PC cards. Specific PC card software drivers are required.*

Media independence allows you to replace a wireless NIC. Because you can separate the radio from the access point, you do not need to replace the entire installed wireless infrastructure if your network requirements change. You can either upgrade an existing solution or add additional functionality.

Media independence lets you take full advantage of your wireless investment by providing a cost-effective migration path to future wireless networking technologies. Media independence also provides flexibility to support new technology and emerging industry standards (such as 802.11) or alternative defacto standards.

System software loaded in the access point detects the type of radio option installed in the NIC slot. When you configure the access point, the software automatically displays the parameters that match the radio option. If you install a different radio option in the slot, you do not need to load radio-specific system software into the access point.

Radio Operation

900 MHz and synthesized UHF radio options are manufactured by Norand. The 900 MHz radio can operate in the United States, Canada, Australia, and South American countries that allow 900 MHz operations.

The UHF radio can operate in licensed or unlicensed frequency bands throughout the world, subject to national regulations. See Appendix A for country-specific frequencies and data rates.

The 2.4 GHz radio option is a member of the RangeLAN2 family of wireless NICs by Proxim, Inc. The Proxim 2.4 GHz radio can operate in areas that allow use of spread spectrum wireless communications at 2.4 GHz, including Australia and countries in North and South America, Europe, and Asia. In many countries operation without a site license is permitted. Consult a Norand Sales Representative, your local distributor, or a national regulatory agency for details.

Access Point Protocol Stack

The access point operates at the MAC sublayer of the Data Link layer of the OSI protocol model. The MAC sublayer provides services to the Logical Link Control (LLC) sublayer of the Data Link layer. By operating at the MAC sublayer, the access point operates transparently to protocols above the MAC sublayer.

Basic Bridge Operation

A bridge is a device that interconnects LANs. Bridges receive frames sent on each attached network and selectively forward frames between LANs, and use Data Link layer addresses to determine whether to forward each frame. Because bridges operate at the Data Link layer they are not required to examine information from the upper layers, which means they can forward traffic from any network layer protocol.

OSI Model and Access Point Protocol Stack

Figure 2-3 shows the OSI model and access point protocol stack.

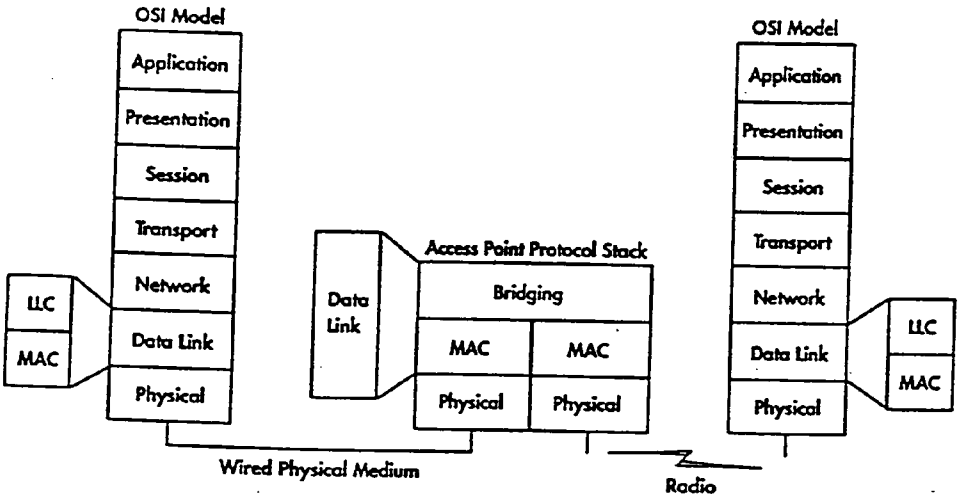


Figure 2-3
OSI Model and Access Point Protocol Stack

Norand divides the MAC sublayer into two functional layers called MACD and MACR.

- MACD is the link protocol, and is responsible for channel access and error free transmission of frames between wireless stations or access points (or both). MACD is media specific and is optimized for the underlying physical medium (wireless radio or wired Ethernet).
- MACR is media independent. It provides facilities for coordination of the infrastructure, roaming of wireless stations between access point coverage areas, and power management to extend the battery life of portable stations.

Figure 2-4 shows the access point protocol stack and communication protocol stacks above the MAC sublayer. The protocol stack above the Bridging layer provides access point management and configuration.

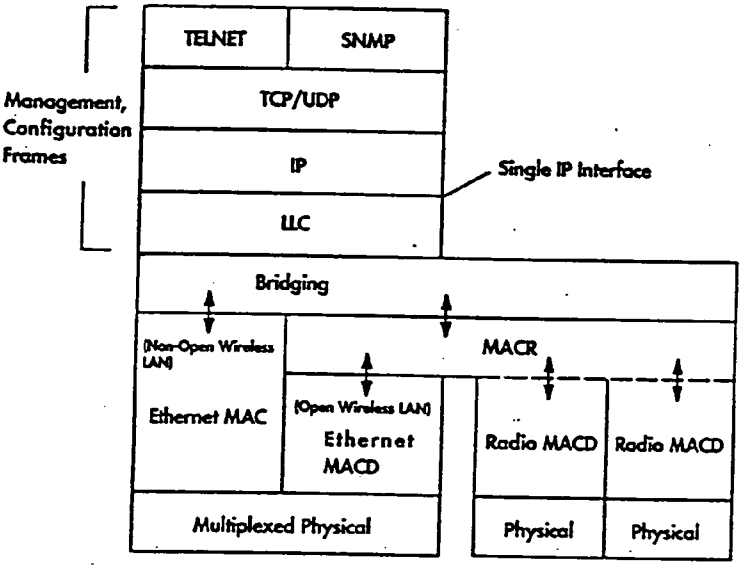


Figure 2-4
Access Point Protocol Stack

Data Flow

Figure 2-5 shows data flow through the access point protocol stack for stations with the 900 MHz, UHF, or Proxim 2.4 GHz radio option.

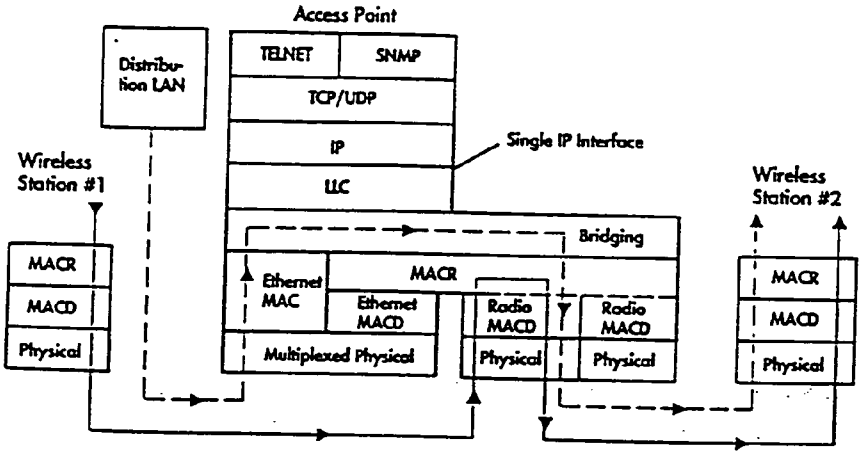


Figure 2-5
Data Flow Through Access Point Protocol Stack

Dashed lines in Figure 2-5 represent the path a frame takes as it travels from a wired station on an Ethernet LAN to a wireless station. The access point *bridges* the frame from the Ethernet LAN to the radio network. (In this document, *bridging* is the translational bridging process of converting open wireless LAN frames to Ethernet frames, and Ethernet frames to open wireless LAN frames.)

Solid lines in Figure 2-5 represent the path a frame takes as it travels from wireless station #1 in the radio network to wireless station #2. Because the path is within the radio network, the access point does not bridge the frame.

Protocols

The wireless infrastructure supports a range of industry-standard communication, management, and configuration protocols, including TCP/IP, NetWare (IPX/SPX), DECnet, and NetBIOS. You can use the standard protocols to transport information from clients (nodes) on the network. This capability does not require knowledge of proprietary tools, application program interfaces, and libraries.

Communication Protocols

The access point is protocol-independent. It uses MAC sublayer addresses to bridge or forward (or bridge *and* forward) frames between stations.

Management and Configuration Protocols

The wireless infrastructure provides management and configuration through a command interpreter and SNMP. You can access the command interpreter through a remote TCP/IP TELNET session or locally through the access point's RS-232 serial diagnostic port for out-of-band management.

SNMP is the most common industry-standard protocol for managing devices on an IP-based network. The resident SNMP agent for NORAND access points and gateways complies with MIB-II standards for information exchange in TCP/IP environments. SNMP-based commands from remote sites are also possible.

TCP/IP is the suite of transport and application protocols that run over IP. The access point and gateway contain an embedded TCP/IP stack. After you assign IP and subnet mask addresses to an access point or a gateway through its configuration menus, you can configure it through a local or remote TELNET session.

You can also update the access point with the latest version of system software through TFTP over the network backbone. Section 9, "System Management," provides more information about TFTP, TELNET, and SNMP.

Section 3

Wireless Infrastructure Operation

About This Section

This section describes how the wireless infrastructure configures itself and operates through a spanning tree. This section also covers related operations including frame forwarding, flooding, filtering, and roaming.

The open wireless LAN by Norand provides extensive system capabilities to resolve unique issues with wireless communications. Because of operational differences among radio options, not all features will provide all of the capabilities discussed in this section. Differences are discussed in the appropriate paragraphs in this section. The open wireless LAN architecture is flexible, allowing new capabilities to be introduced as new wireless media become available.

Wireless Communication Issues

The open wireless LAN resolves a range of unique issues associated with wireless communications, including the following:

- Installation issues and wiring costs
- Radio technology tradeoffs and evolution
- Portable, battery operated wireless stations
- Wired and wireless stations coexisting on the same backbone
- Dynamic radio coverage

Installation Issues and Wiring Costs

A separate segment of the Ethernet physical medium can be installed to specifically support the open wireless LAN installation. Access points and NORAND® gateways can also connect to the site's existing Ethernet medium to provide a transparent extension to an enterprise network.

Access points and gateways are designed to physically connect to 10BASE-T, 10BASE2, and 10BASE5 Ethernet. An access point with the 900 MHz or UHF radio option can operate as a wireless access point, which does not require connection to the Ethernet medium.

Radio Technology Tradeoffs and Evolution

A major issue in radio technology decisions involves range versus speed tradeoffs, with faster radios having reduced communication range. Higher speed radios increase system costs because a 50 percent reduction in the range of a radio results in a need for about four times the number of access points to cover the same area.

Depending on required usage, wireless NICs in the access point enable the system to be adapted to provide different data throughput and coverage tradeoffs. Because the radio separates from the access point, the entire installed wireless infrastructure does not need to be replaced if network requirements change. For example, the following approaches can be employed to meet changing network requirements:

- An additional radio can plug into the access point's second PC card slot to provide a second data channel.
- If new radio technology evolves (such as 802.11), the new radio can plug into the second slot to support incremental growth on the network and also support the existing population on the initial radio.
- If complete migration to new technology is required, only the radio needs to be changed.

In each approach the initial investment in the infrastructure remains intact and the interface to the enterprise network stays the same.

Portable, Battery-Operated Wireless Stations

Because the radio range of a single access point is limited, multiple access points provide coverage over a large area. The access points are installed to provide adjacent areas of coverage, ensuring that as a wireless station moves out of the range of one access point, it roams into the range of another. Figure 3-1 shows a wireless station roaming from coverage area B to A.

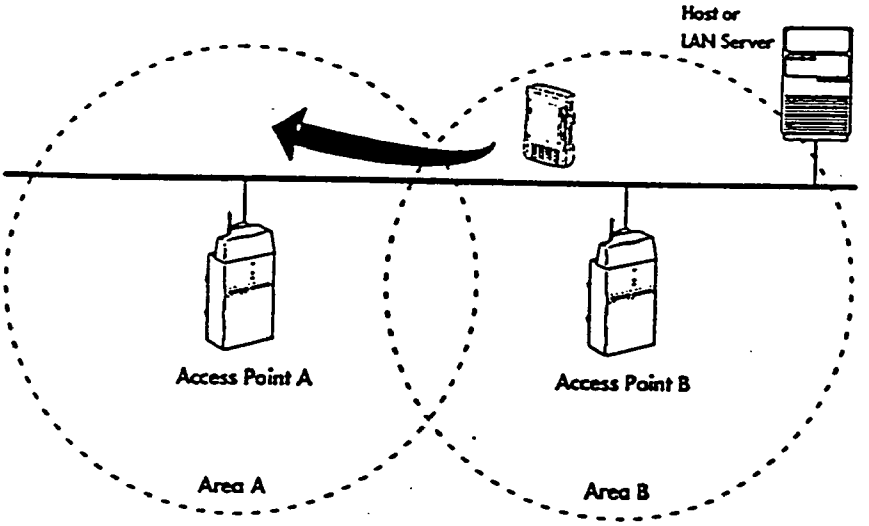


Figure 3-1
Roaming

Areas A and B overlap coverage. When the wireless station leaves the transmission range of access point B, it roams from B to A.

The link with the wireless station is transparently handed from access point to access point without affecting the wireless station's connection to the enterprise LAN. Roaming challenges traditional network design assumptions for the following reasons:

- A roaming wireless station's network address is no longer equal to its physical location.
- Battery-powered mobile computers typically do not maintain continuous connections with the network because advanced power management techniques cycle the radio off when not actively communicating.

Wireless networking software, embedded in the access points, addresses both issues. The access points track the location and status of wireless stations through spanning tree forwarding databases, and manage traffic accordingly. This capability isolates the LAN environment from the issues of mobile devices and transparently integrates the access point based infrastructure into the enterprise network.

Wired and Wireless Stations on Same Backbone

Wired and wireless stations can coexist on the same backbone. An access point operating as a wired bridge provides connectivity to the wireless stations by bridging radio traffic from these devices onto the wired enterprise LAN. The wired bridge converts open wireless LAN frames to Ethernet frames, and Ethernet frames to open wireless LAN frames.

Unicast or multicast flooding options (or both) can be configured for Ethernet LANs through the access point's configuration menus. Programmable and DIX Ethernet filtering options can also be defined to selectively discard Ethernet frames the access point receives on its Ethernet port.

EXAMPLE: In the sample configuration in Figure 3-2, the PEN*KEY® computer and notebook can communicate with the fixed desktop station. To the host and server, the wireless stations appear to be hard-wired to the LAN.

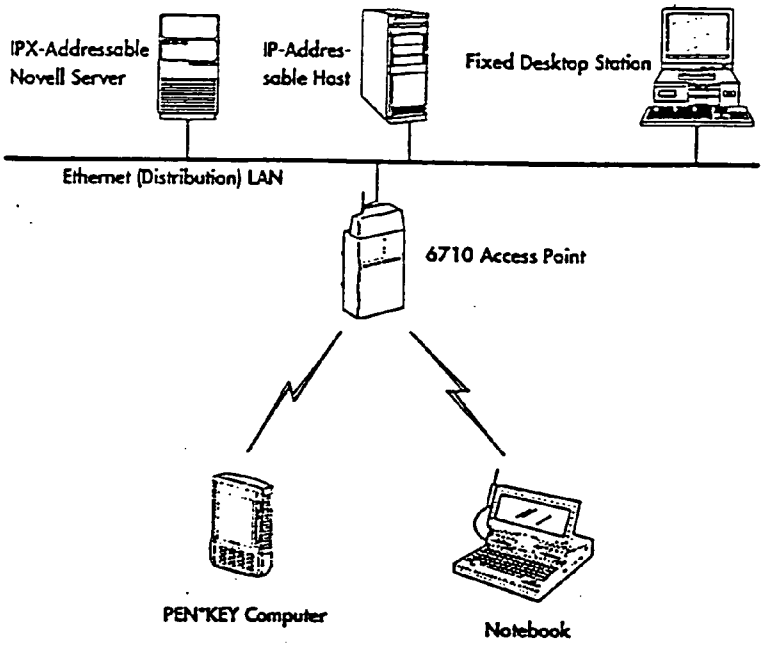


Figure 3-2
Wired and Wireless Stations on Same Backbone

Dynamic Radio Coverage

When access points are powered on, they begin communicating with each other to facilitate the best communications from a wireless station to a server or host. The wireless access point community sets up a hierarchy called a spanning tree, a distributed data structure that optimizes forwarding of messages to wireless stations.

The open wireless LAN spanning tree dynamically organizes the network into a loop-free structure for efficient forwarding of messages. For connectivity, there must be at least one physical path (Ethernet or radio) to each node. If there are multiple possible paths between nodes, the network autoconfigures so that the most efficient link is used. If a link is lost the network dynamically reconfigures to provide an alternative path.

Figure 3-3 shows the physical links in a sample network configuration.

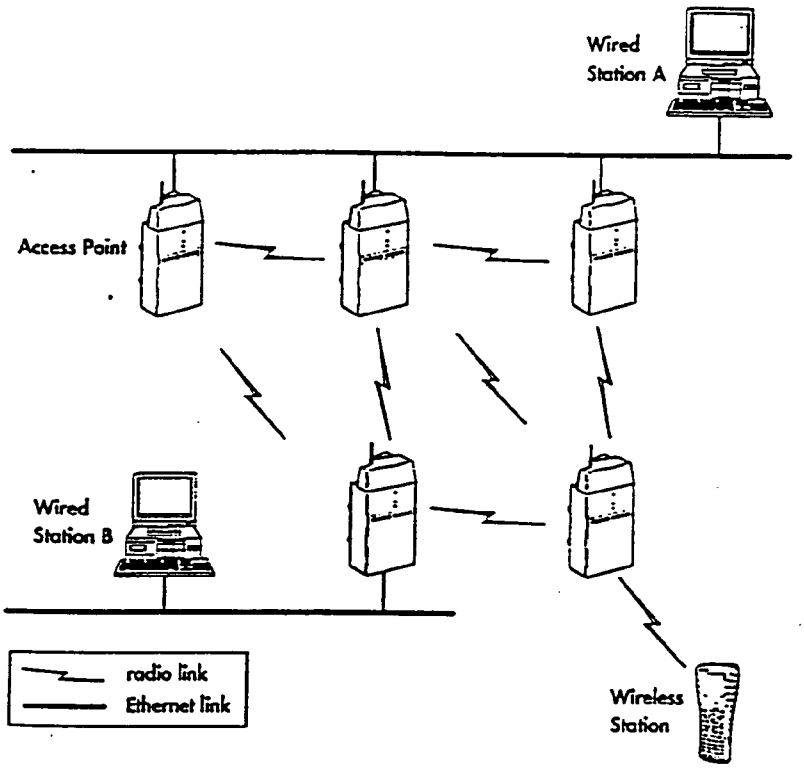


Figure 3-3
Physical Links in Sample Network Configuration

In Figure 3-3, wired station A is not part of the network spanning tree. Wired station B and the Ethernet link it is on can be viewed as part of the spanning tree.

A branch in the spanning tree is a logical link; open wireless LAN frames are forwarded along the branches. Figure 3-4 shows the network in Figure 3-3 organized as a logical network spanning tree. Note that the spanning tree eliminates the loops in the physical topology and uses the most efficient link.

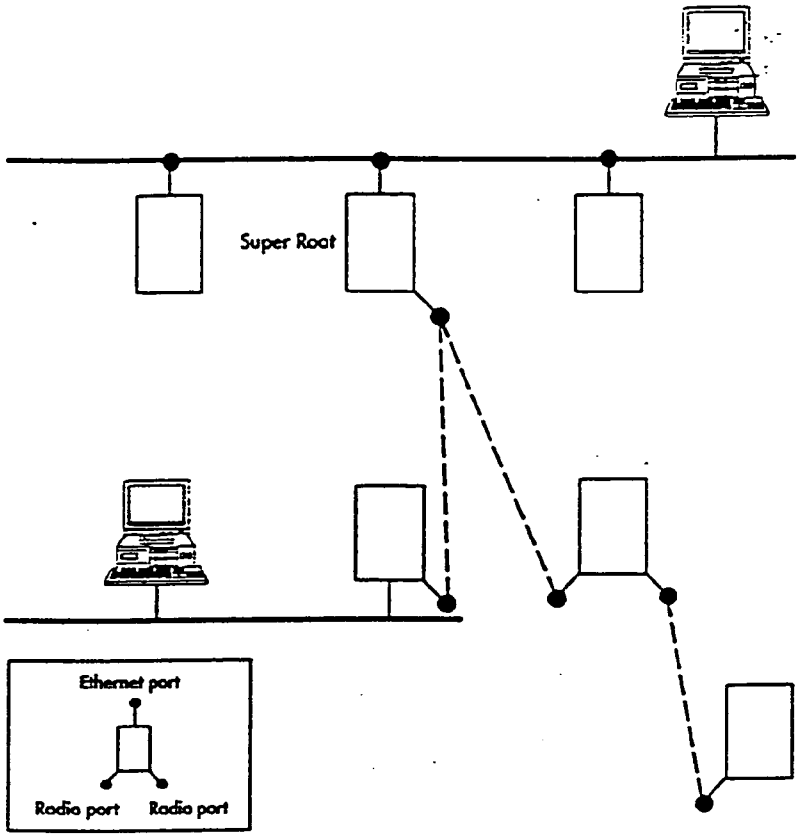


Figure 3-4
Logical Network Spanning Tree

The spanning tree is organized automatically among the nodes of the wireless community. Bridges that are 802.1d compliant also employ a spanning tree architecture. The open wireless LAN spanning tree is designed to overlay 802.1d spanning trees to provide correct operation of wireless nodes in a bridging environment.

Spanning Tree

The open wireless LAN spanning tree can consist of these nodes:

- Multiple access points operating as *wired bridges* and *wireless access points* along the branches of the tree.
- One wired bridge operating as the *super root* of the tree.
- One access point operating as the *designated bridge* for each secondary Ethernet LAN.
- *Wired stations* and *wireless stations* as leaves on the tree.

Figure 3-5 shows a sample network configuration with spanning tree nodes.

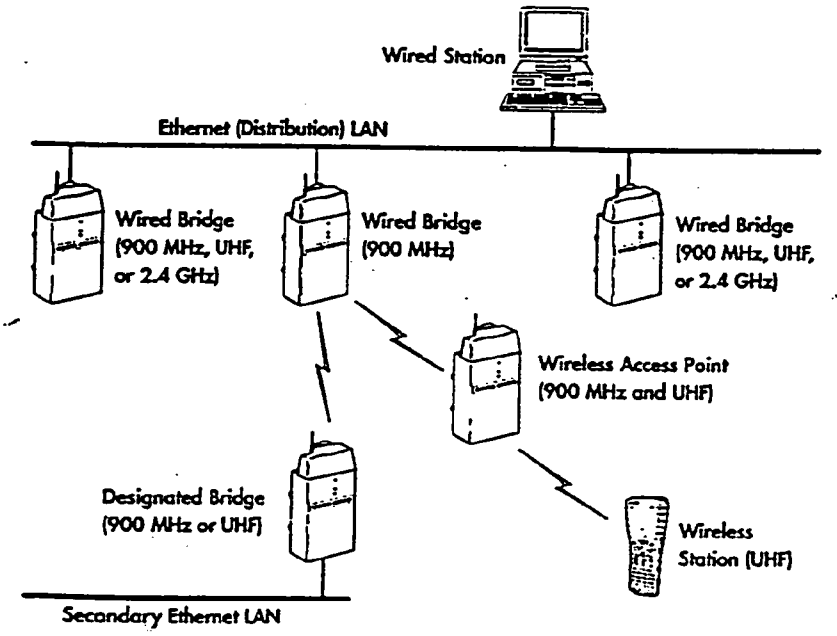


Figure 3-5
Sample Network Configuration

An open wireless LAN node logically attaches to its *parent* through the node's *root port.*

▶ NOTE: *Open wireless LAN nodes are NORAND access points and any wireless station that connects to the network through a NORAND 900 MHz or UHF radio.*

In Figure 3-5 the root port for the wired bridges is the *Ethernet port.* The root port for the wireless access point, designated bridge, and wireless station is the *radio port.* Figure 3-4 on page 3-7 illustrates the port concept.

Wired Bridges

A *wired bridge* is an access point that converts (bridges) an open wireless LAN frame to an Ethernet frame, and an Ethernet frame to an open wireless LAN frame. The wired bridge also forwards open wireless LAN frames to open wireless LAN nodes.

Wireless Access Points

An access point that does not physically connect to the Ethernet medium is a *wireless access point.* It forwards frames through its parent or another wireless access point, or through an access point on the distribution LAN. The parent is the access point to which the wireless access point logically attaches.

▶ NOTE: *The distribution LAN is the physical segment to which the super root physically connects. Usually, the distribution LAN is also the segment to which the primary Ethernet host, LAN server, or NORAND gateway connects.*

Super Root

The open wireless LAN spanning tree must have a root, which is the *super root.* Similarly, each access point is the root of the subtree below the access point. The super root is a wired bridge that operates as the central control point for network-wide parameters, network registration, and other operations.

Super Root Selection

If a network has one super root candidate, that candidate becomes the super root on the distribution LAN.

If a network has two or more super root candidates, the candidate with the highest *root priority* automatically becomes the super root. The range of root priorities is "0" (zero) through "7," where "7" is the highest priority. An access point with priority "0" cannot become a super root.

If two or more access points have the same root priority, the access point with the highest Ethernet address becomes the super root.

EXAMPLE: Figure 3-6 shows an example of the super root selection process. Access points A and B have root priority "3." Access point C has root priority "5." Each access point has a unique Ethernet address.

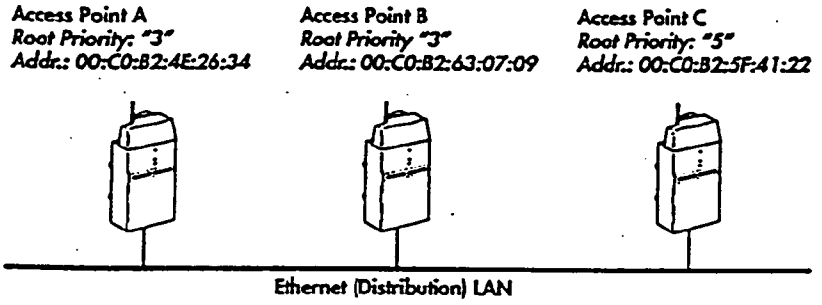


Figure 3-6
Super Root Node Selection Example

Access point C is the super root because it has the highest root priority. If access point C went offline, access points A and B would become super root candidates. However, because B's address is higher than A's, B would become the new super root.

Norand assigns a default value of "1" to the root priority. You can change this value through the access point's configuration menus to achieve a more efficient configuration for your site.

Directing Super Root Selection

Multiple root priority levels prioritize super root selection, which you specify and which is based on location. To direct the selection in networks with more than one super root candidate, you set the root priority of the preferred access point higher than the root priorities of the other access points. When two or more access points are operating in a single network, they constantly communicate to ensure only one super root exists.

Root priority is important if the super root goes offline or is disconnected from the distribution LAN. In this case the spanning tree autoconfigures and selects a new super root. How quickly it makes the selection depends on network size.

Typically, a network is operational within 2 to 3 minutes of when the super root goes offline. Multiple root node candidates allow for *redundancy*, which is the ability of a duplicate device to take over the function of another device.

If you install additional access points, the one with a root priority higher than the current super root's root priority immediately becomes the new super root when it attaches (associates) with the network. A network typically resumes operation within 2 to 3 minutes of when a new super root node is introduced. If several super root candidates have the same root priority, time to operation may increase while the system searches for the candidate with the highest Ethernet address.

Speeding Super Root Selection

Norand recommends that you set the root priority to a nonzero value for two super root candidates on small networks and two or three candidates on large networks. For example, on a small network two access points could have a root priority of "5"; other access points should have a root priority of "0." For convenience, you should assign the highest root priority to the access point that is the most physically accessible. You should also configure network-wide parameters (for example, flooding levels) consistently on all root node candidates.

Designated Bridge

Access points physically connected to a secondary physical LAN, and within the radio coverage area of an access point on the distribution LAN, are candidates to become the *designated bridge* for the secondary LAN. The designated bridge is a particular access point assigned the role of bridging frames destined for or received from the secondary LAN, providing a wireless connection between two unconnected secondary LAN segments.

The secondary LAN can be in the same building as the distribution LAN or in a separate building. Figure 3-5 on page 3-8 shows a single secondary Ethernet LAN.

Currently, an access point with the 900 MHz or UHF radio option can be a designated bridge. For a 2.4 GHz solution, two interbuilding bridges provide similar point-to-point capability by linking Ethernet LANs.

See Section 4, "Network Configurations," for an example of a configuration with interbuilding and intrabuilding bridges. Appendix B, "Recommended Network Products," lists the interbuilding bridge Norand suggests for use with the open system.

Designated Bridge Selection

If a network has one designated bridge candidate, that candidate becomes the designated bridge for the secondary LAN. If a network has two or more candidates, the candidate with the highest *bridge priority* automatically becomes the designated bridge. The range of bridge priorities is "0" through "7," where "7" is the highest bridge priority. An access point with bridge priority "0" cannot become a designated bridge.

EXAMPLE:

Figure 3-7 shows an example of the designated bridge selection process. The two access points connected to the secondary LAN have different bridge priorities ("3" and "5"). Access point B is the designated bridge because its bridge priority is higher than A's priority.

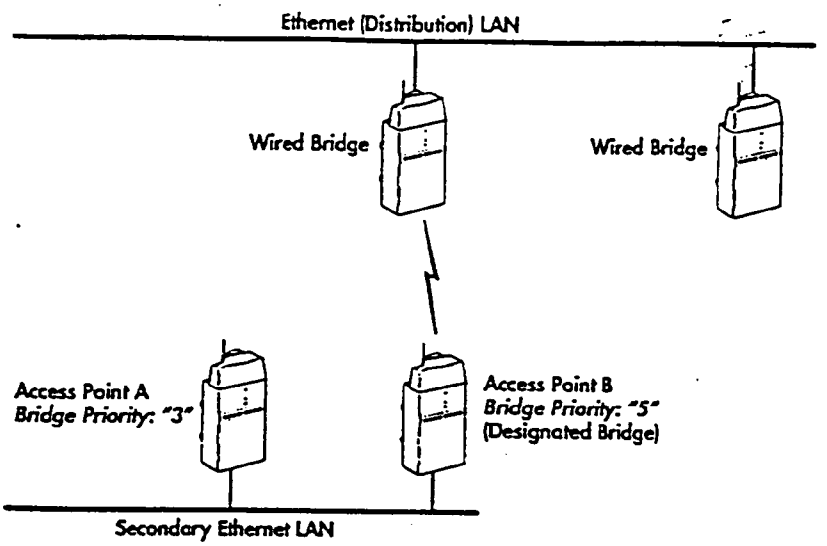


Figure 3-7
Designated Bridge Selection Example

If an access point on the secondary LAN has a higher bridge priority than other access points on the secondary LAN, but is outside the radio coverage area of an access point on the distribution LAN, it cannot become the designated bridge.

Norand assigns a default value of "1" to the bridge priority. If you are installing a designated bridge, you should change this value through the access point's configuration menus to determine the selection of the designated bridge.

If two or more access points have the same bridge priority, the access point with the highest Ethernet address becomes the designated bridge.

You must set a secondary *bridge flag* for the designated bridge's Ethernet port in addition to the bridge priority. Bridge flags allow the designated bridge to be configured to optimize which frame types are bridged over the wireless link. Bridge flags are unicast and multicast flooding options, discussed later in this section.

If a designated bridge goes offline, the remaining candidates determine which one becomes the new designated bridge. The designated bridge is always the access point that meets these criteria:

- Physically connects to a secondary Ethernet LAN
- Is within the radio coverage area of an access point on the distribution LAN
- Has the highest nonzero bridge priority; if it has the same bridge priority as another access point, then it has highest Ethernet address (unless the access point with the highest priority is out of range)
- Has a secondary bridge flag set for its Ethernet port

Configurations

Section 4 shows examples of secondary and multiple secondary LANs.

Secondary Proxim LAN

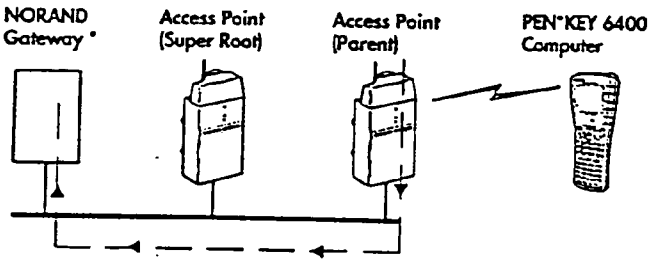
The radio coverage area of an access point with the Proxim 2.4 GHz radio option is a *secondary Proxim LAN*. The designated bridge is the access point with the radio. Default values for access points with the Proxim 2.4 GHz radio are factory-set to enable designated bridging, with flooding disabled.

Wireless Stations

Wireless stations are the end nodes in the spanning tree. Access points forward frames to and receive frames from the wireless stations. When a wireless station is power managed the access point buffers outbound messages and uses a pending message list to communicate that messages are buffered.

Unicast, broadcast, and multicast addresses are supported. A *unicast address* is a unique Ethernet address assigned to a single station. *Broadcast* is a transmission to all wireless stations at the same time. *Multicast* is a form of broadcast through which copies of the frame are delivered to a subset of all possible destinations with a common multicast address.

The wireless station converts, then sends to its parent, information it collects through its keyboard or scanner, for example. If an emulation solution requires a gateway, the gateway forwards the data to the host. Figure 3-8 shows data flow from a terminal emulation station to its parent and then to a NORAND gateway. Dashed lines are data.



* RC4030E Gateway, Wireless Network Access Server on host, 6950 Enterprise Gateway Server, or 6910 Integrated Gateway/Access Point

Figure 3-8
Sample Network Data Flow

LAN Identification Number (Domain)

So they can communicate, open wireless LAN nodes must have the same *LAN ID* (also called *domain*). It may be desirable to have independent networks operating with an overlapping radio frequency (RF) coverage area. In this case, you must set the LAN ID to ensure each network's access points and wireless stations communicate within their own network.

You set the LAN ID through configuration menus for each access point and wireless station. Norand assigns a default LAN ID of "0" to these devices. If you need to change this number to achieve a more efficient configuration for your site, you must change the number for each access point and wireless station in the same network to the new number. Norand strongly recommends that you change the default LAN ID to another value when you initially configure the devices.

▶ NOTE: *For the 900 MHz and UHF radio options, the range of LAN ID numbers is "0" through "254." For the Proxim 2.4 GHz radio option, the range is "0" through "15." (The LAN ID is the Proxim domain ID modulo 16.)*

Netname and Security ID

For link security, the wireless infrastructure provides a network naming capability called *netname*. For the Proxim 2.4 GHz radio option an additional parameter called *security ID* provides separate security to prevent unauthorized PC-compatible computers from associating with access points. (*Associating* is the process a wireless station follows to connect with a single access point at any one time.) Norand supports the security ID for the Proxim 2.4 GHz radio in addition to netname for compatibility with standard Proxim drivers in wireless stations.

Netname and security ID are ASCII strings. You set the strings through configuration menus for the access point and PC-compatible computer. The default netname is blank; the default security ID is "norandowl." So they can communicate, all access points and PC-compatible computers in the same network must have the same netname or security ID, or both. An access point or PC-compatible computer attaches to the network only if its netname or security ID matches the super root's netname or security ID.

Autoconfiguration

For most installations, one of the features of the open wireless LAN is its ability to automatically configure the spanning tree using factory default parameters of the access points.

Within minutes of when access points power up, the network discovers all possible communication paths, develops the spanning tree, and selects the super root. The spanning tree creates a loop-free network topology.

For the network to autoconfigure, Norand suggests that a minimal number of parameters be set:

- Root priority
- LAN ID
- Netname
- Bridge priority for each designated bridge
- For Proxim 2.4 GHz radio option: security ID

Norand also suggests that a minimal number of radio-specific parameters be set:

- For Proxim 2.4 GHz radio: channel and subchannel
- For 900 MHz radio: mode and channel
- For UHF radio: frequency

EXAMPLE 1:

Figure 3-9 shows an example of autoconfiguration. Each access point has the default root priority, LAN ID, and netname or security ID (or both). Each access point has a unique Ethernet address:

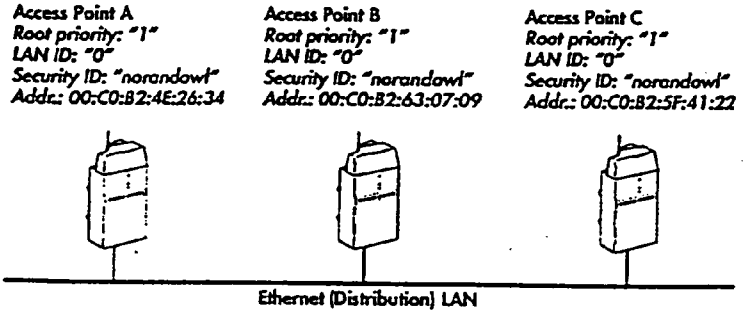


Figure 3-9
Autoconfiguration Through Ethernet Address

Access points A, B, and C are super root candidates because they have the same root priority. Access point B becomes the super root because it has the highest Ethernet address. If access point B went offline, access point C would become the super root because its Ethernet address is higher than access point A's address.

EXAMPLE 2:

Figure 3-10 shows another example of autoconfiguration. Each access point has the default LAN ID and netname or security ID (or both). Access points A and B have the default root priority. Access point C's root priority is "3." Each access point has a unique Ethernet address:

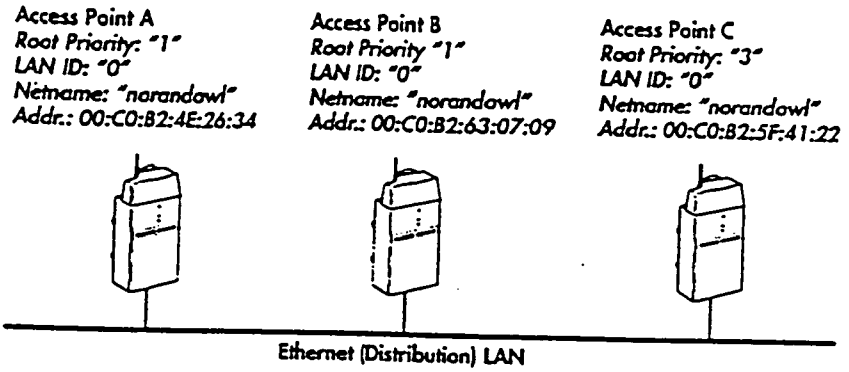


Figure 3-10
Autoconfiguration Through Root Priority

Access point C becomes the super root because it has the highest root priority. If access point C went offline, access points A and B would become super root candidates. B would become the new super root because its Ethernet address is higher than A's address.

Why Change the Spanning Tree Configuration?

The spanning tree can use factory-set default values for the autoconfiguration parameters to create a loop-free topology. However, those values might conflict with an adjacent LAN. To achieve a more desirable configuration, you can change the values through the access point's configuration menus.

Usually the best configuration depends on a network's particular characteristics and traffic loads. In configurations with secondary Ethernet LANs, the root priority and bridge priority can be changed to force a more efficient topology.

In this case the root priority should be set so that the distribution LAN is the "backbone" LAN. Norand or certified providers can review your needs to help you determine the best configuration for your environment.

Inter-Access Point Protocol

The open wireless LAN uses an inter-access point protocol to manage the wireless infrastructure. Open wireless LAN protocol frames are transmitted between NORAND network devices. The DIX Ethernet type of an open wireless LAN protocol frame is hexadecimal 875C.

Hello Frames

On Ethernet links, the super root or designated bridge periodically sends hello frames. On 900 MHz or UHF radio links, all access points periodically send hello frames. The frames help organize the access points into a spanning tree and advertise link availability.

Hello frames randomize around 1-, 2-, or 3-second intervals on an Ethernet link. The multicast destination 802 address for hello frames is always hexadecimal 01:C0:B2:4D:43:4F on Ethernet links.

On open wireless LAN radio links, hello frames can randomize around 1- or 2- second intervals. The interval is adjustable through configuration menus.

Power Management

For the 900 MHz or UHF radio option, the MACR sublayer provides several facilities to support sleeping wireless stations. A sleeping station initially synchronizes on a hello response frame from the wired bridge. Hello frames include a pending message list. Each entry in the list contains the node ID of a wireless station with pending messages. The wireless station calculates the time of the next expected hello response frame and powers down with an active timer interrupt set to wake it just before the next hello response frame is transmitted.

Power management extends the operating period for a given battery charge. By awakening before the next scheduled message, the wireless station misses no messages during a sleep period.

The Proxim 2.4 GHz radio option has a power management mechanism similar to the 900 MHz and UHF radio options. Norand supports the standard Proxim power management facilities for compatibility with the Proxim operating system or standard Proxim drivers in wireless stations.

Attach Frames

Open wireless LAN nodes periodically send unicast attach frames to explicitly associate with the open wireless LAN and refresh connections. Periodic attachment reduces or eliminates the need to flood unicast frames into the radio network.

Data Frames

Data frames normally bridge onto an Ethernet link as regular Ethernet frames with a DIX, 802.3, or SNAP protocol type. The 802 source address of NORAND open wireless LAN frames (DIX 875C) is always the 802 address of the access point that transmitted the frame, which is DIX, 802.3, or 802.3 SNAP over the Ethernet physical medium.

EXAMPLE: A DIX IP frame originating on a radio link bridges onto an Ethernet link with a DIX type of hexadecimal 0800. The 802 source address is the address of the wireless station that originated the frame.

Optionally, you can disable bridging on an access point through its configuration menus. An access point with bridging disabled is a *wired access point* that will not convert Ethernet frames to open wireless LAN frames, and open wireless LAN frames to Ethernet frames. Normally, bridging should not be disabled.

MAC-R Frames

DIX Ethernet frames are used for inter-access point (MAC-R) communications.

Frame Forwarding

The access point maintains a forwarding database with an entry for each node in the subtree rooted at the access point, and entries for inbound nodes or nodes on the distribution LAN. All access points receive frames on the Ethernet link in *promiscuous mode*, which means an access point receives all broadcast frames and frames destined for unicast or multicast addresses. If the destination is in the subtree rooted at an access point, that access point forwards the frame outbound, for example. Otherwise the frame is ignored.

Optionally, flooding levels can be set so that frames are flooded throughout the network if the destination is unknown. Flooding is discussed later in this section.

The database associates unicast Ethernet addresses with *ports*. Each entry contains a destination address and an associated port identifier. When the access point receives a frame, it searches its forwarding database to determine the port of the destination. If the access point finds the destination and if the destination is on another port (other than the one on which the frame arrived), the access point bridges or forwards the frame to the destination port.

With bridging enabled, an access point bridges inbound open wireless LAN frames onto its Ethernet port if the destination is not in its forwarding database. If the destination is in its subtree, the access point bridges a frame from its Ethernet port outbound into the open wireless LAN radio network. An open wireless LAN frame is *inbound* if it is moving toward the super root. The frame is *outbound* if it is moving away from the super root.

EXAMPLE 1:

Figure 3-11 shows how spanning tree nodes route a frame. Dashed lines represent branches. Assumptions are as follows:

- Computer A is sending a unicast IP frame to computer B.
- Each access point and PEN*KEY computer has the 900 MHz or UHF radio option.

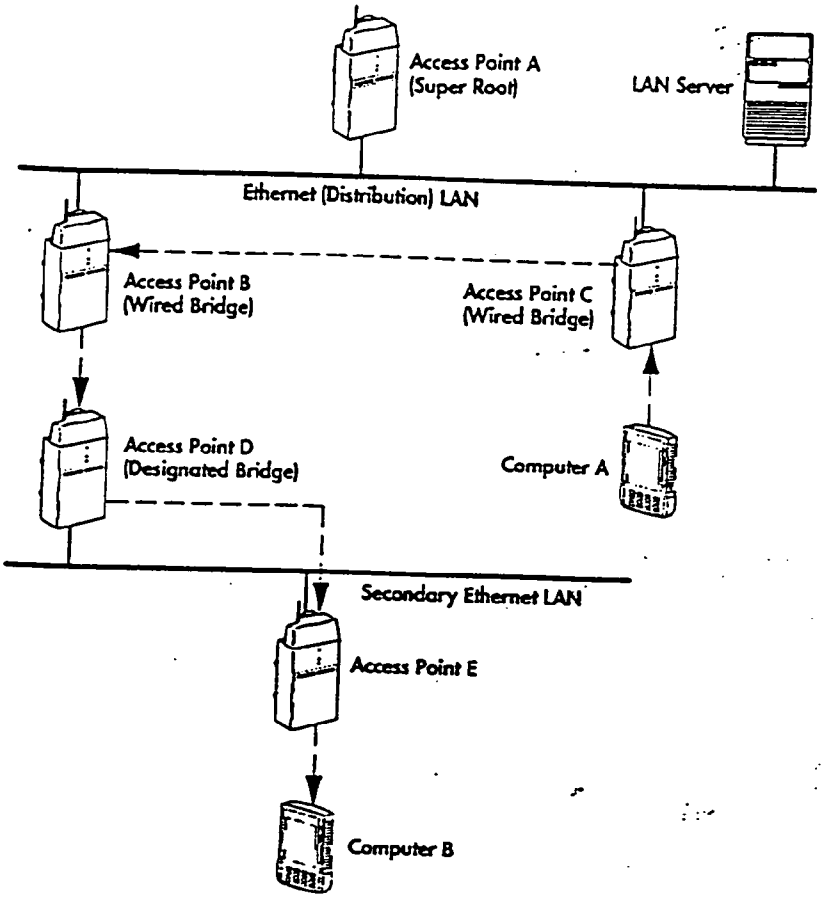


Figure 3-11
Frame Forwarding Between Wireless Stations

Spanning tree nodes route the frame as follows:

1. Computer A sends the unicast IP frame to its parent (access point C) through a wireless link.
2. Access point C bridges the frame onto the distribution LAN. The source and destination addresses are the 802 addresses of computers A and B, respectively. If the frame is DIX, the Ethernet type is 0800.
3. Access points A and B receive the bridged frame. In this case, the super root (access point A) and access point B should have a route table entry for computer B. However, the root entry for the access points' private database is marked as *distributed*. (Distributed means another access point is responsible for bridging outbound frames to the destination.) The super root will not bridge a non-open wireless LAN frame if its route table entry for the destination is distributed.
4. Access point B bridges the IP frame and forwards it over its radio port to access point D. Access point D has a distributed table entry for computer B. Therefore, access point D bridges the frame onto the secondary Ethernet LAN (that is, as a DIX IP frame).
5. Access point E receives the frame because its Ethernet port is in promiscuous mode. Access point E has a route table entry for computer B.
6. Access point E bridges the frame from Ethernet and forwards it over its radio port to computer B.

EXAMPLE 2: If computer B sends an IP frame to the LAN server on the distribution LAN, Figure 3-12 shows how spanning tree nodes route the frame.

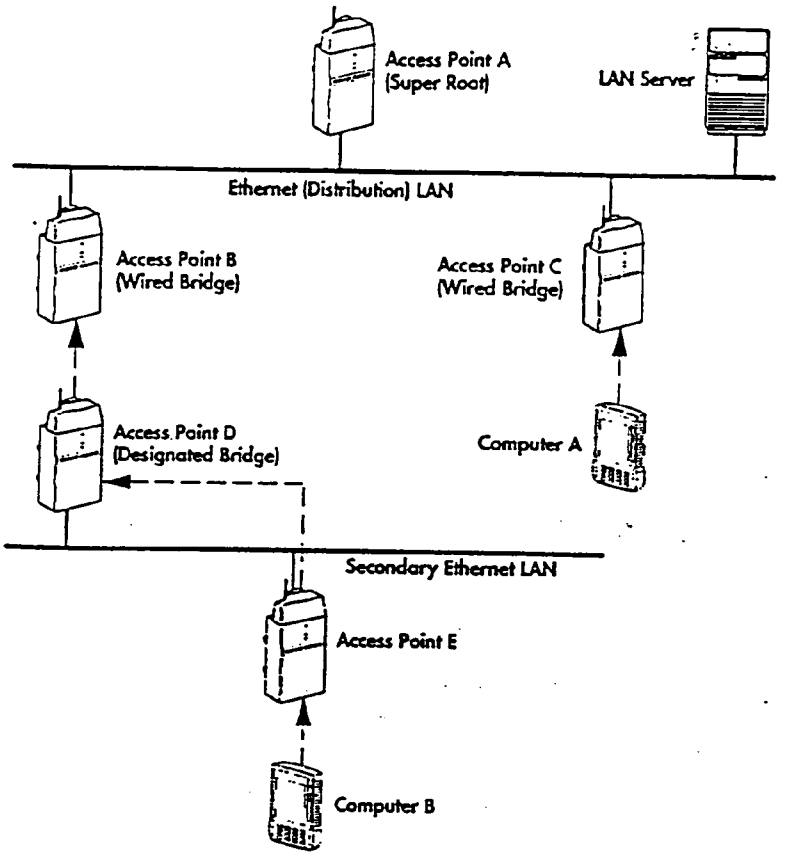


Figure 3-12
Frame Forwarding Between Wireless Station and LAN Server

Spanning tree nodes route the frame as follows:

1. Computer B sends the frame to its parent (access point E) through a wireless link.
2. Access point E bridges the frame onto the secondary Ethernet LAN as a IP frame. Note that if the LAN server was connected to the secondary LAN, the server would receive the bridged frame.
3. Access point D forwards the frame inbound to access point B.

4. Access point B bridges the frame onto the distribution LAN as an IP frame. If the frame is DIX, the Ethernet type is 0800.
5. The server receives the frame. Forwarding is transparent to the server, which has an Ethernet protocol stack.

Flooding

If the access point is unable to find a destination address in its forwarding database (the destination is unknown), the access point floods the frame if configured to do so. *Flooding* is a process where frames received on one port are transmitted on all other ports.

You can configure unicast or multicast flooding options (or both) for the distribution LAN and for each secondary Ethernet LAN. You configure flooding options through the super root's configuration menus. Because access points with a nonzero root priority are candidates to become the super root, each access point with a nonzero root priority should have the same flooding options for consistency.

In many cases flooding puts unnecessary traffic onto the RF medium. For this reason, the default configuration for the access point disables outbound flooding of unicast frames.

To reduce traffic you can limit flooding to a subset of secondary Ethernet LANs. This feature is intended for sites with a mixture of secondary Ethernet LANs and secondary Proxim LANs. Depending on your application, you may want to avoid flooding frames to secondary Proxim LANs but want to flood frames to certain secondary Ethernet LANs.

You can selectively flood the frames by specifying a secondary bridge flag for each secondary LANs' designated bridge. The bridge flag works in conjunction with the flooding levels set for the super root. The designated bridge for a secondary LAN notifies the super root and each access point on the inbound path to the super root that it requires unicast or multicast flooding, or both.

Access points connected to the secondary Ethernet LAN and with a nonzero bridge priority are candidates to become the designated bridge. For consistency, the same flooding options should be set for each candidate.

Complete instructions on how to configure flooding options are in the *6710 Access Point User's Guide (NPN: 961-047-081)* and *6910 Integrated Gateway/Access Point User's Guide (NPN: 961-047-095)*.

Unicast Flooding Options

You can configure the access point to take one of the following actions when it receives unicast frames:

- Discard unicast frames that originate on the distribution LAN if the destination is unknown. The access point forwards unicast frames that originate in the radio network inbound, until the frame arrives at an access point with a route entry for the destination. The access point relays an inbound unicast frame onto the distribution LAN if the destination is unknown.
- Flood unicast frames to the distribution LAN and to each secondary Ethernet LAN that has unicast flooding enabled. For example, an access point forwards, to the distribution LAN and to each secondary Ethernet LAN that has unicast flooding enabled, a unicast frame that originates in the radio network if the destination is unknown.
- Flood unicast frames to the distribution LAN and all secondary Ethernet LANs if the destination is unknown.

Generally, the best option is to disable unicast flooding. However, flooding of unicast frames is required if the network contains one or more secondary Ethernet LANs with wired stations that do not periodically generate traffic.

Multicast Flooding Options

You can configure the access point to take one of the following actions when it receives multicast frames:

▶ NOTE: *Selecting higher multicast flooding levels increases wireless LAN traffic.*

- Discard multicast frames that originate on the distribution LAN.
- Forward, to secondary Ethernet LANs that have multicast flooding enabled, multicast frames that originate on the distribution LAN.

- Flood, throughout the open wireless LAN, multicast frames that originate on the distribution LAN. The access point forwards, to the distribution LAN, multicast frames that originate in the radio network or on a secondary Ethernet LAN.
- Flood, throughout the open wireless LAN, all multicast frames. For example, if a multicast frame originates in the radio network, the access point forwards the frame to the distribution LAN and to each access point on the open wireless LAN. An access point broadcasts the message on each radio port and relays the message to any attached secondary Ethernet LAN.

▶ NOTE: *Broadcast DIX Address Resolution Protocol (ARP) frames (DIX type is hexadecimal 0806) are always forwarded to each access point on the open wireless LAN, even when you set no multicast flooding options.*

Filtering

You can define filtering options to selectively discard Ethernet frames the access point receives on its Ethernet port. *Filtering* is a process that allows only predefined frame types to be forwarded. Filtering prevents the access point from forwarding unnecessary Ethernet frames onto the radio network.

Two filtering options are available: DIX Ethernet and programmable. Complete instructions on how to configure both options are in the *6710 Access Point User's Guide (NPN: 961-047-081)* and *6910 Integrated Gateway/Access Point User's Guide (NPN: 961-047-095)*.

DIX Ethernet Filtering

You can enter a list of up to 15 DIX Ethernet frame types through access point configuration menus. If the list of types is set to "enabled," Ethernet frames with frame types equal to those in the list are forwarded from the Ethernet physical medium to the access point bridging modules. All other frames are discarded. Frame types 875C (open wireless LAN), 0800 (IP), and 0806 (ARP) are always enabled.

If the list of types is set to "disabled," Ethernet frames with frame types equal to those in the list are discarded. Other frames are forwarded to the access point bridging modules within the constraints of any other filtering options.

Filter lists enhance the performance of the access points by keeping unwanted Ethernet frames from being handled and forwarded by the access points. For this reason DIX Ethernet filtering is the preferred filtering option.

The DIX frame type is the two bytes after the source address in an Ethernet MAC header. In the IEEE 802.3 standard the bytes represent the length of the Ethernet frame. Therefore, the DIX Ethernet filtering option is of limited use in 802.3 networks.

Programmable Filtering

You can program unicast or multicast filters (or both) through access point configuration menus. Programmable filters are mainly useful for selectively filtering multicast frames, since the destination address cannot be associated with a single station. That is, the frame must be flooded.

A pattern list and an expression with enabled or disabled masks form the access point's filter. Each pattern list has a fixed size and frame offset. Each list can join to another list through AND, OR, and other operators to build complex filters.

EXAMPLE: You can define a filter to forward only broadcast Novell Service Advertisement Protocol (SAP) frames from a select group of servers. You can also filter all multicast frames except for ARP requests from a group of servers.

Processes

The flowchart in Figure 3-13 summarizes the filtering and flooding processes when the access point receives a unicast frame.

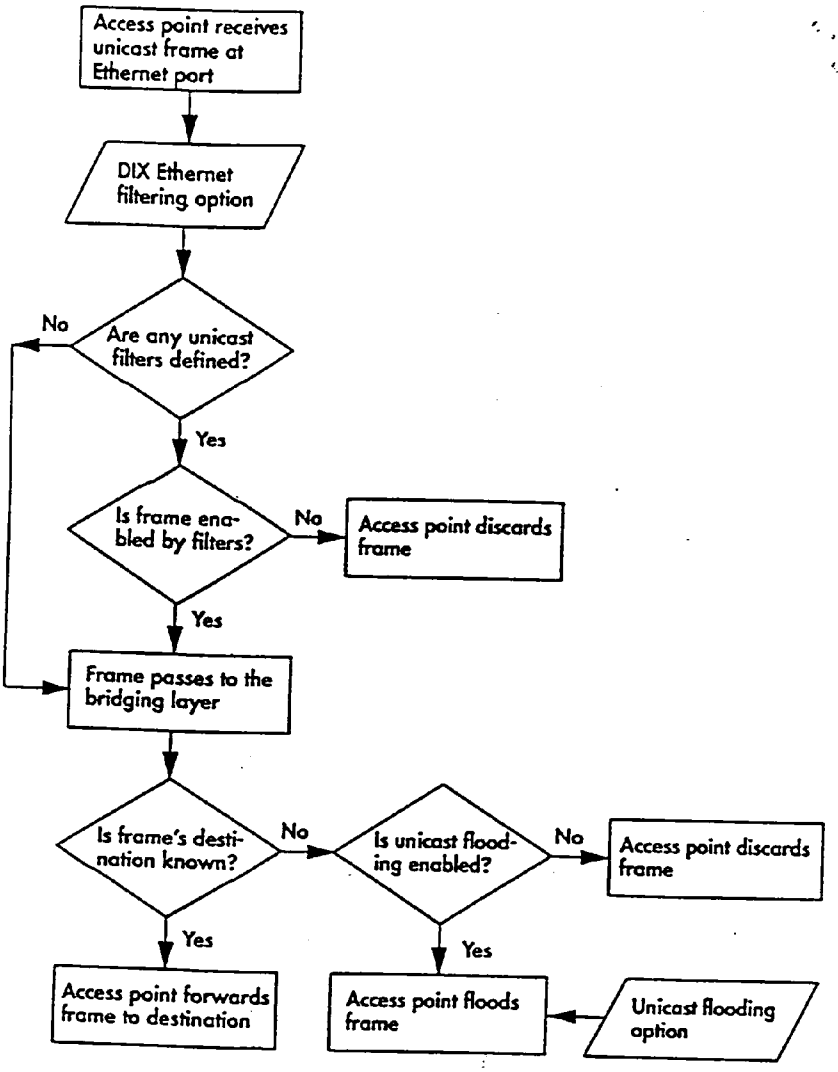


Figure 3-13
Filtering and Flooding Processes

Inter-Access Point Communications

Access points learn the locations of wired and wireless stations through a spanning tree. Because spanning tree information is communicated at the MAC sublayer, access points cannot communicate across IP routers, which connect separate IP subnets. Inter-access point communications at the MAC-R sublayer are not routable.

If an IP router is present, you can enable inter-access point communications by configuring the router to bridge DIX type 0875C packets. Consult a Norand Systems Engineer for more information about enabling bridging on routers.

EXAMPLE: In the configuration example in Figure 3-14, access points on subnet A cannot communicate with access points on subnet B:

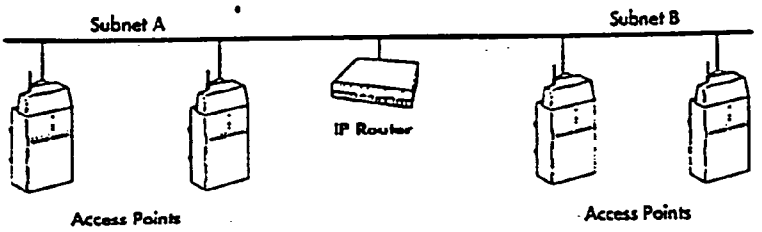


Figure 3-14
Inter-Access Point Communications

See Section 8, "Installation," for general information about routers.

Roaming

An open wireless LAN typically contains multiple access points, which provide an extended, seamless radio coverage area. The coverage areas of multiple access points must overlap to provide uninterrupted wireless access to the Ethernet medium.

Overlapping coverage areas enable a wireless station to move from the coverage area of one access point into the area of another while maintaining LAN connectivity. Wireless stations monitor the reliability of communications with an access point and initiate roaming if a significantly better link can be established with another access point.

Reassociating With Another Access Point

When a wireless station detects it has roamed, it must reassociate with another access point. When the wireless station connects with another access point, the new access point updates its forwarding database and forwards the update to the super root, which forwards it to the previous access point. This updating process ensures frames are sent along the proper branches of the spanning tree to connect to the wireless stations.

When a wireless station roams between access point coverage areas, the new access point begins forwarding frames to the wireless station. The previous access point stops forwarding. A wireless station with the 900 MHz or UHF radio option listens for hello frames on the network and then requests connection to the node with the fastest route to the distribution LAN. The route is determined by the *cost*. The cost in the hello frame is generally an indication of the bandwidth cost to reach the distribution LAN.

As wireless stations roam through the open wireless LAN, they must establish a connection with a single access point (the parent) as their entry point to the wired Ethernet backbone at any one time. When a wireless station approaches a coverage area boundary, it searches for an access point with a better signal and more reliable data throughput.

Updating the Forwarding Database

An access point updates its forwarding database when it receives a frame on a primary or secondary LAN port. The access point makes or updates an entry containing the 802 source address and the source port. If the new source port differs from the previous one, the access point changes its database to indicate the wireless station has roamed from one physical segment to another.

Attach frames on open wireless LAN ports update the forwarding database. The frames are sent each time a wireless station roams, and are periodically sent to maintain the path. Frames are always forwarded to the super root, which generates a detach frame to delete the old path when a wireless station roams.

Use of Bridges Within the Infrastructure

The open wireless LAN architecture is designed to operate correctly when off-the-shelf transparent bridges are used within the Ethernet backbone.

EXAMPLE: The configuration example in Figure 3-15 shows a PEN*KEY computer roaming to an access point on the other side of an off-the-shelf bridge:

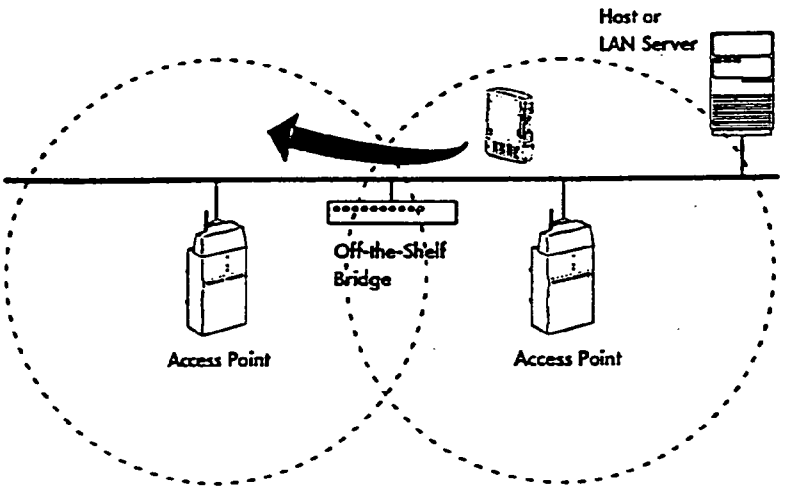


Figure 3-15
Off-the-Shelf Bridge Within Infrastructure

For more information about how bridges operate in general, see Section 8. Appendix B lists network products, including bridges, that Norand recommends.

Section 4

Network Configurations

About This Section

This section shows examples of network configurations with the 900 MHz, UHF, and Proxim 2.4 GHz radio options.

Configurations With 900 MHz, UHF, or Proxim 2.4 GHz Option

Configurations with the 6710 Access Point with the 900 MHz, UHF, or Proxim 2.4 GHz radio option include the following (described on the following pages):

- Configuration with PC-compatible computers
- Configuration with multiple access points.
- Hybrid configuration
- Configuration with remote network management station
- Configuration with TFTP server
- Configuration with RC4030E Gateway
- Modified star configuration
- Configuration with Wireless Network Access Server
- Configuration with 6950 Enterprise Gateway Server

PC-Compatible Computers

In this configuration the 6710 Access Point forwards, over the radio network, frames between PC-compatible computers and the LAN server or IP host (or both) on the Ethernet LAN. In the example shown in Figure 4-1, the access point forwards frames between a PEN*KEY® 6100 and notebook. To the host and server, these computers appear to be hard-wired to the LAN.

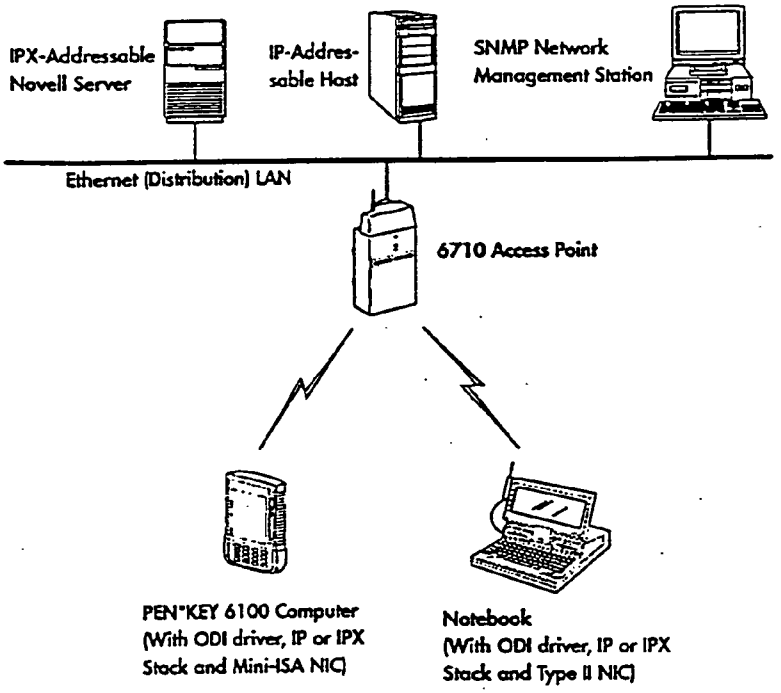


Figure 4-1
Configuration With PC-Compatible Computers

Multiple Access Points

Multiple 6710 Access Points can bridge frames between wireless stations and the Ethernet LAN (Figure 4-2). The open wireless LAN infrastructure enables the wireless stations to roam from the coverage area of one access point to another coverage area without disrupting network service. Roaming is seamless and transparent to the end user and application.

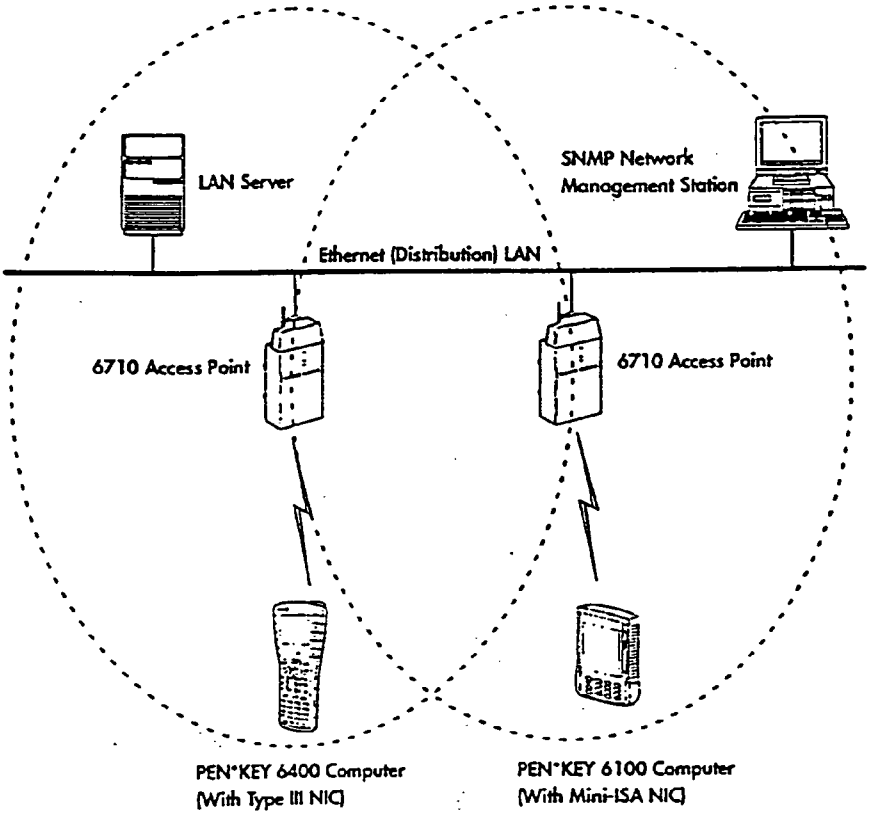


Figure 4-2
Configuration With Multiple Access Points

Hybrid

Different topologies connected together create a hybrid configuration, which results in a larger network span. Figure 4-3 shows a sample configuration with Token Ring and 10BASE2. The router divides the configuration into three separate networks: Token Ring, the company's enterprise network, and the hub with 6710 Access Points.

▶ NOTE: *Appendix B lists hubs and other recommended network products.*

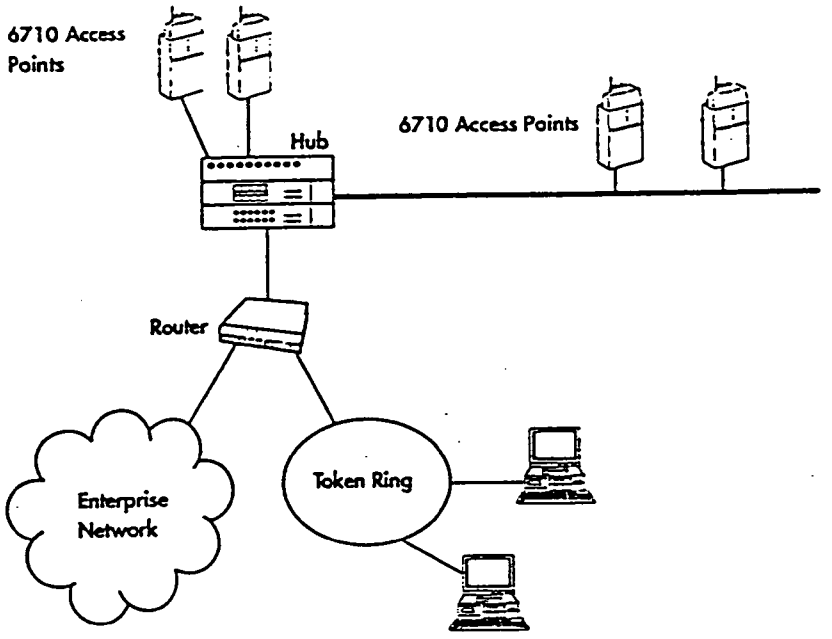


Figure 4-3
Hybrid Configuration

Remote Network Management Station

You can manage the wireless infrastructure through SNMP, which provides a way for network management platforms to query network devices for status and other device information. Figure 4-4 shows a sample configuration with a remote SNMP network management station. The connectivity solution shown is a remote access modem.

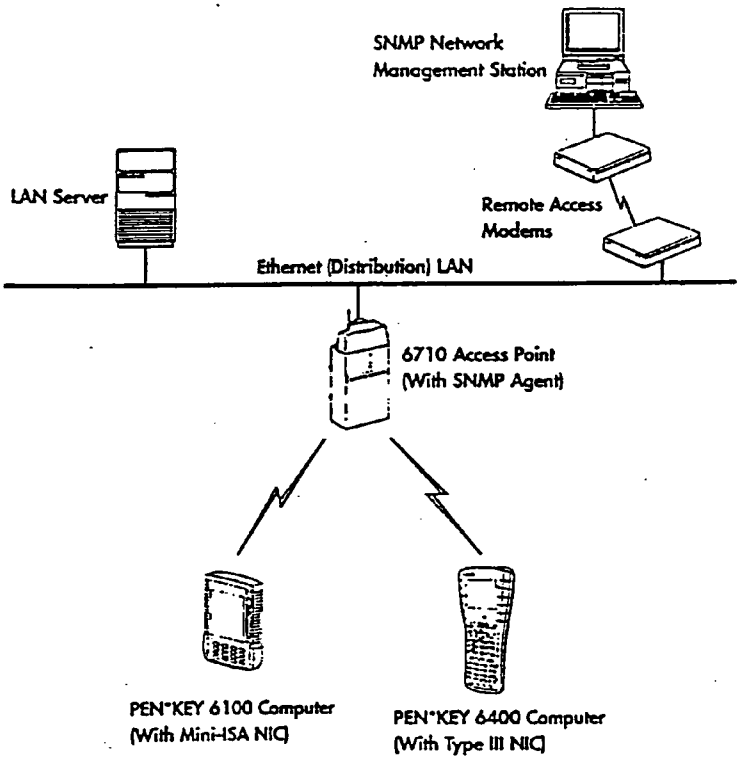


Figure 4-4
Configuration With Remote Network Management Station

PC TFTP Server

You can download the latest version of system software to a 6710 Access Point through a PC TFTP server. Connectivity solutions for the PC TFTP server are direct LAN and Ethernet modem.

Direct LAN

Figure 4-5 shows a PC TFTP server directly connected to the LAN. Each access point is a TFTP client you can access through a TELNET or SNMP session with the access point's IP address.

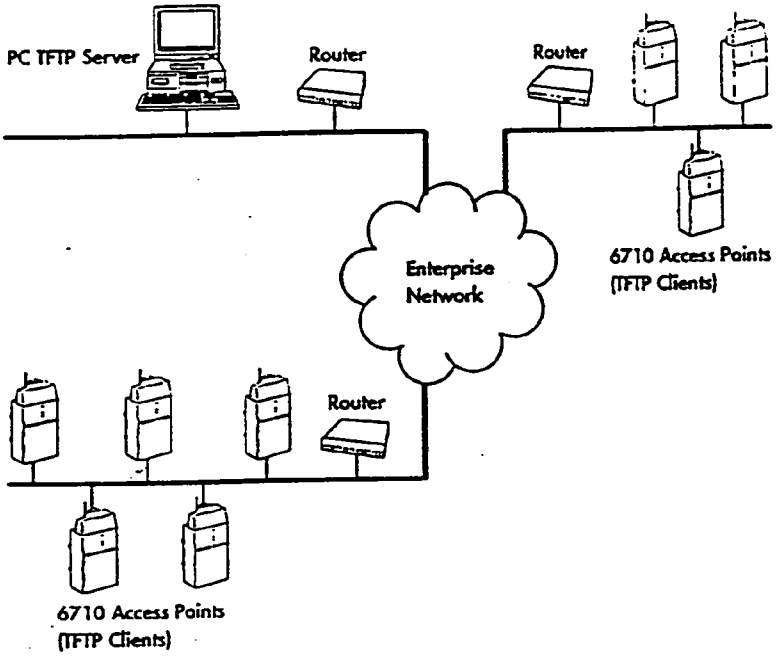


Figure 4-5
Configuration With PC TFTP Server, Direct LAN

Ethernet Modem

Figure 4-6 shows a PC TFTP server connected to an Ethernet modem. Each access point is a TFTP client you can access through a TELNET or SNMP session with the access point's IP address.

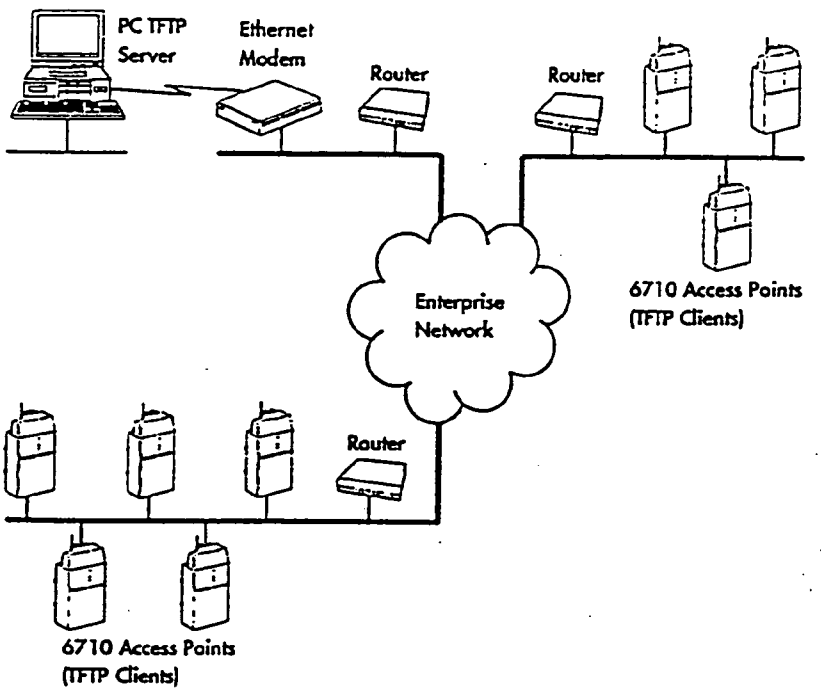


Figure 4-6
Configuration With PC TFTP Server, Ethernet Modem

RC4030E Gateway

Figure 4-7 shows a configuration with an RC4030E Gateway connected to a host running the 3270 or 5250 protocol. The 6710 Access Point forwards, over the radio network, data frames between the terminal emulation stations and the gateway. The gateway converts the data frames into the host protocol.

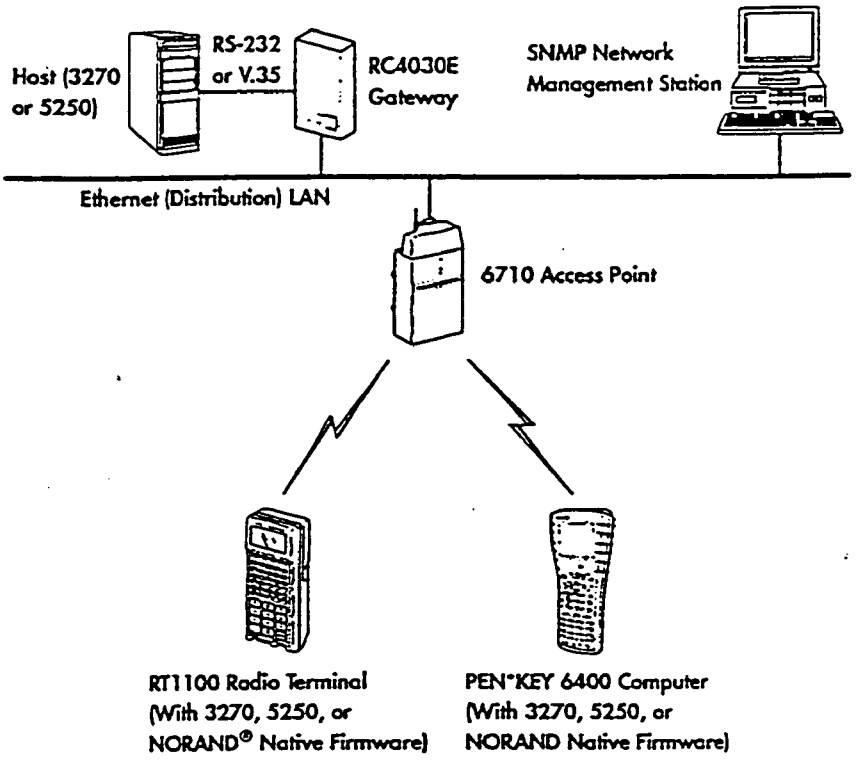


Figure 4-7
Configuration With RC4030E Gateway

Modified Star

Figure 4-8 shows 6710 Access Points and an RC4030E Gateway connected to stacked hubs. A fiber backbone connects the hubs. This configuration forms a modified star. For more information about how hubs and other network communication products operate in general, See Section 8, "Installation."

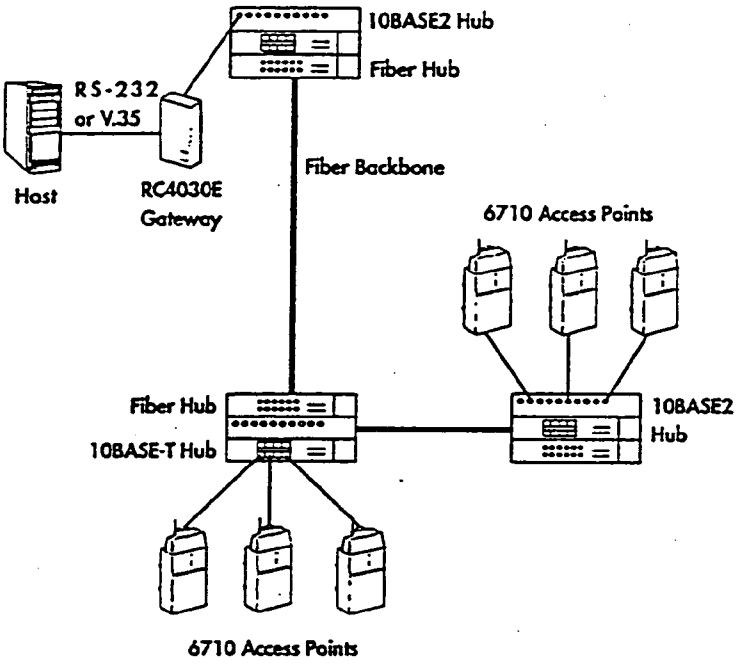


Figure 4-8
Modified Star Configuration

Wireless Network Access Server

Figure 4-9 shows a configuration with Wireless Network Access Server (WNAS) software running on an RS/6000 host. The 6710 Access Point forwards, over the radio network, data frames between the terminal emulation stations and WNAS on the host. WNAS converts the data frames into the TCP/IP TELNET protocol.

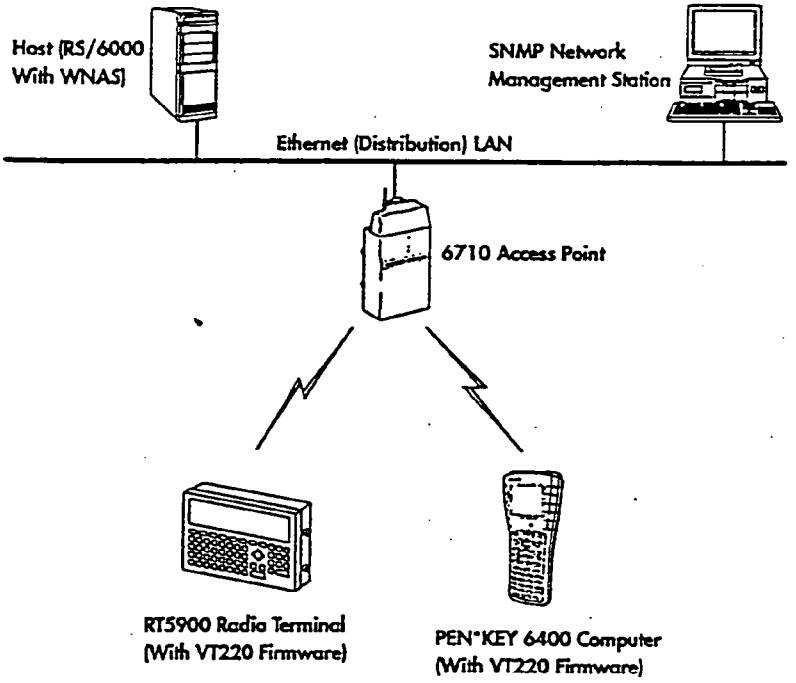


Figure 4-9
Configuration With WNAS

6950 Enterprise Gateway Server

Figure 4-10 shows a configuration with a 6950 Enterprise Gateway Server. The 6710 Access Point forward forwards, over the radio network, data frames between the terminal emulation stations and the gateway server. The gateway server converts the data frames into the TCP/IP TELNET protocol.

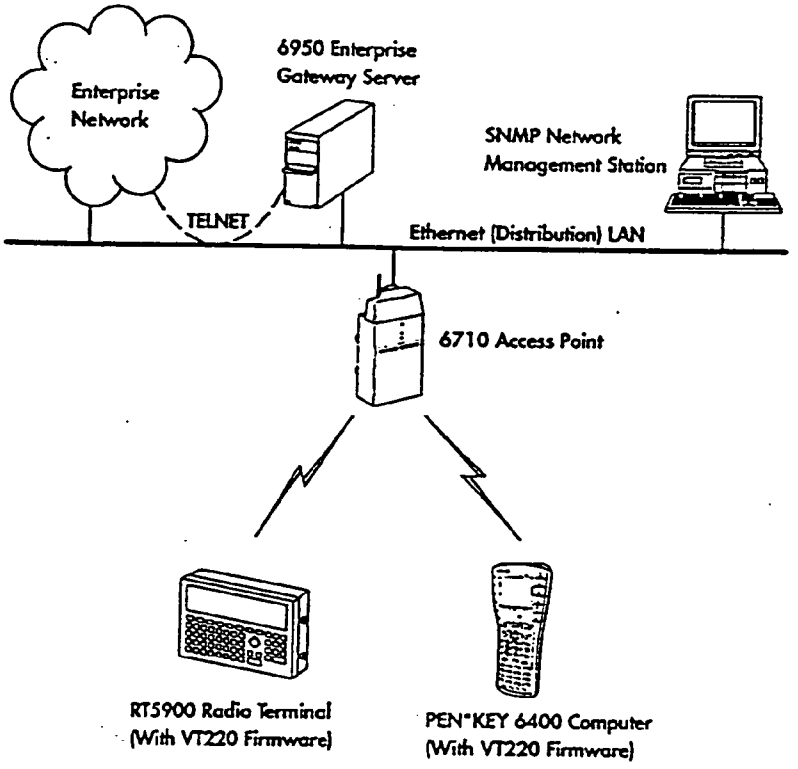


Figure 4-10
Configuration With 6950 Enterprise Gateway Server

Additional Configurations With 900 MHz or UHF Option

Configurations with the 6710 Access Point with the 900 MHz or UHF radio option include the following:

- Secondary LAN
- Multiple secondary LAN
- Wireless access point

▶ NOTE: *The UHF radio option is technically capable of being exercised in these configurations. However, because of bandwidth limitations, these configurations with the UHF radio option are not recommended except in special situations. Contact a Norand Sales Representative for more information.*

Secondary LAN

In a secondary LAN configuration, a designated bridge connects a secondary Ethernet LAN to the distribution LAN through a wireless link. The LANs can be in the same building or in separate buildings.

▶ NOTE: *In general, bridging through a wireless link has lower performance than wired Ethernet.*

Same Building

In the example shown in Figure 4-11 the access point is forwarding, over the radio network, frames between the LAN server and desktop B wired to a secondary Ethernet LAN.

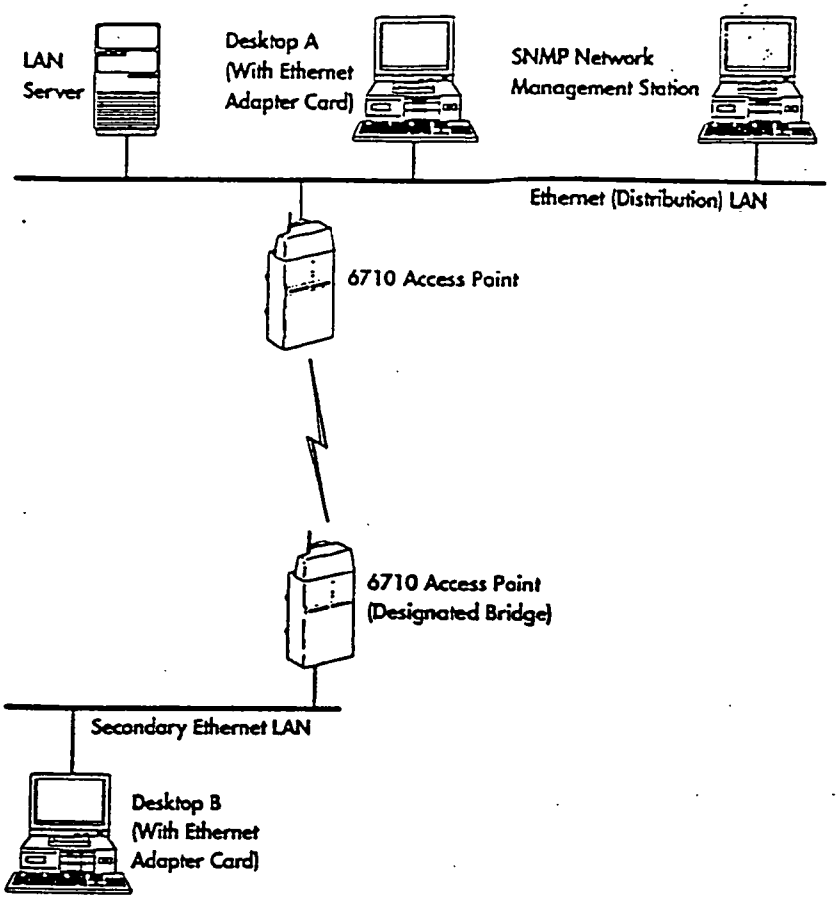


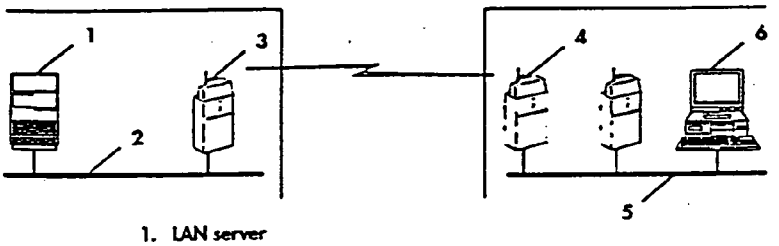
Figure 4-11
Secondary LAN Configuration, Same Building

Separate Buildings

Two access points with the 900 MHz or UHF radio option and standard antennas can connect two Ethernet LANs in separate buildings through a wireless link. This configuration eliminates the need to lay cables between the buildings or lease a line from the phone company. The access points also provide coverage for wireless stations that require connectivity to the LAN.

For best results, you should place the access points and antennas providing the wireless link near windows. Norand or certified providers can supply additional placement information through a formal site survey. For current information about standard antenna availability, consult a Norand Sales Representative.

Figure 4-12 shows a sample configuration where access points with the 900 MHz radio option are forwarding frames between the LAN server on the distribution LAN and the desktop wired to the secondary Ethernet LAN. For best results from the secondary Ethernet LAN, you should assign the highest bridge priority to the access point with the best physical link, which is the access point by the window.



1. LAN server
2. Ethernet (distribution) LAN
3. 6710 Access Point
4. 6710 Access Point (designated bridge)
5. Secondary Ethernet LAN
6. Desktop (with Ethernet adapter card)

Figure 4-12
Secondary LAN Configuration, Separate Buildings

Multiple Secondary LAN

In a multiple secondary LAN configuration, designated bridges connect secondary Ethernet LANs to the distribution LAN through wireless links. In the example shown in Figure 4-13, desktops A and B can communicate with other desktops on their own LAN, or with desktops on separate secondary LANs. The designated bridges forward frames between desktops A and B.

▶ NOTE: *In general, bridging through a wireless link has lower performance than wired Ethernet.*

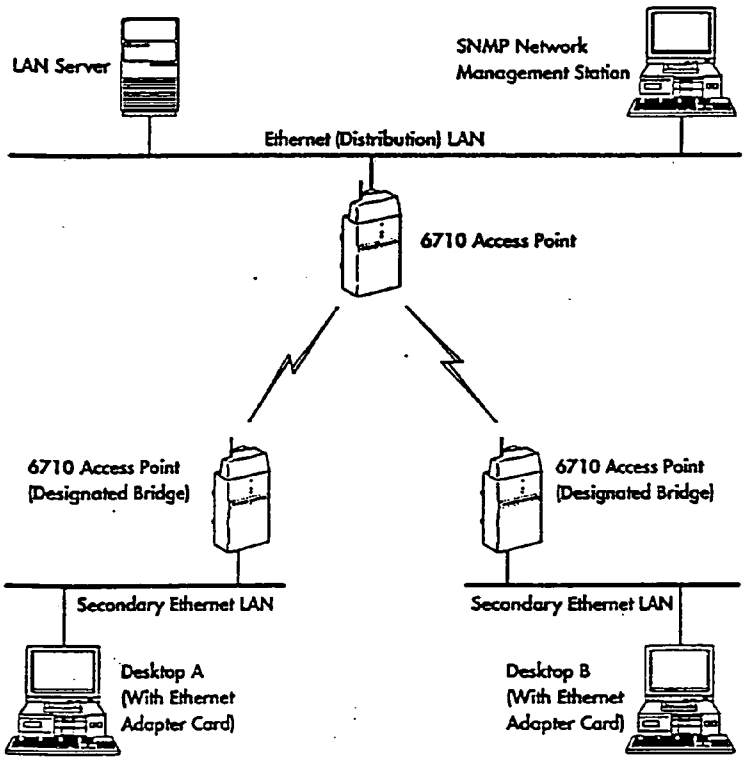


Figure 4-13
Multiple Secondary LAN Configuration

Wireless Access Point

A 6710 Access Point with the 900 MHz or UHF radio option can be a wireless access point that overlaps coverage with a wired bridge. In the example shown in Figure 4-14 the access points are forwarding frames between the wireless stations and the Ethernet LAN over the radio network. Note that the wireless access point does not physically connect to the Ethernet medium.

▶ NOTE: *In general, forwarding through wireless access points has lower performance than wired Ethernet. Section 7, "Wireless Access Points," covers wireless access points and performance issues.*

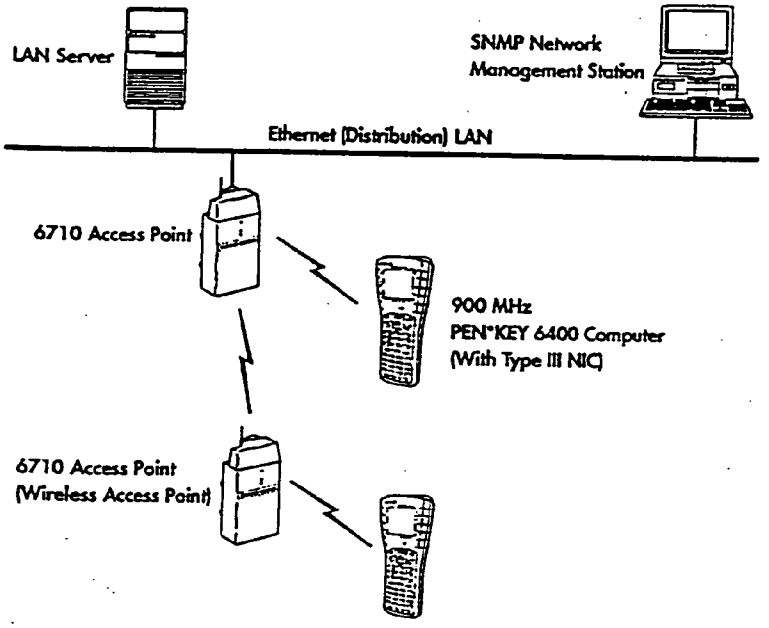


Figure 4-14
Configuration With Wireless Access Point

Additional Configurations With Proxim 2.4 GHz Option

PEN*KEY computers by Norand, and third-party PC-compatible computers with the Proxim 2.4 GHz radio option, are capable of wireless peer-to-peer (ad hoc) connections. The 6710 Access Point with the Proxim 2.4 GHz radio option can be a device in a point-to-point configuration with interbuilding bridges. The following pages describe these configurations.

Wireless Ad Hoc or Peer-to-Peer

A source wireless station establishing a radio link with a destination wireless station creates an ad hoc or peer-to-peer connection. This connection establishes a network that can share files and other resources instantly. Figure 4-15 shows two notebooks and one PEN*KEY 6100 Computer in a peer-to-peer configuration. The notebooks have Proxim 2.4 GHz Type II NICs. The PEN*KEY computer has a Proxim 2.4 GHz mini-ISA NIC in the pod unit attached to the back of the computer.

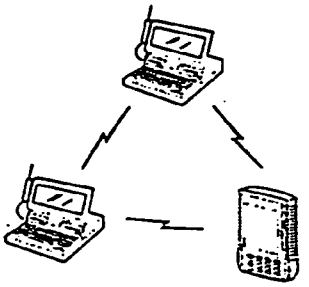


Figure 4-15
Wireless Ad Hoc or Peer-to-Peer Configuration

Point-to-Point

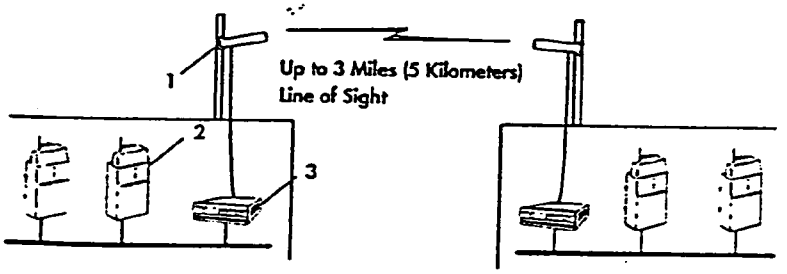
Two interbuilding bridges with high-gain, 2.4 GHz yagi (unidirectional) antennas can connect Ethernet LANs in two separate buildings through a wireless link. This point-to-point configuration eliminates the need to lay cables between the buildings or lease a line from the phone company.

▶ NOTE: *In general, bridging through a wireless link has lower performance than wired Ethernet.*

Each antenna has a range of up to three miles (5 kilometers) line of sight. For best results, you should properly mount each antenna onto an antenna mast or exterior wall. Norand or certified providers can provide additional placement information through a formal site survey.

6710 Access Points coexist with interbuilding bridges on the same Ethernet LAN. To the wireless infrastructure, separate Ethernet LANs with access points and interbuilding bridges operate as a wired network. The interbuilding bridge's architecture supports any protocol or network operating system that supports Ethernet.

Figure 4-16 shows a sample configuration with 6710 Access Points and interbuilding bridges.



1. Yagi antenna (on mast)
2. 6710 Access Point
3. Interbuilding bridge

Figure 4-16
Point-to-Point Configuration, Separate Buildings

Section 5

Network Connectivity

About This Section

Wireless NICs and NORAND® PC-compatible computers provide a range of network connectivity solutions. This section describes these network products and the solutions they provide.

Wireless NICs

NICs for wireless stations that require connectivity to the enterprise Ethernet LAN include the following:

- Type III
- Type II
- Mini-ISA
- ISA

These NICs conform to various PC card and ISA bus standards. To a host or server, wireless stations equipped with NICs appear to be standard network nodes wired to the Ethernet medium. By operating at the Data Link layer, the NICs provide protocol-independent access for mobile users into a wired Ethernet LAN.

▶ NOTE: *Appendix A, "Radio Options," contains wireless NIC specifications and features.*

Type III (900 MHz and Proxim 2.4 GHz)

The Type III wireless NIC (Figure 5-1) is a high-performance adapter for 6710 Access Points and PEN*KEY® 6100 (900 MHz radio option only), 6400, and 6600 Computers. For PEN*KEY computers the NIC supports ODI or NDIS drivers or both, which facilitate mobile applications in environments that support these drivers. For proper network communications, you must configure the NIC's software according to your site's requirements.

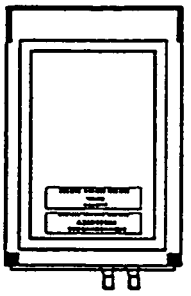


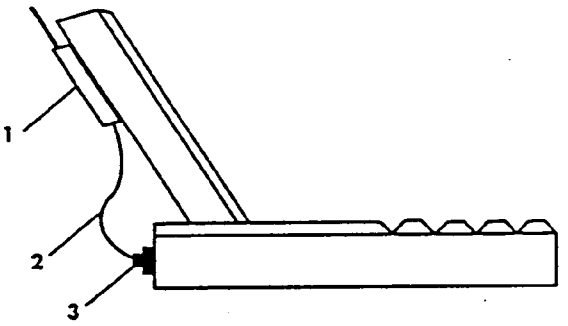
Figure 5-1
Type III Wireless NIC

The Type III NIC plugs into one of the 6710 Access Point's PC card-compatible slots and is field-replaceable. Norand preinstalls the NIC into the PEN*KEY computers; Norand or other qualified service personnel must replace the NIC for these computers.

Type II (Proxim 2.4 GHz)

The Type II wireless NIC is a high-performance adapter for laptop and notebook computers with Type II card slots. The Type II NIC supports ODI and NDIS drivers. Norand provides NIC software on diskette. For proper network communications, you must load the software into the computer and configure it according to your site's requirements.

The Type II NIC consists of a standard Type II card that plugs into the computer's card slot, and an antenna unit that connects to the card through a tether and plug. The antenna unit usually mounts onto the back or side of the computer. For best performance the antenna extends above the top of the notebook. Figure 5-2 shows a notebook with an installed Type II NIC.



1. Antenna unit
2. Tether
3. Plug

Figure 5-2
Notebook With Type II Wireless NIC

Mini-ISA (UHF and Proxim 2.4 GHz)

The mini-ISA wireless NIC is a high-performance adapter for wireless stations in these series: PEN*KEY 6100 Computers (pod solutions), RT1100 and RT1700 Radio Terminals (radio module solutions), and RT5900 Radio Terminals (internal solutions).

For the PEN*KEY 6100 Computer, the mini-ISA NIC supports ODI and NDIS drivers. For proper network communications, you must configure the NIC's software according to your site's requirements. Appendix C, "ODI and NDIS Driver Configurations," shows examples of driver configurations for a PEN*KEY 6100 Computer.

Norand preinstalls the mini-ISA NIC into the pod unit that attaches to the back of the PEN*KEY 6100 Computer (Figure 5-3). Because the NIC is installed in the pod, the computer's PC card slots are accessible for other uses.

1. Pod

Figure 5-3
PEN*KEY 6100 Computer Pod

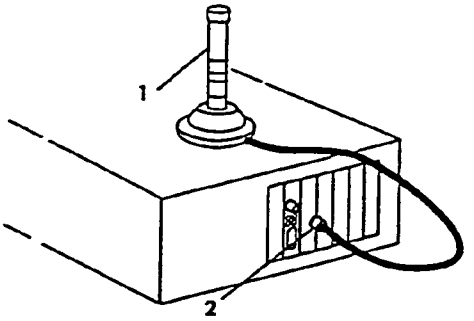
Norand preinstalls the mini-ISA NIC into the field-replaceable radio modules for the RT1100 and RT1700 Radio Terminals. (Appendix A lists radio and scanner modules.) Norand also preinstalls the NIC into the RT5900 Radio Terminal; Norand or other qualified service personnel must replace the NIC.

ISA (Proxim 2.4 GHz)

The ISA wireless NIC is a high-performance adapter for PC AT-bus or PC-compatible computers. The NIC supports ODI and NDIS drivers. Norand provides NIC software on diskette. For proper network communications, you must load the software into the computer and configure it according to your site's requirements.

The ISA NIC consists of a standard network adapter card that plugs into the system unit's ISA bus slot (8 bit or 16 bit), and an antenna unit that connects to the card through a connector.

For best performance the antenna unit sits on the top of the computer or desk. Figure 5-4 shows a computer with an installed ISA NIC; the antenna unit is on top of the system unit. Norand or certified providers can provide additional placement information through a formal site survey.



1. Antenna unit
2. Connector

Figure 5-4
ISA Wireless NIC

PC-Compatible Computers

Portable, hand-held PC-compatible computers by Norand include members of the PEN*KEY family. PEN*KEY computers communicate with 6710 Access Points as part of a wireless network infrastructure (except for PEN*KEY computers in a peer-to-peer or ad hoc configuration). Wireless NICs and software drivers installed in the computers make them appear to be nodes physically connected to the wired Ethernet medium.

PEN*KEY computers are designed for job-specific applications such as forms-based computing. They offer touch, pen, and keyboard data entry, and can capture signatures for transaction verification. The following pages briefly describe PEN*KEY models for the open system. Contact a Norand representative to find out which model would work best in your environment.

PEN*KEY 6100 Computer

The PEN*KEY 6100 Computer (Figure 5-5) is a Windows- and DOS-based computer. For local area communications, the PEN*KEY 6100 uses the Proxim 2.4 GHz radio option.

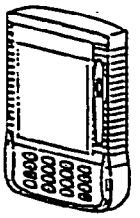


Figure 5-5
PEN*KEY 6100 Computer

The PEN*KEY 6100 Computer's display and keyboard enhance portable computing. The 4.95" diagonal display is touch-activated; a screen stylus is available for signature capture. The keyboard has 16 numeric, tactile keys. You use the keyboard and optional scanner (integrated standard, long-range laser, tethered wand, CCD, or laser) to enter bar code and numeric data.

Two Type II card slots and one Type III slot are other options for the PEN*KEY 6100. More information about these and other options is in the *PEN*KEY 6100 Computer User's Guide (NPN: 961-028-085)*.

PEN*KEY 6400 Computer

The PEN*KEY 6400 Computer (Figure 5-6) is a DOS-based computer that offers wireless access to applications requiring realtime, local-area communications for Ethernet-capable systems. The PEN*KEY 6400 also offers enhanced battery capacity and management. Its lithium ion battery pack incorporates a processor-based logic circuit that performs battery management and charging control.

Figure 5-6
PEN*KEY 6400 Computer

The PEN*KEY 6400 Computer's display and keyboard enhance portable computing. The display size is 2.4" (diagonal). The keyboard has 41 or 51 alphanumeric, tactile keys. You use the keyboards and optional scanner (integrated standard, long-range laser, integrated CCD, tethered wand, CCD, or laser) to enter bar code and numeric data.

Two Type II card slots and one Type III slot are other options for the PEN*KEY 6400. More information about these and other options is in the *PEN*KEY 6400 Computer User's Guide (NPN: 961-028-093)*.

PEN*KEY 6600 Computer

Users enter information on the PEN*KEY 6600 Computer (Figure 5-7) through a standard inductive pen or an optional touch-activated screen. Information can also be entered through an external PS/2 style keyboard, which allows customized software solutions to run with off-the-shelf office automation software packages.

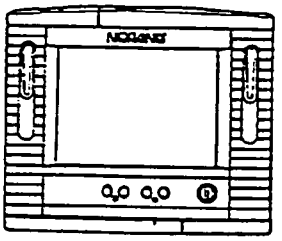


Figure 5-7
PEN*KEY 6600 Computer

The PEN*KEY 6600 is Windows- and DOS-based. Its display has a power-managed backlight and an automatic temperature compensated contrast. Other features include a 486 DX2 50 MHz microprocessor, 2 to 8 MB FLASH expansion modules, and 4, 8, or 16 MB optional internal DRAM expansion modules. Optional modem and scanning options include an RJ11 connection for modems and integrated scanning (both standard and long-range).

The PEN*KEY 6600 can accommodate two Type II card slots and one Type III slot. More information about these and other options is in the *PEN*KEY 6600 Computer User's Guide (NPN: 961-028-084)*.

Development Environments

Because PEN*KEY computers have standard DOS or Microsoft Windows operating systems (or both), you can employ a range of tools to develop custom applications. Their PC architecture opens them to any standard interface.

Most PEN*KEY computers support software developed specifically for pen-based systems. PEN*KEY computers also provide application support for Microsoft C, C++, Visual Basic, and other DOS- and Windows-compatible languages. A major benefit of this open system approach is that you can purchase development tools and software from several vendors, including Norand. This lets you select the equipment and software tools that apply to your development.

Software developer tool kits are available for PEN*KEY computers. The file complement of the tool kits differs among software releases. Tool kits contain DOS or Windows resources (or both) for configuration, power management, communications, and peripherals.

Development information is in these references:

- *PEN*KEY Model 6100 Computer Programmer's Reference Guide (NPN: 977-054-001)*
- *PEN*KEY Model 6200/6300 Computer Programmer's Reference Guide (NPN: 977-054-003)*
- *PEN*KEY Model 6600 Computer Programmer's Reference Guide (NPN: 977-054-002)*

ODI and NDIS Drivers

The wireless infrastructure complies with protocol stacks that support ODI and NDIS driver specifications. Each driver supports multiple standard protocol stacks, which reside above the driver.

EXAMPLE: A PC-compatible computer (such as a PEN*KEY 6100) could have a Proxim 2.4 GHz wireless NIC running the ODI driver with the TCP/IP stack by FTP Software, Inc., and the IPX/SPX stack by Novell. The PEN*KEY computer would communicate with a 6710 Access Point, which bridges the PEN*KEY computer's Ethernet packets onto the wired Ethernet medium. The PEN*KEY computer operates as if it was a node physically connected to the local wired Ethernet segment.

Section 6

Host Connectivity

About This Section

This section provides an overview of NORAND® host connectivity devices and terminal emulation stations. It also describes the terminal emulation protocol stack.

Host Connectivity Devices

Norand provides high-performance gateway products and terminal emulation stations optimized for use over the wireless medium. The gateways connect to or reside within host computer systems and replace existing multiterminal controllers. Gateways provide direct connections to the Ethernet medium. They support network and transport connections over the wireless infrastructure through the 6710 Access Point to the terminal emulation stations.

NORAND host connectivity devices are the RC4030E Gateway, Wireless Network Access Server, and 6950 Enterprise Gateway Server. The following pages describe these products.

RC4030E Gateway

The RCB4030E Gateway (Figure 6-1) is a protocol-dependent device. It operates as a gateway (protocol translator) between the host computer and the terminal emulation stations on the wireless network.

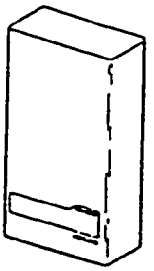


Figure 6-1
RC4030E Gateway

The RC4030E Gateway picks up — via the 6710 Access Point — NORAND packets from terminal emulation stations. The gateway translates the packets into the appropriate host protocol and sends the data to the host through its host port.

You can configure an RC4030E Gateway as one of these controllers:

- IBM 3174 or 3274 Cluster Controller. To the host, the wireless station configured for 3270 terminal emulation appears to be an IBM 3278 Model 2 terminal.
- IBM 5294 or 5394 Control Unit. To the host, the wireless station configured for 5250 terminal emulation appears to be an IBM 5291 Display Station.
- Asynchronous. To the host, the wireless station configured for NORAND Native emulation appears to be an ASCII terminal.

The *RC4030E Gateway User's Guide (NPN: 961-047-087)* describes how to set the controller type and other gateway configuration options. These publications provide more information on terminal emulation:

- *3270 Terminal Emulation Programmer's Reference Guide (NPN: 977-047-040)*
- *5250 Terminal Emulation Programmer's Reference Guide (NPN: 977-047-039)*
- *Native Terminal Emulation Asynchronous Programmer's Reference Guide (NPN: 977-047-038)*

6910 Integrated Gateway/Access Point

The 6910 Integrated Gateway/Access Point (Figure 6-2) combines the functionality of the RC4030E Gateway and 6710 Access Point to support the NORAND Native communications type for small installations.

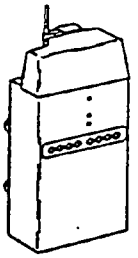


Figure 6-2
6910 Integrated Gateway/Access Point

As an optional wired bridge, the gateway/access point bridges frames between the wired Ethernet LAN and the wireless stations on the radio network. An optional gateway/access point function is to serve as the connection point for several types of wireless stations, which include NORAND terminal emulation stations and PC-compatible computers.

When configured with host options, the gateway/access point picks up data frames from the terminal emulation stations. It translates the frames into the appropriate host protocol and sends the data to the host through the gateway/access point's diagnostic port.

The *6910 Integrated Gateway/Access Point User's Guide (NPN: 961-047-095)* describes how to set host and other gateway/access point configuration options. More information about the NORAND Native communications type is in the *Native Terminal Emulation Asynchronous Programmer's Reference Guide (NPN: 977-047-038)*.

Wireless Network Access Server

The Wireless Network Access Server (WNAS) is a *software* component that provides TELNET capability for VT220 terminal emulation. WNAS is installed and configured on hosts with specified operating systems. When it is infeasible to install WNAS onto the host, the 6950 Enterprise Gateway Server provides similar functionality.

Versions of WNAS support the computer operating systems listed in Table 6-1.

Table 6-1
Operating Systems WNAS Supports

Operating System	Operating System Version
SCO UNIX	4.2 or later
IBM AIX	3.2 or later
HP UX (Series 800)	9.04 or later
Sun Solaris	2.4 or later
IBM OS/2	2.1 or later

WNAS uses the client-server concept: The host computer is the server, and the terminal emulation stations are the clients. This lets software developers write applications for the wireless stations, which are independent of the host computer operating system.

WNAS supports these interfaces:

- VT220 terminal emulation (TELNET) for situations requiring direct connection into existing applications. To the host, the wireless station configured for VT220 terminal emulation appears to be a VT220 terminal.
- NORAND Application Development Kit (ADK) for applications requiring client functionality at the terminal emulation level.

These publications provide more information: *VT220/ANSI Terminal Emulation Programmer's Guide (NPN: 977-047-037)* and *Application Developer's Kit Reference Manual (NPN: 961-051-001)*. The *Wireless Network Access Server User's Guide (NPN: 961-051-006)* describes the WNAS product.

6950 Enterprise Gateway Server

Another host connectivity option is the 6950 Enterprise Gateway Server (Figure 6-3). The gateway server communicates —via the 6710 Access Point — frames from terminal emulation stations running VT220 terminal emulation. The gateway server then translates the packets into a standard TELNET session, and puts the data back onto the Ethernet LAN with the host running TCP/IP.

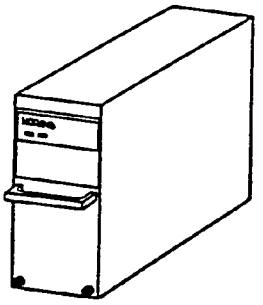


Figure 6-3
6950 Enterprise Gateway Server

When the gateway server is booted it establishes a connection to the terminal emulation station based on the data it gets from system setup files. As each wireless station powers on, it establishes a terminal session with the host. The *6950 Enterprise Gateway Server User's Guide* (*NPN: 961-047-091*) has more information about gateway server operation and setup.

Terminal Emulation Stations

Terminal emulation stations for the open system include the PEN*KEY® 6400 Computer and radio terminals in these series: RT1100, RT1700, and RT5900.

PEN*KEY 6400 Computer

PEN*KEY 6400 Computers can operate as a wireless station running 3270, 5250, VT220, or NORAND Native terminal emulation. More information about how to set up, maintain, and operate the PEN*KEY computer is in the *PEN*KEY 6400 Computer User's Guide (NPN: 961-028-093)*.

RT1100 Radio Terminal

RT1100 Radio Terminals (Figure 6-4) are a series of lightweight "pocket RF" radio terminals designed primarily for retail use. The radio terminal is 6.875" long x 2.625" wide x 1.375" deep and has an elastomer, 47-key keyboard with alphabetic and numeric keys.

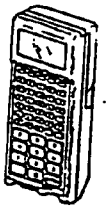


Figure 6-4
RT1100 Radio Terminal

The radio terminal's liquid crystal display has 4, 6, 8, or 9 lines by 12 or 16 characters. (Sizes are adjusted by the user.) The display also has a backlight and full bit-mapped graphics capability.

The radio terminal's scanner and radio modules are interchangeable. You can upgrade the radio terminal by changing its module to accommodate changing application needs or new radio technologies. More information about how to set up, operate, and maintain the radio terminal is in the *1100 Series User's Guide (NPN: 961-047-069)*.

RT1700 Radio Terminal

Radio terminals in the RT1700 Series (Figure 6-5) operate in industrial and retail environments. The radio terminals have an optional vehicle mount, which makes them useful for "pick and run" applications.

The radio terminal is 9.75" long x 2.625" wide x 1.375" deep. It has a elastomer 57- or 37- key keyboard with alphabetic and numeric keys, plus a scanner key. Its black-on-white liquid crystal display has 8, 10, 16, or 21 lines x 16, 21, or 26 characters. (Sizes are adjusted by the user.) The display also has a backlight and full bit-mapped graphics capability.

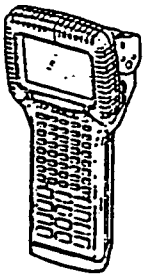


Figure 6-5
RT1700 Radio Terminal

The radio terminal's scanner and radio modules are interchangeable. You can upgrade the radio terminal by changing its module to accommodate changing application needs or new radio technologies. More information about how to set up, operate, and maintain the radio terminal is in the *1700 Radio Terminal User's Guide (NPN: 961-047-068)*.

RT5900 Mobile Mount Radio Terminal

Mobile mount radio terminals in the RT5900 Series (Figure 6-6) meet NEMA 3 standards for ruggedness and durability in harsh environments. The radio terminal can be removed from its mounting bracket and mounted onto a forklift. The radio terminal can also be used on a desktop as a wireless radio terminal for manufacturing processing.

RT5900 Radio Terminals have a tactile 57-key keyboard with alphabetic and numeric keys. The radio terminal's liquid crystal display provides graphics capabilities and sizes from 8 lines by 40 characters up to 25 lines by 80 characters. Sizes are adjusted by the user.

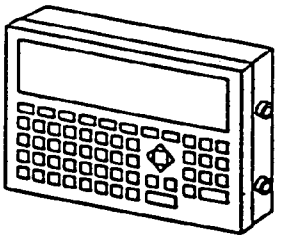


Figure 6-6
RT5900 Mobile Mount Radio Terminal

More information about how to set up, operate, and maintain the radio terminal is in the *RT5980 Radio Terminal User's Guide (NPN: 961-047-092)*.

Host Protocol Support

Terminal emulation stations support IBM 3270 SNA/SDLC, IBM 5250 SNA/SDLC, and NORAND Native terminal emulations through the RC4030E Gateway, and NORAND Native through the 6910 Integrated Gateway/Access Server.

Terminal emulation stations support VT220 terminal emulation through WNAS software or the 6950 Enterprise Gateway Server. WNAS and the gateway server support connectivity to networks with TCP/IP protocols.

Application Integration Tools

You can use a range of tools to develop custom applications for PEN*KEY 6400 Computers and radio terminals. Tools include Microsoft C and the ADK C libraries by Norand.

Terminal Emulation Protocol Stack

The terminal emulation protocol stack provides efficient data exchange over the wireless infrastructure. When compared to off-the-shelf emulations written for standard PCs, the protocol stack promotes large wireless station populations with low response times and improved battery management. The protocol stack also enables terminal emulations to coexist with industry-standard protocols operating over the wireless infrastructure. Figure 6-7 shows the terminal emulation protocol stack.

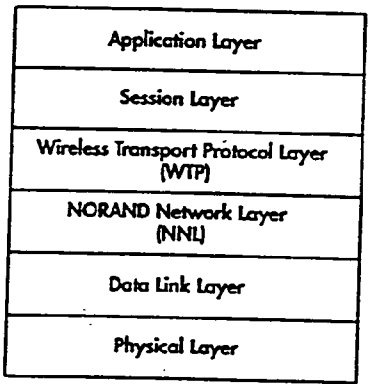


Figure 6-7
Terminal Emulation Protocol Stack

The NNL protocol uses an Ethernet protocol type of hexadecimal 875B. Data frames (such as a frame carrying data from a NORAND RT1700 Radio Terminal to the RC4030E Gateway via a 6710 Access Point) are carried as NNL-type frames. NNL frames are usually not sent to the multicast address; instead they are sent to the MAC (physical) address of the actual network device they are enroute to.

EXAMPLE 1: A data frame from an RT1700 Radio Terminal (with a Proxim 2.4 GHz radio module) to an RC4030E Gateway would have a source address of 00:20:A6:xx:xx:xx (the Ethernet vendor address for Proxim) and a destination address of 00:C0:B2:xx:xx:xx (the Ethernet vendor address for Norand).

EXAMPLE 2: A data frame from an RT1700 Radio Terminal (with a Proxim 2.4 GHz radio module) to a 6950 Enterprise Gateway Server would have a source address of 00:20:A6:xx:xx:xx (Ethernet vendor address for Proxim) and a destination address of 00:00:0C:xx:xx:xx or 00:80:0F:xx:xx:xx (the vendor address for Western Digital or SMC, the manufacturers of the Ethernet adapter card for the gateway server).

Dashed lines in Figure 6-8 show data flow through the terminal emulation protocol stack. The terminal emulation station is set up for 5250 SNA/SDLC, 3270 SNA/SDLC, or NORAND Native terminal emulation. The host connectivity device for these types of terminal emulation is the RC4030E Gateway.

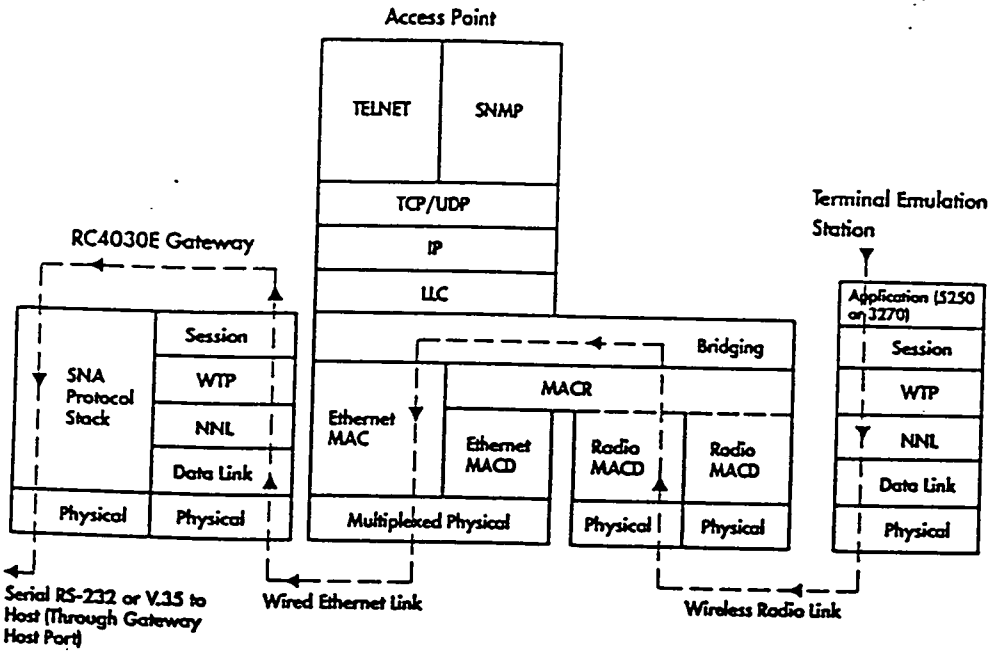


Figure 6-8
Data Flow Through Terminal Emulation Protocol Stack (RC4030E Gateway)

Figure 6-9 shows another example of data flow through the terminal emulation protocol stack. In this figure, the terminal emulation station is set up for VT220 terminal emulation. In this example, the host connectivity device for VT220 terminal emulation is the 6950 Enterprise Gateway Server. Note that the gateway server converts the terminal emulation protocol stack into a TELNET session to the host.

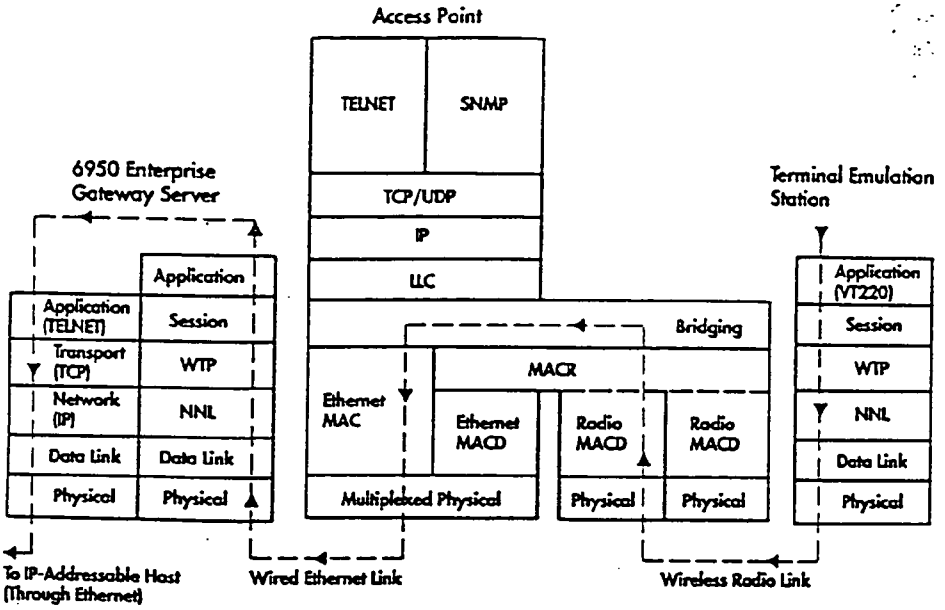


Figure 6-9
Data Flow Through Terminal Emulation Protocol Stack (6950 Enterprise Gateway Server)

Section 7

Wireless Access Points

About This Section

An access point with the 900 MHz or UHF radio option can be a wireless access point, which provides a range of connectivity solutions. This section describes wireless access points and the solutions they provide.

Operation

A *wireless access point* is an access point that does not physically connect to the Ethernet medium. The wireless access point provides a wireless store-and-forward operation (a *hop*) with each frame transmitted twice over the wireless media to reach its destination. Because frames are transmitted twice, the amount of wireless traffic over the radio network doubles.

In general, the throughput of a wireless access point has about half the effective bandwidth of a wired bridge, because all frames received on the radio channel must be forwarded on the same channel. Therefore, using a wireless access point exchanges performance for ease of installation.

Figure 7-1 shows how a wired bridge overlaps coverage with a wireless access point. Note how the coverage areas of the devices overlap more than the areas of wired bridges.

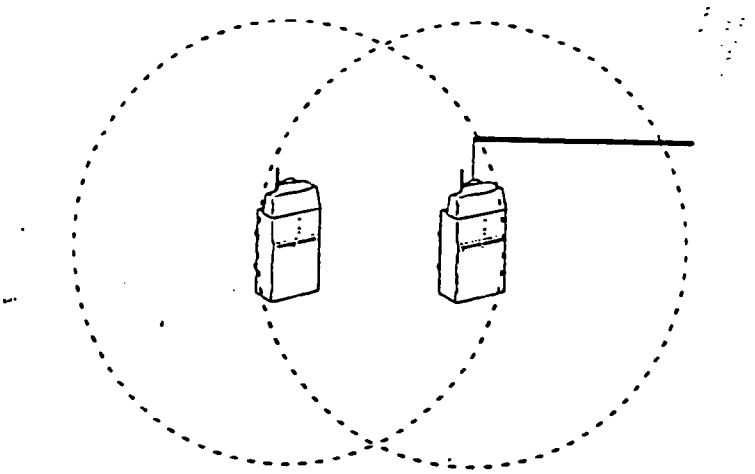


Figure 7-1
Overlapping Coverage of Wired Bridge and Wireless Access Point

When wired and wireless access points overlap coverage, wireless stations will automatically switch between them. Wireless stations base their choice on which device offers the best path. They choose a wired bridge over a wireless access point, for example.

For best coverage, wireless access points are usually mounted high on a wall or post, or on the ceiling, to do the following:

- Expand the coverage area
- Reduce the amount of cable
- Cover fringe areas
- Provide redundancy
- Meet temporary coverage needs

Expanding the Coverage Area

Multiple wireless access points can expand the coverage area. However, each additional wireless access point introduces an additional wireless hop and a corresponding reduction in throughput or an increase in response time. Norand does not recommend extensive use of wireless hops in performance-critical areas. Generally, a single wireless hop will not result in a discernable reduction in performance, unless extensive data transfers are required.

Figure 7-2 shows multiple access points and coverage areas. In the figure the wireless station is associating with wireless access point A. A forwards frames to wireless access point B, which forwards them to access point C (a bridge wired to the Ethernet medium). A network with this many hops would have some performance limitations.

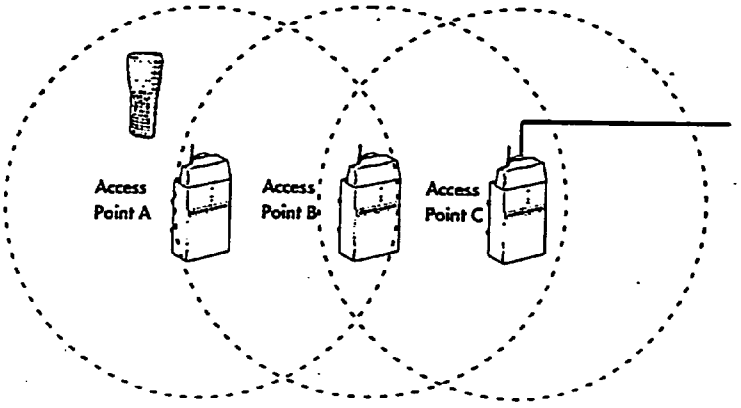


Figure 7-2
Expanding the Coverage Area Through Wireless Access Points

Reducing the Amount of Cable

Wireless access points can reduce the amount of cable needed. The design in Figure 7-3 shows how four wireless access points (in black) reduce the amount of cabling in a warehouse.

▶ NOTE: *This symbol represents an access point:*

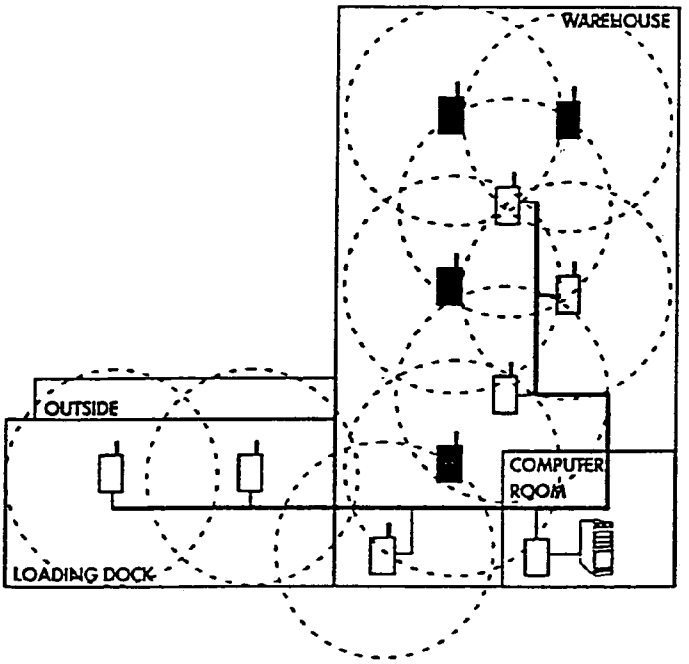


Figure 7-3
Reducing the Amount of Cable Through Wireless Access Points

This example has several advantages:

- It needs more wireless access points, but they are not wired to a physical medium.
- It does not need a repeater.
- It reduces the cable length.
- It results in only one hop from any area.

Wireless access points are not located in the loading dock because the traffic there is heavy and top performance is required.

Covering Fringe Areas

A wireless access point can provide coverage in fringe areas. For example, if an area has marginal coverage, you could mount one or two wireless access points in the area. No cabling would be required.

Figure 7-4 shows how two additional wireless access points (in black) fill in fringe areas on the outside of the loading dock. Heavy dashed circles show their coverage areas.

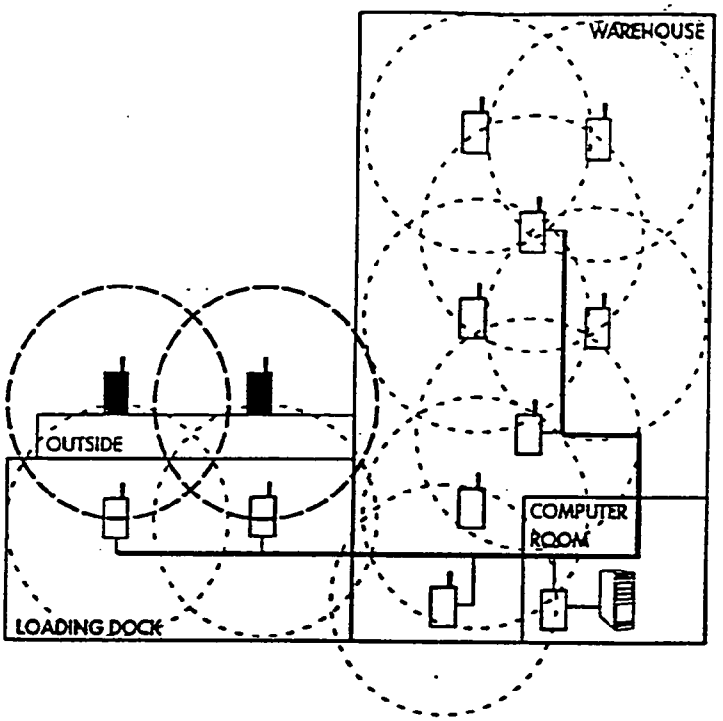


Figure 7-4
Covering Fringe Areas Through Wireless Access Points

Providing Redundancy

Another benefit of a wireless access point is *redundancy*. If properly designed, an installation with wireless access points can ensure that no single device failure or single cable cut can stop service. You can reduce costs by providing redundancy only in selected areas. A site planning to install one or more wireless access points should coordinate the effort with their Norand representative.

Figure 7-5 shows an example of how an access point (in black) wired to 10BASE2 can become a wireless access point and continue coverage if it becomes disconnected from the physical medium.

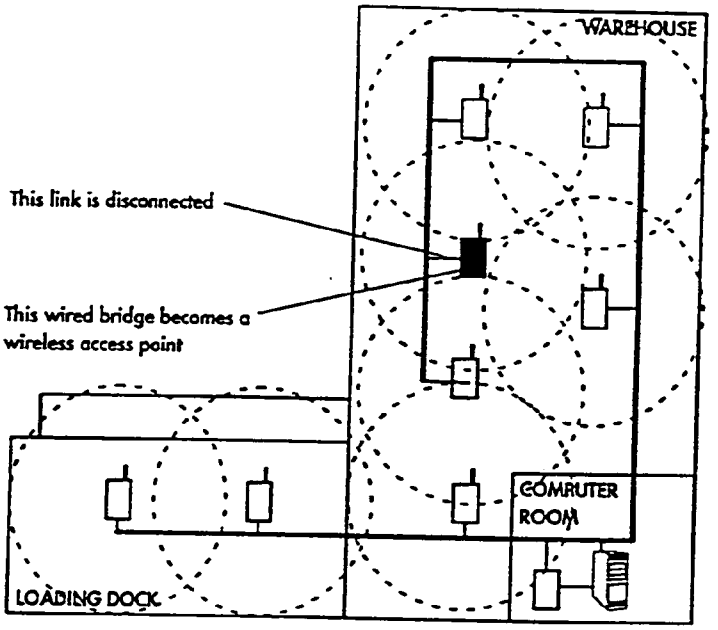


Figure 7-5
Warehouse With Wireless Access Point on 10BASE2

Figure 7-6 shows an example of how an access point wired to 10BASE-T can become a wireless access point and continue coverage if its link to the hub is cut.

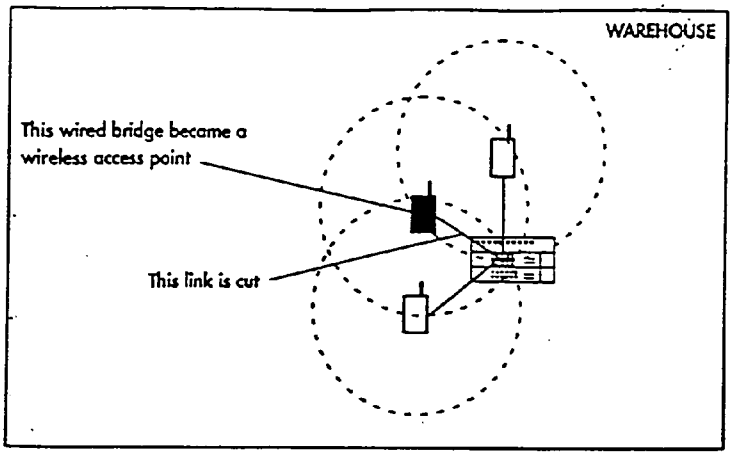


Figure 7-6
Warehouse With Wireless Access Point on 10BASE-T

Meeting Temporary Coverage Needs

Wireless access points can handle a temporary need without the effort or expense of additional cabling. For example, the wireless access point shown in black in Figure 7-7 can provide coverage in a temporary addition.

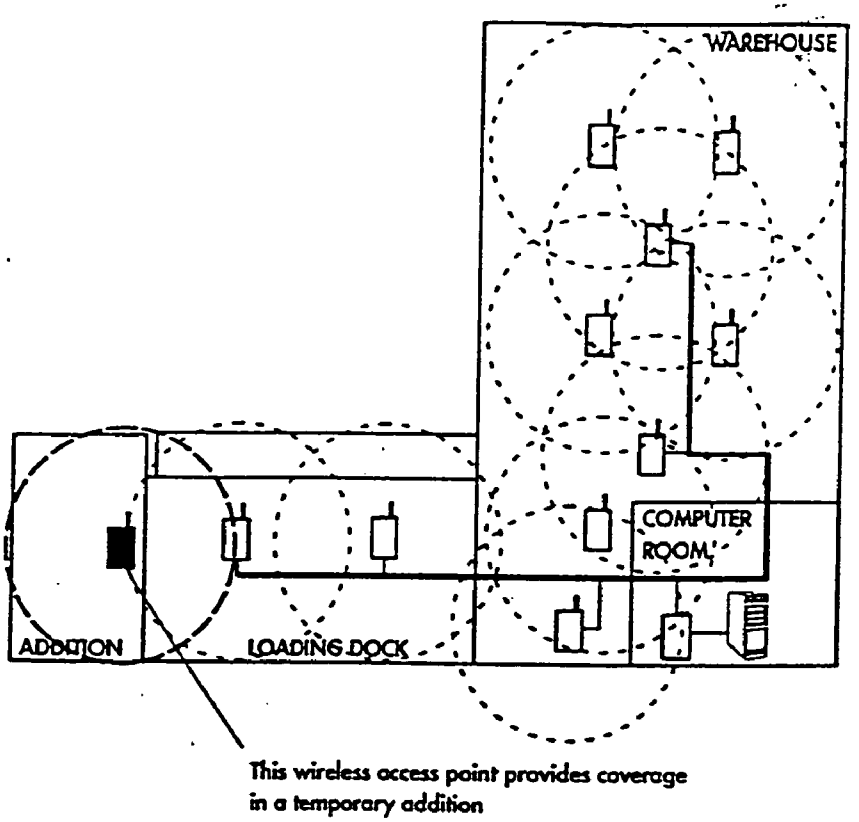


Figure 7-7
Wireless Access Point Meeting Temporary Need

One of the advantages of wireless access points is the ease with which you can install them. To improve coverage in a dead spot, you could temporarily install a wireless access point to see if it helps. If it solves the coverage problem but the performance is unacceptable, you could wire to the physical medium.

If the coverage at your site is satisfactory but the performance of the wireless access points needs to improve, you could connect them to the physical medium. You could also use a wireless access point to experiment with coverage by moving the bridge around, before running cables.

Section 8

Installation

About This Section

Each network installation is unique because each site has different computing equipment and requirements. In general, an installation should emphasize modular cabling systems and a network topology you can easily configure, maintain, and update to meet changing application needs.

Before you install the wireless infrastructure you should consider these strategies:

- Conducting a site survey
- Selecting the Ethernet medium
- Extending the network
- Using existing media
- Using routers

This section has useful overviews of each strategy. The overviews help you design a flexible, open wireless LAN with NORAND® wireless infrastructure components and off-the-shelf network devices such as bridges, routers, and repeaters.

This section also shows how wireless infrastructure components and host connectivity devices connect to Ethernet media. Examples of some simple warehouse and retail installations follow the illustrations.

Conducting a Site Survey

Norand strongly recommends that you conduct a site survey to determine the network solutions for your site. A site survey (such as those conducted by Norand or certified providers) requires special equipment and training. Because it helps you build a system capable of supporting your traffic requirements, the site survey can improve the performance of the open wireless LAN.

A site survey determines the optimum number and placement of 6710 Access Points to provide reliable wireless coverage of your facility. Many factors affect coverage, including the floor plan, building construction, usage needs, and materials or equipment stored or used within the environment.

This section does not provide the detailed information you may need if you have or will have a large network. In this case, you should contact your Norand representative or a certified provider for more information.

Selecting the Ethernet Medium

The following pages provide basic facts about 10BASE2, 10BASE-T, and 10BASE5. For more specific information about these cable types and their electrical and mechanical requirements, refer to the ANSI/IEEE 802.3 standard.

▶ NOTE: *Norand recommends that sites use only ANSI/IEEE 802.3 standard media and equipment. ANSI/IEEE standards promote configuration guidelines with performance expectations. Vendors should be able to provide the ANSI/IEEE standards and specifications that apply to their products.*

10BASE2

10BASE2 or "thinnet" is a commonly-used type of Ethernet medium. It is popular for several reasons including the following:

- It is highly flexible, which makes it easy to work with.
- It is less susceptible to electromagnetic interference than 10BASE-T.
- It does not require hubs (as 10BASE-T does). This lowers the cost of an attachment to the network.

10BASE2 has some disadvantages. One is that the length of each cable segment is limited to about 607 feet (185 meters) or one-tenth of a mile. Repeaters and bridges, however, can extend the length limit. Repeaters and bridges are covered later in this section.

Cable Characteristics

Cables approved by ANSI/IEEE for use with 802.3 10BASE2 Ethernet are:

- 802.3 10BASE2: 50 ohms, stranded tinned core
- RG58 A/U: 50 ohms, stranded tinned core
- RG58 C/U: 50 ohms, stranded tinned core

RG58 and RG58 U coaxial cables have solid center cores. They physically resemble RG58 A/U cable but do not comply with ANSI/IEEE Std 802.3 for 10BASE2. Therefore, RG58 and RG58 U cables should not be used.

▶ NOTE: *When obtaining 10BASE2 coaxial cable, request that it be certified for use with ANSI/IEEE Std 802.3 for 10BASE2. Cheap substitutions are not reliable.*

T-connectors

The 6710 Access Point, RC4030E Gateway, and 6950 Enterprise Gateway Server have industry-standard BNC ports for quick, easy connection to 10BASE2. A T-connector attaches to the BNC port and to the 10BASE2 cable.

T-connectors are available in three impedances: 50 ohms, 75 ohms, and 93 ohms. The difference among them is the diameter of the center pin. Mating T-connectors with different impedances can damage one or both connectors or result in an unreliable connection.

The 10BASE2 RG58 A/U or C/U coaxial cable is 50 ohms. Therefore, you must use 50-ohm T-connectors to connect the fixed-end devices to 10BASE2. A T-connector cover insulates the T-connector from electrostatic discharge when it is not in use.

Cable Terminators

T-connectors on fixed-end devices at each end of the coaxial cable must be fitted with 50-ohm cable terminators, which maintain the impedance of the network. The network will function properly only if each end has an attached cable terminator. The terminator has a BNC coaxial connector.

Many repeaters (and some bridges and routers) have built-in terminators. Bridges and routers often have a switch that enables or disables the terminator, because they are not always at the ends of segments. If the last repeater or bridge on the segment has a built-in terminator, you should not use an external one. The network will not operate properly if more than two terminators are installed on a segment.

Segment Rules

A 10BASE2 cable segment is the length of cable between cable terminators. According to ANSI/IEEE standard specifications the following rules apply:

- The length of each segment can be no longer than about 607 feet (185 meters).
- 30 or fewer network devices can be attached to a segment. Each repeater counts as a network device.
- Cable runs between 10BASE2-connected devices must be 1.64 feet (0.5 meters).

These rules ensure that signal losses are within acceptable limits. They also limit the amount of signal attenuation and distortion on a segment. The standard 185-meter length lets you use 10BASE2 Ethernet components that conform to the ANSI/IEEE standard.

Topology

10BASE2 networks form a bus topology. The coaxial cable forms a line that connects each network device. T-connectors daisy-chain the 10BASE2 cable from one network device to the next, with cable terminators at the ends of each segment. Repeaters join 10BASE2 segments end-to-end; they can also be installed in the middle of a segment. Bridges and routers can join segments in the middle to form X- or T-shaped networks.

Summary of 10BASE2

Table 8-1 summarizes 10BASE2 characteristics. Refer to ANSI/IEEE 802.3 standard specifications for detailed information about design and installation rules.

Table 8-1
10BASE2 Characteristics

Feature	Description
ANSI/IEEE standard	802.3 10BASE2
Data rate	10 Mbps
Topology	Bus (linear)
Maximum cable segment length (without repeaters)	607 feet (185 meters)
Maximum network length (without bridges)	3034 feet (925 meters)
Maximum number of segments	5 (only 3 can be populated)
Maximum number of repeaters	4
Maximum number of network devices per segment	30
Maximum number of network devices per network	1024
Minimum distance between network devices	1.64 feet (.5 meters)
Cable type	RG58 A/U, 0.2 inches diameter, single shielded

10BASE-T

10BASE-T is a popular Ethernet medium. Some reasons are:

- It is easy to install.
- Its star-wired topology makes it easy to troubleshoot because problems can be isolated to a cable or hub (also called a concentrator or multiport repeater).
- A cable break from the hub to a network device disables only the network device at the end of the cable.
- You can easily expand the network by connecting several hubs to each other. This topology is called a modified star configuration, and creates a geographically dispersed network.
- Intelligent hubs provide maintenance, monitoring, and management capabilities unavailable for most other cabling schemes.

The star-wired topology has some disadvantages. One is that additional cabling and connection equipment (such as hubs) are required. Another disadvantage is that a cable break between two connected hubs can bring down many network devices.

Because 10BASE-T uses unshielded cable, problems may develop in electrically-noisy environments such as heavy industrial areas. If this happens the 10BASE-T cables can be shortened, relocated away from the noise source, put in metallic conduits, or shielded. However, the best solution may be to change to another form of Ethernet.

Cable Characteristics

10BASE-T uses unshielded twisted pair (UTP) cable. Sites should use only data grade UTP cable that is verified UTP Category 3, 4, or 5 and installed per IEA/TIA 568–569.

10BASE-T cable is often installed using various telephone wiring methods. The methods followed to form the connection from the device to the hub vary with local practices. However, the connection provided at the fixed-end device's location should always be an RJ45 modular jack.

▶ NOTE: *The cable from the network jack to the network device must be a twisted-pair cable. Avoid using the nontwisted cables often used for telephones.*

The length limit from the network device to the hub is just 328 feet (100 meters). There may already be close to that in the wall extending from the jack back to the hub. So, it is best to keep the cable from the network jack to the network device as short as reasonably possible.

RJ45 Plug

The standard connector for 10BASE-T is an RJ45 modular plug. The plug physically resembles the RJ11 modular plug used to plug most telephones into the wall. The 6710 Access Point, RC4030E Gateway, and 6950 Enterprise Gateway Server have standard RJ45 ports for connection to 10BASE-T.

Connecting to Hubs

A hub is a repeater. On a 10BASE-T network, hubs are multiport repeaters.

Topology

10BASE-T networks form a star topology. Each network device has a separate cable that extends from a central hub to the network device, with the hub at the "center" of the star. Hubs can connect to each other to extend the network beyond the number of network devices that a single hub supports. Multiple hubs connected together form a modified star topology.

Summary of 10BASE-T

Table 8-2 summarizes 10BASE-T characteristics. Refer to ANSI/IEEE 802.3 standard specifications for detailed information about design and installation rules for 10BASE-T.

Table 8-2
10BASE-T Characteristics

Feature	Description
ANSI/IEEE standard	802.3 10BASE-T
Data rate	10 Mbps per second
Topology	Star
Maximum number of segments	5

Table 8-2 (Continued)
10BASE-T Characteristics

Feature	Description
Maximum number of repeaters	4
Maximum cable length between network device and hub	328 feet (100 meters)
Maximum number of network devices/segment	512
Minimum space between network devices	10 feet (.5 meters)
Connector type	RJ45
Twisted pair cable	

10BASE5

10BASE5 is also called "thicknet." Generally, it is only used in specific applications because:

- It uses RG8 coaxial cable, which is heavy and rigid.
- 10BASE5 transceivers (also called media attachment units, or MAUs) tend to be physically heavy.
- 10BASE5 N-connectors and vampire taps are large and difficult to install under field conditions.

10BASE5 does have some advantages. One is that segments can be up to about 1640 feet (500 meters). Also, the cable flow can be easily followed when troubleshooting. And 10BASE5 does not require hubs as 10BASE-T does.

10BASE5 is most often used to cover long distances as a backbone cable. A "backbone" is the main cable installed in a building to provide wired connectivity to separate areas. Backbone cables are usually not designed for direct system access.

▶ NOTE: *Refer to ANSI/EEE 802.3 standard specifications for detailed information about design and installation rules for 10BASE5.*

Extending the Network

Following are standard segment length limits:

- 10BASE2 is 607 feet (185 meters)
- 10BASE-T is 325 feet (100 meters)

The following pages describe how repeaters and bridges can extend these standard segment lengths.

Repeaters

A repeater is a local network device that extends the LAN length and topology by increasing the distance a LAN can extend and joining two different topologies. Repeaters are simple to install and do not need to be configured.

Operation

Repeaters operate at the Physical layer of the OSI model by receiving data from one segment, amplifying the data, and putting the data out on the next segment. A repeater does not process the data and retransmit it, but simply amplifies it along with any noise on the cable. Because of the noise amplification only a few repeaters can be used within a segment. This requirement establishes the maximum length of an Ethernet segment for the different Ethernet media.

The repeater makes all segments operate as though they were a single segment, or network. This is called a single Ethernet "collision domain." A single collision domain makes it possible for network devices connected to any segment in a system of segments linked by repeaters to hear the same signals, and to operate as a single segment.

A repeater prevents signal loss but extends the network so that signals take longer to travel from one end of the cable to the other. The CSMA/CD protocol depends on all network devices to be able to hear transmissions from other devices within certain timing parameters.

If the segment length is longer than the maximum allowed, a network device's moment of listening may not be long enough for it to hear a transmission from another device. The longer the moment, the longer the network device has to wait before it can transmit. This decreases network effectiveness and is why the Ethernet standard allows only four repeaters on 10BASE2 and 10BASE-T networks.

The moment is long enough for the signal to travel five segments and four repeaters.

Repeater Rules

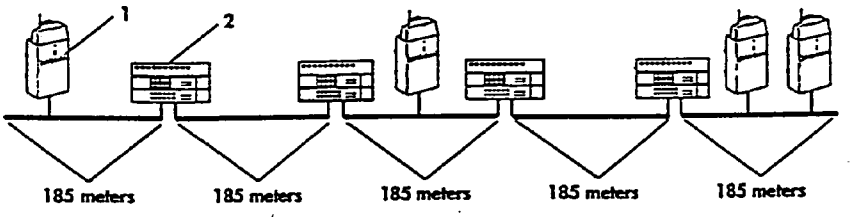
The Ethernet repeater rule is:

> Five segments maximum, interconnected by 4 repeaters (or hubs) in the data transmission path between any two devices in the same collision domain. Three of the segments are populated with nodes, and 2 are interrepeater link segments (for distance). This creates 1 collision domain.

Repeaters on 10BASE2

10BASE2 standards allow a maximum of four repeaters on the network. Four repeaters join five segments. The total network span with five segments is about 3000 feet (925 meters) long. No more than four repeaters can be on the signal path from one network device to any other network device. Figure 8-1 shows a single 10BASE2 network with five segments and four repeaters.

▶ NOTE: *If the repeated network exceeds the Ethernet design limitations (5 segments with 4 repeaters), late collisions and lost packets will occur.*



1. 6710 Access Point
2. Repeater

Figure 8-1
10BASE2 Network With Four Repeaters

Repeaters on 10BASE-T

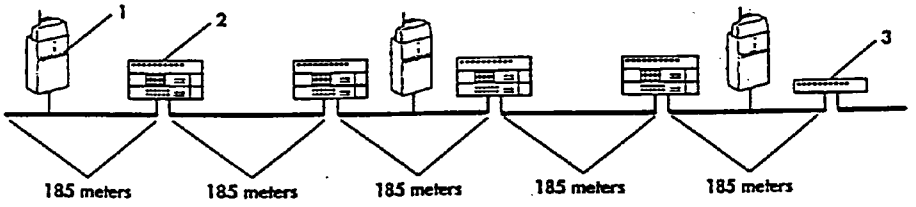
Repeaters with RJ45 (10BASE-T) ports are 10BASE-T hubs. Most hubs support from 8 to 144 ports; however, some vendors offer mini-hubs, which provide 1 to 4 UTP ports for link extension. Each port on the repeater should support the full extent of the ANSI/IEEE specification for 10BASE-T.

Bridges

To go beyond five cable segments for 10BASE2 networks, you need a bridge to receive a packet and then retransmit it on another segment. The number of bridges that can be installed depends on the protocol the bridge is using.

Bridges are intelligent devices that operate at the Data Link layer of the OSI model. They filter packets and provide error checking. Normally, bridges connect segments of similar media and protocol types.

10BASE2 networks can have a maximum of five segments and four repeaters. After the fourth repeater you need to install a bridge to recover the CSMA/CD timing. After the bridge you can install five more cable segments and four repeaters. After the fourth repeater you need to install a second bridge. Figure 8-2 shows a 10BASE2 network with one bridge.



1. 6710 Access Point
2. Repeater
3. Bridge

Figure 8-2
10BASE2 Network With One Bridge

Hybrid Topologies

A larger network span can be created through a hybrid topology, which is created when different topologies are connected.

EXAMPLE: A router can connect 10BASE-T to a bus topology (such as 10BASE2) to form a hybrid topology.

Although 10BASE2 is intended for local use, repeaters can join several segments together to create a larger network span. Similarly, several 10BASE2 segments can be tied into a longer Ethernet backbone.

Some benefits of a hybrid topology are:

- It connects systems that may not be otherwise accessible.
- It connects existing networks so that resources can be shared without installing new cable.

A hybrid topology has some disadvantages.

- It requires bridges and routers to connect the networks.
- More than one person may be required to troubleshoot problems because of different technologies.

Using Existing Media

Some sites already have Ethernet cabling and network devices installed throughout their buildings. The sites may want to use the existing medium for their fixed-end devices, or connect two different topologies. If you are thinking about using the existing cable at your site, you should research answers to the following questions.

- *What type of cable is installed at the site?*

  Sometimes three different types (such as 10BASE5, 10BASE2, or 10BASE-T) are in use. The most important question is which type you will want to connect your fixed-end devices to. The 6710 Access Point and 6950 Enterprise Gateway Server have ports for connection to 10BASE2, 10BASE5, or 10BASE-T. The RC4030E Gateway has ports for connection to 10BASE2 and 10BASE-T.

- *Is the cable physically installed where the devices will be installed?*

  Ethernet cable is usually installed at floor level where most computer equipment is located. So that it can provide maximum radio coverage, the 6710 Access Point is usually installed near the ceiling or in the rafters. Because the RC4030E Gateway has no radio module or antenna, it is usually installed next to the host computer in a computer room.

  Because running connections to access points mounted in the rafters may not be feasible, separate cable can be installed for the access points. Later, the cable with the access points can be bridged to another cable at one point.

- *How heavily utilized are the cable segments the devices will be communicating over?*

  Good response time from wireless stations cannot be expected if fixed-end devices must communicate over an overloaded network. Special instruments can measure network utilization over several days.

Using Routers

Routers are protocol-dependent devices that connect networks and selectively forward packets based on Network layer addresses. For example, a router may forward only IP or IPX packets.

Routers can connect dissimilar protocols and media. They use the Network ID part of the internet address to make the routing decision on each Ethernet packet. Each router exchanges information about the entire network with other routers to maintain current data on the paths through the network. In a network with multiple paths between two devices, a router may select the best path for communications.

Routers forward packets based on path availability, traffic loads, and other factors. If a network device sends two packets to another network device on a network with routers, those packets may take entirely different paths to get from the first network device to the second. Several factors can cause a route from one network device to another to become unavailable.

For example, a network segment may become unavailable if a cable breaks or becomes disconnected, if a repeater or bridge fails, or if a phone line goes down.

Some routers can be configured. For example, inter-access point communications can be enabled by configuring the router to bridge DIX type 0875C packets. NORAND terminal emulation can be enabled by configuring the router to bridge 0875B packets.

Installing Wireless Infrastructure

The following pages describe these installation strategies for the 6710 Access Point as part of the wireless infrastructure:

- Location of the access point
- Mounting options
- Power requirements
- Ethernet connectivity solutions

▶ NOTE: *A person who understands applicable local building codes and is skilled with the tools and equipment used to install FCC Class B electromechanical network devices should install the 6710 Access Point.*

Location

You should locate each 6710 Access Point where it will provide the best performance. The following pages discuss these location strategies:

- Centrally locating the access point
- Using a remote antenna
- Protecting the access point in harsh environments
- Minimizing obstructions
- Resolving other location issues

Centrally Locating the Access Point

The access point should be centrally located within the group of wireless stations. This enables all wireless stations to be within the access point's coverage area. The location for the access point must meet the requirements in Appendix D, "6710 Access Point Specifications."

Using a Remote Antenna

You can use a remote antenna for coverage if you cannot mount an access point in an ideal location. An ideal location in a warehouse, for example, might be near the ceiling at an intersection of aisles. You can mount the antenna near the ceiling in the center of the aisle, but must secure the access point to a column structural support nearby.

NICs that are remote-enabled have a connector for a remote antenna. The antenna attaches to a special connector on the access point's cover either directly or remotely through a coaxial cable. Norand or other qualified personnel must install external antennas.

▶ NOTE: *Only antennas that Norand furnishes may be used with NORAND 6710 Access Points.*

Protecting the Access Point

Norand designed the access point for operation in an enclosed, weather-proof environment. If you need to install an access point in an area that may expose it to excessive heat, humidity, and other harsh elements (such as rain and snow), you can house the access point in an enclosure. The enclosure must meet National Electrical Manufacturing Association (NEMA) standards for environmental protection. Figure 8-3 shows the access point in a NEMA enclosure, which is available from Norand.

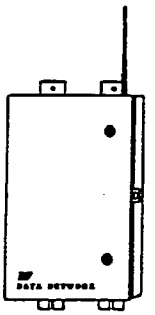


Figure 8-3
NEMA Enclosure

Minimizing Obstructions

You should locate the access point so that a clear line of sight is between the NIC's antenna and the wireless stations. Walls (especially steel reinforced concrete or masonry), floors, office partitions, and other obstructions reduce the effective communications range. You should place the access point where the number of barriers between it and the wireless stations it is communicating with is minimal.

If walls and office partitions prevent you from centrally locating the access point, you should mount the access point as high as possible in another suitable location. In addition, the installed access point must be at least 2 feet from any objects that might negatively affect radio transmissions. Objects include large metal structures and fluorescent lights.

Resolving Other Location Issues

When selecting the best location for each access point keep these other issues in mind:

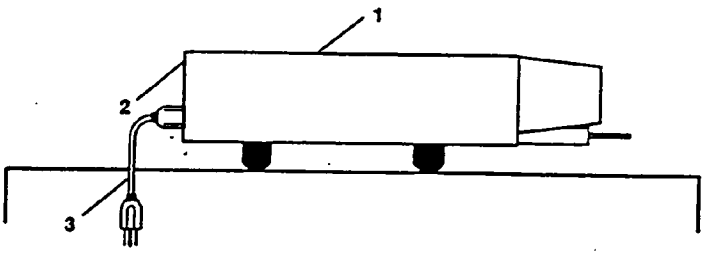
- A person standing on the floor or on a ladder under the access point should be able to easily see the access point's LEDs. The LEDs are useful for troubleshooting and verifying certain operating conditions.
- Leave enough room around each access point so that you can easily connect communication, diagnostic, and power cables to it.
- If possible, another outlet should be available near each access point for LAN test equipment if troubleshooting is necessary.
- The network cabling and access point's power cord must be able to reach the access point after you install it.

Mounting Options

You can mount the access point horizontally on a tabletop, vertically on a wall or post, or on a ceiling. Norand recommends that you mount the access point vertically so that it will be drip-resistant. An access point in any other position must be protected from dripping fluids.

Horizontal Mount

The mounting bracket on the bottom of the access point is not needed for a tabletop installation; you can remove and save the bracket for future use. When the access point is on the tabletop, four rubber feet on the access point's back panel keep it from slipping out of place (Figure 8-4).



1. 6710 Access Point
2. Bottom panel
3. Power cable

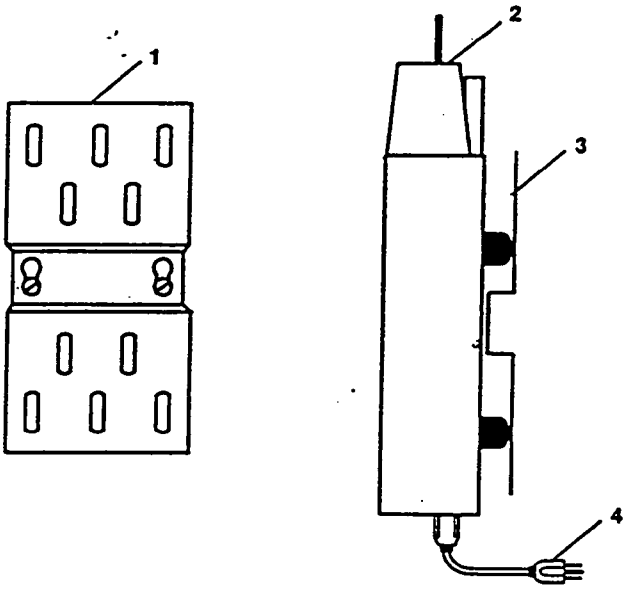
Figure 8-4
6710 Access Point Horizontal Mount

Vertical Mount

Norand supplies a standard mounting bracket with each access point. The bracket secures the access point to a wall, post, or ceiling. For access points mounted vertically, the type of wall or post determines the hardware needed to mount them. Different surfaces (such as dry-wall, wood, concrete block) require different types of screws and other hardware. For these reasons Norand supplies only a mounting bracket with each access point. The site's personnel must supply the screws and other appropriate hardware.

You must remove the access point's standard mounting plate before you mount the access point on a wall or post. The mounting plate is a template. You can use the template to mark the location of the anchors that secure the mounting plate to the surface. After the mounting plate is securely attached to the surface, the access point is reattached to it.

Figure 8-5 shows an access point mounted vertically with its antenna oriented in an upward position. Norand or certified providers can conduct a formal site survey to determine the proper orientation of antennas for best performance.



1. Mounting bracket
2. 6710 Access Point
3. Mounting bracket (side view)
4. Power cable

Figure 8-5
6710 Access Point Vertical Mount

Power Requirements

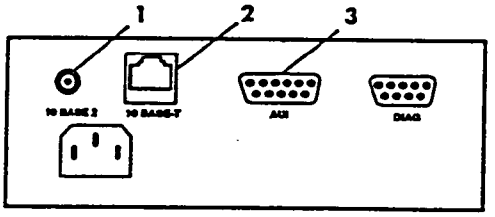
The ac INPUT connector on the bottom panel of the access point is a standard, IEC-type 3-prong connector. The power cord attaches to this connector.

A built-in switching power supply with a source voltage of 85–264 V ac and frequency between 47–63 Hz powers the access point. The power supply supports 110 V ac for operation in domestic markets, and 220 and 240 V ac for international markets. The power supply autosenses the level (110, 220, or 240 V ac) and frequency of the source voltage and operates accordingly.

The dc power cable is 6 feet long. You should locate the access point within 6 feet of the outlet.

Ethernet Connectivity Solutions

The 6710 Access Point connects to the wireless infrastructure through Ethernet and supports 10BASE2, 10BASE5, and 10BASE-T media options. Connections are through the network ports on the access point's bottom panel (Figure 8-6).



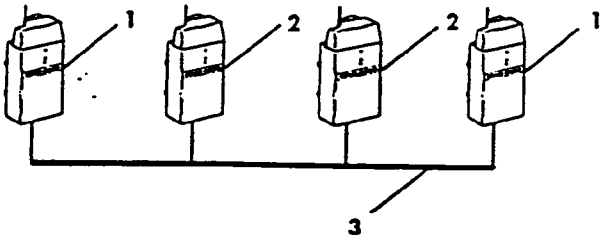
1. BNC port (10BASE2)
2. RJ45 port (10BASE-T)
3. AUI port (10BASE5)

Figure 8-6
6710 Access Point Network Ports

The following pages show how the 6710 Access Point connects to 10BASE2 and 10BASE-T, and how it is installed as a wireless access point.

10BASE2 Connections

The access point can connect to the middle or end of a 10BASE2 segment. Figure 8-7 shows connection options.



1. Access point at end of segment
2. Access point in middle of segment
3. 10BASE2 coax

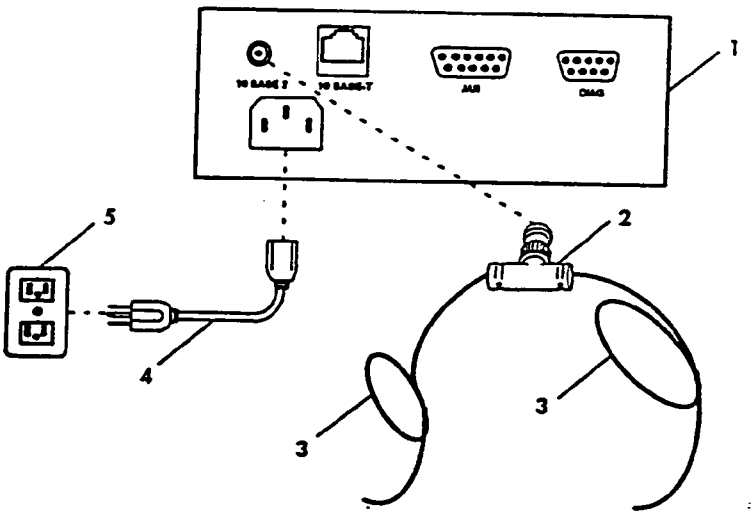
Figure 8-7
6710 Access Point Connection Options

Middle of 10BASE2

An access point that connects to the middle of the 10BASE2 segment requires the following parts:

- T-connector
- 10BASE2 coax
- Power cord
- 110, 220, or 240 V ac outlet

Figure 8-8 shows how the parts connect.



1. 6710 Access Point, bottom panel
2. T-connector
3. 10BASE2 coax
4. Power cord (6 feet long)
5. 110, 220, or 240 V ac

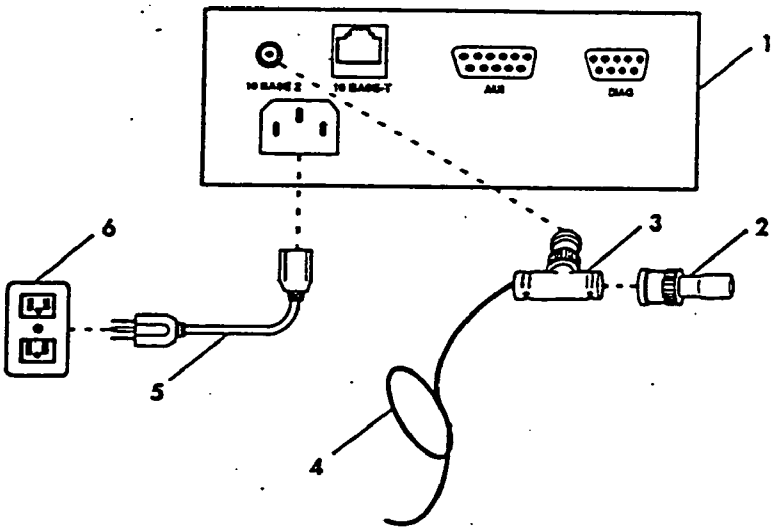
Figure 8-8
6710 Access Point in Middle of 10BASE2

End of 10BASE2

An access point that connects to the end of the 10BASE2 segment requires the following parts:

- T-connector and cable terminator
- 10BASE2 coax
- Power cable
- 110, 220, or 240 V ac outlet

Figure 8-9 shows how the parts connect.



1. 6710 Access Point, bottom panel
2. Cable terminator
3. T-connector
4. 10BASE2 coax
5. Power cord (6 feet long)
6. 110, 220, or 240 V ac

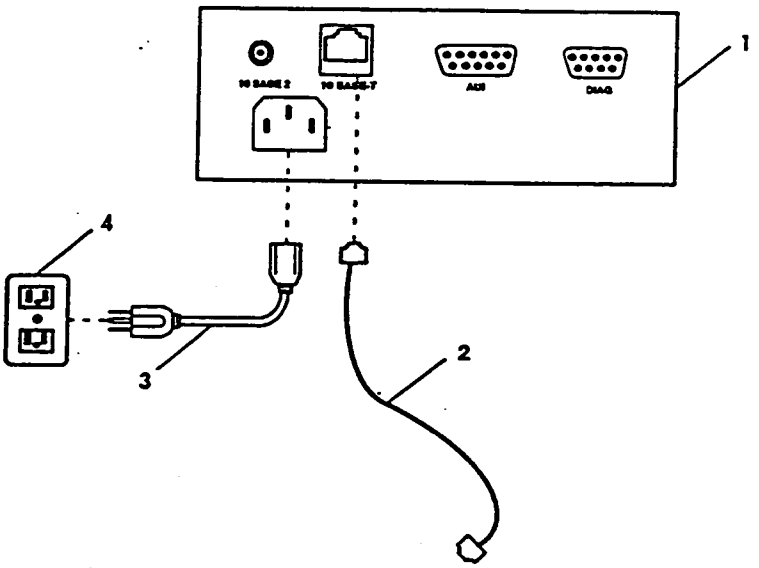
Figure 8-9
6710 Access Point at End of 10BASE2

10BASE-T Connection

An access point that connects to 10BASE-T requires the following parts:

- Cable with RJ45 plugs
- RJ45 jack
- Power cord
- 110, 220, or 240 V ac outlet

Figure 8-10 shows how the parts connect.



1. 6710 Access Point, bottom panel
2. Cable with RJ45 plugs
3. Power cord (6 feet long)
4. 110, 220, or 240 V ac

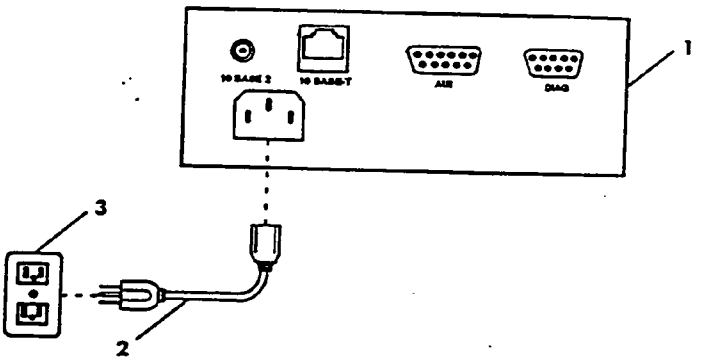
Figure 8-10
6710 Access Point Connected To 10BASE-T

Wireless Access Point Installation

You would typically mount a wireless access point on a ceiling, or high on a wall or post. A wireless access point requires only the following parts:

- Power cord
- 110, 220, or 240 V ac outlet

Figure 8-11 shows how the parts connect.



1. 6710 Access Point, bottom panel
2. Power cord (6 feet long)
3. 110, 220, or 240 V ac

Figure 8-11
6710 Access Point as Wireless Bridge

Installing Host Connectivity Devices

The following pages describe these installation strategies for NORAND gateway products:

- Location
- Mounting options
- Power requirements
- Ethernet connectivity solutions
- Host connectivity solutions

RC4030E Gateway

The following pages describe installation strategies for the RC4030E Gateway.

Location

You would typically locate the RC4030E Gateway next to the host computer in a computer room. The location must meet the requirements in Appendix E, "Host Connectivity Device Specifications."

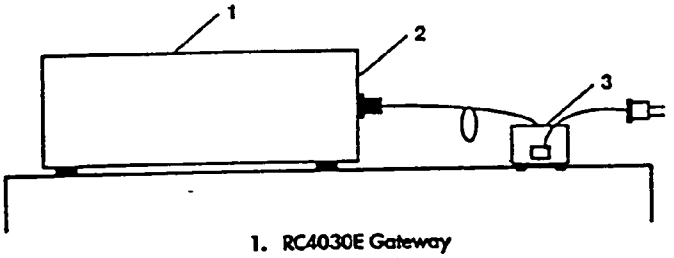
Norand designed the gateway for operation in an enclosed (weatherproof) environment. Norand does not recommend that you locate the gateway in an area that may expose it to harsh operating conditions such as rain, snow, and excessive heat or humidity.

Mounting Options

You can mount the gateway horizontally on a tabletop or vertically on a wall or post. Norand recommends that you mount the gateway vertically so that it will be drip-resistant. A gateway in any other position must be protected from dripping fluids.

Horizontal Mount

The mounting bracket provided with the gateway is not needed for a tabletop installation. When the gateway is on the tabletop, four self-adhesive rubber feet on the gateway's back panel keep it from slipping out of place (Figure 8-12).



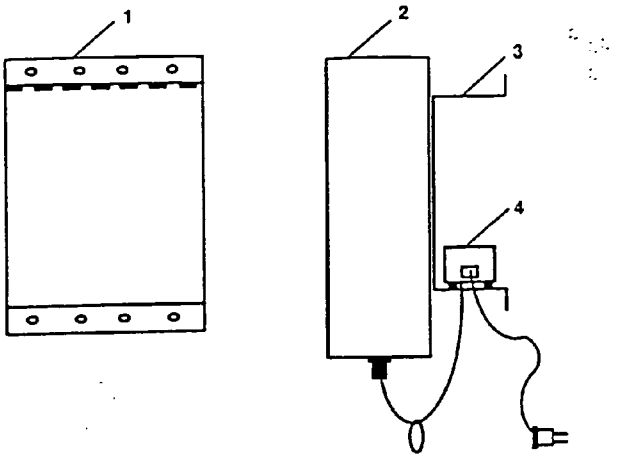
1. RC4030E Gateway
2. Bottom panel
3. NC3100 Power Supply

Figure 8-12
RC4030E Gateway Horizontal Mount

Vertical Mount

Norand provides a mounting bracket with the gateway so that it can mount onto a wall or post. Because the gateway does not require a line of site to the terminal emulation stations, you can mount it at any height. The power supply is usually placed behind the gateway on the mounting bracket.

Figure 8-13 shows a gateway mounted vertically, with the power supply on the bracket.



1. Mounting bracket (front)
2. RC4030E Gateway
3. Mounting bracket (side view)
4. NC3100 Power Supply

Figure 8-13
RC4030E Gateway Vertical Mount

The LEDs on the bottom panel of the gateway are helpful troubleshooting aids. They should be easily visible.

Power Requirements

The RC4030E Gateway requires an NC3100 Power Supply. The power supply has the following characteristics for the United States market:

- Domestic 120 V ac
- 60 Hz power, UL and CSA approval
- Required filtration to meet FCC emissions requirements

The power supply must be located within 6 feet of the gateway and 6 1/2 feet of the power outlet. These distances ensure that the power supply's cables will reach the gateway and outlet. Typically, you would put the power supply on the gateway's wall mounting bracket. If this is infeasible, you can install an extra mounting bracket for the power supply. Norand provides extra mounting brackets.

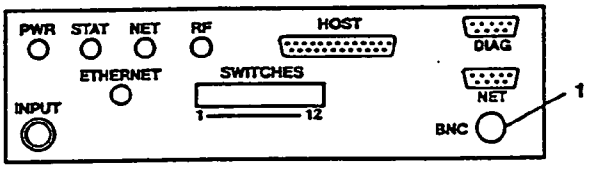
If an outlet cannot be located within 6 feet of the gateway, the dc power cable can be extended. Because of the voltage drop in these cables the extension should not exceed 25 feet. Norand provides assembled dc power extension cables in lengths of 6, 12, and 25 feet.

A different NC3100 Power Supply is used in Europe and other areas having 230 V ac, 50 Hz power with TUV approval. Another type of NC3100 Power Supply is used in Japan and other areas having 100 V ac, 50 or 60 Hz power with MITI approval.

Ethernet Connectivity Solutions

The RC4030E Gateway connects to the wireless infrastructure through Ethernet, and supports 10BASE2 and 10BASE-T media options.

The gateway connects to 10BASE2 through the BNC port on its bottom panel. It connects to 10BASE-T through a media converter. Figure 8-14 shows where the BNC port is located.



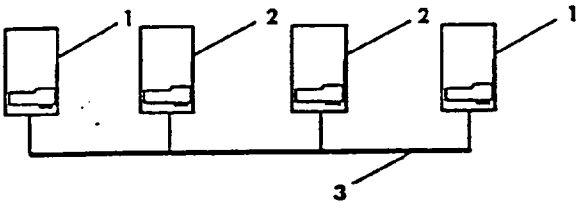
1. BNC port

Figure 8-14
RC4030E Gateway Network Port

The following pages show how the gateway connects to 10BASE2 and 10BASE-T. In some figures the power supply is shown beside the gateway.

10BASE2 Connections

The gateway can connect to the middle or end of a 10BASE2 segment. Figure 8-15 shows connection options.



1. Gateway at end of segment
2. Gateway in middle of segment
3. 10BASE2 coax

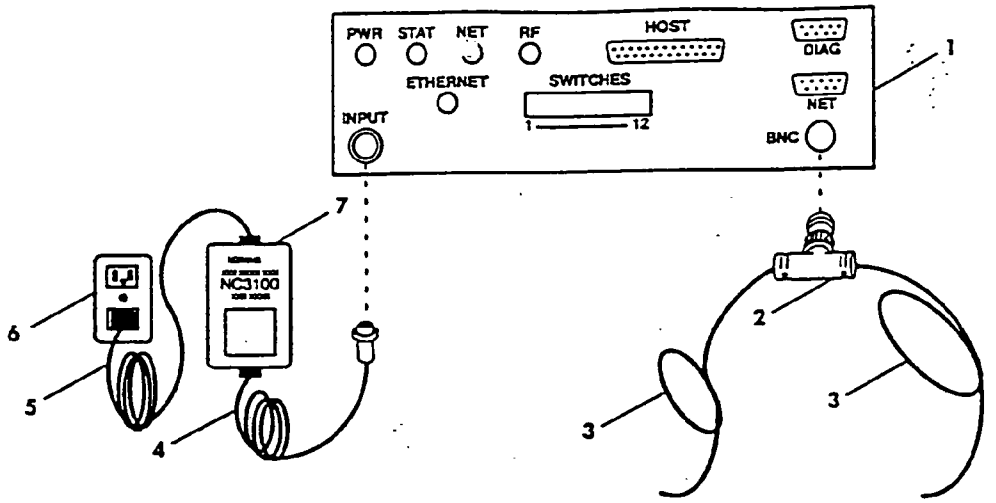
Figure 8-15
RC4030E Gateway Connection Options

Middle of 10BASE2

A gateway that connects to the middle of the 10BASE2 segment requires the following parts:

- T-connector
- 10BASE2 coax
- NC3100 Power Supply and outlet

Figure 8-16 shows how the parts connect.



1. RC4030E Gateway, bottom panel
2. T-connector
3. 10BASE2 coax
4. Power cable, dc (6 feet)
5. Power cable (6 1/2 feet)
6. Outlet
7. NC3100 Power Supply

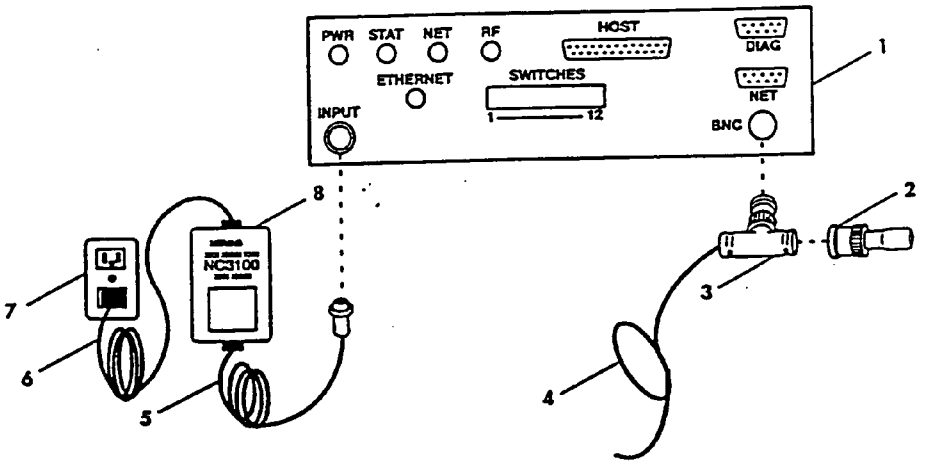
Figure 8-16
RC4030E Gateway in Middle of 10BASE2

End of 10BASE2

A gateway that connects to the end of the 10BASE2 segment requires the following parts:

- T-connector and cable terminator
- 10BASE2 coax
- NC3100 Power Supply and outlet

Figure 8-17 shows how the parts connect.



1. RC4030E Gateway, bottom panel
2. Cable terminator
3. T-connector
4. 10BASE2 coax
5. Power cable, dc (6 feet)
6. Power cable (6 1/2 feet)
7. Outlet
8. NC3100 Power Supply

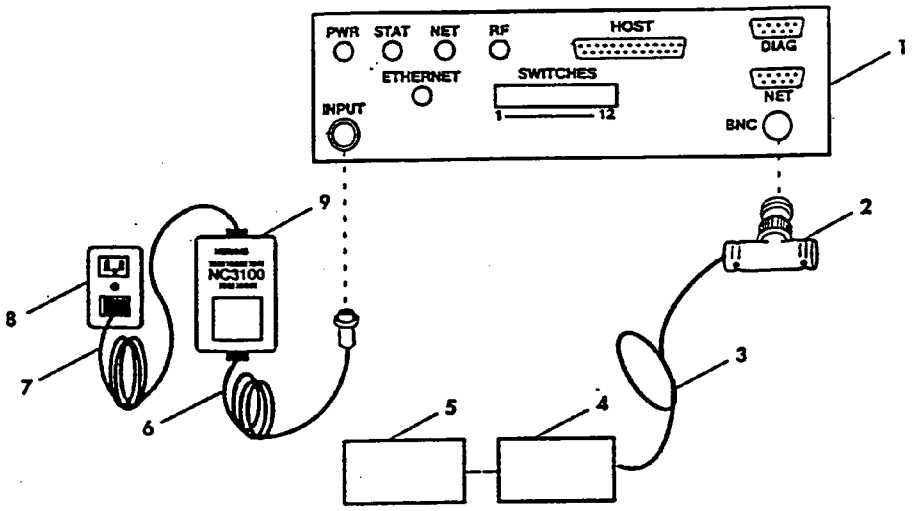
Figure 8-17
RC4030E Gateway at End of 10BASE2

10BASE-T Connection

A gateway that connects to 10BASE-T requires the following parts:

- Media converter
- 10BASE2 coax
- NC3100 Power Supply and outlet

Figure 8-18 shows how the parts connect.

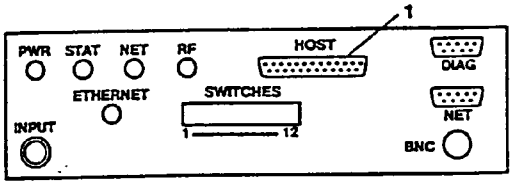


1. RC4030E Gateway, bottom panel
2. T-connector
3. 10BASE2 coax
4. Media converter
5. 10BASE-T hub
6. Power cable, dc (6 feet)
7. Power cable (6 feet)
8. NC3100 Power Supply
9. Outlet

Figure 8-18
RC4030E Gateway Connected to 10BASE-T

Host Connectivity Solutions

Host connection options are V.35 direct and RS-232 direct. The RC4030E Gateway connects, through its 25-pin HOST port, to the 9-pin or 25-pin RS-232 or V.35 port on the host. Figure 8-19 shows where the HOST port is located on the gateway.



1. HOST port

Figure 8-19
RC4030E Gateway HOST Port

The gateway can also connect to the host through modems. Norand provides host and modem cables.

Wireless Network Access Server

Because you would install the Wireless Network Access Server (WNAS) software onto the host, no hardware is needed.

6950 Enterprise Gateway Server

The following pages describe installation strategies for the 6950 Enterprise Gateway Server.

Location

You would typically locate the gateway server next to the host computer in a computer room. The location must meet the requirements in Appendix E.

Norand designed the gateway server for operation in an enclosed (weather-proof) environment. Norand does not recommend that the gateway server be installed in an area that may expose it to harsh operating conditions such as rain, snow, and excessive heat or humidity.

Mounting Options

You can mount the gateway server horizontally on a tabletop. A mounting bracket is not needed for a tabletop installation. When the gateway server is on the tabletop, four self-adhesive rubber feet on its bottom panel keep it from slipping out of place. An optional wall mounting bracket mounts the gateway server on a wall.

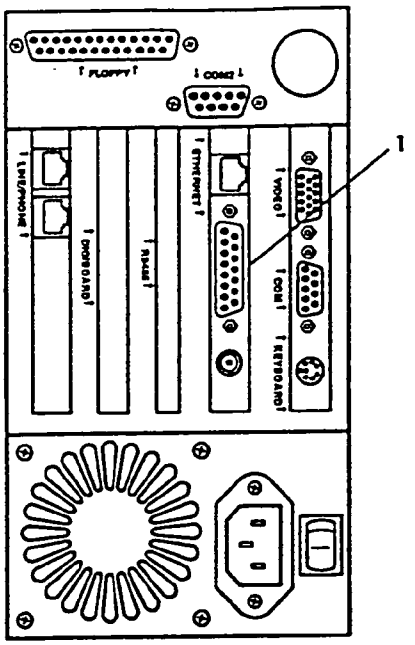
Power Requirements

An internal 65 watt power supply powers the gateway server. The power supply requires an ac power outlet. The power supply autosenses the level (110, 220, or 240 V ac) and frequency of the source voltage and operates accordingly.

You must locate the gateway server within 7 feet of the power outlet. This distance ensures the power cord reaches the gateway server and the outlet.

Ethernet Connectivity Solutions

The gateway server connects to the wireless infrastructure through Ethernet, and supports 10BASE2, 10BASE5, and 10BASE-T media options. Connections are through the network ports on its installed Ethernet adapter card (Figure 8-20).



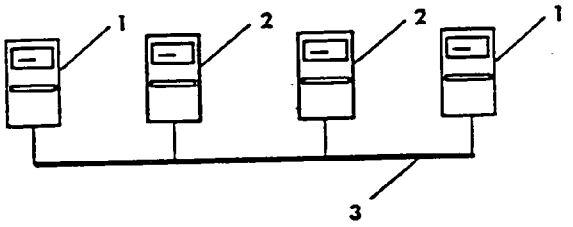
1. Network ports on Ethernet adapter card

Figure 8-20
6950 Enterprise Gateway Server Network Ports

The following pages show how the gateway server connects to 10BASE2 and 10BASE-T.

10BASE2 Connections

The gateway server can connect to the end or middle of a 10BASE2 segment. Figure 8-21 shows connection options.



1. Gateway server at end of segment
2. Gateway server in middle of segment
3. 10BASE2 coax

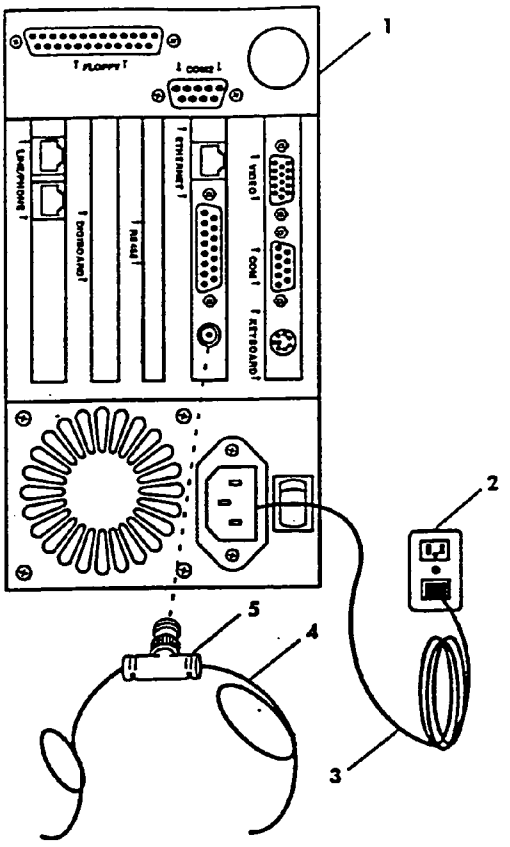
Figure 8-21
6950 Enterprise Gateway Server Connection Options

Middle of 10BASE2

A gateway server that connects to the middle of the 10BASE2 segment requires the following parts:

- T-connector
- 10BASE2 coax
- Power cord and outlet

Figure 8-22 shows how the parts connect.



1. Gateway server, rear panel
2. Outlet
3. Power cord (7 feet)
4. 10BASE2 coax
5. T-connector

Figure 8-22
6950 Enterprise Gateway Server in Middle of 10BASE2

End of 10BASE2

A gateway server that connects to the end of the 10BASE2 segment requires the following parts:

- T-connector and cable terminator
- 10BASE2 coax
- Power cord and outlet

Figure 8-23 shows how the parts connect.

10BASE-T Connection

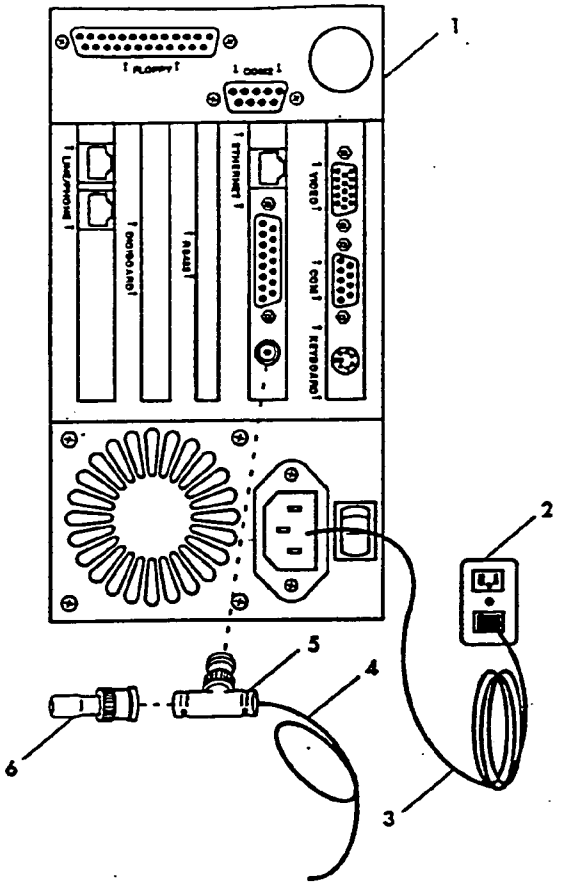
A 6950 Enterprise Gateway Server that connects to 10BASE-T requires the following parts:

- Cable with RJ45 plugs
- RJ45 jack
- Power cord and outlet

Figure 8-24 shows how the parts connect.

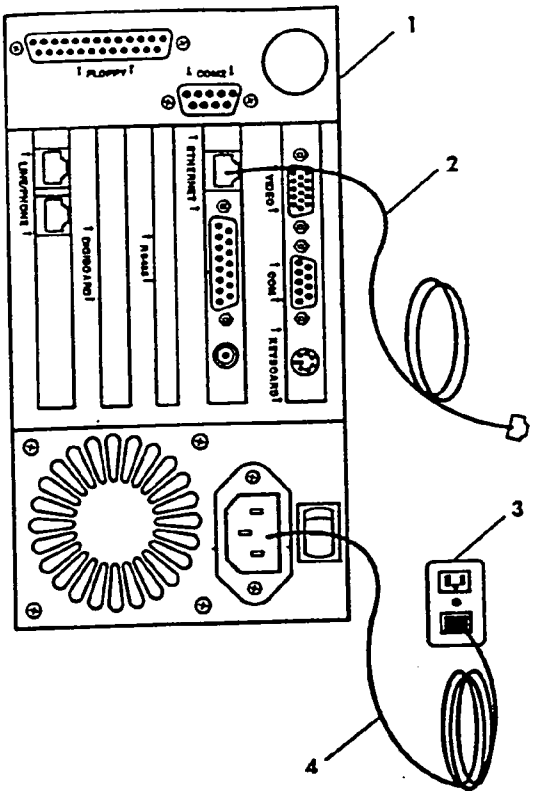
Host Connectivity Solution

Because the gateway server connects to the host through the Ethernet medium, no host cables are needed.



1. Gateway server, rear panel
2. Outlet
3. Power cord (7 feet)
4. 10BASE2 coax
5. T-connector
6. Cable terminator

Figure 8-23
6950 Enterprise Gateway Server at End of 10BASE2



1. Gateway Server, rear panel
2. Cable with RJ45 plugs
3. Outlet
4. Power cord (7 feet)

Figure 8-24
6950 Enterprise Gateway Server Connected to 10BASE-T

Disconnecting Fixed-end Devices

You can disconnect the 6710 Access Point, RC4030E Gateway, and 6950 Enterprise Gateway Server from a network segment without disrupting service. Disconnecting the fixed-end device from 10BASE2 involves disconnecting the T-connector from the device but leaving the 10BASE2 cables connected to the T-connector. You must attach a cable terminator to a T-connector at the end of a cable.

Disconnecting a fixed-end device from 10BASE-T involves pulling the cable with the RJ45 plugs from the fixed-end device and RJ45 jack. Pulling this cable disables only the fixed-end device at the end of the cable.

Installation Examples

The rest of this section contains examples of three different types of installations:

- Warehouse site installation with 10BASE2 coax
- Retail site installation with 10BASE2 coax

The examples include a diagram of the installation and a breakdown of the parts and cabling that would be required.

Warehouse Site

Figure 8-25 shows an example of an installation for a warehouse site. In the figure "AP" is a 6710 Access Point with a 900 MHz or UHF NIC, "GW" is an RC4030E Gateway, and "R" is a repeater. The Ethernet medium is 10BASE2 coax.

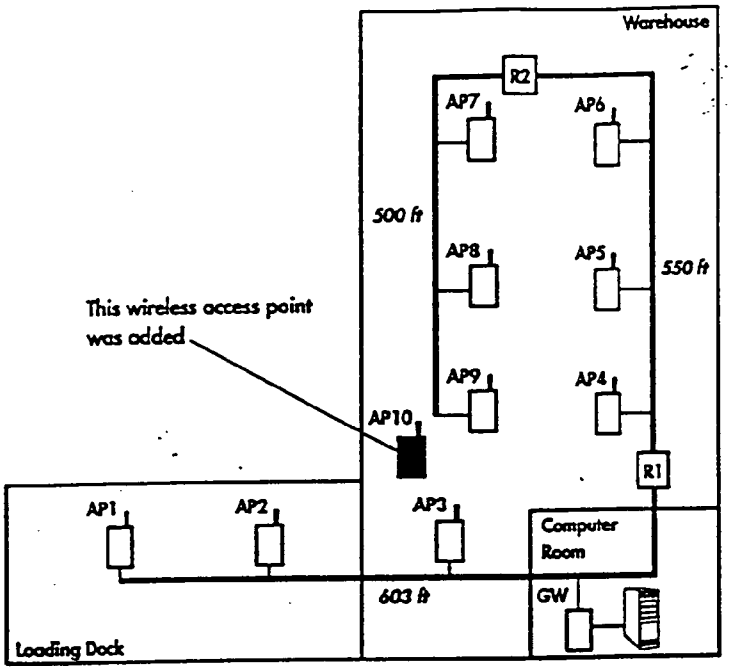


Figure 8-25
Warehouse Site Installation With 10BASE2

A survey for this site revealed two issues. One was the marginal coverage between AP2 and AP3. The other was the possibility of a broken cable disrupting service. The cost of a fully redundant system at this site was not feasible. (Redundancy is providing duplicate devices to immediately take over the function of equipment that fails.)

To resolve the marginal coverage issue, a wireless access point (AP10) was added. The wireless access point covers the area between AP2 and AP3. It also resolves the broken cable issue for the following reasons:

- It provides alternate, wireless routes between AP3 and AP2, between AP9 and AP2, and between AP3 and AP9.
- It covers a cable break between GW and AP3, or between AP3 and AP2.
- It provides communication to GW if a break occurs between GW and R1, or between R1 and AP4.

The wireless access point covers any cable break. For example:

- If the cable from AP6 to AP7 broke, AP6, AP5, and AP4 would communicate with GW through R1.
- AP7, AP8, and AP9 would take the wireless route (AP10 to AP3 and then to GW).

Determining the Parts Required

The following chart lists the parts needed to connect each NORAND fixed-end device to the 10BASE2 cable.

Code	Description	Segment Location	Parts Required
AP1	Access point	End	T-connector, cable terminator
AP2	Access point	Middle	T-connector
AP3	Access point	Middle	T-connector
AP4	Access point	Middle	T-connector
AP5	Access point	Middle	T-connector
AP6	Access point	Middle	T-connector
AP7	Access point	Middle	T-connector
AP8	Access point	Middle	T-connector
AP9	Access point	End	T-connector, cable terminator
AP10	Access point	(Wireless)	(None)
GW	Gateway	Middle	T-connector

Determining Cable Amounts

The warehouse site installation needs about 1,653 feet of coaxial cable to join all network devices. However, segment lengths for 10BASE2 can be no longer than 607 feet. The length was divided as follows:

- 603 feet between AP1 and R1
- 550 feet between R1 and R2
- 500 feet between R2 and AP9

These three segments are joined by two repeaters.

Bill of Materials

A bill of materials (BOM) for the cable and NORAND fixed-end devices in this installation would resemble the following chart.

BOM Item #	Description	Quantity
1	RG58 A/U 10BASE2 coaxial cable	1,653 ft
2	6710 Access Point	10
3	RC4030E Gateway	1
4	Extra mounting bracket (if needed for RC4030E Gateway's power supply)	1
5	Remote antenna for AP10	1
6	NEMA enclosures for AP1 and AP2	2

Retail Site

Figure 8-26 shows an example of an installation for a small retail store. In the figure "AP" is a 6710 Access Point with a Proxim 2.4 GHz NIC and "GW" is an RC4030E Gateway. The Ethernet medium is 10BASE2 coax.

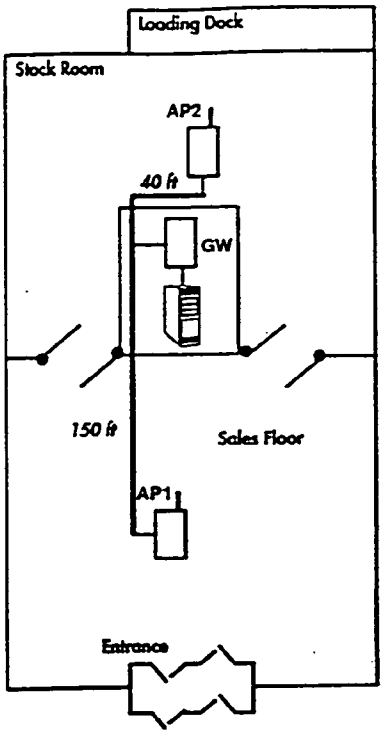


Figure 8-26
Retail Site Installation With 10BASE2

A single 6710 Access Point could cover the sales floor and the stock room. However, the site survey revealed that the concrete wall between the sales floor and stock room prevented effective coverage of both areas from one access point. Because complete coverage is required for the loading dock and front part of the sales floor, access point AP2 was added.

Determining the Parts Required

The following chart lists the parts needed to connect each NORAND fixed-end device to the 10BASE2 cable.

Code	Description	Segment Location	Parts Required
AP1	Access point	End	T-connector, cable terminator
AP2	Access point	End	T-connector, cable terminator
GW	Gateway	Middle	T-connector

Determining Cable Amounts

The retail installation needs about 190 feet of coaxial cable to join all network devices. This is under the maximum segment length for 10BASE2 (607 feet), so no repeater is needed.

Bill of Materials

A BOM for the cable and NORAND fixed-end devices in this installation would resemble the following chart.

BOM Item #	Description	Quantity
1	RG58 10BASE2 A/U coaxial cable	190 ft
2	6710 Access Point	2
3	RC4030E Gateway	1
4	Extra mounting bracket (if needed for RC4030E Gateway's power supply)	1

Section 9

System Management

About This Section

This section describes configuration and management for NORAND® wireless infrastructure components.

Access Point Setup and Configuration

System software parameters for the 6710 Access Point reside in the access point's FLASH ROM. You would use the parameters to set IP addresses, network spanning tree options (such as LAN ID, netname or security ID, and root priority), and other operational features.

You initially configure the 6710 Access Point locally through the access point's DIAG port. After you configure the access point locally, you can access its configuration menus locally or through a remote TELNET session over the network backbone. Complete information about establishing a TELNET session is in the *6710 Access Point User's Guide (NPN: 961-047-081)*.

You do not need special equipment to configure the 6710 Access Point through a TELNET session. However, the access point must be connected to the Ethernet medium (or attached through a wireless link) and have its IP address set as a minimum. (This address must be set locally through the DIAG port.) To manage the access point from a remote location, you would establish a TELNET session with the access point's IP address.

Software Download

Software can be downloaded to an access point through its configuration menus or through SNMP.

Configuration Menus

You can download a new version of system software to a 6710 Access Point through its configuration menus. You can access the menus locally through the access point's DIAG port, or remotely over the network through a PC running server software and TELNET. You can then transfer files from the PC server to the access point (the client device). This feature enables you to download, from a central location, a new version of system software to each access point on a network.

The PC server physically connects to the LAN or to an Ethernet modem, which provides connection to the LAN. The server software Norand recommends for DOS and Windows is TFTP by FTP Software, Inc. All file transfers between the PC TFTP server and the TFTP client (the access point) are through User Datagram Protocol (UDP) packets and TFTP's binary transfer mode.

You establish a TELNET session with the access point's IP address while the TFTP server software is running on the PC. This connection lets you access the TFTP server (through the GET command) and the access point's command line interface for the configuration menus. The series of steps required to download the software to the access point repeats for each access point.

For diagnostic purposes, the access point's command line interface supports commands that show the current status of the file system and allow the file system to be reset if necessary.

The access point operates normally throughout the software download process. While you are downloading a new version of software to the access point, it continues to use the previous version of software until you reboot it through the *reboot* command. After the access point reboots, it uses the new version of software.

SNMP

Software download can also be done through SNMP. This allows a set of access points to automatically perform a set of commands. It also allows the software to be downloaded by multiple access points at a particular time. This method requires a "download script file," which each access point uses as a source of the commands to execute. The "download script file and the software must be on the TFTP server, which could be a PC or an access point.

Setup and Configuration of Host Connectivity Devices

The following pages describe system management for the RC4030E Gateway, 6910 Integrated Gateway/Access Point, WNAS, and 6950 Enterprise Gateway Server.

RC4030E Gateway

System software parameters for the RC4030E Gateway reside in the gateway's FLASH ROM. The parameters set IP addresses, the host type (such as 3270), and communication options (such as group chaining and compression).

You initially configure the RC4030E Gateway locally through its DIAG port. You can then access its configuration options locally or through a remote TELNET session. You do not need special equipment to configure the gateway through a TELNET session. However, the gateway must be connected to the Ethernet medium and have its IP address set as a minimum. To manage the gateway from a remote location, you would establish a TELNET session with the gateway's IP address. Complete information about establishing a TELNET session is in the *RC4030E Gateway User's Guide (NPN: 961-047-087)*.

6910 Integrated Gateway/Access Point

System software parameters for the 6910 Integrated Gateway/Access Point reside in the gateway/access point's FLASH ROM. The parameters set IP addresses, the NORAND Native host type, and communication options (such as group chaining and compression). The parameters also set access point bridging and radio options.

System management for the gateway/access point is the same as for the RC4030E Gateway and 6710 Access Point. Complete information about the gateway/access point is in the *6910 Integrated Gateway/Access Point User's Guide (NPN: 961-047-095)*.

WNAS

After you install WNAS onto the host, you can configure it to meet your site's requirements. Use WNAS configuration parameters to do the following:

- Configure runtime options (such as RS-232 parameters)
- Define the applications to be run from the terminal emulation station
- Determine the menu that each terminal emulation station should display when it powers up
- Set wireless station-specific configurations (such as which menu to use for a specific wireless station number)

You initially configure WNAS locally on the host. You can then access its configuration menus locally or through a remote TELNET session. You do not need special equipment to configure WNAS through a TELNET session. However, WNAS must have an IP address set as a minimum. To manage WNAS from a remote location, you would establish a TELNET session with the IP address. Complete information about establishing a TELNET session is in the *Wireless Network Access Server User's Guide (NPN: 961-051-006)*.

6950 Enterprise Gateway Server

You can configure network options for the 6950 Enterprise Gateway Server through one of the following methods:

- Terminal emulation station within range of the 6710 Access Point
- TELNET session
- Keyboard and monitor attached to the gateway server
- Dumb terminal plugged into the COM2 port

You initially configure the gateway server locally. You can then access its configuration menus locally or through a remote TELNET session. You do not need special equipment to configure the gateway server through a TELNET session. However, the gateway server must be connected to the Ethernet medium and have its IP address set as a minimum.

To manage the gateway server from a remote location, you would establish a TELNET session with the gateway server's IP address. Complete information about establishing a TELNET session is in the *6950 Enterprise Gateway Server User's Guide (NPN: 961-047-091)*.

SNMP

6710 Access Points and RC4030E Gateways are manageable through SNMP. SNMP is an industry-standard protocol that provides a way for network management platforms to query other network devices for status and other device information. The information is typically system identification data, or counters indicating error rates or performance measures in the network device being queried.

SNMP uses the UDP transport-level protocol to provide communications between a network management station and the agent that resides in the managed object. RFC1157 defines the SNMP.

Elements of the SNMP network management system are:

- Network management platform
- SNMP agent
- MIB
- Private NORAND MIBs
- TCP/IP stack

Network Management Platform

A network management platform is a collection of software modules that use SNMP (and the list of objects which can be obtained) to automatically retrieve, display, save, or analyze data. The platform is installed on a network management station, which must meet the requirements outlined in the management platform's user manual.

Norand recommends the OpenView for Windows by Hewlett-Packard (HP) platform to provide network management capability for the open wireless LAN. HP OpenView is a standards-based network management platform. Complete information about HP OpenView for Windows is in the *NORAND Open Wireless LAN with HP OpenView for Windows User's Guide (NPN: 961-051-009)*.

NORAND OWLView for Windows is a separate management application for HP OpenView. OWLView helps manage the open wireless LAN by showing wired and wireless components and connections of NORAND LAN devices. OWLView also shows wireless stations communicating through the open wireless LAN, and periodically updates maps with the status of NORAND devices and connections. Complete information is in the *OWLView for HP OpenView for Windows User's Guide (NPN: 961-051-010)*.

SNMP Agent

An SNMP agent resides at the managed device (such as the 6710 Access Point). The agent accepts SNMP requests from an SNMP network management platform and responds with the requested data. The SNMP agent also services SNMP SET requests and sends unsolicited messages (called traps) when a predefined event occurs.

MIB

A MIB defines the management information that the device supports. MIB-II is the standard MIB defined for SNMP over TCP/IP.

Resident agents for 6710 Access Points and RC4030E Gateways support MIB-II for TCP/IP–based internets. MIB-II is a set of objects an SNMP network management platform can query or set in the SNMP agent of a network device, such as a router or 6710 Access Point. MIB-II provides information about the device, and TCP/IP and SNMP activity for the device.

MIBs are written in Abstract Syntax Notation.1 (ASN.1). ASN.1 is defined in ISO documents 8824.2 and 8825.2. ASN.1 is a machine-independent data definition language that provides an interface to the underlying management information in a system. RFC1155 defines ASN.1 formats, and RFC1213 defines MIB formats.

Because a MIB is written in ASN.1, it can be directly imported into any SNMP network management platform. The MIB defines and describes the management data the fixed-end devices support.

Complete information about MIBs for the 6710 Access Point and RC4030E Gateway is in the *NORAND Management Information Base Reference Manual (NPN: 977-051-002)*.

Private NORAND MIBs

Each 6710 Access Point and RC4030E Gateway maintains a set of private NORAND MIBs. These MIBs apply to both of these devices, or only to a specific device. The MIBs are installed with the network management platform.

A tree structure stores management data (objects). Object identifiers (OIDs) are assigned based on the position of the object in the tree. Standard objects (such as MIB-II) register with the Internet Assigned Numbers Authority (IANA). The IANA is the central registry for various internet protocol parameters such as port, protocol, and enterprise numbers, and options, codes, and types.

Enterprises can register with the IANA for an Enterprise ID, which gives the enterprise its own subtree in the the standard object tree. The Enterprise ID for Norand is 469.

TCP/IP Stack

A TCP/IP stack must be installed on the network management station if it does not already have one. Norand recommends the PC/TCP TCP/IP stack by FTP Software. The stack is available through Norand.

Appendix A
Radio Options

About This Appendix

This appendix describes wireless technology options for the wireless infrastructure. NIC specifications, international frequencies, and data rates are also in this appendix.

Wireless Technology Options

Wireless technology options for the wireless infrastructure are 900 MHz multimode RF, synthesized UHF, and 2.4 GHz frequency hopping (Proxim RangeLAN2). Norand is also developing product capabilities compatible with the emerging IEEE 802.11 standard for wireless LANs, which is currently slated for approval by late 1996. Contact a Norand representative for the current status.

900 MHz Radio Option

The following chart lists NICs and their NORAND® model names for the 900 MHz radio option.

NIC*	Device	Model Name
Type III	6710 Access Point	RM160
	PEN*KEY® 6400	RM160
	PEN*KEY 6600	RM160
	PEN*KEY 6100	RM160
Radio modules	RT1100	RM60, RM70, RM70LR (radio modules)
	RT1700	RM60, RM70, RM70LR (radio modules)

**Consult a Norand representative for availability.*

NIC Specifications

Following are networking specifications for the 900 MHz NIC.

Frequency band:	902–928 MHz spread spectrum direct sequence
Range:	Up to 1300 feet line of sight
Coverage:	100,000–350,000 square feet in typical indoor installations
Data rate:	90, 225, or 450 Kbps (depends on installation)
Channelization:	7 @ 90 Kbps, 1 @ 225, 450 Kbps
Client driver:	ODI
Software compatibility	Requires NORAND communications software resident in the access point
Output power:	250 mW
MAC protocol:	NORAND open wireless LAN MAC radio protocol
Regulatory compliance:	FCC 15.247; Industry Canada RSS 210 (Consult a Norand representative for availability.)

Frequencies and Data Rates

NORAND wireless devices with the 900 MHz option can operate in Australia and in most countries in North and South America. Table A-1 lists various countries and their 900 MHz frequencies. Contact a Norand representative for current information about countries in which the product is approved for use and countries in which submission for type approval is planned.

Table A-1
900 MHz Frequencies – International

Country	Frequencies (MHz)
Australia	919.5, 921.5, 923.5
Canada	907.5, 910.0, 912.5, 915.0, 917.5, 920.0, 922.5

Table A-1 (Continued)
900 MHz Frequencies – International

Country	Frequencies (MHz)
Mexico	907.5, 910.0, 912.5, 915.0, 917.5, 920.0, 922.5
United States	907.5, 910.0, 912.5, 915.0, 917.5, 920.0, 922.5

Table A-2 lists corresponding data rates for Canada, Mexico, and the United States.

Table A-2
Corresponding Data Rates – Canada, Mexico, and United States

Frequency (MHz)	Data Rate (Kbps)
907.5	90
910.0	90
912.5	90
915.0	90
917.5	90
920.0	90
922.5	90
902–928	225
902–928	450

Table A-3 lists corresponding data rates for Australia.

Table A-3
Corresponding Data Rates – Australia

Frequency (MHz)	Data Rate (Kbps)	Channel
919.5	90	34
921.5	90	38
923.5	90	42

Synthesized UHF Radio Option

The following chart lists NICs and their NORAND model names for the synthesized UHF radio option.

NIC*	Device	Model Name
Type II (tethered)	6710 Access Point	RM111
	PEN*KEY 6400	RM111
	PEN*KEY 6600	RM111
Mini-ISA	PEN*KEY 6100	RM211
	RT1100	RM11, RM31 (radio modules)
	RT1700	RM11, RM31 (radio modules)

**Consult a Norand representative for availability.*

NIC Specifications

Following are networking specifications for the synthesized UHF NIC.

Frequency band:	450–470 MHz, four-level FSK (frequency shift keying)
Range:	Up to 3500 feet line of sight
Coverage:	800,000 square feet in typical indoor installations
Data rate:	19.2 Kbps (14.4 Kbps with forward error correction)
Channelization:	20 KHz or 25 KHz
Client driver:	ODI
Software compatibility	Requires NORAND communications software resident in the access point
Output power:	500 mW
MAC protocol:	NORAND open wireless LAN MAC radio protocol
Regulatory compliance:	FCC Parts 15, 90; Industry Canada RSS 119; ETS 300-220; FTZ 2014 (Germany); CE Mark (Europe) (Consult a Norand representative for availability)

Frequencies

NORAND wireless devices with the synthesized UHF option can operate in Europe, the Pacific Rim (except Japan), Australia, and most countries in North and South America. Contact a Norand representative for current information about countries in which the product is approved for use and countries in which submission for type approval is planned.

Proxim 2.4 GHz Radio Option

The following charts list NICs and their NORAND model names for the Proxim 2.4 GHz radio option.

Access Point NIC*	Device	Model Name
Type III	6710 Access Point	RM180

**Requires NORAND communications software resident in the access point.*

Wireless Station NIC*	Device	Model Name
Type III	PEN*KEY 6400	RM180
	PEN*KEY 6600	RM180
Type II	Laptops and notebooks ..	RM185
Mini-ISA	PEN*KEY 6100	RM280
	RT1100	RM80, RM90, RM90LR (radio modules)
	RT1700	RM80, RM90, RM90LR (radio modules)
ISA	Desktops	RM380

**Consult a Norand representative for availability.*

NIC Specifications

Following are networking specifications for the Proxim 2.4 GHz NIC.

Frequency band:	2.401–2.480 GHz spread spectrum frequency hopping
Range:	Up to 500 feet line of sight
Coverage:	25,000 square feet (2,322 square meters) in typical indoor installations
Data rate:	800 Kbps or 1.6 M bps, manual or autoselecting
Client drivers:	ODI and NDIS (version 2.0.1 for DOS and Windows)
6710 Access Point:	Requires NORAND communications software resident in the access point
Ethernet compatibility:	Ethernet packet types and Ethernet addressing
Output power:	100 mW
MAC protocol:	RangeLAN2
Regulatory compliance:	FCC 15.247; Industry Canada RSS 210; European Union ETS 300-328 (Consult a Norand representative for availability.)

Frequencies and Data Rates

NORAND wireless devices with the 2.4 GHz option can operate in most areas that allow use of spread spectrum wireless communications at 2.4 GHz, including Australia and countries in North and South America, Europe, and Asia. Table A-4 lists various countries and their corresponding 2.4 GHz frequencies and data rates. Contact a Norand representative for current information about countries in which the product is approved for use and countries in which submission for type approval is planned.

Table A-4
Proxim 2.4 GHz Frequencies and Data Rates – International

Country	Frequency Band (GHz)	Data Rate (Kbps)
Australia	2.401–2.443	800 or 1600
Canada	2.402–2.480	800 or 1600
Denmark	2.407–2.449	800 or 1600
France	2.447–2.480	800 or 1600
Germany	2.407–2.449	800 or 1600
Hong Kong	2.402–2.480	800 or 1600
Italy	2.407–2.449	800 or 1600
Japan	2.473–2.495	800 or 1600
Mexico	2.452–2.472	800 or 1600
Spain	2.407–2.449	800 or 1600
United Kingdom	2.407–2.449	800 or 1600
United States	2.402–2.480	800 or 1600

Radio Kits

Standard radio kits are available from Norand as factory-installed options. Different radio kits allow the radio network to be customized to provide various data throughput and performance tradeoffs. Consult a Norand representative for the radio kit options currently available.

In accordance with regulations, radio kits are usually shipped with standard antenna kits that use a unique RF (radio frequency) connector. Replacement antennas are available from Norand.

Remote antenna kits are also available from Norand. In accordance with regulations, remote antenna kits must be installed by Norand or other qualified personnel. Only antennas furnished by Norand can be used with NORAND access points.

Performance Tradeoffs

▶ NOTE: *Norand or certified providers can conduct a site survey to help you choose the best radio option for your site. Section 8, "Installation," discusses site surveys.*

When determining the type of radio or radios to use, you must consider the size and physical layout of the site and the amount of traffic that will flow through the network. Note that radio range decreases as radio frequency and data speed increase.

The 900 MHz option is a good compromise between coverage and data rate. It is a good choice for large populations of stations, or for users who want high performance. For PEN*KEY computers, notebooks, and laptops, it is recommended for light to medium density data applications in factories and other large spaces. The 900 MHz radio does not require a site license.

The UHF option has the best coverage but the lowest data rate. It is a good choice for low to medium populations of terminal emulation stations. The UHF radio requires a site license.

The Proxim 2.4 GHz option has the highest data rate but the lowest coverage. It is a good choice for when file transfers are regular or where users have written their own applications, which puts a significant amount of traffic on the air. For PEN*KEY computers, notebooks, and laptops, it is recommended for information-intensive applications requiring high throughput.

Radio and Scanner Modules

Radio and scanner modules for terminal emulation stations in the RT1100 and RT1700 Series are interchangeable. For example, you can change a 900 MHz radio on an RT1700 to a 2.4 GHz radio by changing radio modules; you do not need to replace the entire terminal emulation station. The following chart describes radio modules and scanning capabilities.

Module	Description
RM11	UHF radio
RM31	UHF radio with integrated, standard-range scanner
RM60	900 MHz radio
RM70	900 MHz radio with integrated, standard-range scanner
RM70LR	900 MHz radio with integrated, standard-range scanner (such as used on a forklift)
RM80	2.4 GHz radio
RM90	2.4 GHz radio with integrated, standard-range scanner
RM90LR	2.4 GHz radio with integrated, long-range scanner (such as used on a forklift)

The type of module attached to the terminal emulation station determines the model. For example, an RT1700 with a 2.4 GHz radio module is a model RT1780.

The user activates the scanner module through one of these methods:

- Scanning buttons integrated into the radio module
- Scan button on the RT1700 and PEN*KEY 6400 Computer
- Optional scanning handle for the RT1100 and RT1700

Scanner modules support these bar code symbologies:

ABC Codabar
Codabar
Code 39
Code 93
Code 128
EAN
EAN with add-ons
Encoded Code 39
Extended Code 39
Interleaved 2 of 5
Plessey
Straight2 of 5
UPC
UPC with add-ons

Appendix B

Recommended Network Products

About This Appendix

If you are purchasing a complete networking solution from Norand, refer to this appendix for the products Norand recommends for use with the open wireless LAN. If you are already using products by other vendors, the open wireless LAN should operate correctly with those products.

Modems, Hubs, Bridges, and Transceivers

Products listed in the following chart are available through Norand. For more information about a product refer to its user guide or contact Norand.

Product (Vendor)	Product Name
Ethernet modem (Shiva)	NetModem/E
Hubs (3COM)	LinkBuilder FMS II 12-port twisted pair LinkBuilder FMS 10-port BNC FMS LinkBuilder II 6-port fiber
Hub options (3COM)	Fiber Optic Interface Module Redundant Power System LinkBuilder FMS II Management Module

(Continued on next page)

Product (Vendor)	Product Name
Bridge (3COM)	LinkBuilder Bridge MicroModule
Interbuilding bridge (Proxim)	RangeLINK 2021 with yagi antenna
Transceivers (Transition Networks)	10BASE-T to 10BASE2 10BASE2 to 10BASEF 10BASEF to 10BASE-T

Other Products and Configurations

Following are products and configurations that require special approval through Norand for use with the open wireless LAN. Contact the Advanced Technology Group at Norand for more information.

FDDI
100VGAnyLAN
Fiber Optic Inter Repeater Link (FOIRL)
Routers
Asynchronous Transfer Mode (ATM)
10BASE-T switch
10BASE-F switch

Appendix C

ODI and NDIS Driver Configurations

About This Appendix

This appendix contains examples of ODI and NDIS driver configurations for a PEN*KEY® 6100 Computer with the Proxim 2.4 GHz radio option. The configurations shown are for a Novell NetWare 4.x client and a TCP/IP host using PC/TCP networking software by FTP Software, Inc. NetWare uses only the ODI driver. PC/TCP can use the ODI driver or the NDIS driver.

ODI Driver for NetWare and TCP/IP

The ODI driver for the PEN*KEY 6100 Computer is RL2OEM.COM. The driver uses the network parameters in the NET.CFG file. To load the driver you would load — into AUTOEXEC.BAT — LSL.COM (the link support layer), RL2OEM.COM, and then the protocol stack or driver (IPXODI.EXE for NetWare, and ETHDRV.EXE or VXDINIT.EXE for TCP/IP).

NET.CFG

Following is an example of a NET.CFG file.

```
LINK DRIVER RL2OEM
        INT                     15
# This is the IRQ and I/O base address used by the 2.4 GHz radio.
        PORT                    3F0
        BUS_MODE                1
# STATION_TYPE 0 means station only, 1 is alternate master, 2 is master
# always.
        STATION_TYPE            0
        DOMAIN                  3
        SUBCHANNEL              1
        CHANNEL                 1
# INACTIVITY_MIN/SEC set to 0 will keep the station awake always.
# INACTIVITY_MIN/SEC set to anything > 0 the radio will snooze and will not
# reliably respond to broadcasts.
        INACTIVITY_MIN          0
        INACTIVITY_SEC          0
#       INACTIVITY_SEC          30
# If you are using an access point, set PEER_TO_PEER to "N" for better
# performance.
        PEER_TO_PEER            N
#       FRAME                   ethernet_802.3
        FRAME                   ETHERNET_II
NetWare DOS Requester
        FIRST NETWORK DRIVE = G
        PREFERRED SERVER = ENTERPRISE
        NAME CONTEXT = "o = first_floor"
```

AUTOEXEC.BAT for NetWare

The following lines show some ODI driver files loaded from AUTOEXEC.BAT for a PEN*KEY 6100 Computer in a NetWare environment:

```
lsl.com
rl2oem.com
ipxodi
vlm
```

AUTOEXEC.BAT for TCP/IP

The following lines show some ODI driver files loaded from AUTOEXEC.BAT for a PEN*KEY 6100 Computer in a TCP/IP environment:

```
lsl.com
rl2oem.com
odipkt.com (this is the ODI to packet driver shim provided by FTP Software)
ethdrv (or vxdinit for Windows enhanced mode only)
set pctcp = c:\ftp\pctcp.ini
```

Additional lines would include the path statement for the FTP directory and the SET command for PC/TCP.

NDIS Driver for TCP/IP

The NDIS driver for the PEN*KEY 6100 Computer is RL2OEM.DOS. The driver uses the parameters in the PROTOCOL.INI file, plus PROTMAN.EXE, PROTMAN.DOS, and NETBIND.COM.

PROTOCOL.INI

Following is an example of a PROTOCOL.INI file.

```
[protman]
DriverName = PROTMAN$

[RL2OEM]
drivername = RL2OEM$
;
; The 2.4 GHz radio in the PEN*KEY 6100 Computer uses IRQ 15.
;
INT = 15
;
; The 2.4 GHz radio in the PEN*KEY 6100 Computer uses 0x3f0.
;
PORT = 0x3f0
;
; Valid channel values are 1–15
;
; CHANNEL = 7
;
; Valid sub-channel values are 1–15
;
```

```
;SUBCHANNEL = 7
;
; Valid domain values are 0–15
;
DOMAIN = 7
;
; Valid station_type values are 0, 1, & 2
;
STATION_TYPE = 0
;
; Valid roam_config values are 0, 1, & 2
;
ROAM_CONFIG = 1
;
Valid mac_optimize values are 0 & 1
;
MAC_OPTIMIZE = 1
;
; Valid peer_to_peer values are Y & N
;
PEER_TO_PEER = N
INACTIVITY_MIN = 0
INACTIVITY_SEC = 0

[PKTDRV]
DriverName = PKTDRV$
intvec = 0x60
chainvec = 0 x 62
BINDINGS = RL2OEM
```

CONFIG.SYS

The following lines show some NDIS driver files loaded from CONFIG.SYS for a PEN*KEY 6100 Computer in a TCP/IP environment:

`device = a:\proxim\protman.dos /I:a:\proxim` *(/I specifies the path to `protocol.ini`)*
`device = a:\proxim\rl2oem.dos`
`device = a:\proxim\dis_pkt.gup` *(this is the NDIS to packet driver shim provided by FTP Software)*

Open Wireless LAN Parameters

The following ODI and NDIS parameters affect the PEN*KEY 6100 Computer's performance on the open wireless LAN.

CHANNEL and SUBCHANNEL

CHANNEL sets the hopping sequence of the radio. SUBCHANNEL differentiates between subnetworks These two parameters apply if the PEN*KEY 6100 Computer is an Alternate Master in a peer-to-peer (ad hoc) network. See STATION_TYPE on the next page for a description of Alternate Master.

DOMAIN

The value for DOMAIN must match the LAN ID specified for the 6710 Access Point.

FRAME

FRAME applies to the ODI driver only. Its value must match the frame type of the Ethernet host. Common values are "ETHERNET_II" (DIX Ethernet) and "ETHERNET_802.3."

INACTIVITY_MIN and INACTIVITY_SEC

These parameters set the snooze mode timeout for the Proxim 2.4 GHz radio option. When the radio is in snooze mode, it does not reliably hear broadcast or multicast messages. If the application depends on the PEN*KEY 6100 Computer hearing broadcast or multicast messages when the radio might be snoozing, these parameters must be disabled (set to "0"). Note that these parameters are not related to the computer's power management timeout parameters.

MAC_OPTIMIZE

The value for MAC_OPTIMIZE should be "0" when 20 or fewer PEN*KEY 6100 Computers are communicating with a 6710 Access Point. MAC_OPTIMIZE should be "1" for 21 or more computers. The parameter should be "1" even if 21 or more computers will be communicating with one access point for a short time.

▶ NOTE: *The best value depends on the site's particular operating environment. A Norand Systems Engineer should determine the appropriate value.*

PEER-TO-PEER

If the network is in a peer-to-peer (ad hoc) configuration, the value for this parameter must be "Y." If the PEN*KEY 6100 Computer is communicating through a 6710 Access Point, the value should be "N" for best performance. For most installations, the value will be "N."

ROAM_CONFIG

ROAM_CONFIG determines how the PEN*KEY 6100 Computer uses its internal RSSI values. The value for ROAM_CONFIG should be as follows:

- "0" if the computer will infrequently roam from one AP to another. The computer tries to stay with one access point longer.
- "1" if the access points' coverage areas overlap by some degree.
- "2" when the coverage areas of several access points have a large amount of overlap. The computer would quickly switch to the access point with the best RSSI levels.

▶ NOTE: *The best value depends on the site's particular operating environment. A Norand Systems Engineer should determine the appropriate value.*

STATION_TYPE (Ad Hoc)

The Proxim 2.4 GHz radio option has a mechanism where one Master station coordinates communications among other stations. All stations refer to the Master to determine where and when to hop. If a Master is not present, a station configured as an Alternate Master becomes the Master for a session. If a Master is present, the Alternate Master acts as a Station. Alternate Masters and Masters are usually found in peer-to-peer (ad hoc) networks, which do not contain 6710 Access Points.

Most open wireless LANs contain access points. In almost all cases, the access point is the Master and the PEN*KEY 6100 Computer is a Station. Therefore, STATION_TYPE should be "0" (Station) for the PEN*KEY 6100 Computer in a client-server–based network.

A peer-to-peer network leaves you with the task of configuring each PC-compatible computer on the open wireless LAN as Master, Alternate Master, or Station. In most cases, using the default values for each driver will work. However, you may need to change the configuration for performance or other issues. Following are some factors to consider for a peer-to-peer network:

- At most, one station must act as Master. If additional Masters will be set up, you should configure them as Alternate Masters so only one Master is on the network.
- The Master must be within radio range of the PC-compatible computers.
- The Master should be a station that will be stationary or remain on.
- Best performance results from configuring the fewest number of PC-compatible computers as Masters or Alternate Master.

Appendix D

6710 Access Point Specifications

About This Appendix

This appendix contains product, electrical, and environmental specifications for the 6710 Access Point. This appendix also describes the functionality of the access point's DIAG port.

Product Specifications

Following are product specifications for the access point.

Processor:	AMD 29200 RISC
Memory:	4 MB RAM/2 MB FLASH ROM
Distribution LAN compatibility:	ANSI/IEEE 802.3 (Ethernet communication standard) and DIX Version 2.0
Interface:	10BASE2 (thinnet), 10BASE5 (AUI or thicknet), and 10BASE-T (twisted pair) through ports on bottom panel
Card slots:	Two PC card-compatible slots
Mounting options:	Tabletop, wall, or ceiling

Electrical Specifications

The access point has one IEC connector for industry-standard three conductor ac input. The access point's internal power supply automatically detects the voltage level and frequency of the source power. Following are source power specifications.

Voltages:	Autosensing 110, 220, and 240 V ac
Frequency:	50 to 60 Hz
Safety:	UL/CSA (Underwriters Laboratory/ Canadian Standards Association), United States and Canada; CB (Competent Body) report for Europe

The access point complies with the following standards.

Immunity:	EN (Euro Norm) 50082-1 Generic Immunity Standard and ETS (European Telecommunication Standard) 300–339 Radio Equipment and Systems; Generic EMC (Electromagnetic Compatibility) for Radio Equipment
Emissions:	FCC Class B verified and CISPR* 22 (EN 55022) Class B radiated and conducted emissions under EN 50081-1, Generic Emissions Standard

* Comite International Special des Perturpations Radioelecturques/ International Special Committee on Radio Interference

Environmental Specifications

Following are environmental specifications for the access point.

Approximate size:	14.25 in x 6.80 in x 3.50 in (LWH) (36.19 cm x 17.27 cm x 8.89 cm)
Approximate weight:	3.75 lbs (1.70 kg)
Operating temperature (standard):	–4 °F to 122 °F (–20 °C to 50 °C)
Operating temperature (in NEMA enclosure):	–22 °F to 122 °F (–30 °C to 50 °C)
Humidity:	Up to 90 percent noncondensing

DIAG Port

The DIAG port is a 9-pin serial channel which provides access to the ROM command monitor and FLASH command interpreter. The port also provides the ability to reprogram FLASH, run manufacturing diagnostics, and view and set EEPROM parameters.

Appendix E

Host Connectivity Device Specifications

About This Appendix

This appendix contains product, electrical, and environmental specifications for the RC4030E Gateway and 6950 Enterprise Gateway Server. This appendix also describes the functionality of the RC4030E Gateway's DIAG and HOST ports.

RC4030E Gateway

Product Specifications

Following are product specifications for the RC4030E Gateway.

Processor:	Motorola 68302
Memory:	2 MB RAM/512 KB FLASH
Compatibility:	ANSI/IEEE 802.3 (Ethernet communication standard) and DIX Ethernet
Interface:	10BASE2 (thinnet) through BNC port; 10BASE-T (UTP) through UTP port
Mounting options:	Tabletop or wall

Electrical Specifications

Following are source power specifications for the United States and Canada.

Input voltage:	120 V ac
Output voltages:	+9.0 V dc, +13.5 V dc
Frequency:	60 Hz
Power consumption:	30 watts
Safety:	UL/CSA (Underwriters Laboratory/Canadian Standards Association), United States and Canada; CB (Competent Body) report for Europe

A different power supply is available for Europe and other areas having 230 V ac, 50 Hz power with TUV (Technischer Uberwachungs Verien/ Technical Supervision Society) approval. Another type of power supply is available for Japan and other areas having 100 V ac, 50 or 60 Hz power with MITI approval. Contact your Norand representative for more information about power supplies for international markets.

The RC4030E Gateway complies with the following standards.

Immunity:	EN (Euro Norm) 50082-1 Generic Immunity Standard
Emissions:	FCC Class B verified and CISPR* 22 (EN 55022) Class B radiated and conducted emissions under EN 50081-1, Generic Emissions Standard

* Comite International Special des Perturpations Radioelecturques/ International Special Committee on Radio Interference)

Environmental Specifications

Following are environmental specifications for the RC4030E Gateway.

Approximate size:	12.0 in x 6.4 in x 2.5 in (LWH) (30.48 cm x 16.26 cm x 6.35 cm)
Approximate weight:	4.30 lbs (1.95 kg)
Operating temperature:	–22 to 122 °F (–30 to 50 °C)
Humidity:	5 to 95 percent noncondensing

DIAG Port

The RC4030E Gateway's DIAG port is the channel through which gateway management tasks are done. Tasks include configuring and upgrading the gateway's system software, and checking the gateway's current FLASH and ROM versions. Following are DIAG port parameters.

Connector:	9-pin, D-subminiature female
Clocking:	Asynchronous
Type:	DTE
Interface:	RS-232

HOST Port

The RC4030E Gateway connects to the host computer through the gateway's HOST port. Following are HOST port parameters.

Connector:	25-pin, D-subminiature female
Clocking:	Asynchronous or synchronous
Type:	DTE
Interface:	RS-232 or V.35
Baud rate:	Asynchronous: 300 to 57600 bps Synchronous RS-232 and V.35: 1200 to 64000 bps, and external

6950 Enterprise Gateway Server

Product Specifications

Following are product specifications for the 6950 Enterprise Gateway Server.

Compatibility:	ANSI/IEEE 802.3 (Ethernet communication standard) and DIX Ethernet
Interface:	10BASE2 (thinnet), 10BASE5 (AUI or thicknet), and 10BASE-T (twisted pair) through installed Ethernet network interface card
Card slots:	5-slot back-plane
Mounting options:	Tabletop or wall

Electrical Specifications

The 6950 Enterprise Gateway Server's internal power supply autodetects the voltage level and frequency of the source power. The following chart lists source power specifications.

Input voltages:	Autosensing 110, 220, and 240 V ac
Output voltages:	+5.0 V ac and +12.0 V dc
Frequency:	50 to 60 Hz
Power consumption:	65 watts
Safety:	UL/CSA, United States and Canada; CB report for Europe

The 6950 Enterprise Gateway Server complies with the following standards.

Immunity:	EN 50082-1 Generic Immunity Standard
Emissions:	FCC Class B verified and CISPR 22 (EN 55022) Class B radiated and conducted emissions under EN 50081-1, Generic Emissions Standard

Environmental Specifications

Following are environmental specifications for the 6950 Enterprise Gateway Server.

Approximate size:	16.87 in (17.86 in with handle) x 4.32 in x 8.72 in (LWH) (42.85 cm, 45.36 cm with handle, x 10.97 cm x 22.15 cm)
Approximate weight:	12.00 lbs (5.44 kg)
Operating temperature:	32 to 140 °F (0 to 60 °C)
Humidity:	0 to 95 percent noncondensing

APPENDIX C

OWL NETWORK FRAME FORMATS

General format of an LLC frame.

Pre-amble	Flag	MAC-D Header	MAC-R Header	MAC-R Parms	Length/ Type	LLC Header	LLC Data	CRC	Flag

General Field Definitions for an LLC frame on a radio link.

Physical Layer Header	L bytes
MAC-D Protocol ID	1 byte (hexadecimal 01)
MAC-D Control	1 byte
MAC-D Destination Node ID	2 bytes
MAC-D Source Node ID	2 bytes
MAC-D OWL LAN ID	1 byte
MAC-D Channel Reservation	1 byte
MAC-R Control	2 bytes
MAC-R 802 Destination Address	6 bytes
MAC-R 802 Source Address	6 bytes
MAC-R Sequence	2 bytes
MAC-R Optional Parms	M bytes
802.3 Length or DIX Version 2 Type	2 bytes
LLC DSAP	1 byte
LLC SSAP	1 byte
LLC Control	1 bytes
optional SNAP header	5 bytes
LLC Data	N bytes
Physical Layer Trailer	P bytes

All multi-byte fields are transmitted in "big-endian" byte order (high byte first). Bit 0 is the low-order bit.

16-bit Network Node Identifier.

bit 15	Mulitcast Flag
0	unicast frame
1	multicast or broadcast frame

bit 14-13	Node Type
10	Terminal
01	Access Point (AP)
11	All Nodes

bit 12-0	Node Identifier
all 0's	root node identifer
all 1's	node without a network node identifier or any node

bit 2-0	Port Identifier for Access Point
all 1's	any port

Hexadecimal **2000** is the well-known 16-bit node ID of the root node.
Hexadecimal **DFFF** is the multicast default node ID of a terminal node.
Hexadecimal **BFFF** is the multicast default node ID of an access point.
Hexadecimal **FFFF** is the broadcast node ID for all nodes.

MAC-D Header.

Ethernet MAC-D Header.

802 Hop Destination Address	6 bytes
802 Hop Source Address	6 bytes
DIX Version 2 Type	2 bytes
MAC-D OWL Protocol ID	1 byte (hexadecimal 40)
MAC-D Control	1 byte
MAC-D LAN ID	1 byte
MAC-D Fragment ID	1 byte
MAC-D Length	2 bytes

Radio MAC-D Header.

MAC-D Protocol ID	1 byte (hexadecimal 40)
MAC-D Control	1 byte
MAC-D Destination Node ID	2 bytes
MAC-D Source Node ID	2 bytes
MAC-D LAN ID	1 byte
MAC-D Channel Reservation	1 byte

MAC-D Control Byte (8 bits).

Bits 7-4 in the MAC-D control byte are used to specify the frame type. A MAC-D PDU is classified as either a request or poll frame, depending on the state of the R/P bit. Poll frames are always control frames. A request MAC-D PDU can be either a control or data frame, depending on the state of the CONTROL bit. The MODE bit is set to 1, to indicaste master/slave mode, for any frames sent during a contention free period; otherwise, the MODE bit is 0 to indicate random access mode.

Data frames.

Data request control byte.

bit 7	R/P	0 = request frame
bit 6	CONTROL	0 = data frame
bit 5	START	1= first-in-chain
bit 4	STOP	1 = last-in-chain
bit 3	SEQ	sequence number, modulo 2
bit 2	PRIORITY	0=normal data, 1=high priority
bit 1	(reserved)	must be zero
bit 0	MODE	0=random access, 1=master/slave

The START bit is set ON in the first frame fragment in a series of fragments associated with a single MAC-D PDU.

The STOP bit is set ON in the last frame fragment in a series of fragments associated with a single MAC-D PDU.

Control frames.

Request control byte.

bit 7	R/P	0 = request frame
bit 6	CONTROL	1 = control frame
bit 5-4	TYPE	10 = RFP 00 = ENQ 01 = ABORT 11 = NSP
bit 3-1	(reserved)	must be zero
bit 0	MODE	0=random access, 1=master/slave

Poll control byte.

bit 7	R/P	1 = poll frame
bit 6	(reserved)	must be zero
bit 5-4	TYPE	00 = NSP-ACK 01 = REJECT 10 = CLEAR 11 = POLL
bit 3	SEQ	sequence number, modulo 2
bit 2-1	(reserved)	must be zero
bit 0	MODE	0=random access, 1=master/slave

Non-specific Poll (NSP) frame format.

MAC-D Header	8 bytes
NSP Flags	4 bits (currently must be zero)
NSP Level	4 bits
NSP Rotation	4 bits
NSP Level 0 Expansion Flags	4 bits
NSP Level 1 Expansion Flags	16 bits
NSP Level 2 Expansion Flags	64 bits

NSP-ACK frame format.

MAC-D Header	8 bytes
Priority (total wait time in .1 sec. units)	1 byte
Reservation Request	1 byte

MAC-R Header.

MAC-R Control	2 bytes
MAC-R 802 Destination Address	6 bytes
MAC-R 802 Source Address	6 bytes

If the RELAY bit is 1 in the MAC-R control field, then the MAC-R header includes the 802 hop source address. The hop source address is the 802 address of the relay node which forwarded the inbound packet. The RELAY bit is never ON in outbound packets.

MAC-R Control	2 bytes
MAC-R 802 Hop Source Address	6 bytes
MAC-R 802 Destination Address	6 bytes
MAC-R 802 Source Address	6 bytes

MAC-R Control Bytes (16 bits).

bit 15	Network type	0 = hierarchical, 1 = point-to-point

bit 14	(reserved)	must be zero
bit 13	Outbound Flag	1 = outbound
bit 12	REQ/RSP	0 = request, 1 = response
bit 11	(reserved)	must be zero
bit 10-8	MAC-R PDU Type	(see table below)
bit 7	MAC-R Parms Flag	1 = optional MAC-R parms
bit 6-4	(reserved)	must be zero
bit 3	MAC-R Retry Flag	1 = retry
bit 2	ATTI	1 = attach indication
bit 1	Relay Flag	0=from a child or parent, 1=relayed inbound PDU
bit 0	(reserved)	must be zero

MAC-R PDU Types.

000	Data/R-Data PDU
001	Alert PDU
010	Hello PDU
011	Attach PDU
100	Detach PDU
101	ECHO PDU
110	Registration PDU
111	(reserved)

Optional Bridge Parameters - general format.

1-bit end-of-parms flag	1 = last optional parm
7-bit parm type	(see table below)
1-byte parm length	length of parm value field in bytes
M-byte parm value	(value or list of values)

Optional Parameters.

Parm Type	Parm Length	Description
02h	6 bytes	802 address.
03h	N*4	Detach List. A list of 4-byte detach records. Each record consists of a 2-byte node ID followed by a 2-byte alert ID. The detach ID corresponds to an attach ID.
04h	N*4	Alert List. A list of 4-byte alert records. Each record consists of a 2-byte node ID followed by a 2-byte alert ID. The alert ID corresponds to an attach ID.
05h	N*2	Pending Message List. A list of 2-byte Node IDs.
07h	N bytes	Well-known alias.
08h	N*6 bytes	Remote Attach List. A list of 802 addresses for stations on a secondary LAN.
09h	1 byte	Load Indicator. An indication of the channel load based on frame frequency.
0Ch*	1 or 2 bytes	Awake time (in 100 millisecond units). All 1's denotes forever.
0Dh*	1 or 2 bytes	Awake time offset (in 100 millisecond units). An awake time offset of 0 specifies immediate delivery, even if no awake time is specified.
0Eh*	1 byte	Delivery service type. 0=deliver immediately. 1=store until the node is awake. 2=store until the node is awake; automatically set awake time.
0Fh*	1 byte	Maximum stored message count. The maximum number of hello times that the parent node should store a message for the source child node.
10h	2 bytes	Decendent count.
11h	2 bytes	Device Identifier.
12h	4 bytes	Distributed Clock
13h	N*26 bytes	Port registration list
14h	1 byte	Unicast flooding level
15h	1 byte	Multicast flooding level
16h	N bytes	Network address
17h	N*2 bytes	Port ID list
18h	2 bytes	UHF MAC-D RFP threshold size
19h	2 bytes	UHF MAC-D maximum fragment size
1Ah	2 bytes	Direct Sequence channel ID
20H	2 bytes	Direct Sequence MAC-D maximum fragment size

*Delivery service and awake time parameters (0C, 0D, 0E and 0F) are processed for all unicast messages.

MAC-R Request Packet Formats.

Data (Type 000).

MAC-D Header	
MAC-R Header	
MAC-R Sequence	2 bytes
Optional Parms - Max. stored message count. - Delivery service type. - Wake up time. - Wake up time offset.	N bytes
Length(802.3)/Protocol(DIX)	2 bytes
LLC Header (optional)	
LLC Data (optional)	

Alert (Type 001).

MAC-D Header	
MAC-R Header	
Alert Flags	2 bytes
Alert ID	2 bytes
Node 802 address	6 bytes
Node ID	2 bytes
(reserved)	2 bytes
Optional Parms	N bytes

Detach (Type 100).

MAC-D Header	
MAC-R Header	
Detach Flags	2 bytes
Detach ID	2 bytes
Node 802 address	6 bytes
Node ID	2 bytes
MAC-R Sequence	2 bytes
Optional Parms	N bytes

Echo (Type 101).

MAC-D Header	
MAC-R Header	
flags	2 bytes
Echo ID	2 bytes
Echo source address	6 bytes
Port	1 byte
Path count	1 byte
Path index	1 byte
Response Index	1 byte
Path parms length	2 bytes
Path list	M bytes
Path list parms	N bytes
Optional parms	O bytes
Data	P bytes

Echo path list entry.

802 address	6 bytes
Input port	1 byte
Output port	1 byte
Input port type	1 byte
Output port type	1 byte
Request RSSI	1 byte
Response RSSI	1 byte
Request delay	1 byte
Response delay	1 byte
Nr requests	2 bytes
Nr responses	2 bytes
Flags	1 byte (0x01=response required, 0x02=response from terminal)
Parm count	1 byte
Parm offset	2 bytes

Hello (Type 010).

MAC-D Header	
MAC-R Header	
Optional Parms	N bytes

Attach (Type 011).

MAC-D Header	
MAC-R Header	
Attach Flags	2 bytes
Attach ID	2 bytes
802 Address of Last Parent	6 bytes
Node ID	2 bytes
MAC-R Unicast Data Sequence	2 bytes
Optional Parms - Max. stored message count. - Delivery service type. - Wake up time. - Wake up time offset.	N bytes

Attach Flags.

bit 15-11	(reserved)	must be zero
bit 10	Multicast flood flag	1= if bit 8 is 1, then the secondary LAN requires multicast flooding
bit 9	Unicast flood flag	1 = if bit 8 is 1, then the secondary LAN requires unicast flooding
bit 8	Remote LAN flag	1 = the attaching AP is the designated bridge for a secondary LAN
bit 7	Child	1 = the attach is from a child
bit 6	AP flag	1 = the attaching node is an AP
bit 5	Update sequence flag	1 = sequence transferred from last AP parent
bit 4	Reset sequence flag	1 = reset sequence
bit 3	Remote flag	1 = the attach is for a remote (non-OWL) node
bit 2	ATTI	1 = attach indication
bit 1	Detach pending flag	1 = a detach request is pending for this attach
bit 0	Distributed flag	1 = the path to the source is through a distributed AP

Registration (Type 110).

MAC-D Header	
MAC-R Header	
Network Node ID	2 bytes
(reserved)	4 bytes (must be 0)
REGISTRATION Operation	1 byte (must be 0)
Reason Code	1 byte
Flags	2 bytes
Alias type (07h)	1 byte
Alias length	1 byte
Alias	N bytes
Device ID type (11h)	1 byte
Device ID length	1 byte
Device ID	2 bytes
Network address type (16h)	1 byte
Network address length	1 byte
Network address	N bytes

The network node ID must be set to the multicast ID for the node type (i.e. BFFF or DFFF). The optional alias field can contain a 1 to 16-byte node name. The optional device ID field can contain a 2-byte device identifier. The optional network address field can contain a network address. The address server will set the network node ID field to the next available node ID, for the node type, in the response PDU. If a node ID is not available, the field will be set to all 1's. Note that the MAC-D source node ID is the multicast node ID for the node type (i.e. BFFF or DFFF) on OWL radio ports.

An AP can request a port ID per port by including a registration list as an optional parameter in the registration request PDU. The parm type is hex. 13. The format of a registration list entry is shown below:

Registration list entry.

Ethernet address	6 bytes
Port ID	2 bytes (contains returned port ID)
Alias length	1 byte
Alias	16 bytes
Reason Code	1 byte

Dist_Attach (Type 111).

Ethernet MAC-D Header	
MAC-R Header	
Attach Flags	2 bytes
Attach ID	2 bytes
Optional Parms	N bytes

Bridge Response Packet Formats.

R-Data (Type 000).

MAC-D Header	
MAC-R Header	
MAC-R Sequence	2 bytes
Optional Parms - Max. stored message count. - Delivery service type. - Wake up time. - Wake up time offset.	N bytes
LLC Header (optional)	
LLC Data (optional)	

Alert (Type 001).

MAC-D Header	
MAC-R Header	
Alert Flags	2 bytes
Alert ID	2 bytes
Node 802 address	6 bytes
Node ID	2 bytes
(reserved)	2 bytes
Optional Parms	N bytes

Detach (Type 100).

MAC-D Header	
MAC-R Header	
Detach Flags	2 bytes
Detach ID	2 bytes
Node 802 address	6 bytes
Node ID	2 bytes
MAC-R Sequence	2 bytes
Optional Parms	N bytes

Echo (Type 101).

MAC-D Header	
MAC-R Header	
flags	2 bytes
Echo ID	2 bytes
Echo source address	6 bytes
Port	1 byte
Path count	1 byte
Path index	1 byte
Response Index	1 byte
Path parms length	2 bytes
Path list	M bytes
Path list parms	N bytes
Optional parms	O bytes
Data	P bytes

Hello (Type 010).

MAC-D Header	
MAC-R Header	
Network Node ID	2 bytes
Cost-to-root	2 bytes (0xFFFF = infinity)
Hello Seed	1 byte
Offset	1 byte 0-254 = transmission offset time in hundredths of seconds. 255 = unscheduled.
Root Priority	1 byte
Root Sequence Number	1 byte
Root 802 Address	6 bytes
Hello Period	1 byte (average hello period in tenths of seconds. e.g. 20 = 2 seconds)
Multi-count	1 byte (number of multicast messages which will follow this HELLO)
Flags	1 byte (must be 0)
Bridge Priority	1 byte
Load	1 byte
Optional parms (e.g.) - Pending Message List - Alert List - Distributed Clock	N bytes

Attach (Type 011).

MAC-D Header	
MAC-R Header	
Attach Flags	2 bytes
Attach ID	2 bytes
802 Address of Last Parent	6 bytes
Node ID	2 bytes
MAC-R Sequence	2 bytes
Optional Parms - Max. stored message count. - Delivery service type. - Wake up time. - Wake up time offset.	N bytes

Registration (Type 110).

MAC-D Header	
MAC-R Header	
Network Node ID	2 bytes
(reserved)	4 bytes (must be 0)
REGISTRATION Operation	1 byte (must be 0)
Reason Code	1 byte
Flags	2 bytes
Alias type (07h)	1 byte
Alias length	1 byte
Alias	N bytes
Device ID type (11h)	1 byte
Device ID length	1 byte
Device ID	2 bytes

APPENDIX D

UHF/DIRECT SEQUENCE MAC-D PROTOCOL SPECIFICATION

Introduction.

This document, in conjunction with the OWL network frame format specification, defines the MAC-D link protocol used on Direct Sequence and synthesized UHF radio links in an OWL network. The document defines peer-to-peer random-access operation for exemplary Direct Sequence and UHF links, and master-slave extensions for UHF links. Peer-to-peer random-access periods alternate with master-slave contention-free periods, if the master-slave extensions are enabled. Random-access operation allows access point and terminal nodes to randomly access the channel whenever the channel is idle. The master-slave extensions assume that a single master controls access to the channel during "contention-free periods". Where possible, master-slave extensions are documented in separate sections, which can be ignored for strictly random-access implementations.

Functional requirements and capabilities.

The MAC-D layer:

- accepts transmit requests from the MAC-R layer and passes frames to the physical layer for transmission.
- filters out frames which do not belong to the OWL network of the local device.
- filters out frames which are not directed to the local device.
- forwards frames to the MAC-R layer which are directly addressed to the local device, or are broadcast or multicast to the local device.
- transparently fragments and reassembles unicast MAC-R PDUs, which exceed the maximum MAC-D frame size.
- retransmits lost (i.e. corrupted) unicast MAC-D frame fragments.
- detects and discards duplicate MAC-D data frames and data frame fragments.
- regulates access to the communications channel.
- maintains and provides diagnostic statistics for higher layers.

Terminology and definitions.

MAC-D PDU - MAC-D sub layer protocol data unit (PDU).

MAC-R PDU - MAC-R sub layer PDU.

frame - a physical layer PDU or the MAC-D PDU contained in a physical layer PDU.

fragment - a MAC-D PDU. "Fragment" is typically used to denote a frame which is part of a sequence of frames which contain the data for a single MAC-R PDU.

A frame sequence is a group of 1 or more frame fragments which contain a single MAC-R PDU.

originator - the node which originates the transmission of a frame sequence.

sink - the destination of a unicast frame sequence.

Clear channel detect (CCD) is asserted to indicate that there is not an active transmission within the physical range of the radio.

Busy channel detect (BCD) is the negation of CCD.

A LBT (listen-before-talk) non-persistent channel access algorithm is used to acquire the radio channel.

A CA sequence is the transmission of one or more fragments following a single execution of the LBT channel access algorithm. In the absence of errors, a frame sequence is transmitted as a single CA sequence.

An interframe gap is defined as the minimum idle time allowed between frames which are part of a frame sequence.

An interpoll gap is defined as the time between poll frames which belong to a single CA sequence. The maximum interpoll gap time is BUSY_PULSE_TIME.

Physical layer requirements.

The physical layer:

- prefixes synchronization and framing (i.e. length) bytes and appends FCS bytes to transmitted frames.
- removes synchronization bytes, framing bytes and FCS bytes from received frames.
- discards frames with physical layer errors (i.e. FCS errors).

The physical layer must provide the following primitives:

Open - opens the physical port.

Start - turns the receiver on.

Stop - turns the receiver off.

Transmit - transmits a single MAC-D frame fragment as a single physical layer frame. The physical layer is responsible for prefixing synchronization and framing bytes (i.e. length or delimiter) and for appending FCS bytes to the end of the frame.

Receive_indication - posts a frame without physical layer errors to the MAC-D entity. The physical layer is responsible for removing synchronization and framing bytes (i.e. length or delimiter) from the start of the frame, and FCS bytes from the end of the frame. The receive_indication includes a pointer to a MAC-D PDU and the length of the MAC-D PDU.

MAC-D addresses.

Each MAC-D port is assigned a unicast ethernet address. The MAC-R entity in a device registers the ethernet port address and obtains a 2-byte address for each port. Unicast 2-byte addresses are unique within the domain of an OWL LAN ID. Each MAC-D frame (or frame fragment) contains a 2-byte destination address and a 2-byte source address to identify, respectively, the destination and source port of a MAC-D frame.

The high-order bit of a 2-byte MAC-D address is the multicast bit. The multicast bit is set ON to denote multicast addresses. The low-order 15 bits are divided into a node type (bits 14-13) and a port ID (bits 12-0). The node type can be "AP", "terminal", or "any". The default MAC-D address is a multicast address with the respective node type and a port ID of all 1's. An address of all 1's is the MAC-D broadcast address.

If the multicast bit is set ON in either the destination or source address, then the destination node(s) can not respond (i.e. with a POLL or CLEAR frame). A multicast frame sequence consists of a single EOD frame.

MAC-R sub layer interface.

Each node in the network has a single MAC-R entity which invokes a MAC-D entity per port to send and receive frames on the port.

The MAC-D layer must provide the following service primitives. (Note that primitive names and implementation details may change.)

MACD_init - used to initialize the MAC-D entity.

MACD_hl_bind - used to bind a higher layer entity to the MAC-D entity.

MACD_set_LAN_ID - used to set the LAN ID and (optional) LAN ID mask.

MACD_set_address - used to set the MAC-D 2-byte address.

MACD_get_info - used to obtain MAC-D configuration information.

MACD_transmit_request - used to transmit a MAC-R PDU. Transmit requests are sequenced. The MAC-R entity is limited to a single outstanding transmit request until a "transmit_ok" indication is received from the MAC-D layer.

MACD_transmit_error_indication - used to post the status of a transmit request.

MACD_transmit_ok_indication - used to notify the MAC-R entity that a transmit request has been accepted. The MAC-R entity is limited to a single outstanding transmit request. A request is considered "outstanding" until a transmit_ok_indication is received.

MACD_receive_data_indication - used to post a received MAC-R PDU to the MAC-R entity.

MACD_start - used to put the MAC-D entity in a LISTEN state (i.e. turn on the radio receiver).

MACD_stop - used to put the MAC-D entity in an IDLE state (i.e. turn off the radio receiver).

The MAC-R entity initializes each MAC-D entity with MACD_init, binds to the MAC-D entity with MACD_hl_bind, and puts the MAC-D entity in a LISTEN state with MACD_start.

Initially, the MAC-D entity uses a default multicast address consisting of its node type and a port ID of all 1's. The MAC-R entity uses the ethernet address of the MAC-D entity to obtain a unique 2-byte address from an address server in the OWL root node. The MAC-R entity uses MACD_set_address to set the MAC-D address to the unique value. If the root node changes, the MAC-R entity uses MACD_set_address to reset the MAC-D address to the default value, and the process is repeated.

A MAC-D entity uses the MACD_receive_data_indication primitive to post receive MAC-R PDUs to the MAC-R entity. The receive data primitive includes the MAC-D address of the originator, a MAC-D port identifier, and signal strength information.

The MAC-R entity uses MACD_transmit_request to request a MAC-D entity to transmit a MAC-R PDU. If the size of a MAC-R PDU exceeds the maximum MAC-D frame fragment size, then the PDU is transparently fragmented and re-assembled by the MAC-D originator and sink, respectively. The transmit request primitive includes the destination MAC-D address and a 2-bit P_FLAG parameter. If the multicast bit is ON in the destination address, then the frame can not be fragmented. If P_FLAG is non-zero, then

the MAC-D entity delays a random back off time between 0 and P_FLAG_BACKOFF LBT[pflag_value-1] slot times, before transmitting the PDU.

The MAC-R entity can change the MAC-D state from LISTEN to IDLE (i.e. disable the radio receiver) with MACD_stop. The MAC-R entity puts the MAC-D entity back into the LISTEN state with MACD_start.

MAC-D protocol specification.

MAC-D header.

Destination address.

The destination address specfies the multicast or unicast of the destination (i.e. sink) node(s).

Source address.

The source address is the the address of the originator. The source address may be a multicast address if the MAC-D entity has not been assigned a unique unicast port address.

Protocol ID.

The MAC-D entity discards frames which contain a protocol ID value which it does not support. The protocol field is the first field in the MAC-D header and defines the format of the MAC-D header.

LAN ID.

A MAC-D frame belongs to the MAC-R spanning tree specified by the LAN ID in the MAC-D header. The MAC-D entity discards frames which do not belong to the spanning tree to which it is currently attached after, possibly, updating channel reservation information. The MAC-R entity passes the LAN_ID value to the MAC-D entity during initialization.

Channel Reservation.

The channel reservation byte is used to reserve the communications channel for a unicast frame sequence or for a succeeding multicast frame. The channel reservation value is equal to the total number of data bytes in the sequence divided by 16. The reservation value can not be greater than MAX_CHAN_RESERVE. The channel reservation in RFP and DATA frames is echoed in associated POLL frames.

Control byte.

Bit definitions

R/P (request/poll) - the R/P bit is used to distinguish MAC-D request and poll PDUs. The R/P bit is set OFF in request frames. The R/P bit is set ON in a poll frames.

CONTROL - the CONTROL bit is set ON in control request frames, and is set OFF in data (DATA or EOD) request frames. (The CONTROL bit must be zero in poll frames.)

START - the START bit is set ON in the first data (DATA or EOD) fragment in a frame sequence.

STOP - the STOP bit is set ON in the last data (EOD) fragment in a frame sequence.

TYPE(2) - the START/STOP bits are used as frame "type" bits in control request frames and poll frames.

The R/P, CONTROL, and START/STOP (or TYPE) bits define the frame type.

SEQ - the SEQ bit is used to sequence MAC layer data frames, modulo 2. The SEQ bit is used to detect and discard duplicate data fragments. The SEQ bit is 0 in the first data fragment in a sequence. The SEQ bit in a POLL frame is set to the next expected DATA SEQ.

PRIORITY - the PRIORITY bit indicates the priority - high(1) or normal(0) - of a MAC-R PDU. The MAC-R entity associates a priority with each transmit request. The MAC-D entity in a sink passes the priority to the MAC-R entity with each receive data indication. The PRIORITY bit value is the same for all data frames in a sequence.

MODE - the mode bit is set ON for frames sent in master-slave mode; otherwise, it is set OFF.

MAC-D frame types.

The high-order 4 bits in the MAC-D control byte define the frame type. MAC-D PDUs are categorized as either request or poll frame fragments. Request frames are transmitted by an originator and poll frames are transmitted by the sink.

Request frame types.

A DATA frame is used to send higher-layer data.

An EOD (end-of-data) frame is used to send higher-layer data and is the last data frame in a sequence of one or more data frames. Note that a bracket of data frames may consist of a single EOD frame.

An RFP (request-for-poll) frame is used to request polling from a sink.

An ABORT frame is used to abort the transmission of a frame sequence.

An ENQ (inquiry) frame is used to determine the SEQ state of the sink.

An MSP (master slot poll) frame is used by the master to poll child nodes in master-slave mode.*

Poll frame types.

A POLL frame is used to solicit a data frame from the originator and to return the current SEQ state.

A CLEAR frame is used to return the SEQ state and to inform all listening nodes that the last frame in a frame sequence has been received.

A REJECT frame is used to return an undefined SEQ state or to indicate that a received request frame was invalid.

An MSP-ACK frame is sent in response to an MSP frame from the master, in master-slave mode, to request polling by the master.*

* master-slave mode only.

Frame/packet filtering.

When a MAC-D entity is in a LISTEN state it is continuously listening on its assigned port. The MAC-D entity is always in promiscuous mode and receives all MAC-D layer frames. Frames with physical layer (i.e. FCS) errors are discarded. Valid data frames are re-assembled into a complete PDU and are posted to the MAC-R entity if:

1) the protocol ID and LAN ID in the MAC-D header match the MAC-D protocol ID and LAN ID, and

2) the destination address in the MAC header a) is equal to the unicast address of the local port, or b) is an acceptable multicast or broadcast address.

The high-order multicast bit is set ON in all multicast or broadcast addresses. A frame with a multicast or broadcast destination address is accepted if the node type specifies a group to which the local node belongs (i.e. AP or any) and either a) the port ID is all 1's, or b) the port ID is equal to the ID of the local port.

The MAC-D entity will simply discard a received frame if the protocol ID in the MAC-D header does not match the MAC-D protocol ID (i.e. assigned at compile time).

The MAC-D entity will update channel reservation information and then discard a received frame if the LAN ID in the header does not match the assigned LAN ID.

If an optional LAN ID mask has been set then received data frames with a broadcast destination address will be posted to the MAC-R entity if the LAN ID in the header ANDed with the mask matches the current MAC-D LAN ID ANDed with the mask. The default LAN ID mask is all 1's (i.e. the LAN ID must match exactly). 0 bits in the mask are used to indicate "don't care" bits. The LAN ID mask allows the MAC-R layer to hear broadcast HELLO frames from other LANs.

A LAN ID of all 1's is used as a universal LAN ID. Received data framess with the universal LAN ID and a broadcast destination address are posted to the MAC-R entity.

Appendix 1 contains example filtering code.

Channel access.

The channel access algorithm is either 1) random-access or 2) master-slave, depending on the current channel access mode.

Random-access mode channel access.

A listen-before-talk (LBT) channel access algorithm is executed to gain access to the channel before the first data frame in a CA sequence is transmitted. All other frames in a CA sequence may be sent without executing the channel access algorithm. The idle time between frames which belong to a single CA sequence must be less than the maximum interframe gap time. Note that an RFP frame is first in a unicast frame sequence, whereas an RFP or ENQ frame can be first in a CA sequence.

A CCD slot is a CSMA slot time which includes <busy sense time> + <turnaround time> + <processing delay>. The busy sense time is the elapsed time from the beginning of a transmission until CCD is asserted at a receiver. The turnaround time is the time required by a half-duplex transceiver to switch from a receive state to a transmit state. The processing delay includes the hardware/software processing time required to initiate a transmission.

A CA slot is a CSMA slot time and includes the time it takes an originator to transmit an RFP frame plus the time until the sink transmits a response and BCD is asserted at the originator.

An LBT slot is a weighted average of the CA slot time and the CCD slot time, where weighting is based on the expected worst-case number of nodes and the expected worst-case percentage of hidden nodes. LBT back off times are defined as integer multiples of LBT slots.

CCD idle time is the minimum time that a node must sense the radio channel idle before starting a transmission. The CCD idle time is a function of the access priority associated with each frame sequence. The CCD idle time for high priority frames originated by an access point (AP) is 0.

On wired links, a simple CSMA algorithm forces nodes to detect an idle channel for a CSMA idle time, which exceeds the interframe gap time, before initiating a CA sequence.

On radio links "hidden nodes" can cause throughput to be significantly degraded if a simple CSMA algorithm is used for channel access. The LBT algorithm allows nodes to reserve the channel for a CA sequence. The channel reservation in a request frame is echoed in the associated poll frame. Therefore, the reservation will acquiesce any nodes within range of either the originator or sink. Sleeping nodes must detect an idle radio channel for an LBT idle time which exceeds the maximum interpoll gap time, before accessing the channel. By listening for longer than the interpoll gap time, a node will detect a conversation in progress, if either the originator or sink is in range, even if the active transmitter is "hidden".

MAC-D frames contain a reservation field which is used to reserve the channel for the duration of a frame sequence. The reservation in an RFP, ENQ, or DATA frame indicates the number of untransmitted data bytes in the associated frame sequence. A reservation is calculated as the number of untransmitted data bytes in a frame sequence divided by 16. The actual value stored in a unicast fame is limited to MAX_CHAN_RESERVATION, where MAX_CHAN_RESERVATION is greater than one fragment time. The reservation in a POLL or REJECT frame is always equal to the reservation is the associated RFP, DATA, or ENQ frame, with one exception. The reservation in a retransmitted POLL or CLEAR is always set to the original value. The reservation in a unicast EOD frame and CLEAR frame is 0, except for frames "chained" together by an AP. A multicast frame which has a length greater than MULTI_RFP_THRESHOLD bytes, must be preceded by a muticast RFP frame. The reservation in a multicast RFP frame indicates the length of the associated multicast frame and is not limited to MAX_CHAN_RESERVATION (i.e. because the multicast frame is not fragmented). The reservation in a multicast EOD frame can be set to a short non-zero value (i.e. 2) to reduce channel contention for succeeding multicast frames transmitted as a chained group.

If an originator does not receive an expected response (i.e. a POLL or CLEAR) frame from the sink, during the transmission of a frame sequence, then the originator solicits a retransmission of the last response from the sink by sending an ENQ frame to the sink. The reservation in the ENQ frame is the same as the associated data fragment and, therefore, does not include the length of the (unacknowledged) data fragment. If the data (i.e. DATA or EOD) fragment was lost, then the POLL frame for a previous RFP or DATA frame is retransmitted with the original reservation. If the poll (i.e. POLL or CLEAR) frame was lost, then the poll frame for the current data fragment is retransmitted with the current reservation.

The LBT algorithm requires each node to maintain two variables - RESERVE_TIME and RESERVE_NODE. A MAC-D port operates in promiscuous mode and constantly updates channel reservation information. The channel can be reserved by a received frame if 1) the frame is unicast or multicast, 2) the protocol ID matches, and 3) the LAN ID does not match or the destination address does not equal the address of the local port. The RESERVE_TIME variable is set to the current time plus the time required to send the number of bytes specified in the channel reservation field (including overhead) when a unicast frame is received. The RESERVE_NODE variable is set to the concatenated LAN ID and unicast address of the node which reserved the channel - the source of a request frame or the destination of a poll frame. The RESERVE_TIME variable can be increased by any node, but can only be decreased by

the RESERVE_NODE node. A potential sink node must discard an RFP frame if the RESERVE_TIME is greater than the current time and the originator is not the RESERVE_NODE node. A reservation is canceled if a unicast request frame is received from the RESERVE_NODE node and the LAN ID and destination address specify the local port. Therefore, a sink node can respond to an RFP or ENQ (i.e. with a POLL, REJECT, or CLEAR), if, and only if, the channel is not reserved or if the RFP or ENQ was transmitted by the RESERVE_NODE node.

An idle MAC-D entity resets an LBT retry counter and senses the channel before initiating a CA sequence. If BCD is asserted or the channel is reserved, then the MAC-D entity increments the LBT retry counter, selects a random back off time, and delays for the random back off time before re-accessing the channel; otherwise, the MAC-D entity will start a CA sequence (i.e. by transmitting an RFP frame). If the expected response to an RFP or ENQ frame is lost, then the MAC-D entity will also increment the LBT retry count and delay before attempting to re-access the channel.

The back off time is a random value between 0 and an entry in the TX_BACKOFF_TABLE, indexed by the number of LBT retries. If the channel is reserved, then the remaining reservation time is added to the back off time. A timer is started for the total delay time. The channel is sensed again when the timer expires and the process is repeated. Note that channel reservations are monitored and RESERVE_TIME is updated while the timer is running.

If BCD is not reliable (i.e. can be falsely asserted due to background noise), then BCD should be ignored if the channel does not become reserved within the time required to send a frame of length MULTI_RFP_THRESHOLD bytes.

A node selects a CCD idle time period associated with the priority (high or normal) of a MAC-R PDU for sensing the channel. If the CCD idle time is 0, then the current state of the channel is sensed; otherwise, the channel is sensed busy if BCD is asserted or the channel becomes reserved during the CCD idle time period.

The input which produces BCD is ANDed with a managed boolean variable - BCD_ENABLE. If BCD_ENABLE is FALSE, then BCD is always false. In this case, busy channel assessment is based solely on reservations.

Master-slave mode channel access.

If "master-slave mode" is enabled, then the channel access mode can alternate between peer-to-peer random access and master-slave contention-free access. The channel is considered to be reserved by the master (i.e. access point) during a "contention-free period". All frame sequences are initiated by the master; therefore, the channel reservation and BCD are ignored in master-slave mode. Master-slave mode is described in detail below.

Example random-access transmission sequences.

The following sequences illustrate typical transmission sequences for sending a sequence of frames from an originator to a sink, in random-access mode. The SEQ value for data and poll frames is indicated in parentheses.

Sequence 1 - illustrates a sequence without errors and with no fragmentation:

RFP	——————>	
	<——————	POLL(0)
EOD(0)	——————>	
	<——————	CLEAR(1)

Sequence 2 - illustrates a sequence without errors and with fragmentation:

RFP	——————>	
	<——————	POLL(0)
DATA(0)	——————>	
	<——————	POLL(1)
DATA(1)	——————>	
	<——————	POLL(0)
EOD(0)	——————>	
	<——————	CLEAR(1)

Sequence 3 - illustrates a lost data fragment:

RFP	——————>	
	<——————	POLL(0)
DATA(0)	(lost data fragment)	
ENQ	——————>	
	<——————	POLL(0)
DATA(0)	——————>	
	<——————	POLL(1)
DATA(1)	——————>	
	<——————	POLL(0)
EOD(0)	——————>	
	<——————	CLEAR(1)

In sequence 3, note that the reservation in the retransmitted POLL(0) frame is the same as the reservation in the initial POLL(0) frame.

Sequence 4 - illustrates a lost poll frame:

RFP	——————>	
	<——————	POLL(0)
DATA(0)	——————>	
	(lost poll frame)	POLL(1)
ENQ	——————>	
	<——————	POLL(1)
DATA(1)	——————>	

	<----------	POLL(0)
EOD(0)	---------->	
	<----------	CLEAR(1)

Sequence 5 - illustrates a reserved channel at the sink:

RFP	---------->	
		(channel reserved)
(delay)		
RFP	---------->	
	<----------	POLL(0)
EOD(0)	---------->	
	<----------	CLEAR(1)

Sequence 6 - illustrates a lost EOD frame:

RFP	---------->	
	<----------	POLL(0)
EOD(0)	(lost EOD frame)	
ENQ	---------->	
	<----------	POLL(0)
EOD(0)	---------->	
	<----------	CLEAR(1)

Sequence 7- illustrates a lost CLEAR frame:

RFP	---------->	
	<----------	POLL(0)
EOD(0)	---------->	
	(lost clear frame)	CLEAR(1)
ENQ	---------->	
	<----------	CLEAR(1)

Sequence 8 - illustrates a multicast transmission:

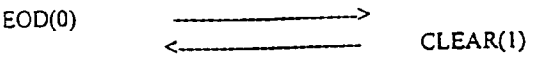
EOD(0)	---------->

Sequence 9 - illustrates a multicast transmission which exceeds MULTI_RFP_THRESHOLD bytes:

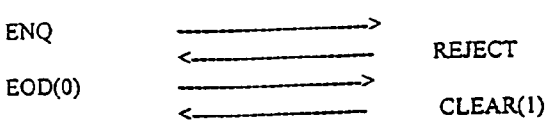
RFP	---------->
EOD(0)	---------->

The multicast RFP frame, in sequence 9, is used to reserve the channel for the duration of the data (EOD) frame.

Sequence 10 - illustrates a unicast transmission where the initial RFP/POLL seqeunce is (optionally) omitted:



Sequence 11 - illustrates a unicast transmission where the initial RFP/POLL seqeunce is (optionally) omitted; the EOD frame is lost; and the sink does not have a receive SEQ table entry for the originator:



In sequence 11, a channel access is required before the first EOD transmission and the ENQ transmission; however, a channel access is not required for the second EOD transmission.

In general, the LBT algorithm must be used to access the channel before each RFP or ENQ frame is transmitted by the originator. The LBT algorithm must also be used to access the channel before a single-fragment EOD frame is transmitted without a preceding RFP.

Note that the sink must cache the SEQ state of the originator for a time greater than MAX_ENQ_RETRY_TIME after it transmits the CLEAR frame, in case the CLEAR frame is lost. A sink retransmits the CLEAR if it receives an ENQ and the SEQ state of the originator is known; otherwise the sink responds with a REJECT frame.

MAC-D States.

The state of the MAC-D entity depends on the "channel state" and "channel access mode". The channel state can be IDLE, LISTEN, TRANSMIT, or RECEIVE. The channel access mode can be RANDOM-ACCESS or MASTER-SLAVE. If the mode is MASTER-SLAVE, then the master-slave state is OUTBOUND or one of several inbound states; otherwise, the master-slave state is IDLE.

MAC-D operation in the TRANSMIT and RECEIVE states depends on the channel access mode. For master-slave mode, the operation also depends on whether the node is functioning as a master or slave, which is a compile-time option.

High-level MAC-D channel states.

Each MAC-D entity is in one of the following high-level states.

IDLE - The MAC-D entity is not enabled.

LISTEN - The MAC-D entity is enabled and is receiving frames in promiscuous mode. Note that the MAC-D entity can not enter the RECEIVE or TRANSMIT states until it has been assigned a unique unicast port address.

RECEIVE - The MAC-D entity is receiving a unicast frame sequence. The unicast receive state machine is active. In master-slave inbound mode, the RECEIVE state is also entered for slot polling.

TRANSMIT - the MAC-D entity is transmitting a frame sequence. The TRANSMIT type is set to MULTI_TX or UNI_TX, for multicast and unicast transmissions, respectively. If the type is UNI_TX, then the unicast transmit state machine is active.

Note that the MAC-D entity is still receiving frames in promiscuous mode in both the RECEIVE and TRANSMIT states.

Transitions to the RECEIVE and TRANSMIT states can only be made from the LISTEN state. For example, node must complete or abort an active frame sequence before it can transition to/from the RECEIVE state from/to the TRANSMIT state. The unicast transmit and receive state machines cannot be active concurrently.

The MAC-D entity must maintain a transmitter_busy variable to ensure that physical layer transmit requests are single threaded. The transmitter_busy variable is TRUE if, and only if, a physical layer transmission is in progress.

Random-access protocol state machines.

Multicast frames.

No state machine is required to send or receive multicast and broadcast frames. Multicast and broadcast PDUs are transmitted as a single frame when the channel is available. Received multicast or broadcast frames are simply posted to the MAC-R layer or are discarded. Note that a received frame is determined to be multicast if the multicast bit is set ON in either the destination or source address. A multicast frame is always an EOD request frame with both the START bit and the STOP bit set ON. If the size of a multicast frame is more than MULTI_RFP_THRESHOLD bytes, then the orginator must transmit an RFP frame, with a broadcast destination address, to reserve the channel for the duration of the multicast frame.

The reservation field in the RFP frame is set to the number of bytes in the multicast data frame, divided by 16 (i.e. the reservation can be greater than MAX_CHAN_RESERVE).

Frame sequencing.

The state machines below do not include states which reflect the value of send and receive sequence variables - VS and VR. Each data frame (DATA or EOD) is sequenced with a 1 bit SEQ number. The originator stores the value of the SEQ bit for the current frame fragment in VS. The sink stores the value of the next expected SEQ in VR. The originator resets VS to 0 when it sends an RFP frame. The receiver resets VR to 0 when it receives an RFP frame. The sink increments VR (mod 2) each time a data (DATA or EOD) frame is accepted. The originator sets VS to the value of the SEQ bit in a received POLL or CLEAR frame. Note that the SEQ bit in a POLL or CLEAR frame is set to the next expected SEQ value in a data frame. (See the example transmission sequences.) The sink must cache its VR value after it sends a CLEAR frame (i.e. in a receive SEQ state table) so that it can correctly respond to an ENQ frame if the CLEAR frame is lost.

The MAC-D control byte contains a START bit and a STOP bit. The START bit is set on in a first-in-chain (FIC) frame fragment. The STOP bit is set on in a last-in-chain (LIC) frame fragment. The STOP bit is always on in an EOD frame and off in a DATA frame. Both the START and STOP bits are set on in an only-in-chain (OIC) frame. OIC (i.e. EOD) frames are sent in a single fragment.

A MAC-D entity can be implemented so that an OIC EOD frame can be sent without a preceding RFP frame. In this case, the MAC-D entity must also cache its VS value for each active sink node (i.e. in a transmit SEQ state table). Instead of resetting VS to 0, the sink sets VS to the cached value for the current sink before sending the single-fragment EOD frame. A preceding RFP must be transmitted if the size of the frame is greater than RFP_THRESHOLD bytes or if the SEQ value for the sink is not cached. Note that RFP_THRESHOLD should be small (i.e. 0 to 20) to avoid sending larger frames without a preceding channel reservation.

VS is used to store the SEQ value for the originator in the TRANSMIT state machine. VR is used to store the SEQ value for the sink in the RECEIVE state machine. The orginator should delete its transmit SEQ state table entry for the sink when it enters the transmit state machine (i.e. after storing the SEQ value in VS). Likewise, the sink should delete its receive SEQ state table entry for the originator when it enters the receive state machine. The transmit and receive SEQ state table entries are (re)created at the end of a successful frame sequence (i.e. after a CLEAR is received or transmitted, respectively). Therefore, a state table entry will not exist if a frame sequence can not be completed.

Note that the following state tables assume that RFP_THRESHOLD is less than the maximum fragment size; however, the protocol can easily be extended to accomodate an RFP_THRESHOLD value which is greater than the maximum fragment size.

Unicast transmit state machine.

The MAC-D entity starts the unicast transmit state machine to send a unicast sequence of 1 or more frame fragments. Only timer events and unicast poll frames destined to the local port are passed to the transmit state machine.

The MAC-D entity can be implemented so that the RFP/POLL sequence which precedes a unicast transmission is not required for a data frame of less than RFP_THRESHOLD bytes. The second state transition table below descibes the transmit states if the RFP is omitted. An RFP is always required if the originator does not have an entry for the sink in its transmit SEQ state table or if the frame size is greater than RFP_THRESHOLD bytes.

State descriptions:

(LISTEN) - The MAC-D entity does not have an outstanding MAC-R transmit request. The MAC-D state is LISTEN or RECEIVE.

TX_PEND - The MAC-D entity has an outstanding MAC-R transmit request and is waiting to access the channel before entering the transmit state machine. The MAC-D state is LISTEN or RECEIVE.

TX_IDLE_R - Entry point from the TX_PEND state if the PDU size is not less than RFP_THRESHOLD bytes, the sink SEQ state is unknown, or (optionally) an RFP is always required.

RDY_RFP - The state machine has a sequence of 1 or more frame fragments to transmit and is waiting to acquire the channel before sending an RFP.

S_RFP - The state machine has sent an RFP frame and is waiting for a POLL frame.

RDY_ENQ - A DATA frame or the POLL response was lost and the state machine is waiting to acquire the channel before sending an ENQ.

RDY_ENQ_E - An EOD frame or the CLEAR response was lost and the state machine is waiting to acquire the channel before sending an ENQ.

S_ENQ - The state machine has sent an ENQ frame to determine the status of the current DATA frame and is waiting for a POLL frame.

S_ENQ_E - The state machine has sent an ENQ frame to determine the status of the current EOD frame and is waiting for a POLL or CLEAR frame.

S_DATA - The state machine has sent a DATA frame and is waiting for a POLL frame.

S_EOD - The state machine has sent an EOD frame and is waiting for a CLEAR frame.

State descriptions for the optional transmit state table for omitting the initial RFP:

TX_IDLE_NR - (optional) entry point from the TX_PEND state, if the PDU size is less than or equal to RFP_THRESHOLD bytes and the sink SEQ state is known.

S_EOD_NR - The state machine has sent an OIC EOD FRAME, without a preceding RFP frame, and is waiting for a CLEAR frame.

RDY_ENQ_NR - The EOD frame or the CLEAR response was lost and the state machine is waiting to acquire the channel before sending an ENQ.

S_ENQ_NR - The state machine has sent an ENQ frame to determine the status of the current EOD frame and is waiting for a REJECT or CLEAR frame.

Transmit state counters.

The state machine maintains four counters: 1) retry_count is used to count the number of consecutive RFP or ENQ frames which are transmitted (i.e. without receiving a response); 2) data_retry_count is used to count the number of times that a single data fragment is retransmitted in a CA sequence; 3) abort_count is used to count the number of times that a transmission sequence is aborted. An aborted sequence is

restarted at the beginning if abort_count is less than the maximum value; otherwise, an ABORT_RETRY error is returned. 4) lbt_retry_count is incremented whenever the channel is sensed busy or an expected response is not received. It is reset to 0 when a poll frame is received. A random delay is calculated as a function of lbt_retry_count. A CHAN_ACCESS error is returned if lbt_retry_count exceeds the maximum value. (CHAN_ACCESS errors are not included in the state machine logic.)

Transmit state timers.

A POLL_TIMEOUT timer is started following the transmission of an RFP, ENQ, DATA, or EOD frame, when a response is expected. Note that the timer must be started immediately after the associated transmisssion completes. The time-out value is larger than the interframe gap time plus the time required to transmit a POLL or CLEAR frame. If the POLL_TIMEOUT timer expires before an expected response is received, a retry count (i.e. retry_count or data_retry_count) is incremented and a recovery action is initiated, if the retry count has not exceeded the maximum value.

A TX_BACKOFF timer is used to implement a random delay before (re)trying a transmission.

The higher layer can, optionally, specify a transmit timeout in an OMDPI transmit request. If a timeout is specified, then the MAC-D entity starts a TX_TIMEOUT timer per transmit request, when it receives the request. If the timer expires before the transmission is complete, then the transmission is aborted and a TX_TIMEOUT error is returned. The required number of TX_TIMEOUT timers is equal to MAX_TX_REQUESTS, the maximum number of outstanding transmit requests the MAC-D entity can queue.

The state machine must maintain a "current pointer" variable which points to the current frame fragment, in a sequence of frames. The current pointer is advanced if, and only if, a POLL for the next frame in the sequence is received. If more than one transition is specified when a POLL frame is received, the state of the current pointer determines which transition should be taken.

A "delay" function causes a random delay which is a function of lbt_retry_count. The delay function is used when the expected response to an RFP or ENQ frame is lost, or when the expected response to an EOD frame is lost if the EOD was sent without a preceding RFP (i.e. see state S_EOD_NR).

Receive frame events which are not specified should be ignored.

Transmit entry states.

state	event	action	next state
(LISTEN)	a PDU is passed to the MAC-D layer for transmission	reset abort_count; reset lbt_retry_count; execute channel access alogirthm	TX_PEND
TX_PEND	channel free; the length is not less than the RFP threshold or the SEQ state for the destination is unknown; set MAC-D state to TRANSMIT	set MAC-D state to TRANSMIT	TX_IDLE_R
TX_PEND	channel free; the length is less than the RFP threshold and the SEQ state for the destination is known	set MAC-D state to TRANSMIT	TX_IDLE_NR
TX_PEND	channel busy	increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	TX_PEND

State table if an RFP is required.

state	event	action	next state
TX_IDLE_R	(an RFP is required)	reset retry_count; reset data_retry_count; set VS number to 0; delete transmit SEQ table entry for sink; send RFP	S_RFP
RDY_RFP	channel acquired	send RFP frame; start POLL_TIMEOUT timer	S_RFP
	channel busy	increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_RFP
S_RFP	POLL_TIMEOUT timer expires; max. RFP retries exceeded	return RFP_RETRY error	(LISTEN)
	POLL_TIMEOUT timer expires	increment retry_count; increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_RFP
	POLL(0) received	reset lbt_retry_count; send first DATA frame; start POLL_TIMEOUT timer	S_DATA
	POLL(0) received	reset lbt_retry_count; send single EOD frame; start POLL_TIMEOUT receive timer	S_EOD
S_DATA	POLL_TIMEOUT timer expires	reset retry count; execute channel access algorithm	RDY_ENQ
	POLL received	reset lbt_retry_count; increment VS, mod 2; reset data_retry_count; send next DATA frame; start POLL_TIMEOUT timer	S_DATA
	POLL received	reset lbt_retry_count; increment VS, mod 2; reset data_retry_count; send next EOD frame; start POLL_TIMEOUT timer	S_EOD
S_EOD	POLL_TIMEOUT timer expires	reset retry_count; execute channel access algorithm	RDY_ENQ_E
	CLEAR received; EOD frame accepted	return good	(LISTEN)
RDY_ENQ	Channel acquired	send ENQ; start POLL_TIMEOUT timer	S_ENQ
	Channel busy	increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_ENQ
S_ENQ	POLL_TIMEOUT timer expires; max. retry_count exceeded	return ENQ_RETRY error	(LISTEN)

	POLL_TIMEOUT timer expires	increment retry count; increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_ENQ
	REJECT received; max. abort_count not exceeded	reset lbt_retry_count; increment abort_count	TX_IDLE_R
	REJECT received; max. abort_count exceeded	return ABORT_RETRY error	(LISTEN)
	POLL for next DATA frame received (i.e. POLL was lost)	reset lbt_retry_count; reset data_retry_count; increment VS, mod 2; send next DATA frame; start POLL_TIMEOUT timer	S_DATA
	POLL for next EOD frame received (i.e. POLL was lost)	reset lbt_retry_count; reset data_retry_count; increment VS, mod 2; send next EOD frame; start POLL_TIMEOUT timer	S_EOD
	POLL for current frame received (i.e. DATA frame was lost)	reset lbt_retry_count; increment data_retry_count; sent current DATA frame; start POLL_TIMEOUT timer	S_DATA
	POLL for current frame received (i.e. DATA frame was lost); max. data_retry_count exceeded	return DATA_RETRY error	(LISTEN)
RDY_ENQ_E	Channel acquired	send ENQ; start POLL_TIMEOUT timer	S_ENQ_E
	Channel busy	increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_ENQ_E
	unicast request frame received	return CLEAR_ABORT error.	(LISTEN)
S_ENQ_E	POLL_TIMEOUT timer expires; max. retry_count exceeded.	return CLEAR_LOST link error to indicate that the MAC-D entity cannot determine if the PDU was received by the sink and that the physical connection may be lost.	(LISTEN)
	POLL_TIMEOUT timer expires	increment retry count; increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_ENQ_E
	unicast request frame received	return CLEAR_ABORT error.	(LISTEN)
	REJECT received; max. abort_count not exceeded	reset lbt_retry_count; increment abort count	TX_IDLE_R
	REJECT received; max. abort_count exceeded	return ABORT_RETRY error	(LISTEN)
	POLL received for current EOD frame (i.e. EOD was lost); max. data_retry_count exceeded	return DATA_RETRY error	(LISTEN)
	POLL received for current EOD frame (i.e. EOD was lost).	reset lbt_retry_count; increment data_retry_count; send current EOD frame; start POLL_TIMEOUT timer	S_EOD

	CLEAR received; EOD frame accepted	return good	(LISTEN)

State table if an RFP is not required.

state	event	action	next state
TX_IDLE_NR (optional)	(an RFP is not required)	reset data_retry_count; set VS to transmit SEQ table value for sink; delete transmit SEQ table entry; send OIC EOD	S_EOD_NR
S_EOD_NR	POLL_TIMEOUT timer expires	increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_ENQ_NR
	CLEAR received; EOD frame accepted	return good	(LISTEN)
	REJECT received	return INVALID_SEQ error	(LISTEN)
RDY_ENQ_NR	Channel acquired	send ENQ; start POLL_TIMEOUT timer	S_ENQ_NR
	Channel busy	increment lbt_retry_count; delay(lbt_retry_count); execute channel access algorithm	RDY_ENQ_NR
	unicast request frame received	return CLEAR_ABORT error	(LISTEN)
S_ENQ_NR	POLL_TIMEOUT timer expires; max. retry_count exceeded	return CLEAR_LOST link error to indicate that the MAC-D entity can not determine if the PDU was received by the sink and that the physical connection may be lost; delete transmit SEQ state table entry for the sink	(LISTEN)
	POLL_TIMEOUT timer expires	increment retry count; increment lbt_retry_count; delay (lbt_retry_count); execute channel access algorithm	RDY_ENQ_NR
	unicast request frame received	return CLEAR_ABORT error	(LISTEN)
	out-of-sequence CLEAR received (i.e. EOD was lost); max. data_retry_count exceeded	return DATA_RETRY error	(LISTEN)
	out-of-sequence CLEAR received (i.e. EOD was lost)	reset lbt_retry_count; increment data_retry_count; resend EOD frame	S_EOD_NR
	REJECT received (i.e. EOD was lost); max. data_retry_count exceeded	return DATA_RETRY error	(LISTEN)
	REJECT received (i.e. EOD was lost).	reset lbt_retry_count; increment data_retry_count; resend EOD frame	S_EOD_NR
	CLEAR received; EOD frame accepted	return good	(LISTEN)
	POLL received	return INVALID_SEQ error	(LISTEN)

Unicast receive state machine.

Only unicast request frames with a destination address equal to the address of the local port address are passed to the unicast receive state machine. Note that unicast frames which do not contain the destination address of the local port are passed to the channel reservation algorithm while the receive state machine is running. Multicast and broadcast frames are simply posted to the MAC-R entity, if a buffer is available, and are not passed to the receive state machine. Note that a frame is determined to be multicast if the multicast bit is set ON in either the destination or source address.

An RX_TIMEOUT timer is started when the MAC-D entity enters the unicast receive state machine. The timer is reset each time a valid request frame is received from the originator of the current unicast frame sequence. The MAC-D entity returns to the LISTEN state if the timer expires. The RX_TIMEOUT value must be greater than the maximum time an originator can take to send a series of ENQ frames to determine the status of a data fragment before aborting a unicast transmission with an ENQ_RETRY error.

Only one frame sequence may be in progress at a time. An ORIGINATOR variable contains the address of the node which originated the current frame sequence. A VR variable contains the current SEQ state of the frame sequence. The receive state machine is started when an RFP frame is received. The ORIGINATOR variable is set to the source address of the RFP frame and VR is set to 0. A sink node will transmit a POLL frame if, and only if, it is in the receive state machine and the source of the associated request frame is the ORIGINATOR node.

Each node must maintain a SEQ state table which contains an entry for each node which has recently originated a valid frame sequence. An entry in the table is discarded when the associated node originates a new frame sequence or an entry can be discarded after it has been in the table for at least an RX_TIMEOUT time period. Note that the table will never have an entry for the current ORIGINATOR node when the receive state machine is running. An entry for the current ORIGINATOR node is added to the table when a CLEAR frame is transmitted and the receive state machine is exited. The entry contains the ORIGINATOR address and the VR value. A MAC-D entity will respond to an ENQ frame with a CLEAR frame if the SEQ state table contains an entry for the source address of the ENQ frame. A MAC-D entity will respond to an ENQ frame with a REJECT frame if an entry for the source address is not in the table and the source address is not the current ORIGINATOR node (i.e. if the receive state machine is running).

If a preceding RFP is not required, the MAC-D entity can respond to an OIC EOD frame, with a CLEAR or REJECTframe, without entering the receive state machine. The receive SEQ state table entry for the originator is updated with the value in a CLEAR frame. The receive SEQ entry is deleted (and a REJECT frame is transmitted) if the EOD frame is not in sequence.

The receiver must reserve enough buffers for a maximum size sequence of frames before sending a POLL frame in response to an RFP frame. Either the entire frame sequence is received successfully or queued fragments from a paritial sequence are discarded after an RX_TIMEOUT time period.

State descriptions:

RX_IDLE - The receiver is not receiving a sequence of unicast frame fragments. The MAC-D entity is not in the RECEIVE state (i.e. it is in the LISTEN or TRANSMIT state).

RX_BUSY - The receiver has sent a POLL frame and is waiting for the next frame in a sequence. The RX_TIMEOUT timer is running and the MAC-D entity is in the RECEIVE state. The MAC-D entity returns to the LISTEN state when the receive state changes to RX_IDLE.

state	event	action	next state
RX_IDLE	RFP received; channel available	start RX_TIMEOUT timer; set ORIGINATOR variable to source address; set VR to 0; delete SEQ state table entry for the originator; send POLL(0)	RX_BUSY
	RFP received; channel reserved	(ignore)	RX_IDLE
	EOD, DATA with START bit set OFF; channel available	delete SEQ state table entry for the source address; send REJECT	RX_IDLE
	EOD, DATA with START bit set OFF; channel not available	delete SEQ state table entry for the source address	RX_IDLE
	(optional) only-in-chain (i.e. START/STOP bits set ON) EOD	set VR to EOD SEQ+1 (mod 2); send CLEAR; update receive SEQ state table entry for originator with VR; post rx PDU to MAC-R; (Note that the receive SEQ state table value is not checked before the frame is accepted.)	RX_IDLE
	ENQ received; channel available; an entry for the originator is in the SEQ state table	Send CLEAR	RX_IDLE
	ENQ received; channel available; no entry for the originator is in the SEQ state table	Send REJECT	RX_IDLE
	ENQ received; channel reserved	(ignore)	RX_IDLE
RX_BUSY	DATA received from active ORIGINATOR; SEQ equals VR	reset RX_TIMEOUT timer; queue fragment; increment VR, mod 2; send POLL (with SEQ equal to VR)	RX_BUSY
	EOD received from active ORIGINATOR; SEQ equals VR	stop RX_TIMEOUT timer; increment VR, mod 2; send CLEAR; add SEQ state table entry for originator; post rx PDU to MAC-R	RX_IDLE
	DATA or EOD received from active ORIGINATOR; SEQ is not equal to VR	(invalid) flush the current sequence	RX_IDLE
	RX_TIMEOUT timer expires	flush the current sequence	RX_IDLE

	only-in-chain EOD received from inactive node	send CLEAR with EOD SEQ+1; update receive SEQ state table entry for originator; post rx PDU to MAC-R; (Note that the receive SEQ state table value is not checked before the frame is accepted.)	RX_BUSY
	not only-in-chain EOD or DATA received from inactive node	delete SEQ state table entry for the source address	RX_BUSY
	RFP received from inactive node	flush the current sequence; reset RX_TIMEOUT timer; delete SEQ state table entry for originator; set ORIGINATOR variable to source address; set VR to 0; send POLL(0)	RX_BUSY
	ENQ received from active ORIGINATOR; channel available	reset RX_TIMEOUT timer; send POLL	RX_BUSY
	ENQ received from active ORIGINATOR; channel reserved	reset RX_TIMEOUT timer; (ignore)	RX_BUSY
	ENQ received from a node which is not the ORIGINATOR; channel available; an entry for the source is in the SEQ state table	Send CLEAR	RX_BUSY
	ENQ received from a node which is not the ORIGINATOR; channel available; no entry for the source is in the SEQ state table	Send REJECT	RX_BUSY
	ENQ received from a node which is not the ORIGINATOR; channel reserved	(ignore)	RX_BUSY

Frame SEQ States.

All unicast MAC data frames are sequenced with a 1-bit sequence number (SEQ). The sequence number is used to detect lost data frames and duplicate data frames.

The MAC-D entity in each node must maintain transmit and receive SEQ state tables for unicast messages. The receive SEQ state table contains an entry for each active MAC-D originator node. The (optional) transmit SEQ state table contains an entry for each active sink node. Each entry consists of a 1-bit SEQ state variable and a network address. Only unicast frames affect state table entries. Receive table entries must be kept for a period longer than the maximum transmit retry time for a single data frame. The

maximum retry time is equal to the maximum ENQ retry time times the maximum number of data retries (i.e. MAX_DATA_RETRY_COUNT). An entry in the transmit SEQ state table can be kept until the space is required for a new entry. If the originator does not have an entry in its transmit SEQ state table for a sink, then the originator must send an RFP frame before sending a data frame. The RFP causes both the originator and sink to, respectively, set the transmit and receive SEQ state to 0. The SEQ state tables are initialized to empty on power-up and whenever the MAC-D address is set.

Receive SEQ states.

SEQ State descriptions:

SEQ_0 - the sink expects the next DATA or EOD packet to have a SEQ number of 0.

SEQ_1 - the sink expects the next DATA or EOD packet to have a SEQ number of 1.

SEQ_UNKNOWN - the sink is not in the receive state machine and does not have an entry in its receive SEQ state table for the originator.

The MAC-D sink caches receive SEQ state variables for active originator nodes. The variable can be set to one of three states listed above. A state of SEQ_UNKNOWN applies to all nodes which do not have entries in the sink's receive SEQ state table. The sink sets the SEQ bit in a POLL or CLEAR frame to denote the next frame that the sink expects. The sink enters the receive state machine when it receives an RFP. The sink maintains a VR variable, while it is in the receive state machine, which contains the current receive SEQ state for the active ORIGINATOR. VR is initialized to 0 and is incremented, modulo 2, each time a data frame is received.

The sink responds to an ENQ as follows: a) a POLL frame, with the SEQ set to VR, is returned if the sink is in the receive state machine and the ENQ is from the active ORIGINATOR; b) a REJECT frame is returned if the ENQ is not from the active ORIGINATOR and the SEQ state is SEQ_UNKNOWN; c) a CLEAR frame is returned if the sink is not in the receive state machine and the SEQ state is SEQ_0 or SEQ_1.

If the sink exits the receive state due to a time out (i.e. a data frame was not received in response to a POLL frame within an RX_TIMEOUT time period) then the receive SEQ state for the sink node is set to SEQ_UNKNOWN. If the sink exits the receive state after sending a CLEAR frame, then the SEQ state for the sink node is equal to the value of the SEQ bit in the CLEAR frame (i.e. the VR value).

The SEQ state of the originator overrides the SEQ state of the sink. If an RFP is not required, the sink always accepts an OIC EOD frame and returns a CLEAR. For example, if a sink receives an OIC EOD, with a SEQ equal to 0, the sink will always return a CLEAR with a SEQ of 1 and set its receive SEQ state to SEQ_1 for the originator.

Transmit SEQ States.

SEQ State descriptions:

SEQ_0 - the originator sends the current data frame with a SEQ number of 0 and expects a POLL or CLEAR with a SEQ number of 1.

SEQ_1 - the originator sends the current data frame with a SEQ number of 1 and expects a POLL or CLEAR with a SEQ number of 0.

SEQ_UNKNOWN - the transmit SEQ state for the sink is unknown and the originator must send an RFP frame to establish the SEQ state.

The originator sets the SEQ bit in transmitted data frame fragments is to the current VS value for the sink. If an RFP is used, the VS value us initialized to 0 before the first data fragment is transmitted. If an RFP is not used, the originator sets VS to the value in the transmit SEQ state table entry for the respective sink node. The state variable can be in one of the three states listed above. The UNKNOWN state applies to all nodes which do not have entries in the originator's transmit SEQ state table. If the state is UNKNOWN, the originator must send an RFP, to set the SEQ state to 0, before sending a data frame. The SEQ state entry for the sink should be deleted when the originator enters the transmit state machine. The VS value for the sink is updated and a stored in a new transmit SEQ state table entry when the CLEAR is received.

A MAC-D entity does not have to maintain a transmit SEQ state table if an RFP is always used. In this case, the transmit SEQ state is always UNKNOWN.

MAC-D Error Codes.

Transmit Request Codes.

ERR_TX_GOOD	0x00	
Interface errors:		
ERR_TX_DISABLED	0x01	- the MAC-D entity is disabled.
ERR_TX_BUSY	0x02	- the maximum number of outstanding transmit requests has exceeded.
ERR_TX_MAX_LENGTH	0x03	- the lenght is greater than the maximum MAC-R PDU size.
Protocol errors:		
ERR_TX_POLL_SEQ	0x11	- the transmission was aborted because a POLL or CLEAR was receive with an invalid sequence.
Link errors:		
ERR_TX_RFP_RETRY	0x21	- the maximum number of RFP retries was exceeded.
ERR_TX_ENQ_RETRY	0x22	- the maximum number of ENQ retries was exceeded.
ERR_TX_DATA_RETRY	0x23	- the maximum number of data retries was exceeded.
ERR_TX_CLEAR_LOST	0x24	- a CLEAR was not received from the sink (i.e. and the maximum number of ENQ retries was exceeded).
Channel access errors:		
ERR_TX_ABORT_RETRY	0x41	- the transmission was aborted and restarted the maximum number of times.
ERR_TX_CHAN_ACCESS	0x42	- the maximum number of channel access retry errors was exceeded.
ERR_TX_CLEAR_ABORT	0x43	- a CLEAR was not received from the sink and the transmission was aborted by a request frame from a third node.

MAC-D constants/variables.

MAX_TX_REQUESTS - 1 to 3, the maximum number of outstanding MAC-R transmit requests that the MAC-D entity can queue.

MAX_FRAG_SIZE = 250 bytes, is the maximum size of a unicast frame fragment.

MAX_SDU_SIZE = 1600 bytes, is the maximum size of a MAC-R PDU.

MAX_PDU_SIZE = 1608 bytes, is the maximum size of a MAC-D PDU.

MAX_MULTI_SIZE = 1600 bytes, is the maximum size of a multicast frame.

DEF_RFP_THRESHOLD - default RFP threshold. Unicast transmit request which exceed the RFP threshold must be transmitted with a preceding RFP.

LBT_SLOT_SIZE is the LBT slot time and is dependent on the bit rate and radio characteristics.

INTERFRAME_GAP is the maximum idle time allowed between frames in a CA sequence and is dependent on the radio type.

BUSY_PULSE_TIME is the maximum interpoll gap time and is a function of the bit rate.

POLL_TIMEOUT is the time an originator waits for an expected poll frame from a sink. The value is equal to the INTERFRAME_GAP time, plus the worst-case time required to send a poll frame.

RX_TIMEOUT is the maximum time that a sink will remain in the unicast receive state machine without receiving a valid request frame in the current frame sequence.

The following values may be different for terminal and AP implementations:

AP constants:

CCD_IDLE_TIME_HIGH = 0, is the CCD idle time for the high(1) priority data.

CCD_IDLE_TIME_NORMAL is the CCD idle time for the normal(0) priority data.

MAX_RFP_RETRY_AP = 20, is the maximum number of times that an originator will send an RFP frame to an AP sink, without receiving a POLL frame, before aborting a frame sequence.

MAX_RFP_RETRY_TERM = 7, is the maximum number of times that an originator will send an RFP frame to a terminal sink, without receiving a POLL frame, before aborting a frame sequence.

MAX_ENQ_RETRY_AP = 10, is the maximum number of times that an originator will send an ENQ frame to an AP sink, without receiving a POLL or CLEAR frame, before aborting a frame sequence.

MAX_ENQ_RETRY_TERM = 7, is the maximum number of times that an originator will send an ENQ frame to a terminal sink, without receiving a POLL or CLEAR frame, before aborting a frame sequence.

MAX_DATA_RETRY_AP = 10, is the maximum number of times that an originator will retransmit a single DATA or EOD frame in a single CA sequence, when the sink is an AP.

MAX_DATA_RETRY_TERM = 7, is the maximum number of times that an originator will retransmit a single DATA or EOD frame in a single CA sequence, when the sink is a terminal.

MAX_ABORT_RETRY = 5, is the maximum number of times that an originator will restart a frame sequence that has aborted by the sink (i.e. with a REJECT).

MAX_LBT_RETRY = 30, is the maximum number of consecutive times that an originator will delay due to a busy or reserved channel before aborting a frame sequence.

TX_BACKOFF_TABLE[lbt_retry_count] = an array of maximum delay values, indexed by the LBT retry count. The actual delay is a random number between 0 and the maximum delay. "lbt_retries" is incremented whenever the expected response to an RFP or ENQ is not received or whenever the channel is sensed busy or reserved before initiating a transmission. "lbt_retries" is initialize to 0 when a transmit request is received and whenever a poll response (i.e. POLL, CLEAR, REJECT) is received from the active sink. Entries in an AP backoff table are lower than the corresponding entries in a terminal backoff table, to prioritize channel access for an AP.

Terminal constants:

CCD_IDLE_TIME_HIGH

CCD_IDLE_TIME_NORMAL

MAX_RFP_RETRY = 10

MAX_ENQ_RETRY_AP = 7

MAX_DATA_RETRY = 7

MAX_ABORT_RETRY = 7

MAX_LBT_RETRY = 30

TX_BACKOFF_TABLE[lbt_retry_count]

Master-slave Extensions.

Contention free periods.

A master-slave extension of the MAC-D protocol allows an AP to, optionally, function as a master for contention-free channel access, in environments with a single AP per coverage area. A master-slave set (MSS) consists of an AP "master" and "slave" nodes, where a slave node is any terminal node attached to the AP. If master-slave mode is enabled then a contention-free period (CFP) is initiated, per MSS, each time a scheduled MAC-R HELLO frame is transmitted by the AP master. The maximum duration of a CFP is less than, or equal to, the inter-HELLO period, and consists of an a) outbound phase, followed by b) a inbound phase. Multicast messages and pending messages are delivered during the outbound phase. Slave nodes are "polled" by the master, during the inbound phase. A CFP is immediately followed by a peer-to-peer random channel access period or the start of the next CFP.

OWL MAC-D provider interface (OMDPI) extensions.

The MAC-R entity in an AP sets the contention-free start (CF-START) flag ON in an OMDPI transmit request, for a HELLO PDU, to indicate the start of the current CFP to the MAC-D entity. The AP uses "chaining" to associate outbound transmissions with the outbound phase of the CFP. (In random-access mode, the MAC-R entity in an AP "chains" transmit requests to reduce channel contention for outbound

transmissions associated with scheduled HELLO packets.) The MAC-R entity sets the CHAIN flag ON, in each OMDPI transmit request, except the last one, which is associated with the transmission of a scheduled HELLO response packet. The MAC-D interface in the AP should allow at least two outstanding transmit requests, to minimize the latency between chained transmissions. The CHAIN flag is set OFF in the last transmit request in a group of chained transmit requests. The MAC-D entity independently initiates the "inbound phase" of the CFP immediately follows the transmission of the last frame in the chained group. (Note that the HELLO frame is the last frame if no multicast or pending messages are associated with the HELLO frame.) The inbound phase ends when the CFP ends. The protocol used during the inbound phase is described in detail below.

Contention-free channel access rules are simple. The master owns the channel for the duration of the CFP and initiates all transmissions without sensing the channel (i.e. BCD and the channel reservation are ignored). A child node may not initiate a transmission during a CFP. A child node responds to the master without sensing the channel, during a CFP.

The master starts a CFP TIMEOUT timer at the start of each CFP. The CFP ends after the CFP_TIMEOUT timer expires, as soon as any active frame sequence is completed. The master broadcasts a CFP_END frame, at the end of the CFP, to mark the end of the CFP and the start of a random access period.

A CFP is established in a child node in two ways. The MODE bit, in the MAC-D header is always set to 1 in any frame transmitted by the master (or a slave) during a CFP. A child node enters a CFP whenever it is in random-access mode and receives a frame, which originated in its MSS, with the MODE bit set to 1.

A child node also maintains a CFP_START timer, and enters a CFP whenever the timer expires. The timer is synchronized to received HELLO frames, and is set to expire at least 1 LBT slot time before the next scheduled HELLO frame. CFP_START timer operation is described in detail below.

The current CFP for a slave node ends a) if it receives a frame from its parent with the MODE bit set to 0, b) at the start of the next CFP, or c) if the node detects an idle channel for CFP_TIMEOUT milliseconds, where CFP_TIMEOUT is slightly greater than twice the BUSY_PULSE_TIME (e.g. 350 milliseconds for UHF). The MAC-D entity in a child node examines the MODE bit in all received frames which originate within the MSS, even if the destination address is for another node. A CFP_TIMEOUT timer is started at the beginning of a CFP and restarted whenever a frame is received with the MODE bit set ON.

A random-access period follows the end of a CFP, if the CFP ends before the start of the next CFP. The MODE bit in the MAC-D header is set to 0 for any frames transmitted during a random-access period. A child node enters random-access mode as soon as it receives a frame from its parent AP with the MODE bit set to 0. In the absence of other traffic, the AP will send a CFP_END frame, which is a CLEAR frame, with a) a channel reservation of 0, b) a MAC-D broadcast destination address, and c) a MODE of 0, to mark the end of the master-slave period and the start of the random access period. A child node can also enter random access mode if it does not receive a frame from its parent AP within CFP_TIMEOUT milliseconds.

A sleeping terminal can wake up and transmit (i.e. in random-access mode) within BUSY_PULSE_TIME milliseconds, in the absence of other traffic.

A child may initiate a random-access transmission at any time during a random-access period, even if the transmission will not complete before the start of the next CFP. Note that the AP sets the offset field in a scheduled HELLO packet to indicate how much the HELLO packet was delayed. If a child node is unable to transmit a queued inbound frame to its parent during the random-access period, it must wait until the next inbound phase.

CFP_START timer operation.

A logicial CFP_START timer, maintained by the MAC-R entity in a slave node, is set to expire at least one LBT slot time before the next scheduled HELLO period. Whenever the timer expires, the MAC-R entity calls MACD_cfp_indicate(), to indicate the start of a new CFP to the MAC-D entity, and immediately restarts the timer with the calculated interval for the next HELLO period. The timer is restarted whenever a HELLO packet is received (i.e. with an adjusted interval) to maintain sychronization with the parent AP. If the timer expires and MAX_HELLO_LOST consecutive HELLO packets have been missed from the parent AP, then the MACD_cfp_indicate() is not called and the timer is not automatically restarted (i.e. the node goes into an unattached state).

The MAC-D entity (re)starts the CFP_TIMEOUT timer whenever MACD_cfp_indicate() is called and whenever it receives a frame with the master/slave mode bit set ON. Note that the MAC-D entity will exit the current CFP when the CFP_TIMEOUT timer expires or when it receives a frame with the mode bit set OFF.

Contention-free polling protocol.

The master-slave MAC-D protocol can provide contention-free polling for up to 200 terminal nodes. (Wireless APs are not supported if contention-free polling is used.) After all multicast and pending frames have been transmitted in the current CFP, the AP transmits a MAC-D master slot poll (MSP) frame to start a master-slave "inbound phase". The time immediately after an MSP transmission is divided into discrete response time slots, where each slot is slightly larger than the time required to transmit an MSP-ACK frame (e.g. 24 milliseconds for UHF). A child node, which has inbound data to send, responds to an MSP frame, with an MSP-ACK frame, in its designated response slot. (OWL MAC-D frame formats are defined in "owlfram.doc".)

An MSP contains a "slot count" field which contains the number of responses slots. A "flags" field in the the MSP contains response mode bits which define the response mode. a) If the response mode is RANDOM, then a slave randomly chooses one of the "slot count" response slots. b) If the response mode is POLL_LIST, then the MSP contains a list of addresses and the "slot count" field contains the number of addresses in the list. Each address corresponds to a response slot. A slave may respond in a slot if its address matches an address in the list. The relative location of the slave's address in the list defines the relative position of its response slot. c) If the response mode is ADAPTIVE, then a slave determines its response slot offset by processing a set of "expansion flags" contained in the MSP.

The MSP "expansion flags" field contains 4 expansion factor bits and a set of bit flags organized into a tree with up to 4 levels, level 0 to level 3. Expansion flag bits and bytes , in an MSP PDU, are numbered from 0 to N in network order (e.g. left to right).

The first bit (i.e. bit 0) defines the expansion factor for level 0. If it is set ON, then level 0 slots are expanded by 4; otherwise, level 0 slots are expanded by 2. Likewise, bits 1 through 3 define the expansion factors for levels 1 through 3, respectively. (Currently, level 3 cannot be expanded.)

The low-order 4 bits of the first expansion flags byte (i.e. bits 4-7) represent four base slots at level 0, numbered from 0 to 3. If each of the 4 bit flags is 0, then the MSP is followed by four response slots. If a level 0 bit flag is set ON, then the base slot is expanded and the sub tree rooted at the bit defines the number of expansion slots derived from the base slot. In general, if a bit is set ON at level j, then the sub tree rooted at the bit defines the number of derived expansion slots. The bits at level 3 are always 0 (and therefore are not included in the MSP). Figure 1 shows the tree for a set of expansion flags with a value of hex. 0D 83 80 04. Note that the expansion factor bits are all 0, therefore the expansion factor for each level is 2.

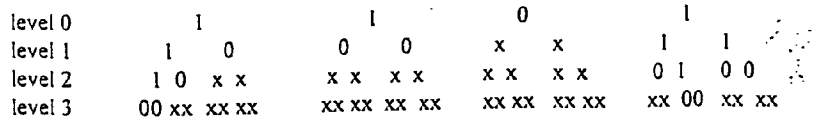


Fig. 1

The "x" bits in the example expansion tree, in figure 1, represent "don't care" bits, and are set to 0. The bits at level 3 are implicitly set to 0. The expansion tree allocates 12 response slots (i.e. 1 for each 0 bit), with 4, 2, 1, and 5 slots allocated, respectively, per sub tree.

The base slot for a slave node is its address, modulo 4, where 0 is the first slot. For example, a slave node with an address of hex. 4007 is assigned to base slot 3. If the expansion flag for the base slot is set ON, the slave node continues to recursively derive a smaller sub tree, rooted at the next level, until the root bit of the derived sub tree is 0. At each level, the node shifts out the previous significant bits of its address and takes its address, modulo the expansion factor, to determine the root bit of the next sub tree. The node responds in the slot which follows any slots allocated by all previous sub trees. For example, in figure 1, a slave node with address hex. 4007 responds in the ninth slot.

A slave node calls **node_slot_offset(nsp, address)**, where "nsp" points to an MSP PDU and "address" is the 16-bit slave address to determine its relative slot number. The function returns a negative value if the node cannot respond in a slot; otherwise, it returns the slot number for the slave node, where 0 is the first response slot. Example code for node_slot_offset() is shown in appendix 3.

The polling algorithm assumes that the AP can detect a collision, if more than 1 node responds in the same slot. If a collision occurs in a slot at level j, then that slot is expanded, by the expansion factor for level j, to level j+1 . If no collisions occur in a set of expanded slots at level j+1, then the slots are compressed to level j.

The total number of response slots can be large, in a heavily loaded network. If the expansion tree defines a large number of slots, then the master limits the number of "active" response slots to a "window" of slots within the total slots. The MSP "slot offset" field contains the offset of the first active slot in the window, and the "slot count" field contains the number of slots in the window. The active slot count is always less than or equal to MAX_RSP_SLOTS (i.e. 8). The master resends the same expansion flags until all of the response slots have been active.

The AP master polls all nodes which successfully respond to an MSP. The master adds the source address, in any received MSP-ACK, to an internal polling list. After the MSP response period ends, the AP sends a "master POLL" frame (i.e. a POLL frame with the MODE bit set to 1) to each node in its list, in priority order. The POLL frame initiates a unicast OWL MAC-D frame sequence. The reservation in the master POLL frame is set to the reservation request value in the corresponding MSP-ACK frame. If a node has inbound data queued it responds to a master POLL frame with a DATA or EOD frame; otherwise, it responds with an ABORT frame. An ABORT causes the node to be deleted from the AP polling list. If a slave responds to a POLL with a DATA frame, then the master immediately sends another POLL frame, to the slave, for the next data fragment. Ifa slave responds with an EOD frame, then the master sends a CLEAR frame to the slave. A slave node is deleted from the polling list whenever the master successfully receives a frame from the slave node (even if the frame is received in random-access mode). If the terminal has more than 1 frame to send, it must wait for the next MSP cycle, or for a random-access period.

An MSP-ACK frame has a "wait time" field which is set to the time that the associated transmit request has been queued, in milliseconds, just before the MSP_ACK is transmitted. The master subtracts the "wait time" value from the current time and enters the difference in the polling list entry for the slave node. Polling list entries are sorted so that the lowest entries are polled first.

After polling each node in its list, the AP sends the next MSP. The inbound phase ends when no MSP-ACKs are received and no collisions are detected in any response slots. A polling cycle is guaranteed to terminate because slots with collisions are not expanded at the highest MSP level. (In the worst case, a slot is allocated per terminal because the maximum number of slots, 256, exceeds the maximum number of terminals.)

In the absence of other outbound traffic, the AP sends an CFP_END frame to mark the end of the CFP (and the inbound phase). Note that it is possible that the inbound phase may not end before the start of the next CFP. In this case, the AP saves its polling list, which contains an entry for each node which successfully responded to an MSP but did not get polled. The AP first polls those nodes in its list before transmitting the first MSP in the next inbound phase.

Initially, a device does not have a unique 2-byte address. A slave node, without a 2-byte address, cannot respond in an MSP response slot. Instead, it must contend for the channel in a random-access period. The master must guarantee periodic random-access on a heavily loaded channel. A master can guarantee a random-access period, for example, by limiting the duration of each CFP, to guarantee a random-access period between successive CFPs, or by skipping every Nth CFP.

Slave state machines.

Slave transmit state machine.

The state transition table below specifies the slave MAC-D protocol operation for the transmission of unicast inbound data in master-slave mode. The slave discards frames which are not from the master therefore only frames from the master are passed to the slave transmit state machine.

Note that the master is responsible for error recovery. If a POLL, DATA, or EOD frame fragment is lost, the master will resend a POLL frame to solicit the (re)transmission of a data fragment. The master checks BCD one CA slot time after sending a POLL, and (re)transmits the next POLL (i.e. to the same node or another node) immediately if BCD is not asserted. The master waits for, at most, BUSY_PULSE_TIME milliseconds, if a response is not received, before sending the next POLL frame. The master limits the number of POLL retries for a single node to MS_POLL_RETRIES (i.e. 3).

In the table below, "other frame" is used to denote any frame other than previously listed frame types for the state. It is assumed that POLL and CLEAR frames are addressed to the station (i.e. POLL or CLEAR frames addressed to another station fall into the category of "other frames").

Note that the current CFP ends if the CFP_TIMEOUT timer expires or if a frame is received from a node in the MSS with the mode bit set to 0. The CFP_TIMEOUT timer is restarted each time a frame is received with the MODE bit set to 1.

State	Event	Action	Next State
(LISTEN)	a PDU is passed to the MAC-D layer for transmission	reset abort_count; set VS number to 0; delete transmit SEQ table entry for sink	MS_TX_PEND
	POLL received	send ABORT	(LISTEN)
MS_TX_PEND	MSP received	calculate slot offset; send MSP_ACK with SEQ equal to VS	MS_POLLING
	CFP ends/random-access period begins		TX_PEND
	POLL received	send ABORT	MS_TX_PEND

MS_POLLING	POLL received; data frame is more than 1 fragment	send first DATA frame fragment	MS_S_DATA
	POLL received; data frame is 1 fragment	send single-fragment EOD frame	MS_S_EOD
	MSP received	calculate slot offset; send MSP_ACK with SEQ equal to VS	MS_POLLING
	CFP ends/random-access period begins		RDY_RFP
MS_S_DATA	POLL received for current DATA fragment	retransmit the current DATA fragment	MS_S_DATA
	POLL received for next DATA fragment	increment VS, mod 2; send next DATA fragment; start MS_POLL_TIMEOUT timer	MS_S_DATA
	POLL received for next EOD fragment	increment VS, mod 2; send last EOD fragment; start MS_POLL_TIMEOUT timer	MS_S_EOD
	MSP received; max. abort_count exceeded	return ABORT_RETRY error	(LISTEN)
	MSP received; max abort_count not exceeded	increment abort count; calculate slot offset; set VS to 0; send MSP_ACK with SEQ equal to VS	MS_POLLING
	other frame received with MODE=1; max. abort_count exceeded	return ABORT_RETRY error	(LISTEN)
	other frame received with MODE=1; max. abort_count not exceeded	Set VS number to 0; increment abort count	MS_TX_PEND
	CFP ends; max. abort_count not exceeded	Set VS number to 0; increment abort_count	TX_PEND
	CFP ends; max. abort_count exceeded	return ABORT_RETRY error	(LISTEN)
MS_S_EOD	CLEAR received; EOD frame accepted	return good	(LISTEN)
	other frame received with MODE=1	return CLEAR_ABORT error	(LISTEN)
	CFP ends	return CLEAR_ABORT error	(LISTEN

Slave receive state machine.

The receive state machine for unicast frame sequences in slave mode is identical to the receive state machine for random-access mode, except that the channel is never busy (i.e. BCD and the channel reservation are ignored).

Master state machines.

Master mode timers.

An MM_RESPONSE timer is started when the master sends a POLL frame and expects to receive a data fragment in response. The timer is started, initially, with a short timeout equal to the CA slot time. If a frame reception is in progress when the timer expires (i.e. if BCD is asserted), the timer is restarted to wait for the end of the transmission; otherwise, the next POLL frame is immediately transmitted.

An MSP_SLOTS timer is started immediately after the transmission of an MSP frame. The interval is equal to the number of MSP response slots times the duration of one slot.

A CFP_TIMEOUT timer can be used to limit the duration of a CFP.

Master transmit state machine

The transmit state machine for unicast frame sequences in master mode is identical to the transmit state machine for random access mode, except that the channel is never busy and a random delay is not used between RFP and ENQ retries.

Master receive state machine.

The state transistion table below describes the operation of the master, in master-slave mode, for receiving inbound unicast frames. The master enters the MM_OUTBOUND state at the start of each CFP and transitions to one of the master-slave master polling (MMP) states at the start of the inbound phase.

The master must maintain a poll list (described above), which contains an entry for slave nodes which have successfully responded to an MSP frame with an MSP_ACK frame. Entries in the list are aged and discarded after MSP_ACK_TIMEOUT seconds. An entry is deleted when the master sends the first POLL to the associated node. An entry is also deleted if a frame is received from the associated node in random-access mode.

There is an immediate transition in the MMP_NEXT state, depending on the state of the poll list and two variables, CFP_start_pending and CFP_end_pending. If CFP_end_pending is TRUE, then the current CFP ends as soon the MAC-D entity completes any active unicast frame sequence. If CPF_start_pending is TRUE, then the next CFP is started as soon as the MAC-D entity completes any active unicast frame sequence. CFP_end_pending is set to TRUE if a CFP is active and either the CFP_TIMEOUT timer expires or the MAC-R entity initiates the next CFP. CFP_start_pending is set to TRUE when the MAC-R entity initiates the next CFP.

If the master MAC-D entity receives a transmit request while in the MMP_S_POLL state, the master can start the outbound transmission as soon as polling is complete for the current node, and then return to the MMP_S_POLL state when the outbound transmission is complete. The transmission is executed according to the unicast transmission state table, shown above, except that the MODE bit is set ON in each frame in the sequence.

State	Event	Action	Next State
MM_OUTBOUND	transmission of the last outbound CF frame completes; polling list is empty	reset collision_count; set poll_list to empty; send initial MSP; start MSP_SLOT(slots) timer	MMP_S_MSP
	transmission of the last outbound CF frame completes; polling_list is not empty	set VR to 0; reset retry_cnt; send master POLL to the first node in the polling list; start MM_RESPONSE	MMP_S_POLL

MMP_S_MSP	MSP_SLOT timer expires; collision_count=0; polling_list is empty	send CFP_END CLEAR frame	(LISTEN) random-access mode
	MSP_SLOT timer expires; collision_count > 0; polling list is empty	adjust slots; send next MSP; start MSP_SLOT(slots) timer	MMP_S_MSP
	MSP_SLOT timer expires; polling_list is not empty	sort poll list by priority; reset retry count; send master POLL to the first node in the polling list	MMP_S_POLL
	MSP_SLOT timer expires; next CFP starts	save polling list	MM_OUTBOUND
MMP_NEXT	CFP_end_pending is TRUE; CPF_start_pending is FALSE	save polling list; send CFP_END frame	(LISTEN) random-access mode
	CFP_end_pending is TRUE; CPF_start_pending is TRUE	save polling list	MM_OUTBOUND
	CFP_end_pending is FALSE; poll list is not empty	set VR to 0; send POLL to the next node in the poll list	MMP_S_POLL
	CFP_end_pending is FALSE; poll list is empty; collision_count > 0	adjust slots; send next MSP; start MSP_SLOT(slots) timer	
	CFP_start_pending is FALSE; poll list is empty; collision_count = 0	send CFP_END (CLEAR) frame	(LISTEN) random-access mode
MMP_S_POLL	receive DATA frame from the polled node	increment VR; reset retry_cnt; send next master POLL to the current node	MMP_S_POLL
	receive EOD frame from the polled node	send CLEAR; save polling list	MMP_NEXT
	receive ABORT frame from the polled node		MMP_NEXT
	MM_RESPONSE timer expires; max. retry count exceeded		MMP_NEXT
	MM_RESPONSE expires; max. retry count not exceeded	increment retry count; resend POLL	MMP_S_POLL
	CFP_TIMEOUT timer expires	set CFP_end_pending flag	MMP_S_POLL
	next CFP start indication is received from the MAC-R layer	set CFP_end_pending flag; set CPF_start_pending flag	MMP_S_POLL

Appendix 1 - Receive frame coding example.

The macd_phy_rx_done "C" routine, shown below, is used to process all received frame events. (Note that the physical layer discards received frames with errors and removes physical layer framing bytes.) The code does not include unicast transmit, receive state machine logic, or master-slave extentsions. A frame is passed to the unicast receive state machine with rx_idle or rx_busy. A frame is passed to the unicast transmit state machine by calling the state function pointed at by cb->tx_state.state.

```
#define MACD_BROADCAST_ADDRESS 0xffff
#define MACD_BROADCAST_LAN_ID   0xff

typedef struct {
  uchar_t protocol_id;
  uchar_t lan_id;
  uchar_t dst[2];
  uchar_t src[2];
  uchar_t ctl;
  uchar_t len;
} macd_header_t;

void macd_phy_rx_done(macd_control_t *cb, o_event_t *evt)
{
  macd_header_t *hdr;
  unsigned len;
  unsigned short dst, src, node_type, port_id;

  /* set the MAC-D PDU (i.e. hdr) pointer and length variables */
  hdr=(macd_header_t *) evt->buffer;
  len=evt->length;
  src=NTOI(hdr->src);  /* convert the network format to a 2-byte unsigned integer */
  dst=NTOI(hdr->dst);  /* convert the network format to a 2-byte unsigned integer */

  /* if the frame size is invalid - discard */
  if (len < MACD_HEADER_SIZE || len > cb->max_rx_size)
  {
    ++cb->mib.invalid_frame_count;
    (void)o_evt_free_chain(evt); /* discard the event */
  }
  /* if the protocol ID does not match - discard*/
  else if (hdr->protocol_id != MACD_PROTOCOL_ID)
  {
    ++cb->mib.invalid_protocol_id_count;
    (void)o_evt_free_chain(evt);
  }
  /* if the LAN ID does not match */
  else if (hdr->lan_id != cb->lan_id)
  {
    ++cb->mib.invalid_lan_id_count;
    if (MACD_IS_UNICAST(hdr))
    {
      lbt_set_reservation(hdr);
      (void)o_evt_free_chain(evt);
```

```
  }
  /* broadcast packets are posted to MAC-R if the LAN ID is
    0xff or if the masked LAN IDs match */
  else if (dst==MACD _BROADCAST_ADDRESS)
  {
    if (hdr->lan_id==MACD_BROADCAST_LAN_ID ||
      (hdr->lan_id & cb->lan_id_mask==cb->lan_id & cb->lan_id_mask))
    {
      /* post the buffer to the MAC-R layer. */
      MACR_receive_data_indication(cb, evt);
    }
    else
      (void)o_evt_free_chain(evt);
  }
  else /* multicast - but not broadcast */
    (void)o_evt_free_chain(evt);
}
/* else, if it is multicast */
else if (dst & 0x8000 || src & 0x8000)
{
  node_type=MACD_NODE_TYPE(dst);
  port_id=MACD_PORT_ID(dst);

  /* Is it addressed to an acceptable multicast address (i.e. for an AP)? */
  if ((node_type==O_NODE_ANY || node_type==O_NODE_AP) &&
      (port_id==O_PORT_ID_ANY ||
        port_id==MACD_PORT_ID(cb->address)))
  {
    /* post the buffer to the MAC-R layer. */
    MACR_receive_data_indication(cb, evt);
  }
  else
    (void)o_evt_free_chain(evt);
}
/* else, if it is unicast and addressed to me */
else if (dst==cb->address)
{
  if (MACD_IS_REQUEST_FRAME(hdr))
  {
    /* cancel the channel reservation if it is reserved by the
      source node */
    lbt_check_reservation(cb->lan_id,src);

    switch(cb->state)
    {
    case LISTEN:
      rx_idle(cb,evt);    /* rx_idle may change the high-level state to RECEIVE */
      break;

    case RECEIVE:
      rx_busy(cb, evt);
      break;

    case TRANSMIT:
```

```
        if (cb->tx_state.state==RDY_ENQ_E || cb->tx_state.state==S_ENQ_E)
          post_tx_done(ERROR_CLEAR_LOST);
        else
          transmit_abort(cb);  /* abort and retry later */
        cb->state=LISTEN;
        rx_idle(cb,evt);
        break;

      default:
        (void)o_evt_free_chain(evt);
        break;
      }

    }
    else /* poll frame */
    {
      switch (cb->state)
      {
      case TRANSMIT:
        if (src==cb->tx_state.dst)
          cb->tx_state.state(cb,evt);  /* pass the frame to the unicast transmit state machine */
        else
          (void)o_evt_free_chain(evt);
        break;

      case RECEIVE:
        (void)o_evt_free_chain(evt);
        break;

      default:
        (void)o_evt_free_chain(evt);
        break;
      }
    }
  }
  /* else, if it is unicast and not addressed to me */
  else
  {
    lbt_set_reservation(hdr);
    (void)o_evt_free_chain(evt);
  }
}
```

Appendix 2 - management variables.

MAC-D statistics.

The Direct Sequence/UHF MAC-D statistics are specific to the Direct Sequence/UHF radio cards. Media-independent statistics are kept in the MIB II interface group and Norand interface group on the AP. Note that frames which do not pass a CRC check are discarded and do not affect any MAC-D statistics.

Statistics which can be kept at the MAC-R layer:

MAC-D service time array - an array of counters which can be used to derive a histogram of MAC-D transmit request service times.

transmit request count - the total number of MAC-R transmit requests

transmit length errors - the total number of transmit requests which are rejected because the the maximum length was exceeded.

transmit request errors - the total number of rejected transmit requests.

access errors - the number of times that a transmit request failed because the channel was busy or reserved for the maximum number of retries.

RFP retry errors - the number of times that a transmit failed because the RFP retry count was exceeded.

ENQ retry errors - the number of times that a transmit failed because a POLL was lost (i.e. after max. ENQ retries).

lost CLEAR errors - the number of times that the status of a transmit request is unknown because the CLEAR was lost (i.e. after max. ENQ retries).

transmit timeout errors - the number of times that a transmit request failed because the time limit expired.

transmit abort errors - the number of times that a transmit request failed because the transmission was aborted (i.e. and restarted) too many times.

data retry errors - the number of times that a transmit request failed because a data fragment was retried the max. number of times.

tx protocol aborts - the number of times that a transmit request failed due to a protocol error.

Statistics which must be kept on the radio card:

tx frames - total number of transmitted frames.

tx data frames - total number of transmitted unicast DATA or EOD frames.

rx frames - total number of received frames.

rx data frames - total number of received unicast DATA or EOD frames.

rx overruns - the number of (i.e. data) frames discarded due to a lack of receive buffers.

channel access count - the number of times that the channel access algorithm was executed (i.e. before initiating a transmission).

channel busy count - the number of times that the channel was busy because RX detect was asserted.

channel reserved count - the number of times that the channel was reserved.

transmit abort count - the number of times that a transmission in progress was aborted.

invalid POLL count - the number of times that an invalid POLL or CLEAR frame was received, in response to a DATA or EOD frame.

receive timeout errors - the number of times that the receive state machine timed out (i.e. during the reception of a unicast packet).

invalid lan id count - the number of frames received with the wrong LAN ID.

rx access count - the number of times that the channel access algorithm was executed before responding to an RFP or ENQ frame.

rx busy count - the number of times that an RFP or ENQ was ignored because RX detect was asserted.

rx reserved count - the number of times that an RFP or ENQ was ignored because the channel was reserved.

rx restart count - the number of times that an RFP was received from the originator of the active conversation in the receive state (i.e. a DATA frame was followed by an RFP frame).

rx abort count - the number of times that an active conversation was aborted because an RFP frame was received from a node other than the originator.

reservation errors - the number of times that the originator or sink, of an active conversation, received a frame from some other node while the channel was reserved.

short frame count - the number of frames received which are too short.

long data frame count - the number of data frames received whichare too long.

protocol ID errors - the number of frames received which have an invalid protocol ID.

invalid frame count - the number of frames received with an invalid MAC-D header.

tx sequence errors - the number of POLL or CLEAR frames received with an invalid sequence number.

rx sequence errors - the number of out-of-sequence DATA or EOD frames received.

ENQ reject count - the number of times that a REJECT was sent in response to an ENQ frame.

DATA reject count - the number of times that a DATA or EOD frame was ignored because the reciever was not in the receive state.

MAC-D initialization variables.

- mode.
- channel.
- CSMA enable/disable.

Appendix 3 - master-slave coding examples.

The following code defines algorithms for calculating the total number of slots in an MSP response period and for calculating the response slot offset for a terminal.

```
#include <stdio.h>
#include <string.h>

typedef unsigned char uchar_t;
typedef unsigned short ushort_t;

/* 4 slots at level 0 */
#define MSP_L0_MASK  3
#define MSP_L0_SLOTS  4

/* default expansion factor for each level */
#define DEF_EXP_FACTOR 4

#define MSP_MAX_LEVEL 3
#define MSP_MAX_RSP_SLOTS 16

#define MSP_FLAG_RANDOM   0x01
#define MSP_FLAG_POLL_LIST 0x02

#define MSP_EXP_FACTOR(msp,lvl) \
((msp->exp_slots[0] & (0x8 >> lvl)) ? 4 : 2)

#define MSP_EXP_MASK(msp,lvl) \
((msp->exp_slots[0] & (0x8 >> lvl)) ? 3 : 1)

#define MSP_TEST_FLAG(flags,bit) (flags[bit>>3] & (0x80 >> (bit & 7)))
#define MSP_SET_FLAG(flags,bit) (flags[bit>>3] |= (0x80 >> (bit & 7)))

/*****************************************************************************
 * msp_expand_slot - recursively expands a branch of MSP expansion flags
 *****************************************************************************/
static unsigned msp_expand_slot(uchar_t *flags,
                    unsigned flag_cnt,
                    unsigned start_bit,
                    unsigned count,
                    unsigned *exp_factor,  /* 2 or 4 per level */
                    unsigned level,
                    slot_list_t *slp)
{
  unsigned slots=0;
  uchar_t *next_flags;
  unsigned next_flag_cnt;
  unsigned next_start_bit;
  unsigned index;
  unsigned i, test;

  next_flags = flags + (flag_cnt >> 3);
  next_flag_cnt = flag_cnt * exp_factor[level];  /* 2X or 4X */
```

```
next_start_bit = start_bit * exp_factor[level];

index = start_bit >> 3;        /* convert bit index to byte index */
test = 0x80 >> (start_bit & 7); /* point to the bit in the indexed byte */

for (i=0; i<count; i++)
{
  if (test & flags[index])
  {
    if (level==MSP_MAX_LEVEL-1)
    {
      slots+=exp_factor[MSP_MAX_LEVEL-1];
      if (slp)
        sl_add_slots(slp, exp_factor[MSP_MAX_LEVEL-1],
                (uchar_t)MSP_MAX_LEVEL,
                (uchar_t)next_start_bit);
    }
    else
      slots+=msp_expand_slot(next_flags,next_flag_cnt,
                    next_start_bit,exp_factor[level],
                    exp_factor,level+1,slp);
  }
  else
  {
    slots++;
    if (slp)
      sl_add_slots(slp,1,(uchar_t)level,(uchar_t)(start_bit+i));
  }

  test >>= 1;
  if (test==0)
  {
    test=0x80;
    ++index;
  }

  next_start_bit += exp_factor[level];
 }

 return slots;
}

/*****************************************************************************
 * msp_calc_tot_slots - calculates the total slots for the "msp" and
 *   builds a slot list if "slp" is not NULL
 *****************************************************************************/
unsigned msp_calc_tot_slots(o_msp_t *msp, slot_list_t *slp)
{
  uchar_t *flags;
  unsigned start_bit;
  unsigned slots;
  unsigned test;
  int i;
  unsigned exp_factor[MSP_MAX_LEVEL+1];
```

```
for (i=0; i <= MSP_MAX_LEVEL; i++)
  exp_factor[i]=MSP_EXP_FACTOR(msp,i);

slots=0;
flags=msp->exp_slots+1;
start_bit=0;
test=0x80;

/* expand the level 0 slots */
for (i=0; i < MSP_L0_SLOTS; i++)
{
  if (test & msp->exp_slots[0])
  {
    slots += msp_expand_slot(flags, exp_factor[0] * MSP_L0_SLOTS,
                start_bit, exp_factor[0],
                exp_factor, 1, slp);
  }
  else
  {
    slots++;
    if (slp)
      sl_add_slots(slp,1,0,(uchar_t)i);
  }

  start_bit += exp_factor[0];
  test >>= 1;
 }

 return slots;
}

/*****************************************************************************
 * node_slot_offset - calculates the response slot offset for a slave node
 *****************************************************************************/
int node_slot_offset(o_msp_t *msp, ushort_t node_id)
{
  uchar_t *flags;
  unsigned flag_cnt;
  unsigned start_bit;
  unsigned offset, i;
  unsigned slots;
  unsigned test;
  unsigned level;
  unsigned start_slot;
  unsigned exp_factor[MSP_MAX_LEVEL+1];
  unsigned exp_mask[MSP_MAX_LEVEL+1];

  start_slot=NTOI(msp->slot_offset) & 0xfff;

  for (i=0; i <= MSP_MAX_LEVEL; i++)
  {
    exp_factor[i]=MSP_EXP_FACTOR(msp,i);
    exp_mask[i]=MSP_EXP_MASK(msp,i);
  }
```

```
offset=node_id & MSP_L0_SLOTS-1;

slots=0;
flags=msp->exp_slots+1;
start_bit=0;
test=0x80;

/* expand previous slots */
for (i=0; i<offset; i++)
{
  if (test & msp->exp_slots[0])
  {
    /* recursive level 1..N expansion */
    slots += msp_expand_slot(flags, MSP_L0_SLOTS*exp_factor[0],
                             start_bit, exp_factor[0],
                             exp_factor, 1, NULL);
    start_bit += exp_factor[0];
  }
  else
    slots++;

  test >>= 1;
}

if (test & msp->exp_slots[0])
{
  /* level 1 expansion */
  start_bit=offset*exp_factor[0];
  node_id /= MSP_L0_SLOTS;
  offset=node_id & exp_mask[0];;

  flag_cnt=exp_factor[0] * MSP_L0_SLOTS;
  if (offset)
    slots += msp_expand_slot(flags, flag_cnt, start_bit,
                             offset, exp_factor, 1, NULL);

  /* level 2..N-1 expansion (level N cannot be expanded) */
  level=1;
  while (MSP_TEST_FLAG(flags,start_bit+offset) && level < MSP_MAX_LEVEL-1)
  {
    flags += flag_cnt >> 3;
    flag_cnt *= exp_factor[level];
    start_bit = (start_bit+offset) * exp_factor[level];
    node_id /= exp_factor[level-1];
    offset=node_id & exp_mask[level];
    ++level;
    slots+=msp_expand_slot(flags, flag_cnt, start_bit,
                           offset, exp_factor, level, NULL);
  }

  if (MSP_TEST_FLAG(flags,start_bit+offset))
  {
    node_id /= exp_factor[level-1];
```

```
        offset=node_id & exp_mask[level];
        slots += offset;
      }
    }

    start_slot=NTOI(msp->slot_offset) & 0xfff;

    return (int)(slots - start_slot);
  }
```

1. A communication network providing wireless communication within a premises, the wireless network comprising:

a wired network operating according to a wired protocol, the wired network having a first network segment and a second network segment;

a wireless terminal having a wired network protocol address;

a first access point coupled to the first network segment;

a second access point coupled to the first network segment; and

a data link tunnel that communicatively couples the second access point to the first access point when the wireless terminal is in wireless communication with the second access point.

2. The communication network of claim 1 wherein the first access point is connected to the first network segment.

3. The communication network of claim 2 wherein a protocol tunnel communicatively couples the first access point to the second network segment.

4. The communication network of claim 3 further comprising a third access point connected to the second network segment.

5. The communication network of claim 4 wherein the wireless terminal has a wired network protocol address respective to the third access point.

6. The communication network of claim 5 wherein the first access point and the third access point are communicatively coupled with a protocol tunnel.

7. The communication network of claim 6 wherein routed communication through the data link tunnel uses a different protocol scheme than when routed through the protocol tunnel.

8. The communication network of claim 1 wherein the wired network operates under the Internet protocol.

9. The communication network of claim 1 wherein the data link tunnel operates across the wired network.

10. The communication network of claim 1 wherein the data link tunnel operates across a radio link.

11. The communication network of claim 1 wherein routed communication from the first tunnel is not bridged onto the second network segment.

12. A communication network comprising:

a wired network having a first network subnet and a second network subnet;

a first tunnel coupling the first network subnet with the second network subnet;

a roaming terminal communicatively coupled with the firs network subnet; and

a second tunnel concatentated with the first tunnel to provide a logical extension of the first subnet for the roaming terminal.

13. The communication network of claim 12, wherein the communication network further sends a data message destined to the roaming terminal as a first message under a first network protocol, the first message encapsulating a second message under a second network protocol, the second message encapsulating a message under the wired network protocol.

14. The communication network of claim 13 wherein the second network protocol is a wireless network protocol.

15. The communication network of claim 13 wherein the second network protocol is a wired network protocol.

16. The communication network of claim 12 wherein the wired network operates under an Internet protocol.

17. The communication network of claim 12 wherein the second tunnel operates across the wired network.

18. The communication network of claim 12 wherein the second tunnel operates across a radio link.

19. The communication network of claim 12 wherein a routed communication from the first tunnel is not bridged onto the second network subnet.

20. A communication network comprising:

a wired network having a first network subnet and a second network subnet;

a first tunnel coupling the first network subnet with the second network subnet;

a roaming terminal communicatively coupled with the first network subnet; and

a second tunnel concatentated with the first tunnel to provide a logical extension of the first subnet for the roaming terminal without requiring the dynamic assignment of pseudo addresses.

21. The communication network of claim 20, wherein the communication network further sends a data message destined to the roaming terminal as a first message under a first network protocol, the first message encapsulating a second message under a second network protocol, the second message encapsulating a message under the wired network protocol.

22. The communication network of claim 21 wherein the second network protocol is a wireless network protocol.

23. The communication network of claim 21 wherein the second network protocol is a wired network protocol.

24. The communication network of claim 20 wherein the wired network operates under an Internet protocol.

25. The communication network of claim 20 wherein the second tunnel operates across the wired network.

26. The communication network of claim 20 wherein the second tunnel operates across a radio link.

27. The communication network of claim 20 wherein a routed communication from the first tunnel is not bridged onto the second network subnet.

28. A communication network providing wireless communication within a premises, the wireless network comprising:

a wired network operating according to a wired protocol, the wired network having at least a first network segment and a second network segment;

a wireless terminal having a wired network protocol address;

a first fixed access point connected to the second network segment;

a second fixed access point connected to the second network segment; and

a data link tunnel that communicatively couples the first and the second fixed access points via wireless communications only such that bridging communication data onto the second network segment is avoided when

communications between one of the fixed access points and the wireless terminal require communication with the other fixed access point.

29. The communication network of claim 28 wherein the data link tunnel comprises a radio link between the first and the second fixed access points.

30. The communication network of claim 28 wherein the wireless terminal is a roaming terminal.

31. The communication network of claim 30 wherein the first fixed access point encapsulates a message in a packet for transmission via the data link tunnel to the second fixed access point such that the message is supplied to the wireless terminal without the use of pseudo addresses which are dynamically assigned to roaming terminals.

32. The communication network of claim 28 further comprising a router that couples the first and the second network segments.

33. The communication network of claim 28 wherein the first network segment and the second network segment have different sub-network addresses.

34. The communication network of claim 28 wherein the wired network operates according to an internet protocol.

35. A communication network comprising:

a wired network having a first network access point and a second network access point;

a data link tunnel communicatively coupling the first network access point with the second network access point via wireless communications only; and

a roaming terminal communicatively coupled to the first network access point wherein communications from the roaming terminal pass within the data link tunnel to the second network access point.

36. The communication network of claim 35 wherein the data link tunnel comprises a radio link between the first and the second network access points.

37. The communication network of claim 35 wherein the roaming terminal is a wireless terminal.

38. The communication network of claim 37 wherein the first network access point encapsulates a message in a packet for transmission via the data link tunnel to the second network access point such that the message is supplied to the wireless terminal without the use of pseudo addresses which are dynamically assigned to roaming terminals.

39. The communication network of claim 35 wherein the wired network operates according to an internet protocol.

* * * * *