

E-5

```

Procedure Init1
when router i initializes itself
do begin
  set a link state table with costs of adjacent links:
   $N \leftarrow i; N_i \leftarrow x \mid l_x^i < \infty;$ 
  for each  $x \in N_i$ 
  do begin
     $N_i \leftarrow N \cup x; tag_x^i \leftarrow null;$ 
     $s_x^i \leftarrow null; p_x^i \leftarrow null; D_x^i \leftarrow \infty$ 
  end
   $D_i^i \leftarrow 0; s_i^i \leftarrow null; p_i^i \leftarrow null; tag_i^i \leftarrow correct$ 
  for each  $j \in N$  call Init2(x, j)
  for each  $n \in N_i$  do add  $(0, i, 0, i)$  to  $LIST_i(n)$ 
   $x \leftarrow$  retransmission time;  $y \leftarrow$  hello count;
   $s \leftarrow$  retransmission count;
  call Send
end

Procedure Init2(x, j)
begin
   $D_{jx}^i \leftarrow \infty; p_{jx}^i \leftarrow null; s_{jx}^i \leftarrow null; seqno_{jx}^i \leftarrow 0$ 
end

Procedure Send
begin
  for each  $n \in N_i$ 
  do begin
    if  $(LIST_i(n))$  is not empty
    then send messages with  $LIST_i(n)$  to  $n$ 
    empty  $LIST_i(n)$ 
  end
end

Procedure Message
when router i receives a message on link (i, k)
begin
  if  $(k \notin N_i)$  do
  begin
     $N_i \leftarrow N_i \cup k;$ 
     $l_k^i \leftarrow$  cost of new link;
    if  $(k \notin N)$  begin
       $N \leftarrow N \cup k; tag_k^i \leftarrow null;$ 
       $D_k^i \leftarrow \infty; p_k^i \leftarrow null; s_k^i \leftarrow null;$ 
      for each  $x \in N_i$  do call Init2(x, k)
    end
    for each  $(i, k, l_k^i)$  do
      send update  $(0, k, D_k^i, p_k^i)$ 
    end
  end
  reset HelloTimer;
  for each entry  $(u_j^k, j, RD_j^k, rp_j^k) \mid i \neq j$ 
  do begin
    if  $(j \notin N)$ 
    then begin
      if  $(RD_j^k = \infty)$  then delete entry
    else begin
       $N \leftarrow N \cup j;$ 
      for each entry  $x \in N_i$  call Init2(x, j)
       $tag_j^i \leftarrow null;$  call DT
    end
  end
  also begin
     $tag_j^i \leftarrow null;$ 
  end
end
for each entry  $(u_j^k, j, RD_j^k, rp_j^k)$  left  $\mid i \neq j$ 
do case of  $u_j^k$ 
0: call Update(j, k)
1: call ACK(j, k)
end
call Send
end

Procedure Create_RList(seqno)
begin
   $seqno \leftarrow seqno + 1; NeighborSet \leftarrow N_i$ 
   $bitmap[p] \leftarrow 0; RetransmissionTimer \leftarrow x$ 
  add updates to RList
end

Procedure Delete_RList(seqno)
begin
  set  $bitmap[seqno] \leftarrow 1; delete \leftarrow 1$ 
  for all  $n \in N_i$  begin
    if  $(bitmap[seqno] = 0)$  delete  $\leftarrow 0$ 
  end
  if  $(delete = 1)$  delete RList[seqno] end

Procedure Update_RList(seqno)
begin
  reset RetransmissionTimer
  send update RList[seqno];
end

Procedure Clean_RList(seqno)
begin
  for all entries in RList
  delete RList[seqno];
end

Procedure Connectivity
when HelloTimer expires
begin
   $HelloCount[k] \leftarrow HelloCount[k] + 1;$ 
  if  $(HelloCount[k] < y)$  then reset HelloTimer;
  also begin
     $N_i \leftarrow N_i - k$ 
    call Delete_RList(k)
     $l_k^i \leftarrow \infty$ 
     $tag_k^i \leftarrow null$ 
    delete column for k in distance table
    update routing table
  end
end

Procedure Timeout(i, k)
when RetransmissionTimer expires
begin
   $RetransmissionCounter \leftarrow RetransmissionCounter - 1;$ 
  if  $(RetransmissionCounter < z)$ 
  call Update_RList(k)
  else begin
     $N_i \leftarrow N_i - k$ 
    call Delete_RList(k)
     $l_k^i \leftarrow \infty$ 
     $tag_k^i \leftarrow null$ 
    delete column for k in distance table
    update routing table
  end
end

Procedure DT
when distance table update has to be done
begin
   $D_{jk}^i \leftarrow l_k^i + D_{jk}^k; p_{jk}^i \leftarrow p_j^k;$ 
  (2) for all neighbors b
  do begin
    if k is in the path from i to j in
    the distance table through neighbor b
    then  $D_{jb}^i \leftarrow D_{kb}^i + D_{j^i}^b; p_{jb}^i \leftarrow p_j^k$ 
  end
end

```

Figure 1: Protocol Specification

The link-cost table of node i lists the cost of relaying information through each neighbor k , and the number of periodic update periods that have elapsed since node i received any error-free messages from k .

The cost of a failed link is considered to be infinity. The way in which costs are assigned to links is beyond the scope of this specification. As an example, the cost of a link could simply be the number of hops, or the addition of the latency over the link plus some constant bias. The cost of the link from i to k (i, k) is denoted by l_k^i .

The message retransmission list (MRL) specifies one or more retransmission entries, where the m^{th} entry consists of the following:

- The sequence number of an update message

- A retransmission counter that is decremented every time node i sends a new update message
- An *ack-required* flag (denoted by a_{km}^i) that specifies whether node k has sent an ACK to the update message represented by the retransmission entry
- The list of updates sent in the update message

The above information permits node i to know which updates of an update message have to be retransmitted and which neighbors should be requested to acknowledge such retransmission. Node i retransmits the list of updates in an update message when the retransmission counter of the corresponding entry in the MRL reaches

```

Procedure ACK(n)
when router i receives an ACK on link (i, k)
begin
  call Delete_RList(n);
  RetransmissionCounter ← z;
end

Procedure Update(i, k)
when router i receives an update on link (i, k)
begin
  send ACK to neighbor k;
  RetransmissionCounter ← z;
  RetransmissionTimer ← x;
  (0) begin
    update=0;
    RTEMPi ← φ;
    DTEMPi,b ← φ for all neighbors b
  (1) for each triplet (j, Djk, pjk) in Vk,i, j ≠ i do
    call procedure DT
  (3) begin
    if there are b and j such that
      (Djb < Dji) or (Djb > Dji) and (b = sji)
    then call RT.Update
  end
  (4) begin if (RTEMPi ≠ φ) then
    for each neighbor b do begin
      for each triplet t = (j, Dji, pji) in RTEMPi
      do begin
        if b is not in the path from i to j
        then DTEMPi,b ← DTEMPi,b ∪ t;
      end
    end
    send DTEMPi,b to neighbor b;
  end
end
end

Procedure RT.Update
when routing table has to be updated
begin
  find minimum of the distance entries DTmin
  if (Djsj = DTmin) then ns ← sj
  else ns ← b | (b ∈ Ni and Djb = DTmin);
  x ← j;
  while (Dxns = Min{Dxbi | b ∈ Ni}
  and Dxns < ∞ and tagxi = null)
  do x ← pxns;
  if (pnsi = i or tagnsi = correct)
  then tagji ← correct else tagji ← error
  if (tagji = correct) then begin
    if (Dji ≠ DTmin or pji ≠ pnsi) then begin
      seqno ← seqno + 1;
      add (0, j, DTmin, pnsi, seqno) to LISTi(x) ∀ x ∈ Ni;
      call Clean_RList(seqno)
      call Create_RList(seqno)
    end
    Dji ← DTmin; pji ← pnsi; sji ← ns
  end
  else begin
    if (Dji < ∞) then begin
      seqno ← seqno + 1;
      add (0, j, ∞, null, seqno) to LISTi(x) ∀ x ∈ Ni;
      call Clean_RList(seqno)
      call Create_RList(seqno)
    end
    Dji ← ∞; pji ← null; sji ← null
  end
end

```

Figure 2: Protocol Specification (Cont..)

zero. The retransmission counter of a new entry in the MRL is set equal to a small number (e.g., 3 or 4).

2.3 Information Exchanged among Nodes

In WRP, nodes exchange routing-table update messages (which we call “update messages” for brevity) that propagate only from a node to its neighbors. An update message contains the following information:

- The identifier of the sending node.
- A sequence number assigned by the sending node.
- An *update list* of zero or more updates or ACKs to update messages. An update entry specifies a destination, a distance to the destination, and a predecessor to the destination. An ACK entry specifies the source and sequence number of the update message being acknowledged.
- A *response list* of zero or more nodes that should send an ACK to the update message.

In the event that the message space is not large enough to contain all the updates and ACKs that a node wants to report, they are sent in multiple update messages. An example of this event can be the case in which a node identifies a new neighbor and sends its entire routing table.

The response list of the update message is used to avoid the situation in which a neighbor is asked to send multiple ACKs to the same update message, simply because some other neighbor of the node sending the update did not acknowledge.

The first transmission of an update message must ask all neighbors to send an ACK, of course, and this is accomplished by specifying the “all-neighbors address,” which consists of all 1’s.

When the update message reports no updates, the “empty address” is specified; this address consists of all 0’s and instructs the receiving nodes not to send an ACK in return. This type of update message is used as a “hello message” from a node to allow its neighbors to know that they maintain connectivity, even if no user messages or routing-table updates are exchanged.

As we explain subsequently, an ACK entry refers to an entire update message, not an update entry in an update message, in order to conserve bandwidth.

2.4 Routing-Table Updating

Figures 1 and 2 specify important procedures of WRP used to update the routing and distance tables.

A node can decide to update its routing table after either receiving an update message from a neighbor, or detecting a change in the status of a link to a neighbor.

When a node i receives an update message from its neighbor k , it processes each update and ACK entry of the update message in order.

In WRP, a node checks the consistency of predecessor information reported by *all* its neighbors each time it processes an event involving a neighbor k . In contrast, all previous path-finding algorithms [4, 10, 14] check the consistency of the predecessor only for the neighbor associated with the input event. This unique feature of WRP accounts for its fast convergence after a single resource failure or recovery as it eliminates more temporary looping situations than previous path-finding algorithms.

2.4.1 Processing an Update

To process an update from neighbor k regarding destination j , the distance and the predecessor entries in the distance table are updated. A flag (tag) is set to specify that this entry in the table has been

changed. A unique feature of WRP is that node i also determines if the path to destination j through any of its other neighbors $\{b \in N_i | b \neq k\}$ includes node k . If the path implied by the predecessor information reported by node b includes node k , then the distance entry of that path is also updated as $D_{jb}^i = D_{kb}^i + D_j^k$ and the predecessor is updated as $p_{jb}^i = p_j^k$. Thus, a node can determine whether or not an update received from k affects its other distance and routing table entries.

To update its distance and predecessor for destination j (Procedure RT_Update), node i chooses a neighbor p that has reported routing information such that:

- The path from p to j (which is implied by the predecessor information reported by p) does not include node i
- $D_{jp}^i \leq D_{jx}^i$ for any other neighbor x , and $D_{yp}^i \leq D_{yx}^i$ for any other neighbor x and for every node y in the path from i to j .

The above means that node i chooses node p as its successor to a destination j if that neighbor appears to offer a smallest-cost loop-free path to j and all the intermediate nodes in the path to j .

When node i sends an update message, it updates its ack-status table and message retransmission list. For each destination j for whom there is an update being reported, node i sets the ack-required flag for all its neighbors. It also adds an entry in the update-retransmission list containing the sequence number given to the update message, and starts the retransmission timer for the entry.

2.4.2 Sending New and Retransmitted Update Messages

Node i sends a new update message after processing updates from its neighbors or detecting a change in a link to a neighbor. Whenever node i sends a new update message, it must

- Decrement the retransmission counter of all the existing entries in the MRL
- Delete the updates in existing entries in the MRL that are included in the new update message
- Add an entry in the MRL for the new update message

When the list of updates of a MRL entry is emptied by the transmission of a new update message, node i erases that entry from the MRL.

When the retransmission counter for a retransmission entry m in the MRL expires, node i sends an update message with a new sequence number, an update list containing the list of updates of the retransmission entry, and a response list specifying those neighbors who did not acknowledge the update message earlier (i.e., every neighbor k for whom $a_{km}^i = 1$). The retransmission counter of existing entries in the MRL is not modified.

Note that, based on the above retransmission strategy, there is no limit on the number of times node i would retransmit an update message to an existing neighbor. However, as we discuss below, node i stops considering node k as its neighbor after it fails to communicate with it for some finite amount of time.

2.4.3 Processing an ACK

An ACK entry in an update message refers to another entire update message, i.e., it acknowledges all the updates included in the update message bearing the referenced sequence number. Therefore, it is up to the node whose update message is being acknowledged to ascertain which updates are implied by a received ACK.

To process an ACK from neighbor k , node i scans its MRL for the sequence number matching the sequence number specified in the ACK received. Whenever a match is found, node i resets the ack-required flag for neighbor k ; if $a_{pm}^i = 0$ for entry m and every neighbor p of node i , the retransmission entry is deleted. This scheme obtains short ACKs at the expense of additional processing.

Node i may receive an ACK for an update message whose retransmission entry has been erased after sending a more recent update message for the same destinations. In that case, node i simply ignores the ACK.

2.4.4 Handling Topology and Link-Cost Changes

To ensure that nodes know that they have connectivity even when they do not transmit user messages or routing-table updates for some time, every node i must periodically send an update message reporting no changes (hello messages). Acknowledgments are not required for such update messages, and they can be very short (e.g., a byte for control information and a byte for the node identifier, since the control information can imply that there is no sequence number, update list, or response list in the message). Alternatively, a node may retransmit an update message if it is not too long. When a node k comes up, it transmits a hello message.

Given that short periodic update messages are transmitted by every node, the failure of a link to a neighbor is detected by the lack of any user or update messages being received from that neighbor over a period of time equal to a few update-message transmission periods. Similarly, new links and nodes are detected by means update messages or user messages.

When node i receives an update or user message from node k and node k is not listed in its routing table or distance table, it adds the corresponding entry to its distance or routing table for destination k . An infinite distance to all destinations through node k is assumed, with the exception of node k itself and those destinations reported in node k 's updates, if the message received from k was an update message. In addition, node i notifies node k of the information in its routing table. This information can be transmitted in one or multiple update messages that only node k needs to acknowledge.

When a link fails or a link-cost changes, node i recomputes the distances and predecessors to all affected destinations, and sends to all its neighbors an update message for all destinations whose distance or predecessor change.

2.5 Example

The following example illustrates the working of WRP. Consider a four node network shown in Figure 3(a). All links and nodes are assumed to have the same propagation delays. Link-costs are as indicated in the figure. Node i is the source node, j is the destination node and nodes k and b are the neighbors of node i . The arrows next

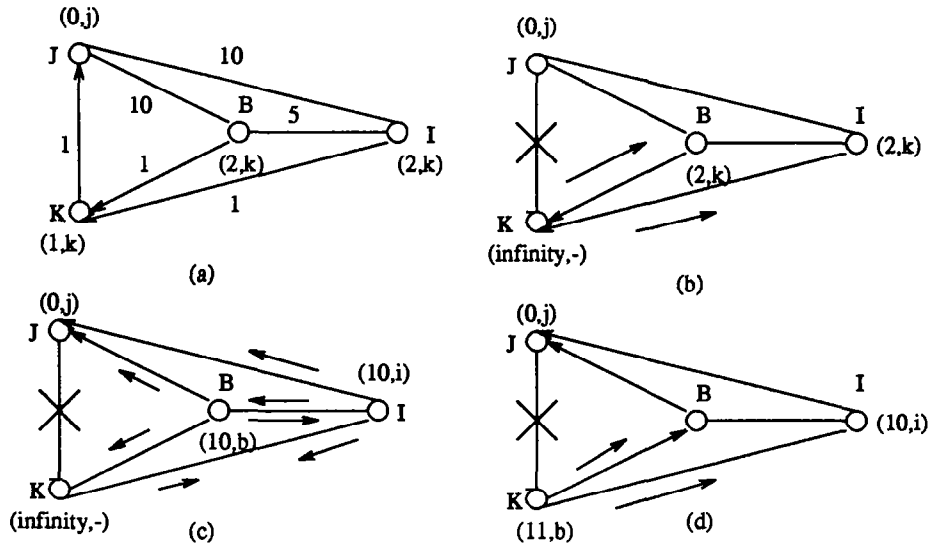


Figure 3: Example of the algorithm's operation

to links indicate the direction of updates messages and the label in parentheses gives the distance and the predecessor to destination j . Each update will be acknowledged by an ACK message from the neighbor. ACKs are not shown in the figure. The figure focuses on update messages to destination j only.

When link (j, k) fails, nodes j and k send update messages to their neighboring nodes as shown in Figure 3(b). In this example, node k is forced to report an infinite distance to j as nodes b and i have reported node k as part of their path to destination j . Node b processes node k 's update and selects link (b, j) to destination j . This is because of step(2) of WRP which forces node b to purge any path to node j involving node k . Also, when i gets node k 's update message, i updates its distance table entry through neighbor k and checks for the possible paths to destination j through any other neighboring nodes. Thus, a node examines the available paths through its other neighboring nodes and updates the distance and the routing table entries accordingly. This results in the selection of the link (i, j) to the destination j (Figure 3(c)). When node i receives neighbor b 's update reporting an infinite distance, node i does not have to update its routing table as it already has correct path information (Figure 3(d)). Similarly, updates sent by node k reporting a distance of 11 to destination j will not affect the path information of nodes i and b . This illustrates how step(2) of WRP helps in the reduction of the formation of temporary loops in the explicit paths.

3 Simulation Results

To gain insight into the average-case performance of WRP in a dynamic environment, we have simulated its operation using an actor-based, discrete-event simulation language called *Drama* [15], together with a network simulation library. Link failures and recoveries are simulated by sending link status message to the nodes at the end points of the appropriate links. Node failures can be treated as all links connecting to that node going down at the same time and

the link cost changes can be treated as a link failing and recovering with a new link cost. The connectivity of a mobile node is said to be lost when a node does not hear from a mobile node for a certain period of time. The connectivity with a node will be reestablished when a node hears from a mobile node again. Mobility is modeled as an arbitrary set of failures and recoveries of a mobile node at random points in time. All simulations are done assuming unit propagation time and zero packet processing time at each node. If a mobile node fails when the packets are in transit, the packets are assumed to get dropped.

Our goal is to compare the performance of WRP against the performance of routing protocols based on DBF, DUAL, and ILS. To reduce the complexity of the simulation, we have eliminated those features of the protocols that were common to all; these features concern the reliable transmission of updates over unreliable links, and the identification of neighbors. Accordingly, our simulation assumed that, for any of the protocols simulated, any update message sent over an operational link is received correctly, and that a node always receives enough user messages to know that it continues to have connectivity with a neighbor. According to these assumptions, there is no need to account for acknowledgments, retransmissions of updates, or periodic transmissions of update messages.

However, our intent in running the simulations was to obtain insight on the comparative overhead of different protocols that necessarily require the transmission of acknowledgements to update messages. We approached this problem in the following manner: In a packet radio network, the same update messages sent by a node is received by all its neighbors i.e., each update message is broadcast to a node's neighbors. However, to guarantee the reliable transmission of updates, each neighbor must send an acknowledgement to the sender of the update. Therefore, under the assumption that no errors or collisions occur in the network channel, counting the number of acknowledgements received for a single update broadcast to all neighbors is much the same as counting the number

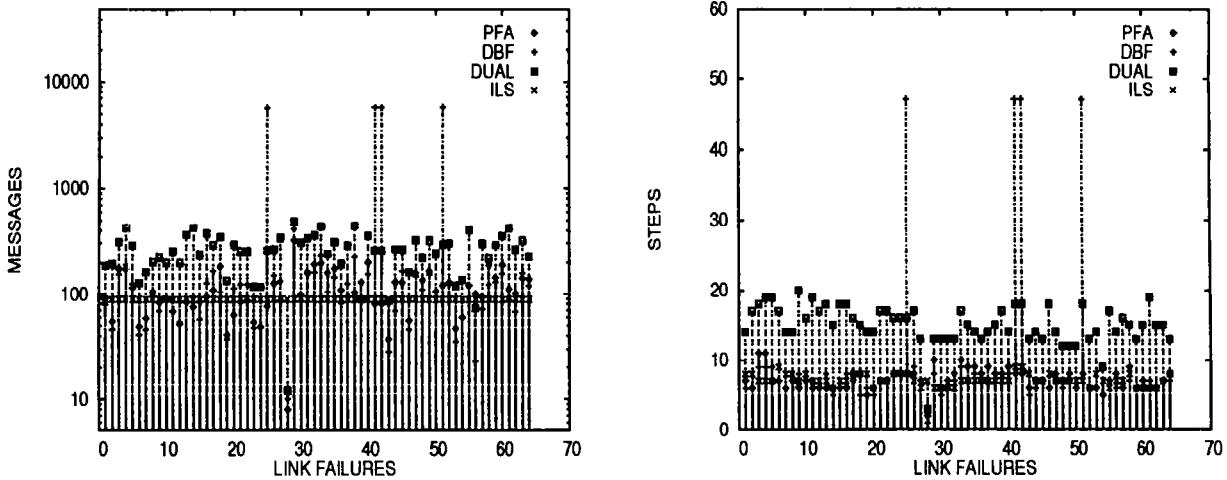


Figure 4: ARPANET Link Failure

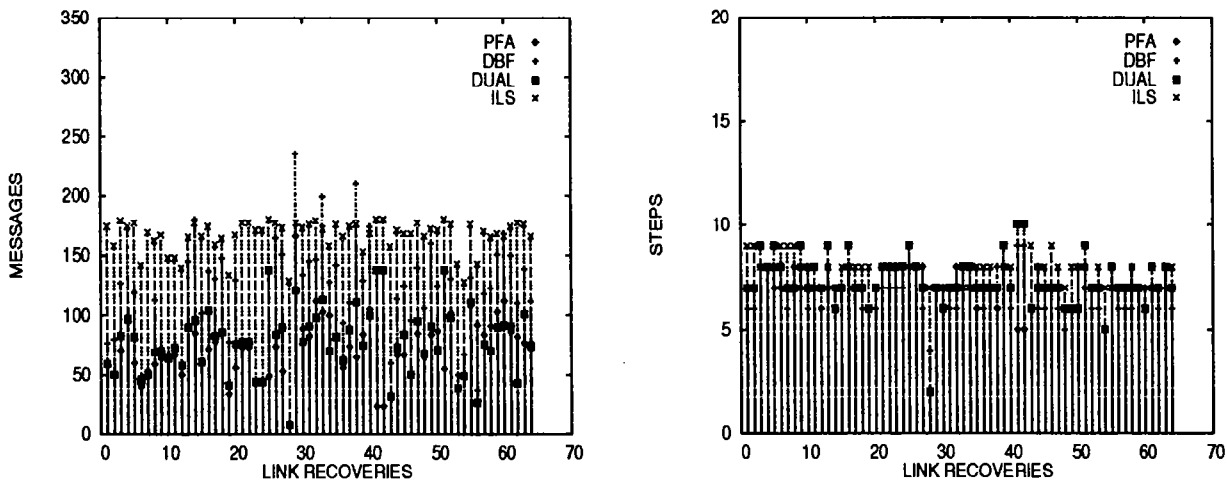


Figure 5: ARPANET Link Recovery

of updates sent by a node to its neighbors on a point-to-point basis and with no acknowledgements—the two counts differ only by one. Accordingly, we simulated the routing protocols’ operation in a packet-radio network using the same point-to-point links typical of wireline networks. The message count obtained from the simulation runs is not the exact number of updates and acknowledgements sent by each protocol, but accurately reflects the relative differences among protocols.

The resulting simplified version of WRP we simulated is called “path finding algorithm” (PFA), and is the same basic algorithm first described in [13]. Similarly, ILS, DBF, and DUAL correspond to the ideal case of the best protocols that could be designed based on these algorithms.

To simulate the routing algorithm, a node receives a packet and responds to it by running the routing algorithm, queueing the outgoing packets and processing the updates one at a time in the order in which they arrive. Drama’s internals ensure that all the packets at a given time are processed before new updates are generated.

The simulations were run on several network topologies such as *Los-Nettos*, *Nsfnet* and *Arpanet*. We chose these topologies to com-

pare the performance of routing algorithms for well-known cases given that we cannot sample a large enough number of networks to make statistically justifiable statements about how an algorithm scales with network parameters. The *los-nettos* topology has 11 nodes, a diameter of 4 hops, and each node has at most four neighbors. The *Nsfnet* topology has 13 nodes, a diameter of 4 hops, and each node has at most 4 neighbors. The *ARPANET* topology has 57 nodes, a diameter of 8 hops, and each node has a maximum of four neighbors.

For the routing algorithms under consideration, there is only one shortest path between a source and a destination pair and we do not consider null paths from a node to itself. Data are collected for a large number of topology changes to determine statistical distribution. The statistics has been collected after each failure and recovery of a link. To obtain the average figures, we make each link (or node) in the network fail and count the number of steps and messages needed for each algorithm to converge. Then the same link (node) is made to recover and the process is repeated. The average is taken over all failures and recoveries. Again, this message count is not exact, but the relative difference from one

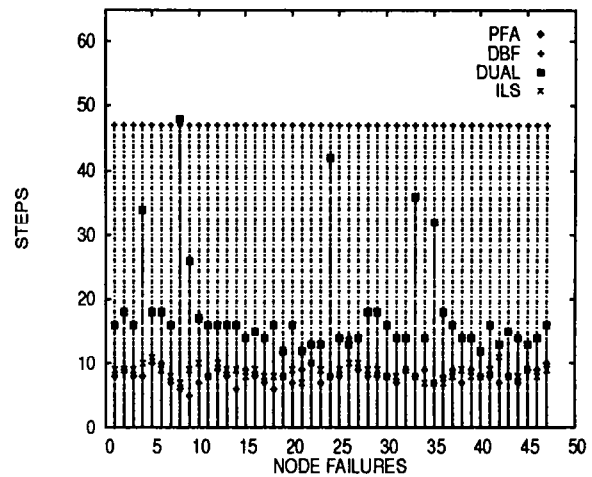
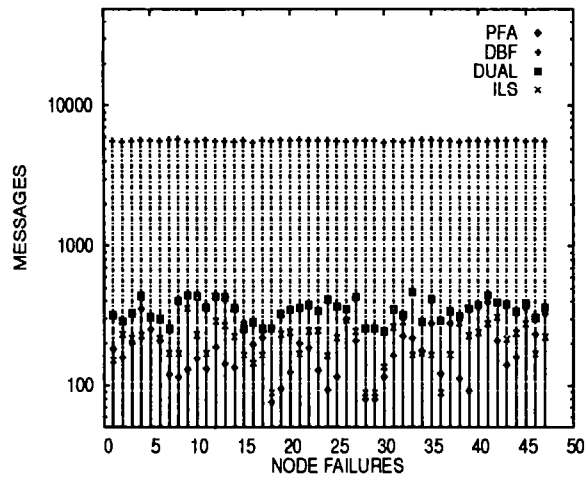


Figure 6: ARPANET Node Failure

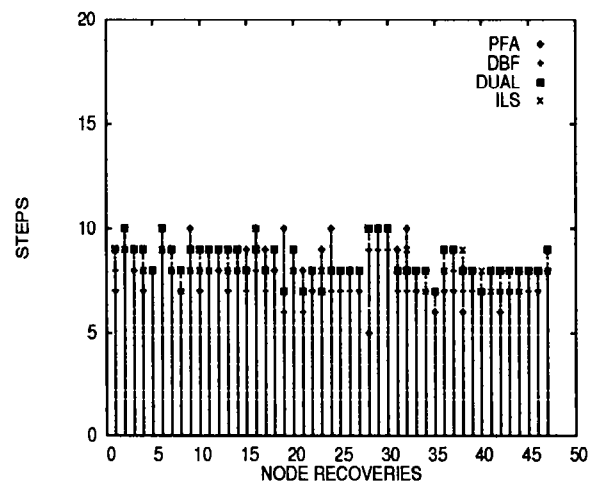
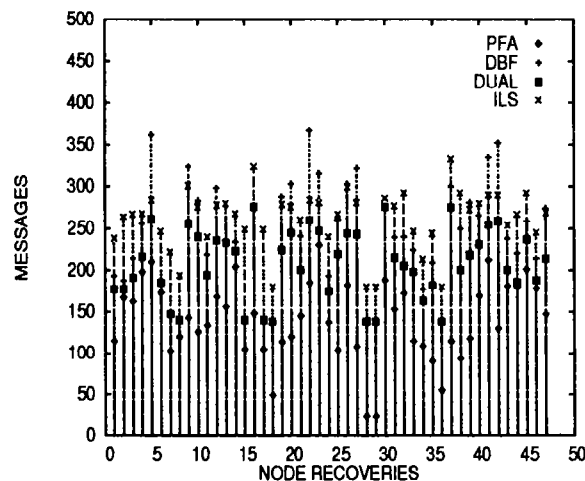


Figure 7: ARPANET Node Recovery

protocol to another is accurate.

3.1 Total Response to a Single Resource Change

The graphs in Figures 4 and 5 depict the number of messages exchanged and the number of steps required before PFA, DBF, DUAL, and ILS converge for every link failing and recovering in the ARPANET topology. We focus more on the results for the ARPANET topology, because of its larger size. Similar graphs for every node failing and recovering are given in Figures 6 and 7 respectively. All topology changes are performed one at a time and the algorithms were allowed to converge after each such change before the next resource change occurs. The ordinates of the graphs represent the identifiers of the links and the nodes while the data points show the number of messages exchanged after each resource change (graphs on the left hand side) and the number of steps needed for convergence (graphs on the right hand side) in each of these figures.

For a single resource failure, PFA outperforms DUAL. This is because, PFA does not use an internodal coordination mechanism that spans several hops to achieve loop freedom. The performance of PFA is comparable to that of ILS after resource failures. The

performance of PFA and DUAL is much better than that of ILS after resource recoveries. The counting-to-infinity problem of DBF can be clearly seen in both resource failures and resource recoveries. Given that both resource recoveries and failures will occur in the WRP, PFA offers the best total response to single topology changes, in terms of both update messages and time required to obtain correct routing tables after a topology change.

3.2 Dynamics with Mobile Nodes

We incorporated mobility to the existing fixed network topology by making the links fail and come back up arbitrarily at random points in time. The network is assumed to be fully connected with potential links. At startup, the topology is initialized to some well known topology such as *los-nettos*, *Nsfnet* or *ARPANET*. After initialization, to simulate the movement of a node, a node is assumed to have failed at its previous location and reappear in its new location. Node failure is simulated as all the links associated with that node going down at the same time. The gradual movement of a node from one location to another is simulated by means of link failures and additions. When a link fails, it can be assumed that a node is no

longer in the neighborhood of its previous neighbor. The addition of a new link is viewed as a movement of a node wherein, a node reappears in the new neighborhood.

The links are chosen at random from the set of all the existing links in the fully connected network. Selecting any particular link is equally likely. The probability of a link failing or recovering is also equally likely. We also have imposed an additional condition in our simulations that a node at any given time cannot have more than x neighbors. Here, x indicates the degree of the node. This condition is imposed in order to make sure that all the links pertaining to one node alone will not be active. This helps in simulating the mobility more closely.

The average number of messages and the average message length for each of these algorithms are obtained by varying the interarrival time between two events (Figures 8–10). An event can be either a link failure or a link recovery. For the purpose of event generation, we consider a fully connected topology and start off with a given initial topology. Since any random link can fail or recover at any time, our model simulates mobility closely.

The above results indicate that the routing algorithm of WRP outperforms all other algorithms which we have simulated, namely, DBF, DUAL and ILS. We were not able to simulate ILS algorithm for ARPANET topology due to limited resources. The statistics about the average number of messages and the average message length have been collected for all the above mentioned topologies for all the four algorithms by varying the interarrival time between events (failures and recoveries).

In all cases, the average number of messages for DBF and DUAL are more than that of WRP. This is because, DBF suffers from counting-to-infinity problem and DUAL uses an interneighbor coordination mechanism to achieve loop-freedom and this synchronization mechanism spans the entire diameter of the network. ILS sends maximum number of messages since the complete topology information has to be transmitted to each node every time the topology changes.

The average length of each message is the highest in DUAL as compared to all other algorithms. The average message length in case of ILS is almost constant since it always sends the complete topology information. Even though we do not have simulation results for ILS in case of ARPANET topology, we can extrapolate the results from the other two network topologies and can expect similar behavior for ARPANET topology also.

4 Conclusion

A new routing protocol, WRP, for a packet radio network has been presented. This protocol is based on a path-finding algorithm which substantially reduces the number of cases in which routing loops can occur. A mechanism has been proposed for the reliable exchange of update messages. The performance of the routing algorithm in WRP has been compared with that of an ideal topology broadcast algorithm (ILS), DUAL and DBF for highly dynamic environment through simulations. The simulation results show that WRP provides about 50% improvement in the convergence time as compared to DUAL. The results indicate that WRP is a good alternative for

routing in packet radio networks.

References

- [1] D. Beyer *et. al.*, "Packet Radio Network Research, Development and Application", *Proc. SHAPE Conference on Packet Radio*, Amsterdam, 1989.
- [2] D. Beyer, "Accomplishments of the DARPA SURAN Program", *Proc. IEEE MILCOM 90*, Monterey, California, Dec. 1990.
- [3] D. Bertsekas and R. Gallager, *Data Networks*, Second Ed. Prentice Hall, Inc. 1992.
- [4] C. Cheng, R. Reley, S. P. R. Kumar and J. J. Garcia-Luna-Aceves, "A Loop-Free Extended Bellman-Ford Routing Protocol without Bouncing Effect", *ACM Computer Communications Review*, 19 (4), 1989, pp.224–236.
- [5] Charles E. Perkins and Pravin Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", *ACM SIGCOMM*, Vol.24 (No.4), Oct. 1994, pp.234–244.
- [6] M. Scott Corson and Anthony Ephremides, "A Distributed Routing Algorithm for Mobile Wireless Networks", *ACM J. of Wireless Networks*, Jan. 1995, pp. 61–81.
- [7] J.J. Garcia-Luna-Aceves, "A Fail-Safe Routing Algorithm for Multihop Packet-Radio Networks", *IEEE INFOCOM* April, 1986.
- [8] J.J. Garcia-Luna-Aceves, "Loop-Free Routing Using Diffusing Computations", *IEEE/ACM Trans. Networking*, Vol.1, No. 1, Feb. 1993, pp.130–141.
- [9] J. Hagouel, "Issues in Routing for Large and Dynamic Networks", *IBM Research Report*, RC 9942 (No. 44055), April 1983.
- [10] P.A. Humblet, "Another Adaptive Shortest-Path Algorithm", *IEEE Trans. Comm.*, Vol.39, No.6, June 1991, pp.995–1003.
- [11] B.M. Leiner, D.L. Nielson and F.A. Tobagi, *Proc. IEEE*, Packet Radio Networks Special Issue, Jan. 1987.
- [12] J. Moy, "OSPF Version 2", *RFC 1583*, March 1994.
- [13] Shree Murthy and J.J. Garcia-Luna-Aceves, "A More Efficient Path-Finding Algorithm", *28th Asilomar Conference*, Pacific Grove, CA, Nov. 1994.
- [14] B. Rajagopalan and M. Faiman, "A Responsive Distributed Shortest-Path Routing Algorithm within Autonomous Systems," *Journal of Internetworking: Research and Experience*, Vol. 2, No. 1, March 1991, pp. 51–69.
- [15] W. T. Zaumen, "Simulations in Drama", *Network Information System Center, SRI International*, Menlo Park, California, Jan. 1991.

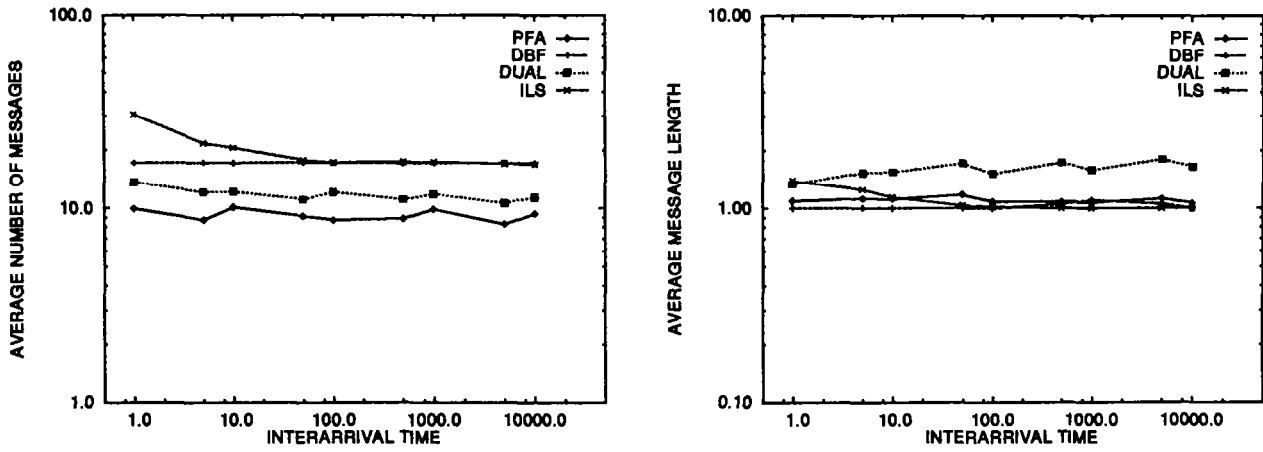


Figure 8: Los-Nettos

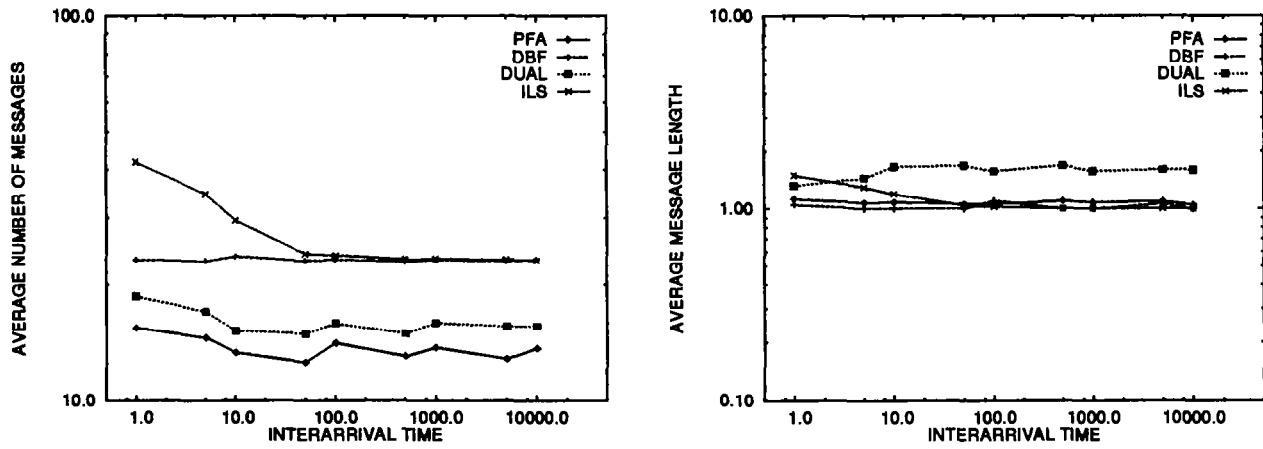


Figure 9: Nsfnet

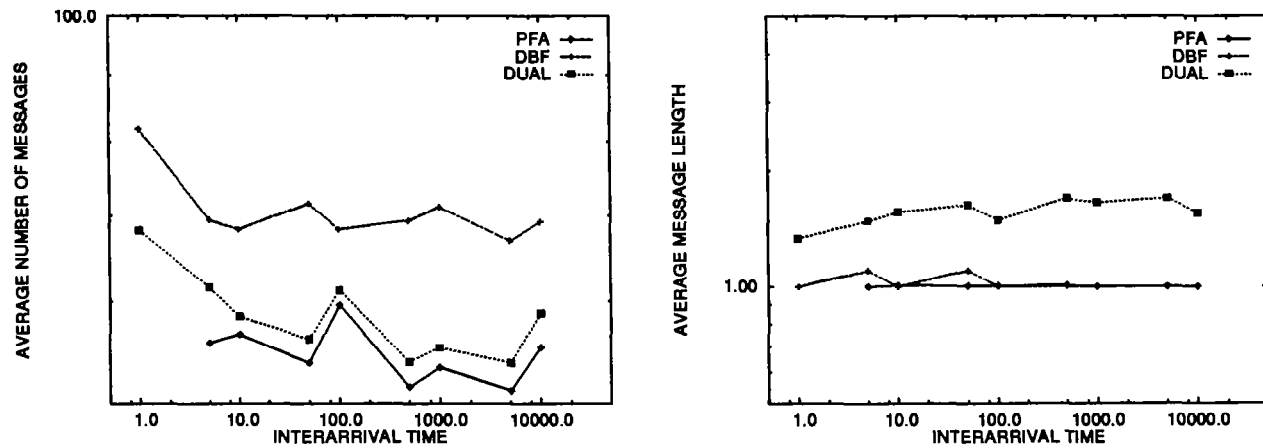


Figure 10: ARPANET

United States Patent [19]

[11] **Patent Number:** **5,673,252**

Johnson et al.

[45] **Date of Patent:** **Sep. 30, 1997**

[54] **COMMUNICATIONS PROTOCOL FOR REMOTE DATA GENERATING STATIONS**

0263421 4/1988 European Pat. Off. .
 0366342 5/1990 European Pat. Off. .
 2205260 10/1980 Germany .
 2060843 4/1981 Germany .

[75] **Inventors:** **Dennis F. Johnson; Don Marcynuk; Erwin Holowick**, all of Winnipeg, Canada

OTHER PUBLICATIONS

WO93/14585 (PCT/US93/00014) with International Search Report.

[73] **Assignee:** **Itron, Inc.**, Spokane, Wash.

Primary Examiner—Dang Ton
Attorney, Agent, or Firm—Patterson & Keough, P.A.

[21] **Appl. No.:** **451,386**

[22] **Filed:** **May 26, 1995**

[57] **ABSTRACT**

Related U.S. Application Data

[63] Continuation of Ser. No. 247,988, May 23, 1994, abandoned, which is a continuation-in-part of Ser. No. 124,495, Sep. 22, 1993, abandoned, which is a continuation of Ser. No. 732,183, Jul. 19, 1991, which is a continuation-in-part of Ser. No. 480,573, Feb. 15, 1990, Pat. No. 5,056,107.

A method for communicating data between a central data terminal, a plurality of intermediate data terminals, a plurality of remote cell nodes, and a plurality of network service modules, using a plurality of frames with each frame having a plurality of channels. The plurality of intermediate data terminals transmit IDT-synchronization signals to the plurality of remote cell nodes on a first channel of the frame. The plurality of remote cell nodes transmit RCN-synchronization signals to the plurality of network service modules on a second channel of the frame. The network service modules transmit data from a plurality of physical devices, using radio waves, as NSM-packet signals to the plurality of remote cell nodes using a fourth channel of the frame. The plurality of remote cell nodes store the incoming NSM-packet signals and, responsive to a first polling signal transmitted in a third channel of the frame from a particular intermediate data terminal, transmit the NSM-packet signals to the intermediate data terminal as RCN-packet signals on a fifth channel of the frame. The intermediate data terminal in turn stores the RCN-packet signals received from the plurality of remote cell nodes and, responsive to a second polling signal transmitted from the central data terminal on a sixth channel of the frame, transmits the RCN-packet signals as an IDT-packet signal on a seventh channel of the frame to the central data terminal.

[51] **Int. Cl.⁶** **H04J 3/16**
 [52] **U.S. Cl.** **370/94.1; 370/95.2; 370/100.1**
 [58] **Field of Search** 370/95.1, 95.2, 370/95.3, 105.1, 105.4, 105.2, 94.1, 94.2, 94.3, 60, 60.1, 61, 100.1, 110.1, 85.1, 85.6, 85.08; 375/354, 355, 356, 357, 359, 363, 365, 366, 372; 340/825.5, 825.51, 825.52, 825.08; 455/13.3, 13.4, 51.1, 54.2, 58.1

[56] **References Cited**

U.S. PATENT DOCUMENTS

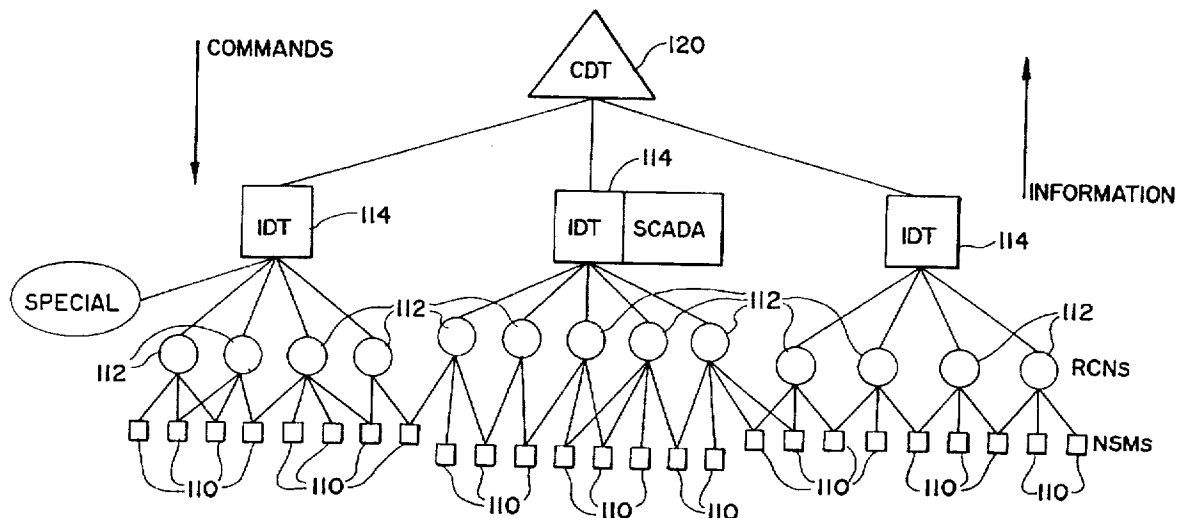
1,987,889 1/1935 Beverage et al. 342/367
 3,705,385 12/1972 Batz 340/152
 3,786,423 1/1974 Martell 340/151
 3,860,870 1/1975 Richards et al. 370/11
 4,013,962 3/1977 Beseke et al. 370/11

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

0244384 11/1987 European Pat. Off. .

85 Claims, 39 Drawing Sheets



5,673,252

Page 2

U.S. PATENT DOCUMENTS							
4,040,046	8/1977	Long et al.	340/310	4,815,106	3/1989	Propp et al.	375/257
4,190,800	2/1980	Kelly, Jr. et al.	340/310.02	4,839,642	6/1989	Batz et al.	340/825
4,361,851	11/1982	Asip et al.	358/84	4,887,259	12/1989	Morita	370/60
4,388,690	6/1983	Lumsden	364/483	4,952,928	8/1990	Carroll et al.	340/825.54
4,495,596	1/1985	Sciulli	364/900	4,958,645	9/1990	Cadell et al.	128/903
4,661,804	4/1987	Abel	340/539	5,014,213	5/1991	Edwards et al.	364/483
4,707,679	11/1987	Kennon et al.	340/310	5,056,107	10/1991	Johnson et al.	375/200
4,734,680	3/1988	Gehman et al.	340/539	5,086,292	2/1992	Johnson et al.	340/637
4,780,910	10/1988	Huddleston et al.	455/617	5,132,968	7/1992	Cephus	370/94.1
4,783,623	11/1988	Edwards et al.	324/156	5,166,664	11/1992	Fish	340/539
4,799,059	1/1989	Grindahl et al.	340/870.03	5,239,575	8/1993	White et al.	379/107
4,799,062	1/1989	Sanderford et al.	342/450	5,264,828	11/1993	Meiksin et al.	340/539
4,804,938	2/1989	Rouse et al.	340/310	5,381,136	1/1995	Powers et al.	340/539

Fig. 1

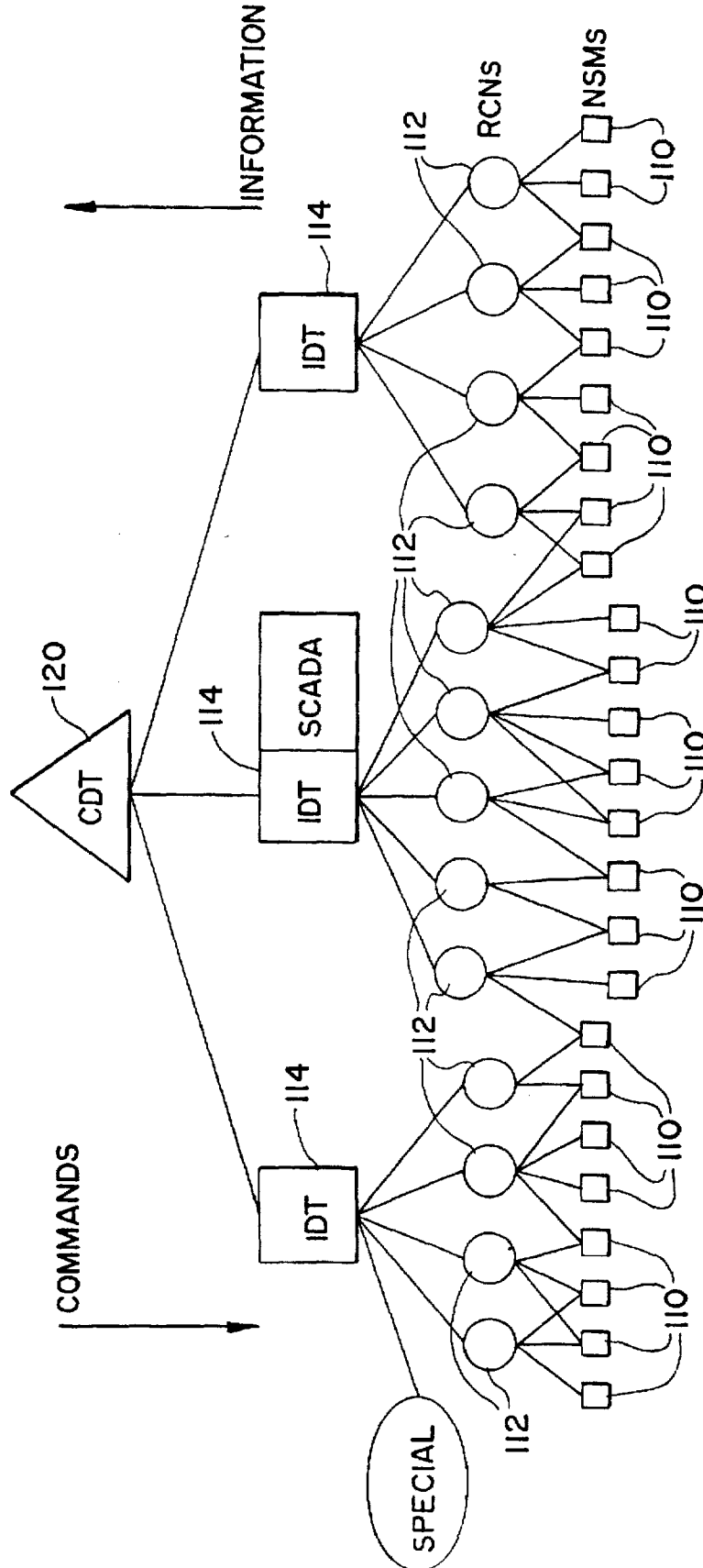


Fig. 2

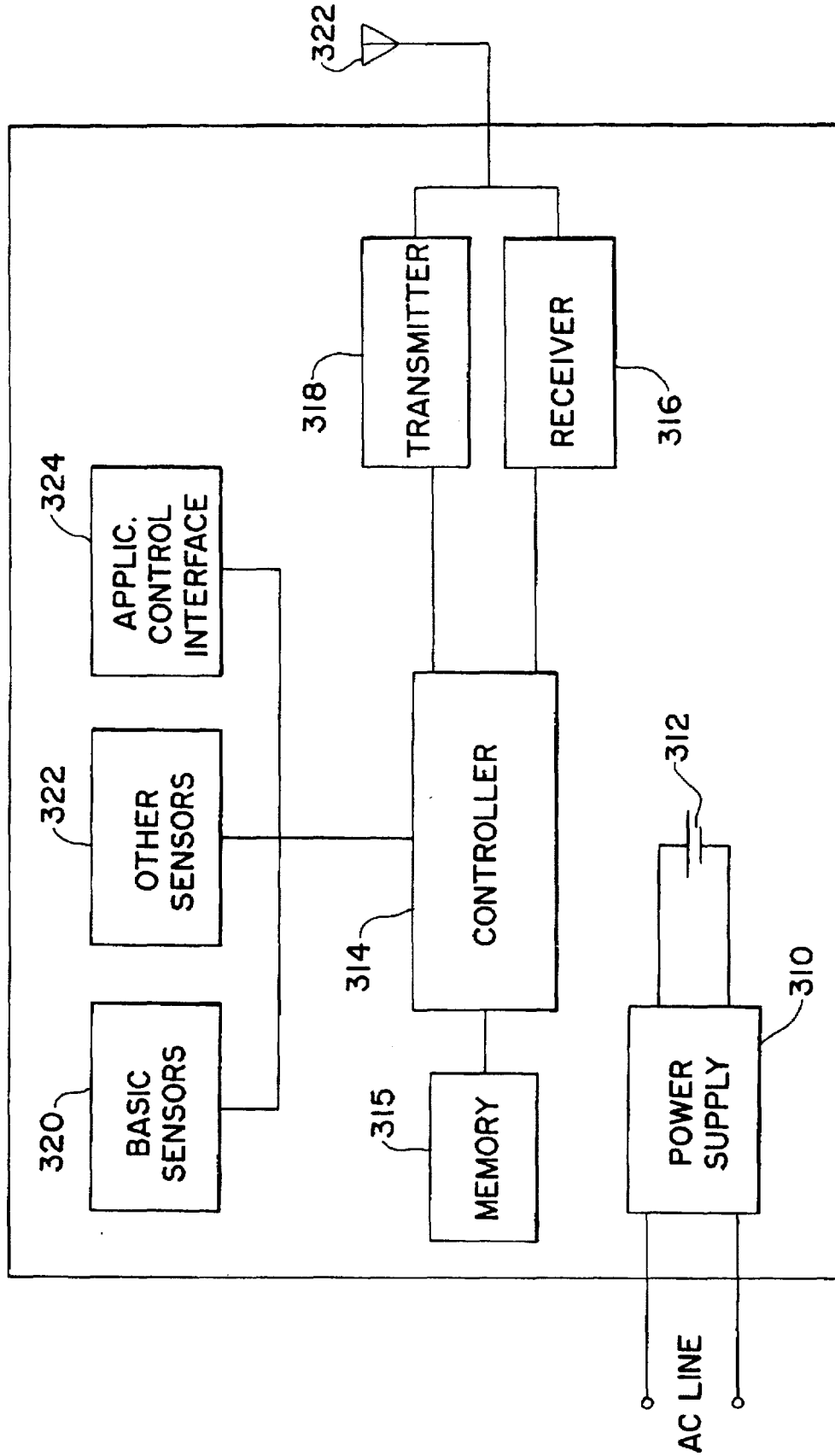


Fig. 3

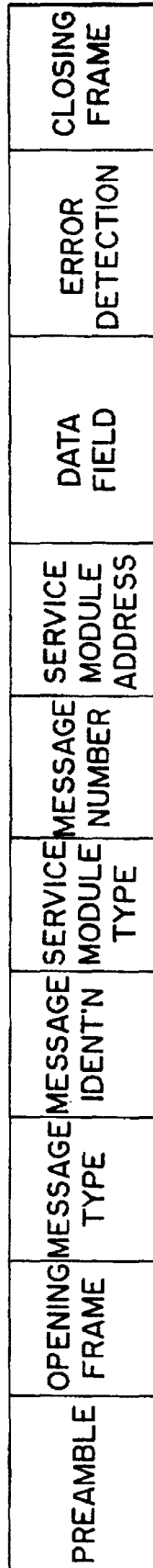


FIG. 4

<u>GROUP</u>	<u>APPLICATION</u>
1. - BILLING	1.1 BASIC MONTHLY BILLING (RES./COMMERCIAL) 1.2 TIME OF USE BILLING (RES./COMMERCIAL) 1.3 DEMAND BILLING (RES./COMMERCIAL) 1.4 CONSOLIDATED BILLS FOR MULTI-SITE CUSTOMERS E.G. BILLBOARD OPERATORS (COMMERCIAL) 1.5 PREPAYMENT CUSTOMER SUPPORT (RESIDENTIAL)
2. - SPECIAL READS	2.1 CUSTOMER INQUIRIES (1-3 MONTH DAILY CONSUMPTION RECORD AVAILABLE FOR EACH CUSTOMER) 2.2 OPENING ACCOUNTS (CURRENT READING AVAILABLE) 2.3 CLOSING ACCOUNTS (CURRENT READING AVAILABLE) 2.4 REFUND AUTHORIZATION ON PREPAYMENTS
3. - UNAUTHORIZED SERVICE USE	3.1 SOURCE OF LOSSES 3.2 METER TAMPER DETECTION AND MONITORING 3.3 REVENUE DIVERSION ESTIMATION
4. - GRID STATUS	4.1 DISTRIBUTION GRID OUTAGE (FUSE, RECLOSER, SECTIONALIZER, DISTRIBUTION TRANSFORMER) 4.2 INDIVIDUAL SERVICE LOSS 4.3 RESTORATION NOTIFICATION
5. - SERVICE QUALITY	5.1 OUTAGE INFORMATION 5.2 VOLTAGE ON LINES
6. - GRID CONFIGURATION MANAGEMENT	6.1 CAPACITOR BANK SWITCHING 6.2 TRANSFORMER LOAD MANAGEMENT 6.3 FEEDER LOAD MANAGEMENT 6.4 SECTIONALIZER CONTROL
7. - LOAD CONTROL	7.1 AIR CONDITIONERS 7.2 WATER HEATERS 7.3 POOL PUMPS/HEATERS
8. - SERVICE CONTROL	8.1 SERVICE CONNECT 8.2 SERVICE DISCONNECT 8.3 SERVICE LIMITATION
9. - LOAD SURVEY	9.1 15 MINUTE RESOLUTION LOAD SURVEY
10. - SUB-STATION MONITORING SUB-SCADA AND SCADA	10.1 TRANSFORMERS (TEMPERATURE, VOLTAGE, DEMAND, ETC.) 10.2 OTHER
11. - LOAD CURTAILMENT MONITORING AND NOTIFICATION	11.1 HIGH RESOLUTION READINGS ON DEMAND 11.2 CENTRALIZED CONTROL FOR MULTIPLE-STATE GEOGRAPHIC AREAS 11.3 CUSTOMER NOTIFICATION (CRT, PAGE, RADIO)

LIGHT I/P
TO RECHARGE
BATTERY

Fig. 5

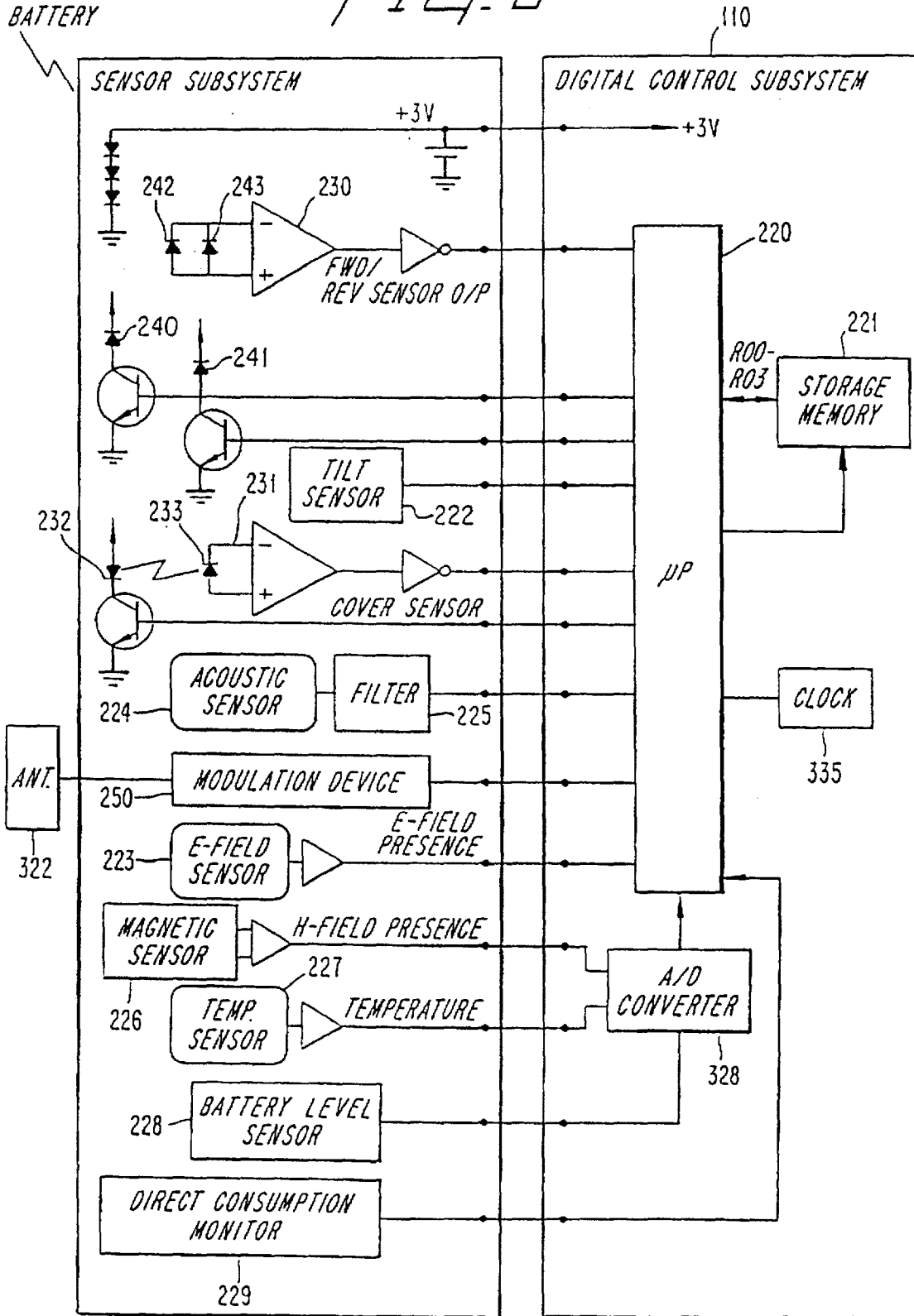


Fig. 6

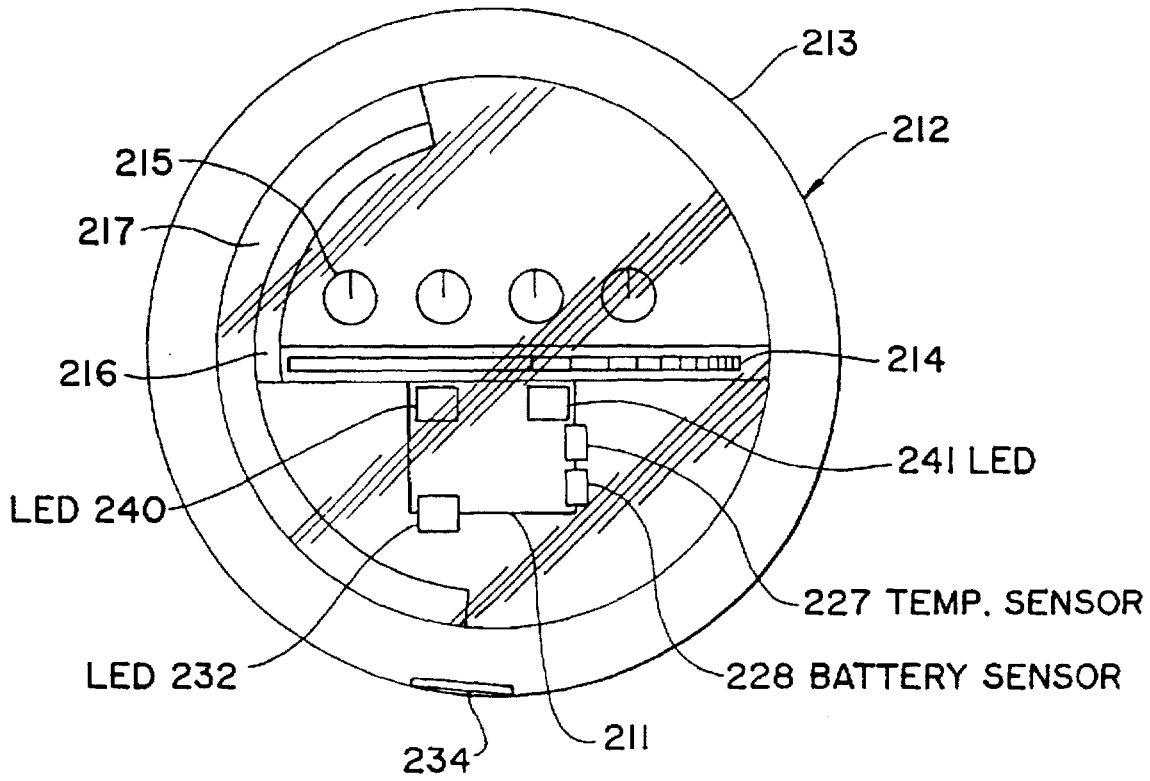


Fig. 7

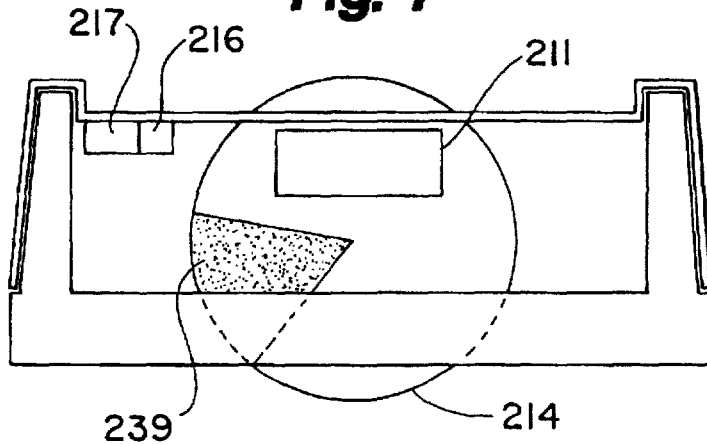


Fig. 8

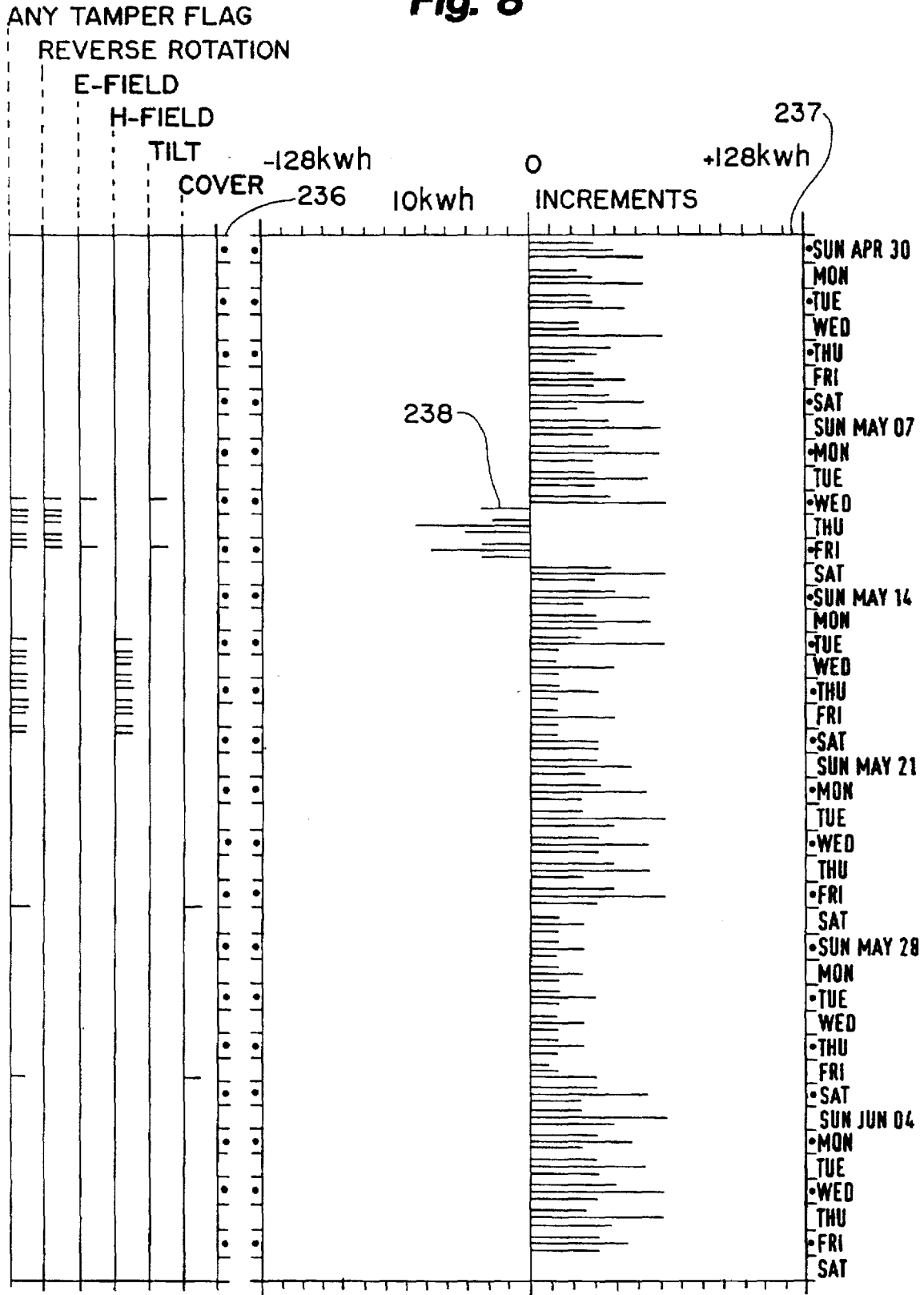


Fig. 9

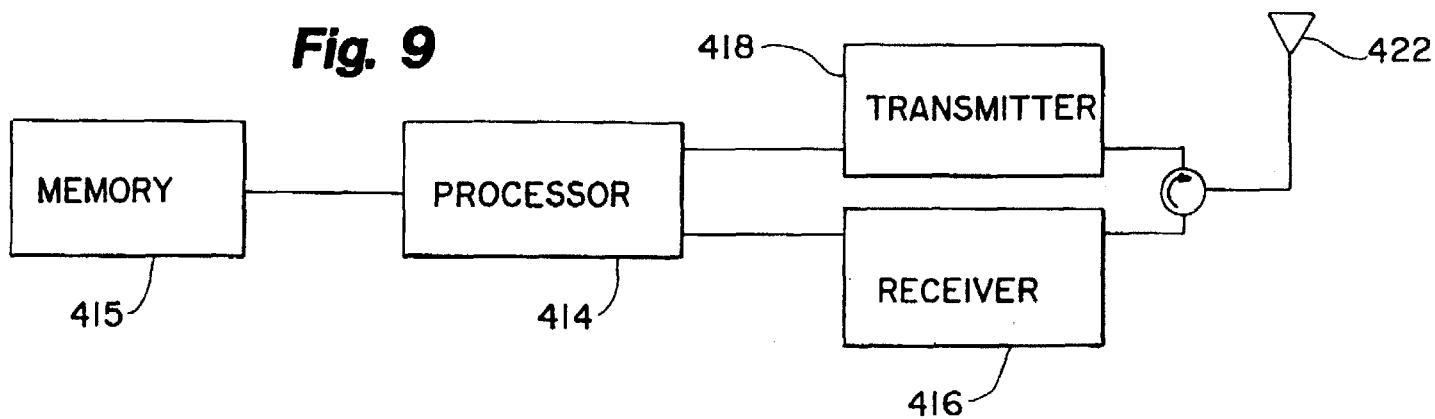


Fig. 10

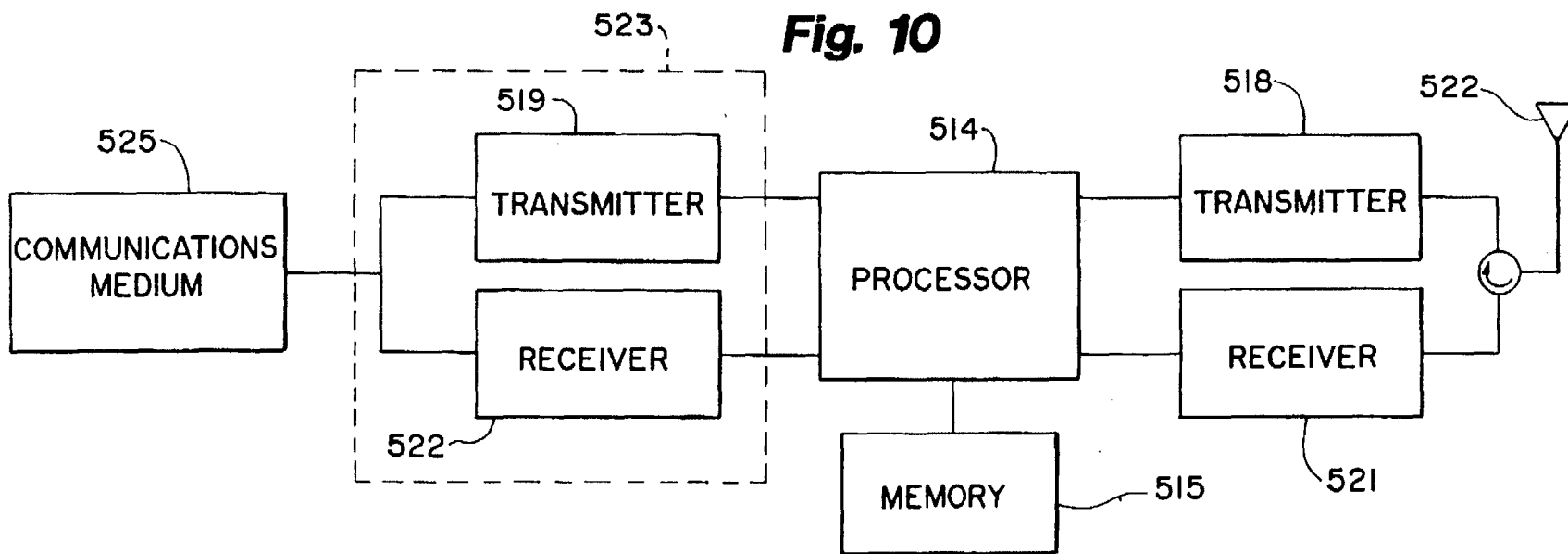


Fig. 11

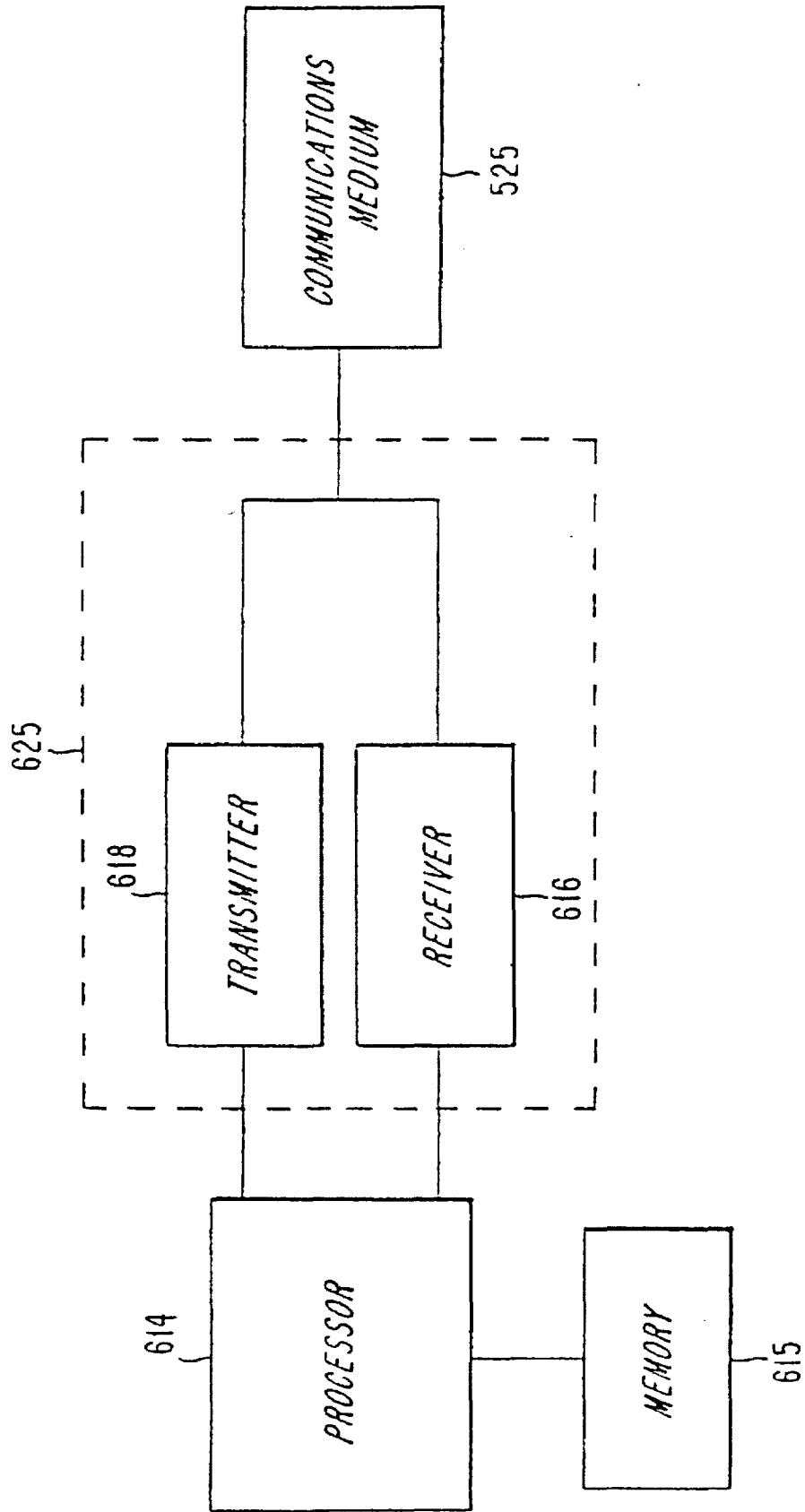
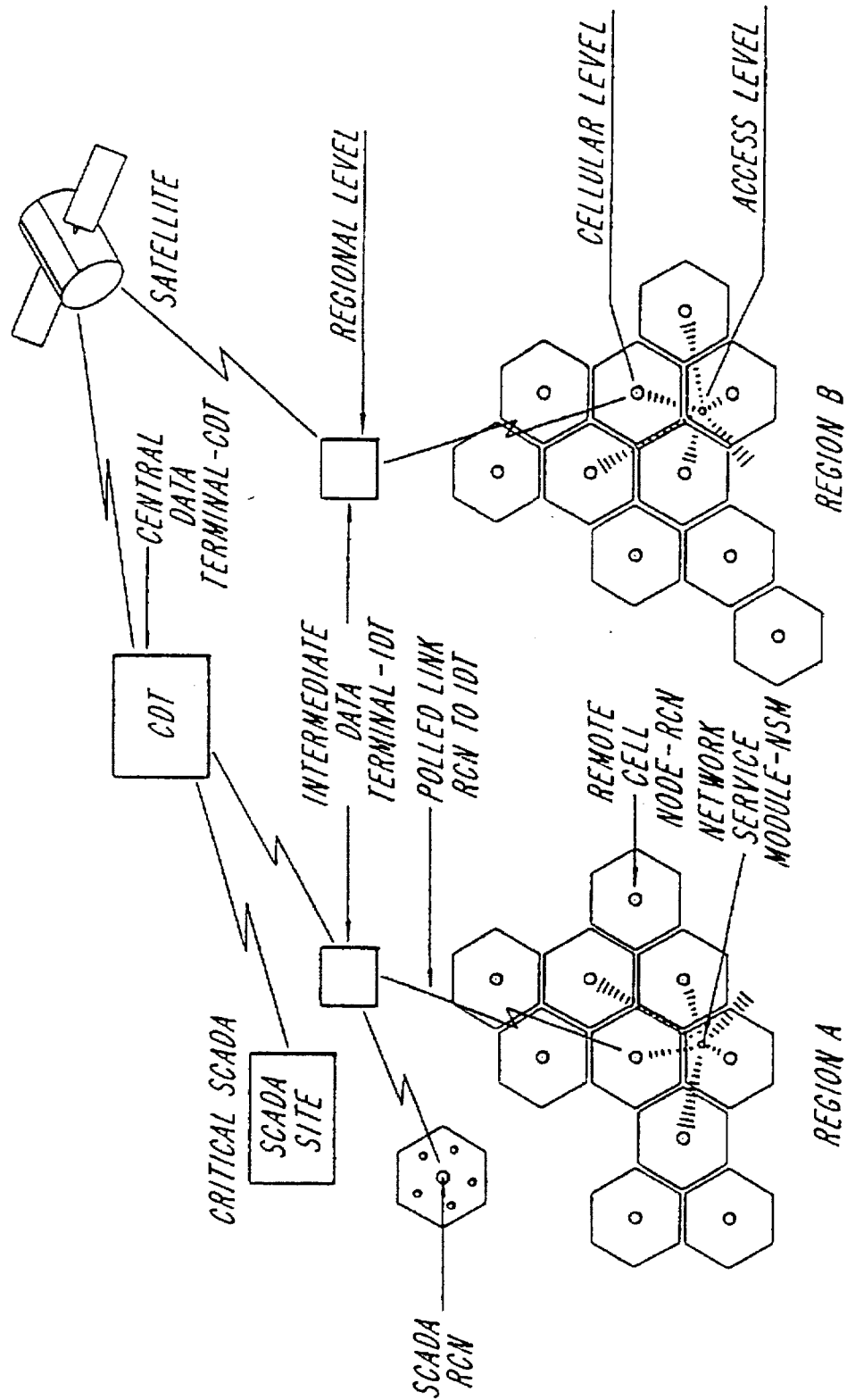


FIG. 12



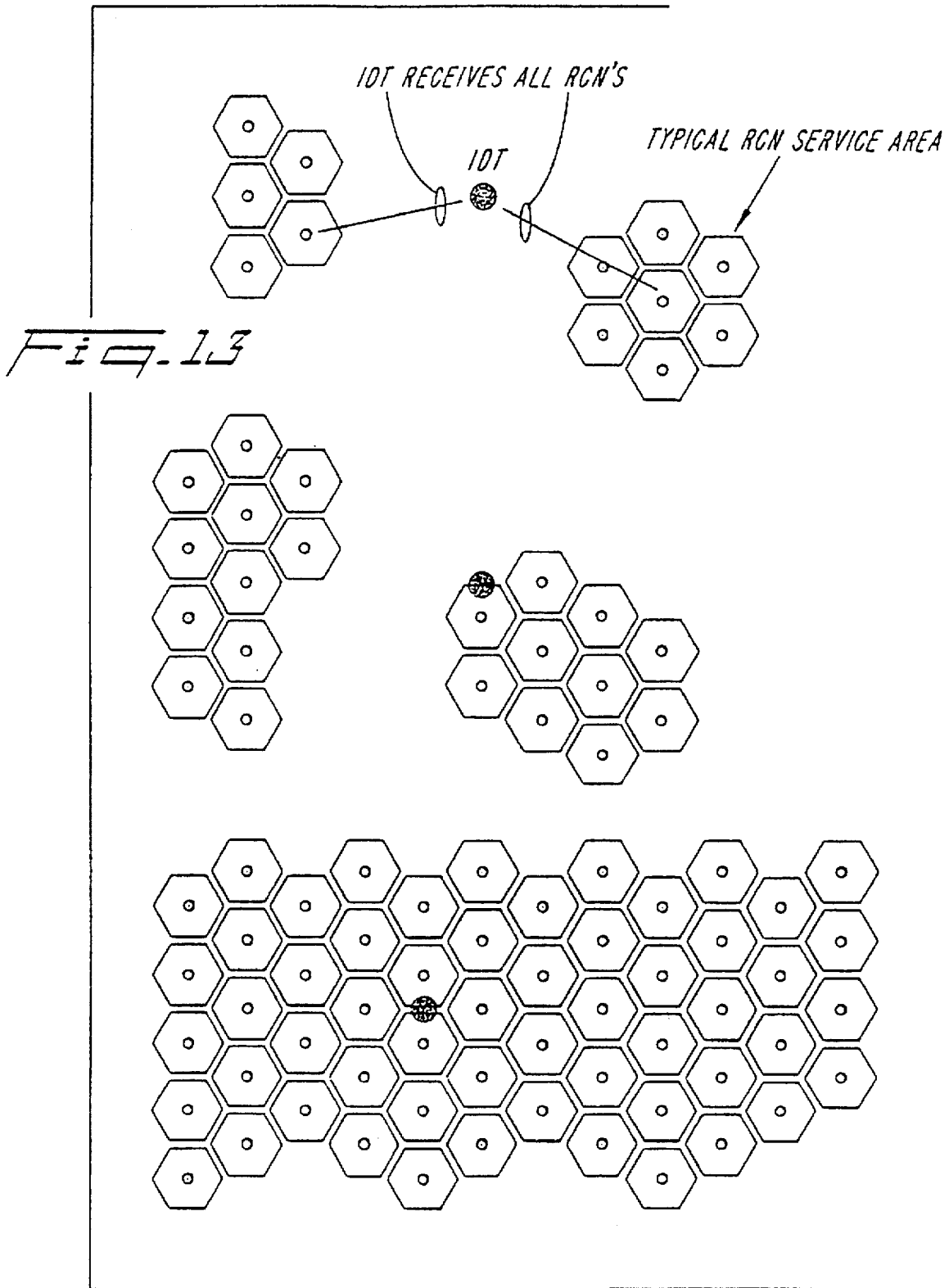


FIG. 14

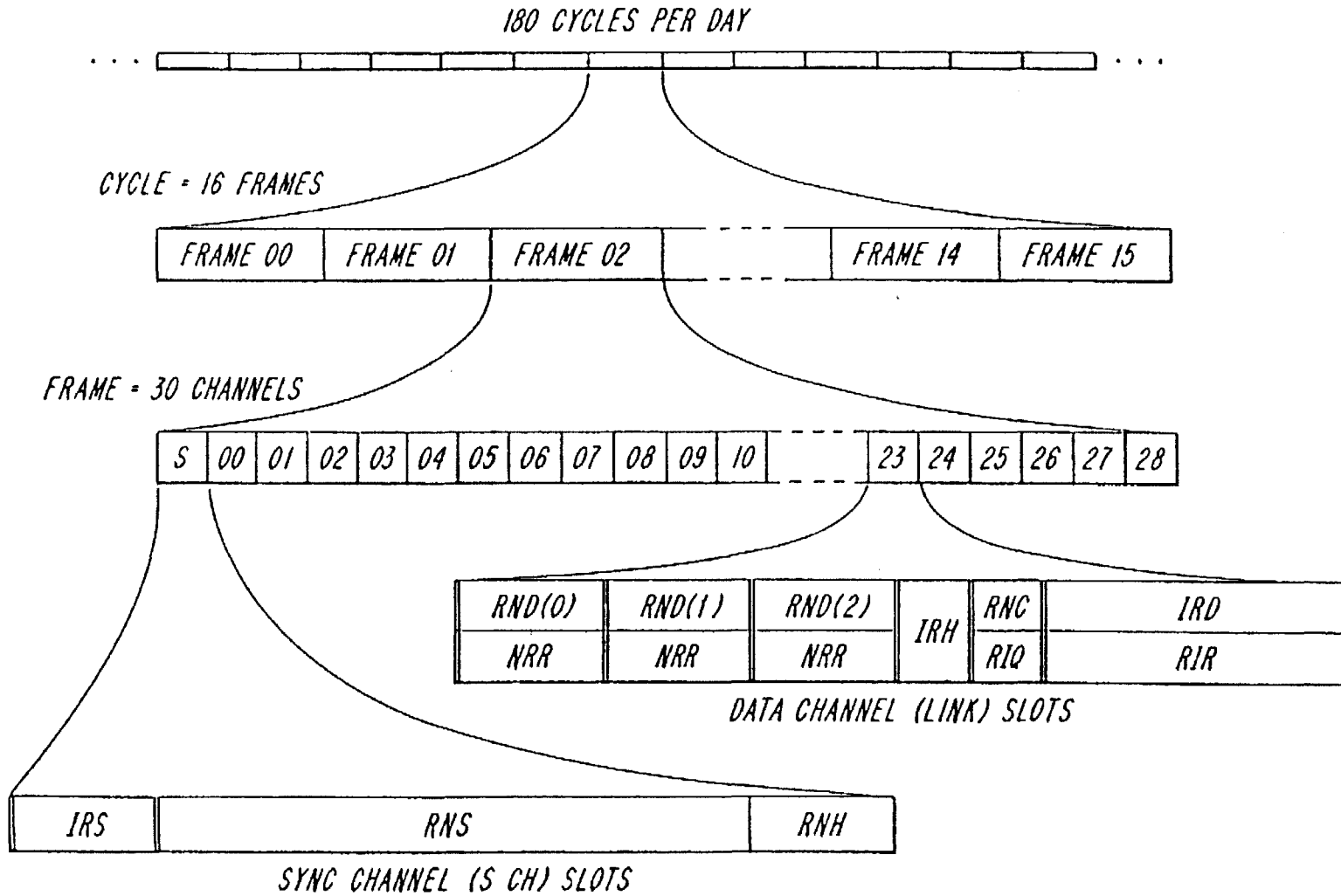


Fig. 15A

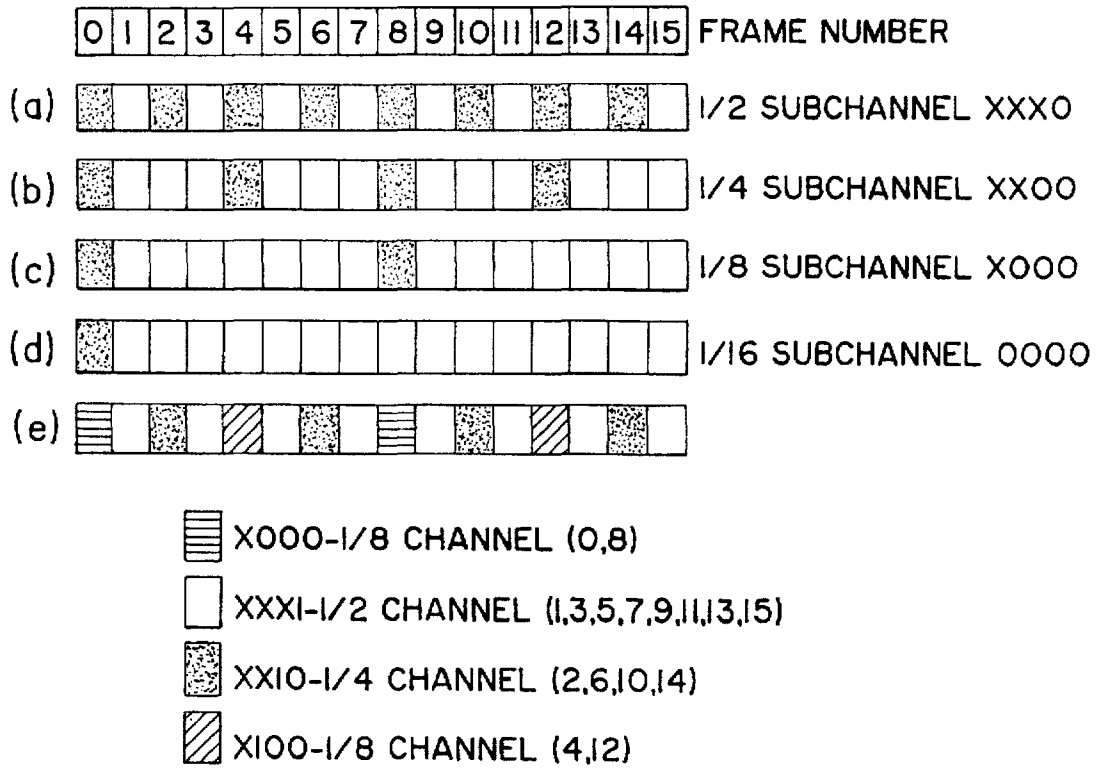
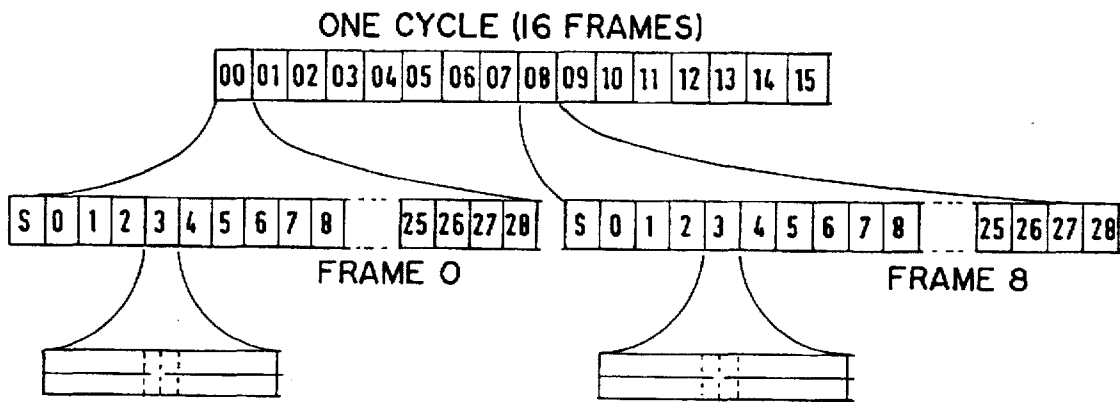


Fig. 15B



(f) THE TWO RND SLOTS PER CYCLE WHICH COMPRISE RND (0).3.XOOO

FIG. 16

LEADING GUARD	PREAMBLE	DATA LINK PACKET	TRAILING GUARD
			STOP BIT

FIG. 17

DATA CHANNEL SLOT DESCRIPTION										
SLOT		LEADING GUARD		PREAMBLE		DL PACKET				TRAILING GUARD
NAME	SIZE (ms)	Tx DELAY (ms)	Rx DELAY (ms)	Tx (ms)	Rx (min) (ms)	SIZE (ms)	BITS		÷	(ms)
							FLAG	PKT		
NRR	100	6±1	8	12	8	80	8	152	F	2±1
RND	100	8	7±1	0	0	90	6	84 ⁺	F	2
RNC	100	8	7±1	0	0	90	6	84 ⁺	F	2
IRH	100	8	8	10	8	80	8	152	F	2
RIQ	100	8	8	10	8	80	8	152	F	2
IRD	500	8	8	10	8	480	8	952	V	2
RIR	500	8	8	10	8	480	8	952	V	2

± - FIXED VS. VARIABLE LENGTH PACKET
⁺ - NOT A MULTIPLE OF 8

FIG. 18

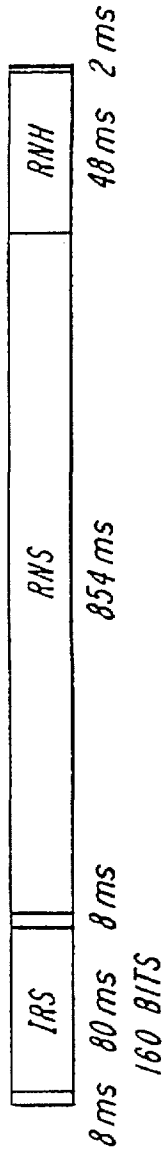


FIG. 19

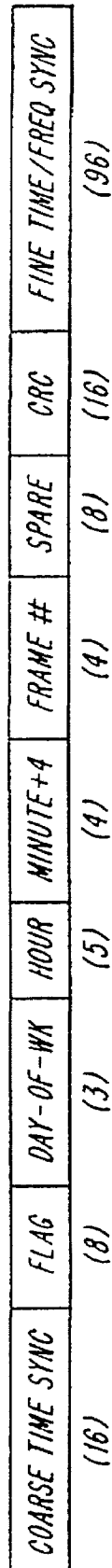


FIG. 20

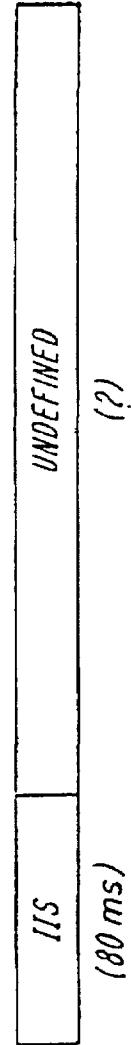


FIG. 21

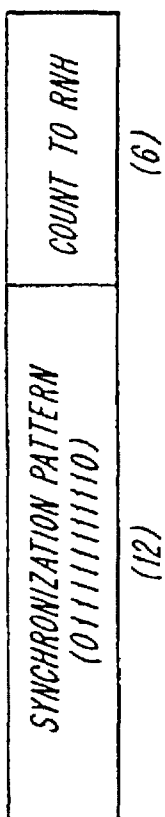


FIG. 22

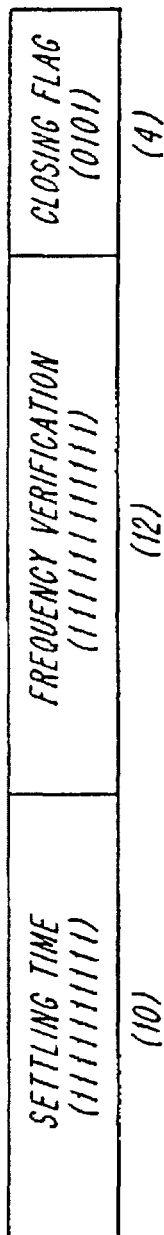


FIG. 23

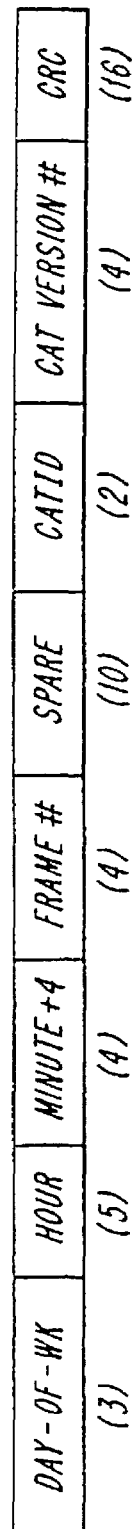


FIG. 24

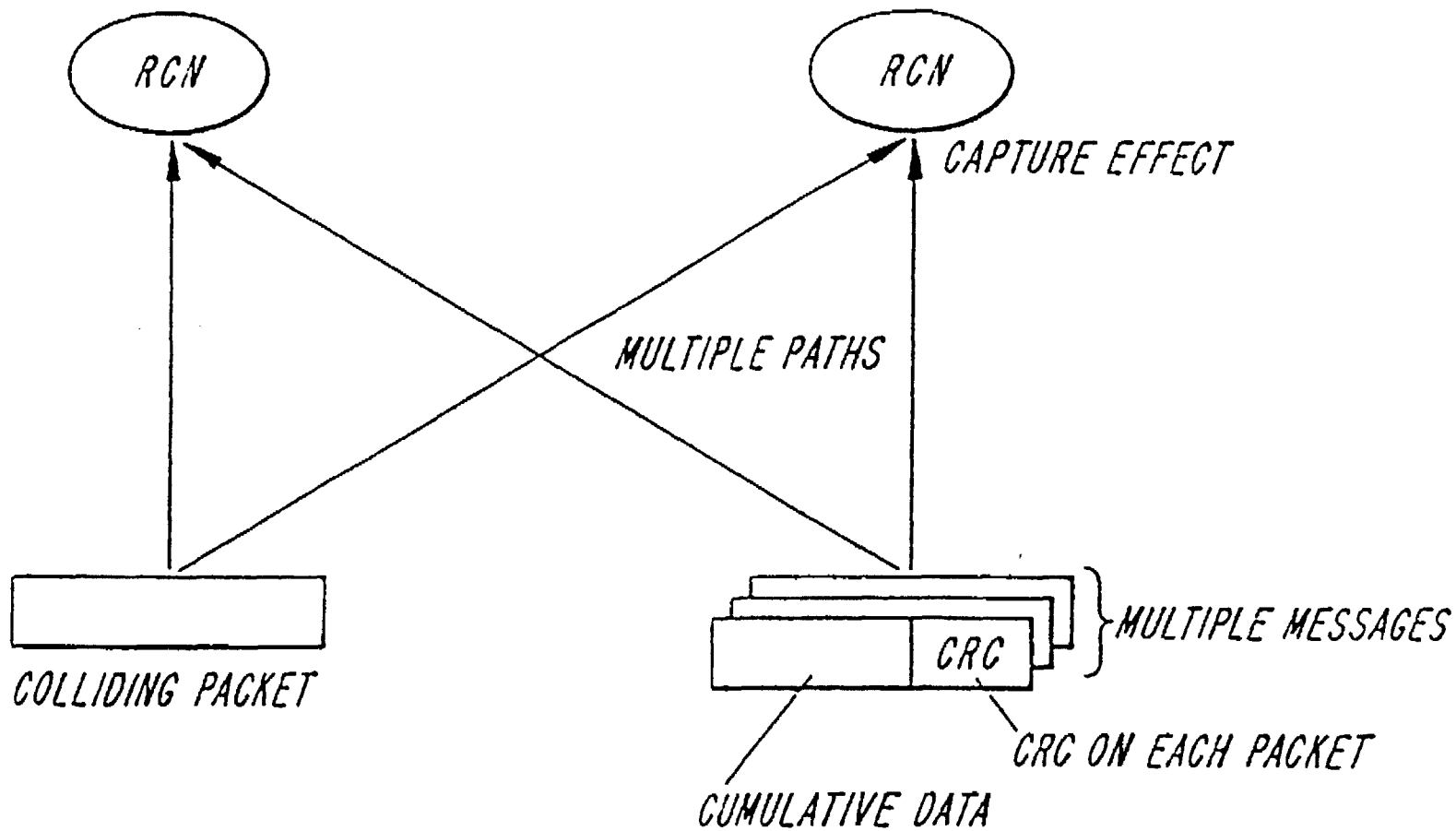


FIG. 25

FLAG	ADDRESS	CONTROL	INFORMATION	FCS	FLAG
(8)	(8)	(8/16)	(VARIABLE)	(16/32)	(8)
01111110	EXTENDABLE			CRC	01111110

FIG. 26

FLAG	LENGTH	NETWORK MESSAGE	CRC
(8)	(8)	(VARIABLE \leq 928)	(16)

FIG. 27

FLAG	RCNADR	CONTROL	LENGTH	NETWORK MESSAGE	CRC
(8)	(24)	(24)	(8)	(VARIABLE \leq 880)	(16)

FIG. 28

FLAG	RCNADR	CONTROL	NETWORK MESSAGE	CRC
(8)	(24)	(8)	(80)	(16)

FIG. 29

FLAG	ADDRESS (NSMTYPE + NSMADR)	CONTROL	NETWORK MESSAGE	CRC
(8)	(40)	(16)	(80)	(16)

FIG. 30

FLAG	RCNADR	CONTROL	NETWORK MESSAGE	CRC
(8)	(24)	(24)	(24)	(16)

FIG. 31

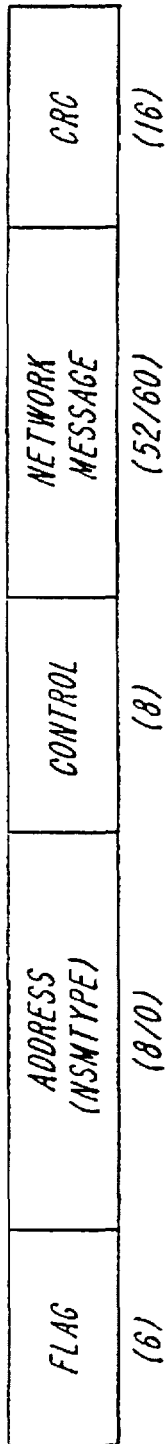


FIG. 32

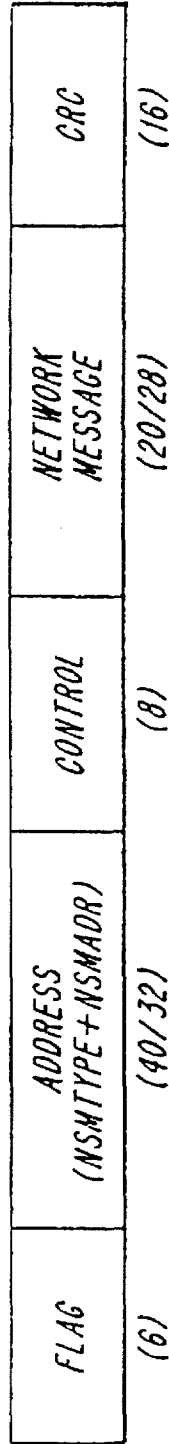
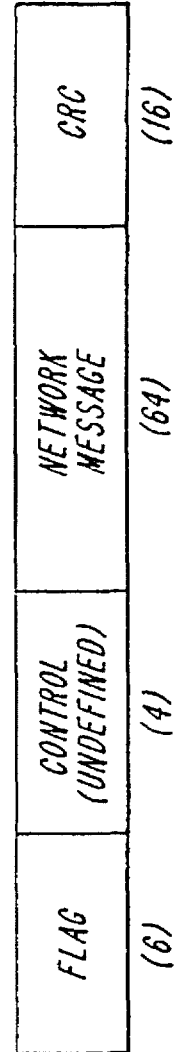


FIG. 33



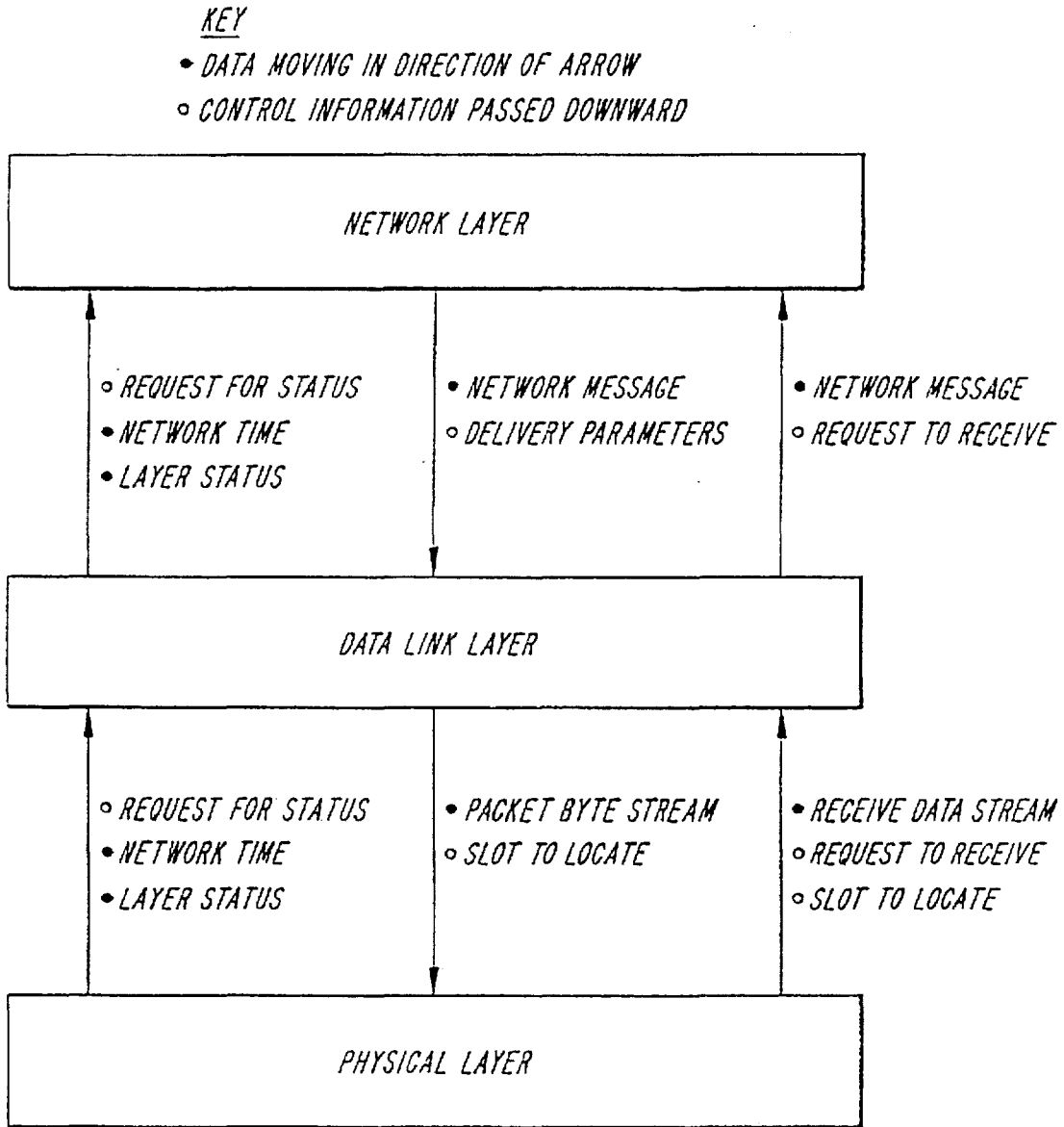


FIG. 34

<i>NODE</i>	<i>NO. OF ADDR BITS</i>	<i>SIZE OF ADDR SPACE</i>
<i>NSM</i>	32	4 BILLION
<i>RCN</i>	24	16 MILLION
<i>IDT</i>	16	64 K

FIG. 35

0000 1000 0000 0000 0000 0011 TIER ADDRESS PATTERN
 0000 1011 0000 0000 0000 1111 TIER ADDRESS MASK

XXXX 1X00 XXXX XXXX XXXX 0011 LOGICAL TIER ADDRESS
--

1101 1100 0010 1111 1001 0011 E.G. SELECTED ADDRESS
 1101 1100 0010 1111 1001 0101 E.G. NON-SELECTED ADDRESS

FIG. 36

FIG. 37

COMMON FIELDS OF BROADCAST MESSAGES	
MESSAGE TYPE	CODE INDICATING FORMAT AND CONTENT OF REST OF MESSAGE, MAY BE CONSIDERED TO BE A COMMAND OPCODE
NSM TYPE	RESTRICTS MESSAGE RECIPIENTS TO PARTICULAR CLASS OF NSM, MAY BE OMITTED IF SUBCHANNEL IS DEDICATED TO ONE NSM TYPE
VERSION SEQUENCE NUMBER	EACH TIME ANY MESSAGES IN THE CIRCULATION LIST CHANGES, THIS NUMBER IS INCREMENTED (MODULO ??); NSMs CAN ABORT SEARCH FOR MESSAGES IF PERFORMED PREVIOUSLY
LIST POSITION	POSITION OF CURRENT MESSAGE IN COMPLETE CIRCULATION LIST, TOGETHER WITH LIST LENGTH, MAY BE USED TO OPTIMIZE SEARCH
LIST LENGTH	NUMBER OF MESSAGES IN CIRCULATION LIST

FIG. 38

SUBCHANNEL SIZE	1/16, 1 RND SLOTS PER CHANNEL
LIST LENGTH	68 MSGS (4 RATE CLASSES, 17 MSGS EACH)
CIRCULATION TIME	9 HOURS 4 MINUTES

FIG. 39A

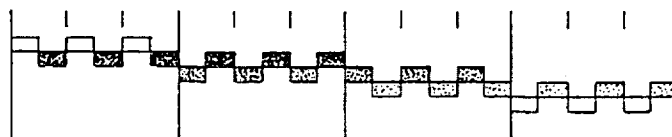


FIG. 39B

SUBCHANNEL SIZE	1/4, 3 RND SLOTS PER CHANNEL
LIST LENGTH	8 MSGS
CIRCULATION TIME	5 MINUTES, 20 SECONDS (AVG)
MSG LIFESPAN	6 TRANSMISSIONS OVER 32 MINUTES
LIST LIFESPAN	50% TURNOVER (4 MSGS) EVERY 16 MINUTES
MAX THRUPUT	360 MSGS/DAY OR 15 MSGS/HR

FIG. 40

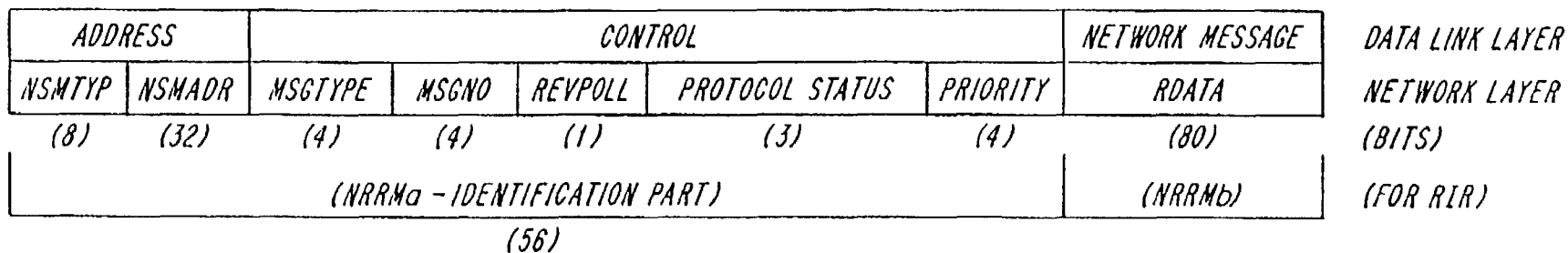


FIG. 42

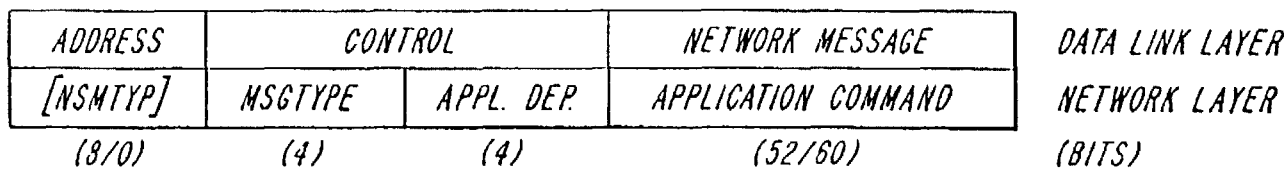


FIG. 43

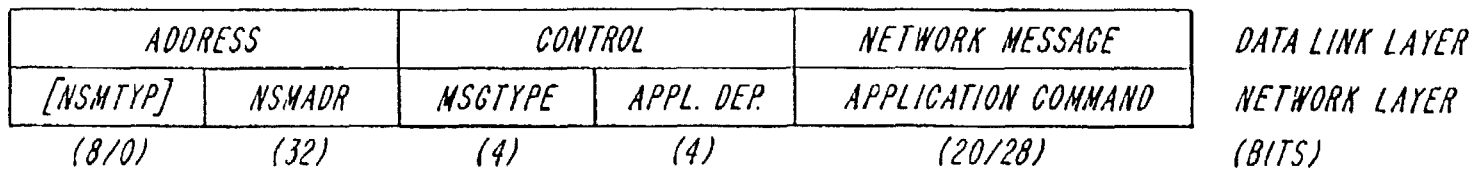


FIG. 41

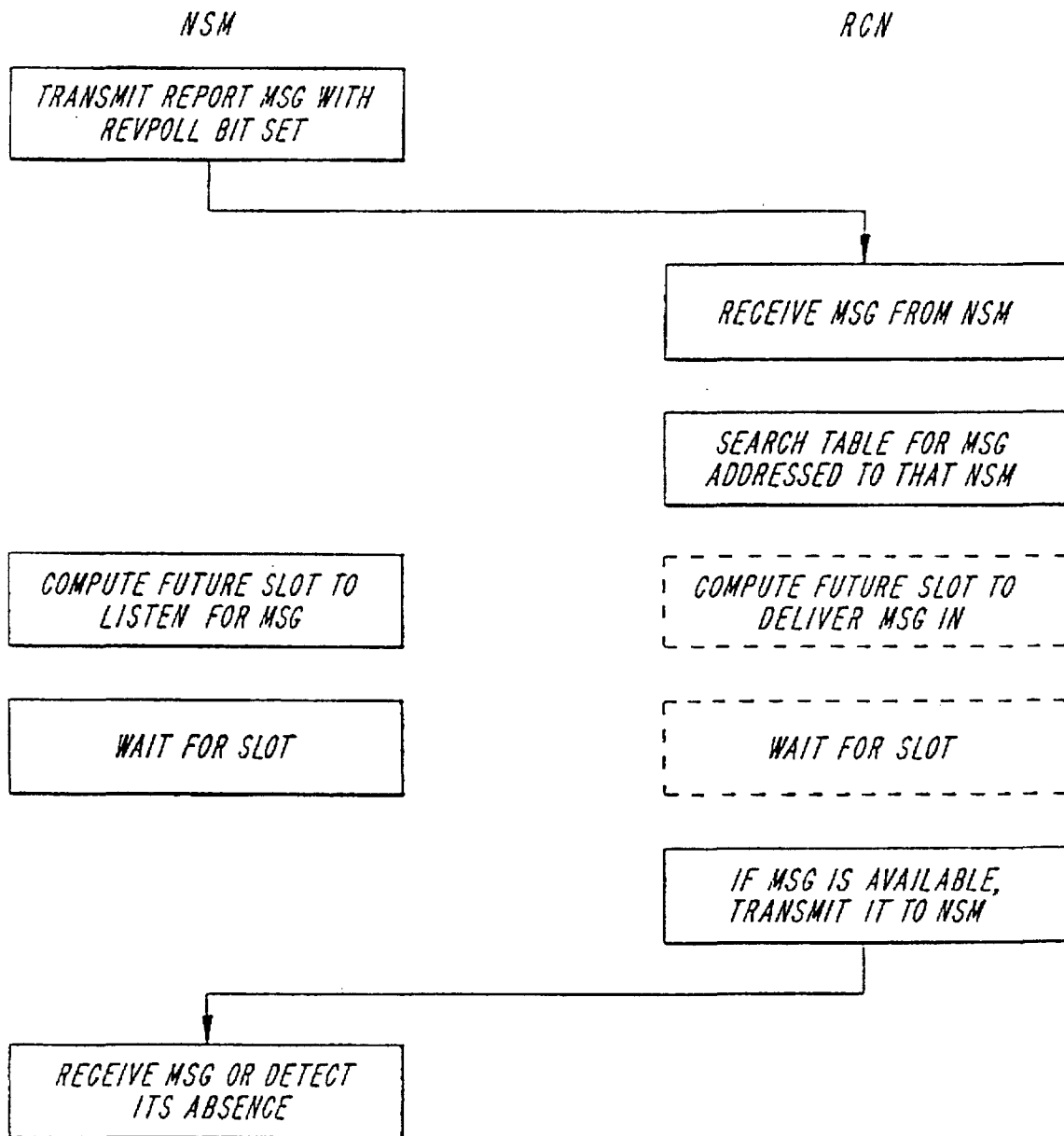


FIG. 44

<i>ADDRESS</i>	<i>CONTROL</i>			<i>NETWORK MESSAGE</i>				<i>DATA LINK LAYER</i>
<i>NSMTYP</i>	<i>APPLTYP</i>	<i>CAT ID</i>	<i>UNUSED</i>	<i>BITMAP</i>	<i>B/C SUBCHAN</i>	<i>REV POLL SUBCHAN</i>	<i>CAT VERS</i>	<i>NETWORK LAYER</i>
(8)	(4)	(2)	(2)	(28)	(10)	(10)	(4)	(BITS)

FIG. 45

<i>RND SLOT</i>	<i>CHANNEL</i>	<i>SUBCHANNEL SIZE/LOCATION</i>
<i>ss: 0, 1, 2</i> <i>3 ⇒ ALL</i>	<i>ccc: 0..7</i>	<i>0 yyyy RND[ss].ccc.yyyy</i> <i>10 yyy RND[ss].ccc.xyyy</i> <i>110 yy RND[ss].ccc.xxyy</i> <i>1110 y RND[ss].ccc.xxyy</i> <i>11110 RND[ss].ccc.xxxx</i>
<i>ss = 3</i>	<i>ccc = 7</i>	<i>11111 NO SUBCHANNEL ASSIGNED</i>
(2)	(3)	(5)

(BITS)

FIG. 46

<i>RCNADR</i>	<i>CONTROL</i>	<i>LENGTH</i>	<i>NETWORK MESSAGE</i>				<i>DATA LINK LAYER</i>
			<i>NSM MSG IDs</i>	<i>INTERMEDIATE CRC</i>	<i>RCN STATUS</i>	<i>NSM MSG CONTENTS</i>	<i>NETWORK LAYER</i>
(24)	(24)	(8)	(56 X N ITEMS)	(16)	(24)	(VARIABLE)	(BITS)

FIG. 47

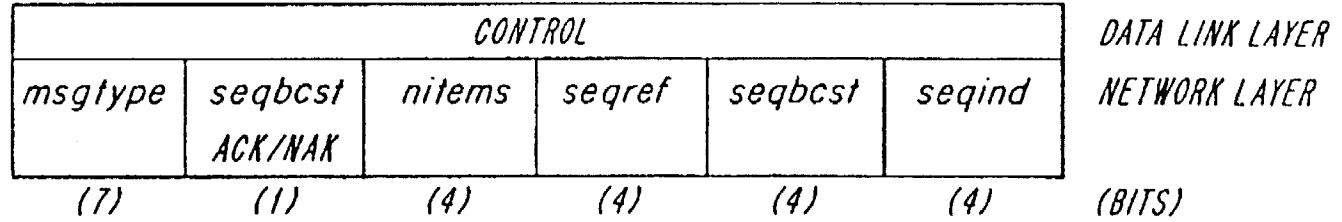
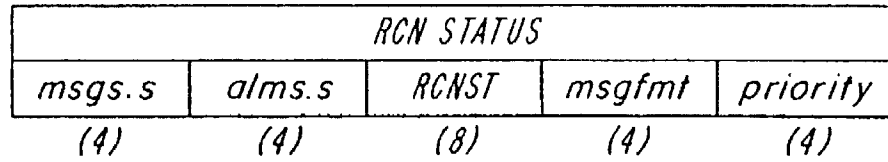


FIG. 48



MAXIMUM NUMBER OF NSM MSGS PER RIR			
<i>msgfmt</i>	<i>tags</i>	<i>NSM msg size (bits)</i>	<i># NSM msgs/report</i>
0	<i>p, v</i>	144	5
1	<i>p, v, f, s, t</i>	176	4
2	<i>p, v, f, d, t</i>	176	4
3	<i>p, v, f, s, d, t, c, crc</i>	200	4
4	<i>p, v, f, s, d, t</i>	184	4
5	<i>p, v, d, t</i>	168	5

FIG. 49

FIG. 50

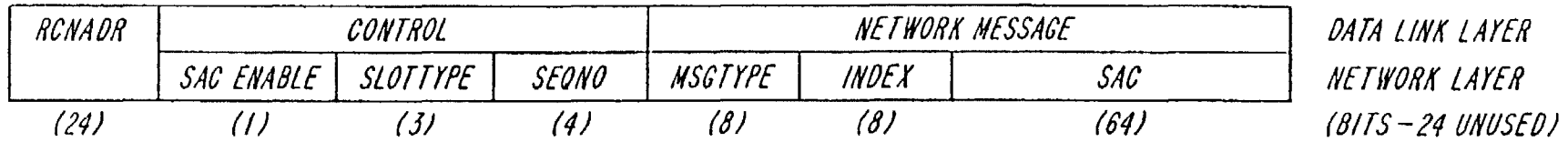


FIG. 51

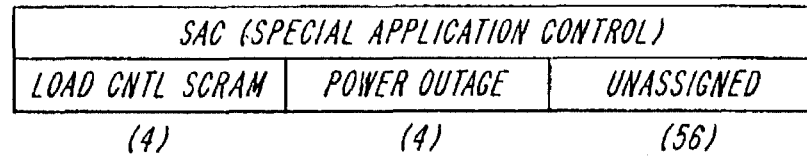


FIG. 52

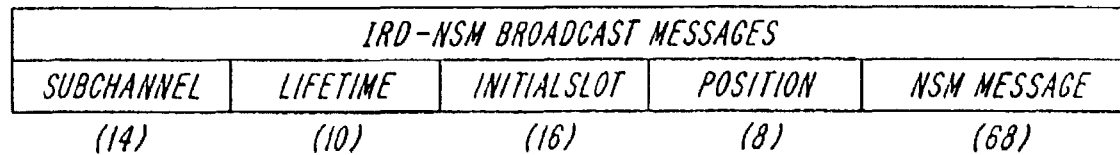


FIG. 53

SUBCHANNEL				LIFETIME		INITIALSLOT		
FRAME #	MASK	CHANNEL	SLOT	UNITS	LENGTH	DAY-OF-WEEK	CYCLE #	FRAME #
(4)	(4)	(3)	(3)	(2)	(8)	(4)	(8)	(4)

FIG. 54

IRD-NSM REVERSE POLL MESSAGES						
4 RCNADRS	SUBCHANNEL	LIFETIME	PARMS	CRITERIA	NSM MESSAGE	UNUSED
(96)	(14)	(10)	(8)	(16)	(68)	(4)

FIG. 55

REPETITIONS	HASHPARMS			USE CRITERIA
	HASH ALGORITHM	PRIORITY	CHAINING	
(2)	(3)	(1)	(1)	(1)

FIG. 56

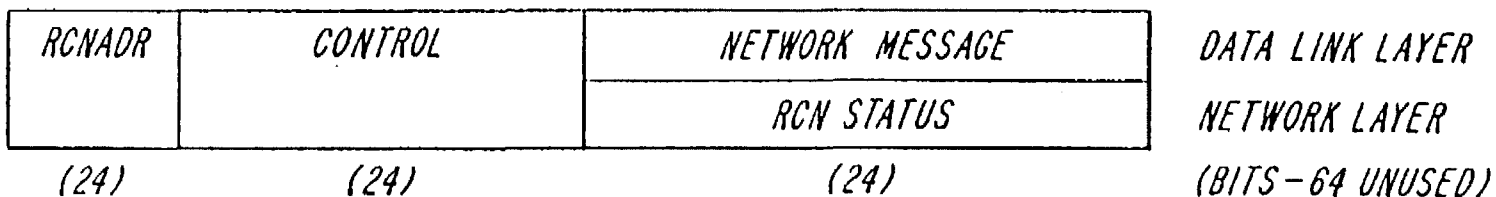
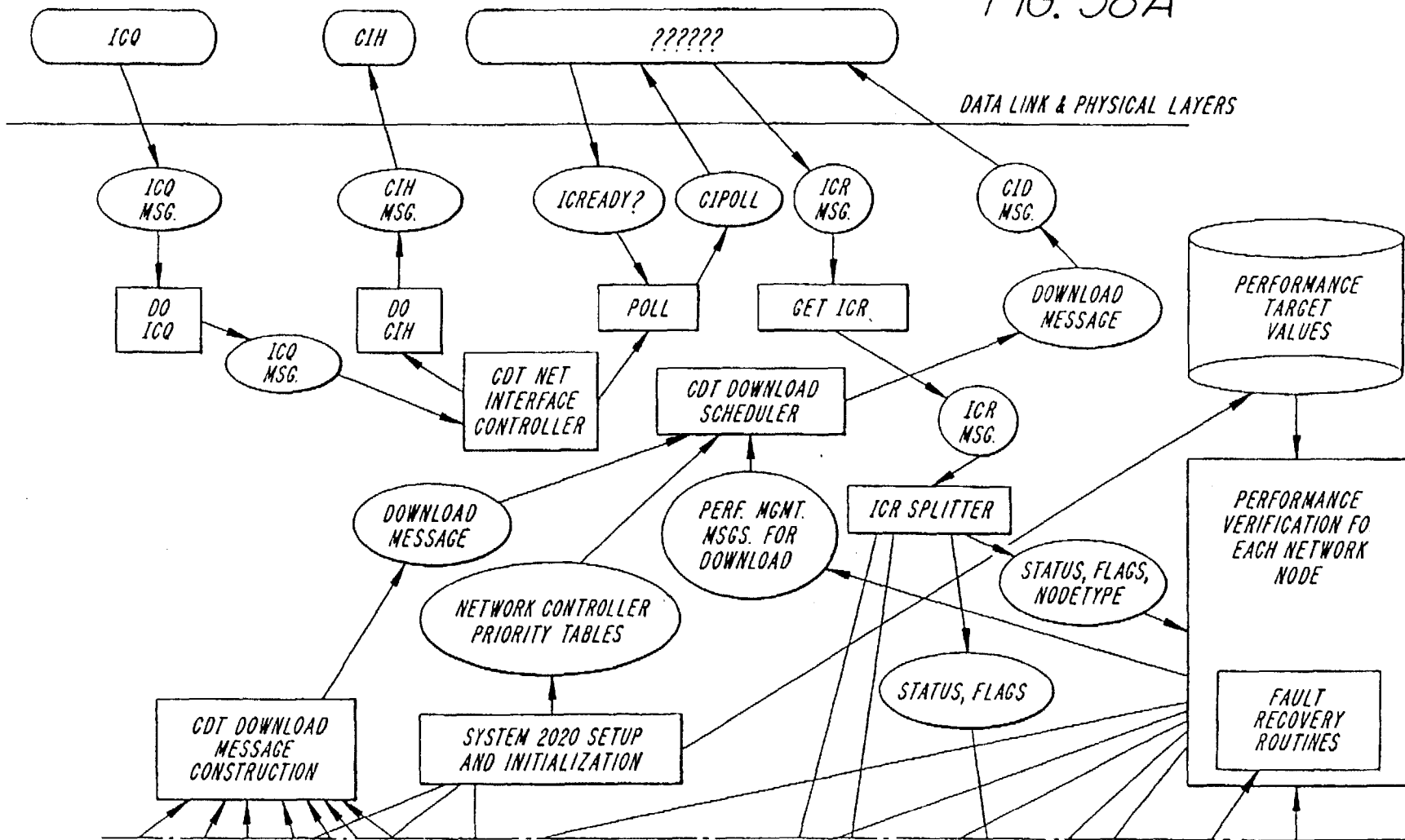


FIG. 57

<i>NSM MESSAGE PRIORITY</i>		
<i>PRIORITY FIELD</i>	<i>ACTUAL PRIORITY</i>	<i>COMMENT</i>
<i>0</i>	<i>NSM-LOW</i>	<i>· ACTUAL PRIORITY ASSIGNED AT RCN</i>
<i>15</i>	<i>NSM-HIGH</i>	<i>· ACTUAL PRIORITY ASSIGNED AT RCN</i>
<i>1..14</i>	<i>1..14</i>	<i>· EXPLICIT PRIORITY INCLUDED IN MESSAGE</i>

FIG. 58A



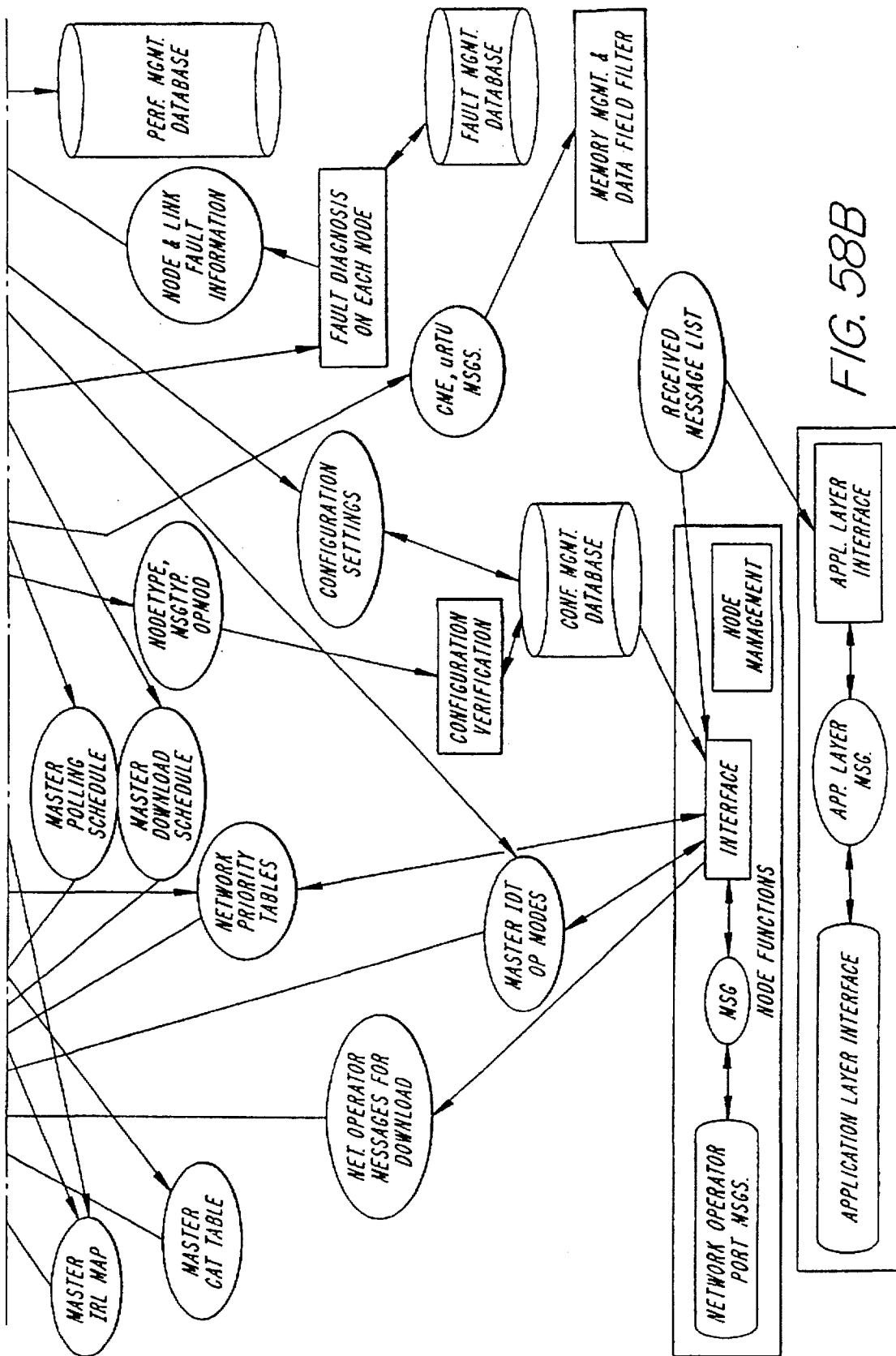
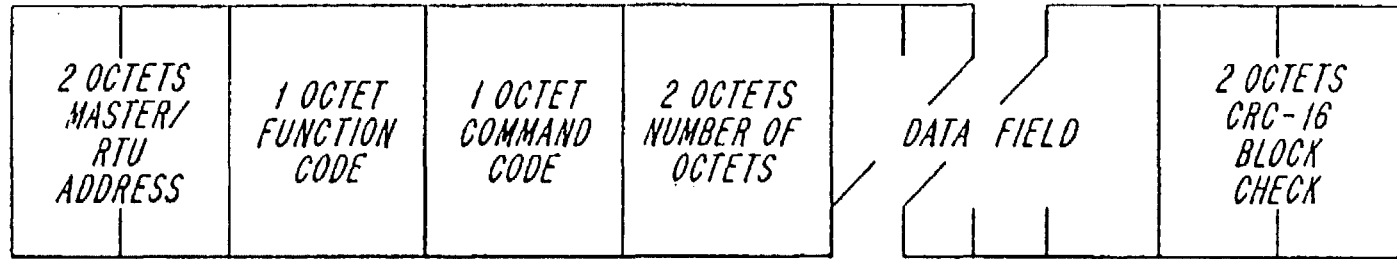


FIG. 58B

FIG. 59

2.4 MASTER MESSAGE FORMAT(S)

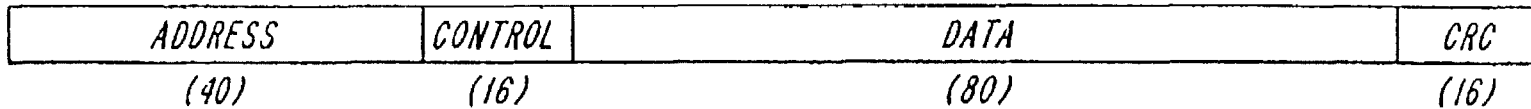


CURRENT SYSTEM 2020 MESSAGES HAVE THE FORMAT :

INDIVIDUALLY ADDRESSED
NSM MESSAGE



NSM REPORT MESSAGE

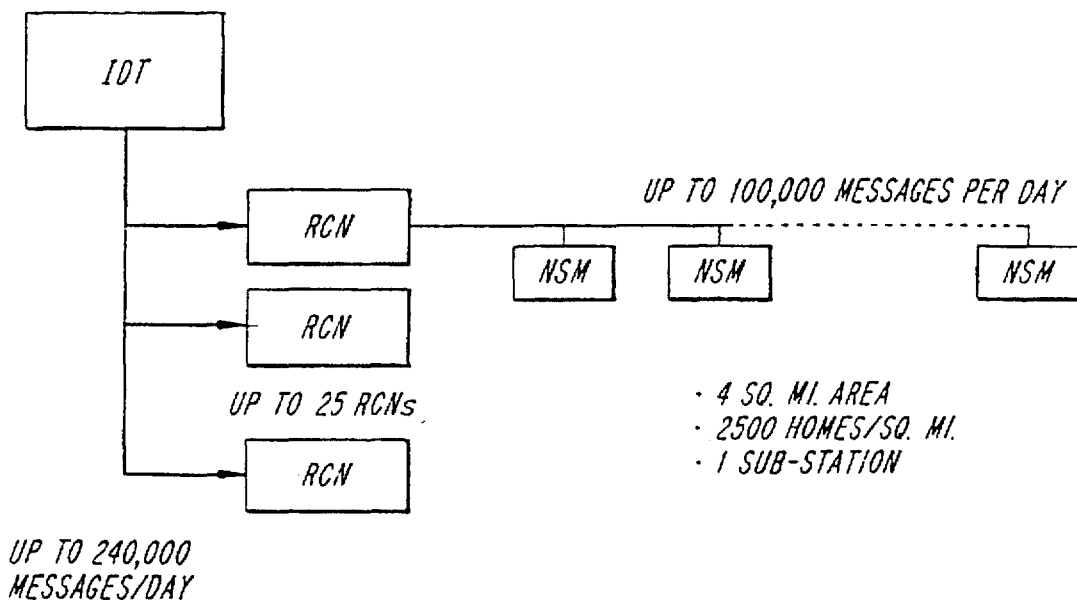


SELECT BEFORE OPERATE SEQUENCE

FIG. 60

	BASIC SYSTEM SERVICES			DSM		DA SERVICES		CBM
	AMR OR TOU	LOAD SURVEY	LOAD CONTROL	INDUSTRIAL METERING	LOAD MANAGEMENT	LMM	FMM	
NUMBER OF FEEDERS	10	10	10	10	10	10	10	10
NUMBER OF NSM'S	10,000	100	6000	100	50	30	100	40
REPORTS/DAY/NSM	3	32	1	48	6	96	FAULT DEPEND. EG. 2X	2
REPORT MESSAGES/DAY	30,000	3200	6000	4800	300	2880	200	80
ALARM MSGS/DAY								
BROADCAST MESSAGES/DAY			60		24			
POLL MESSAGES/DAY/NSM						2	0.1	
CONTROL MESSAGES/DAY/NSM		1	1/30		6			2
RESPONSE TIME		INDIRECT	INDIRECT			10 SEC	10 SEC	10-30 SEC
DOWNSTR. MESSAGES/DAY		100	200		300	60	10	80
TOTAL MESSAGES/DAY	30,000	3300	6200	4800	600	3000	210	160
TOTAL 39,500 MESSAGES/DAY				TOTAL 5400 DSM MSGS/DAY		TOTAL 3450 DA MSGS/DAY		

FIG. 61



<u>DIRECTION</u>	<u>MESSAGE TYPE</u>	<u>#NSM's</u>	<u>REPORTS/ DAY/NSM</u>	<u>MESSAGES/ DAY</u>
UPSTREAM	METER READING REPORT	10,000	3	30,000
	LOAD SURVEY REPORT	100	32	3200
	LCM REPORT	2000	1	2000
	VOLTAGE MONITOR	60	24	1440
	FAULT MONITOR ALARMS	320	2	640
	SUB-TOTAL		12,480	
DOWNSTREAM	TOU SCHEDULE			96
	LOAD CONTROL			48
	LOAD CURTAIL			48
	ENABLE/DISABLE LCM	100		100
	ENABLE/DISABLE LSURVEY	100		100
	SUB-TOTAL			
TOTAL				<u>37,672</u>

FIG. 62

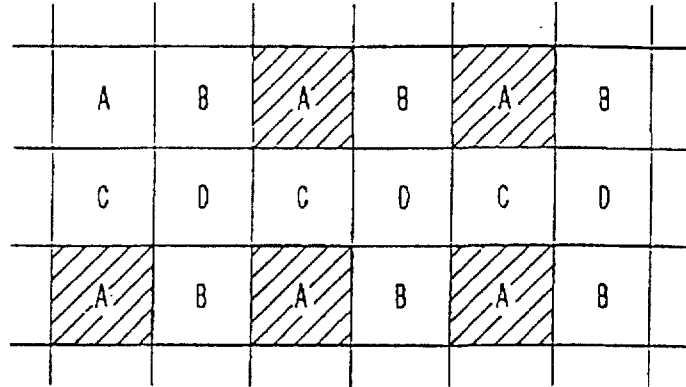


FIG. 63

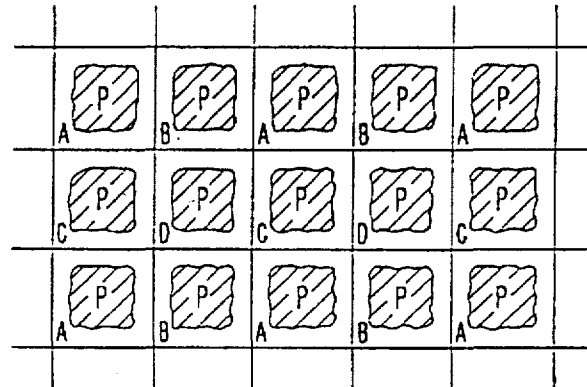


FIG. 64

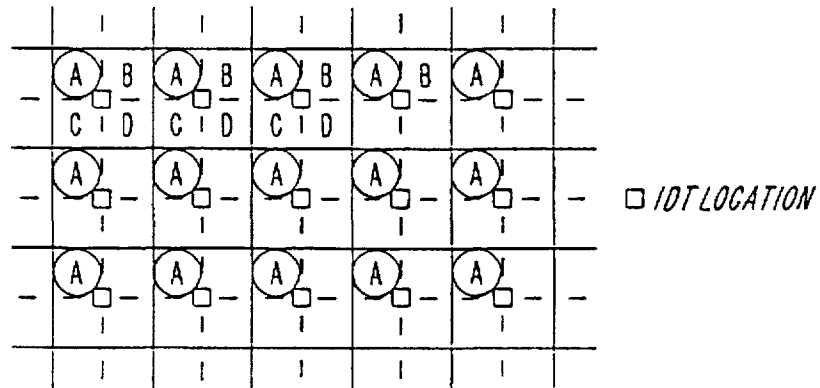
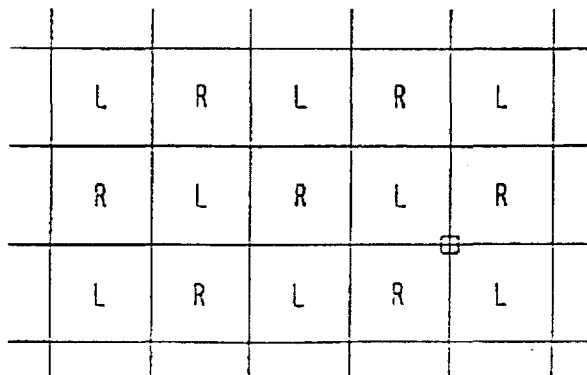


FIG. 65



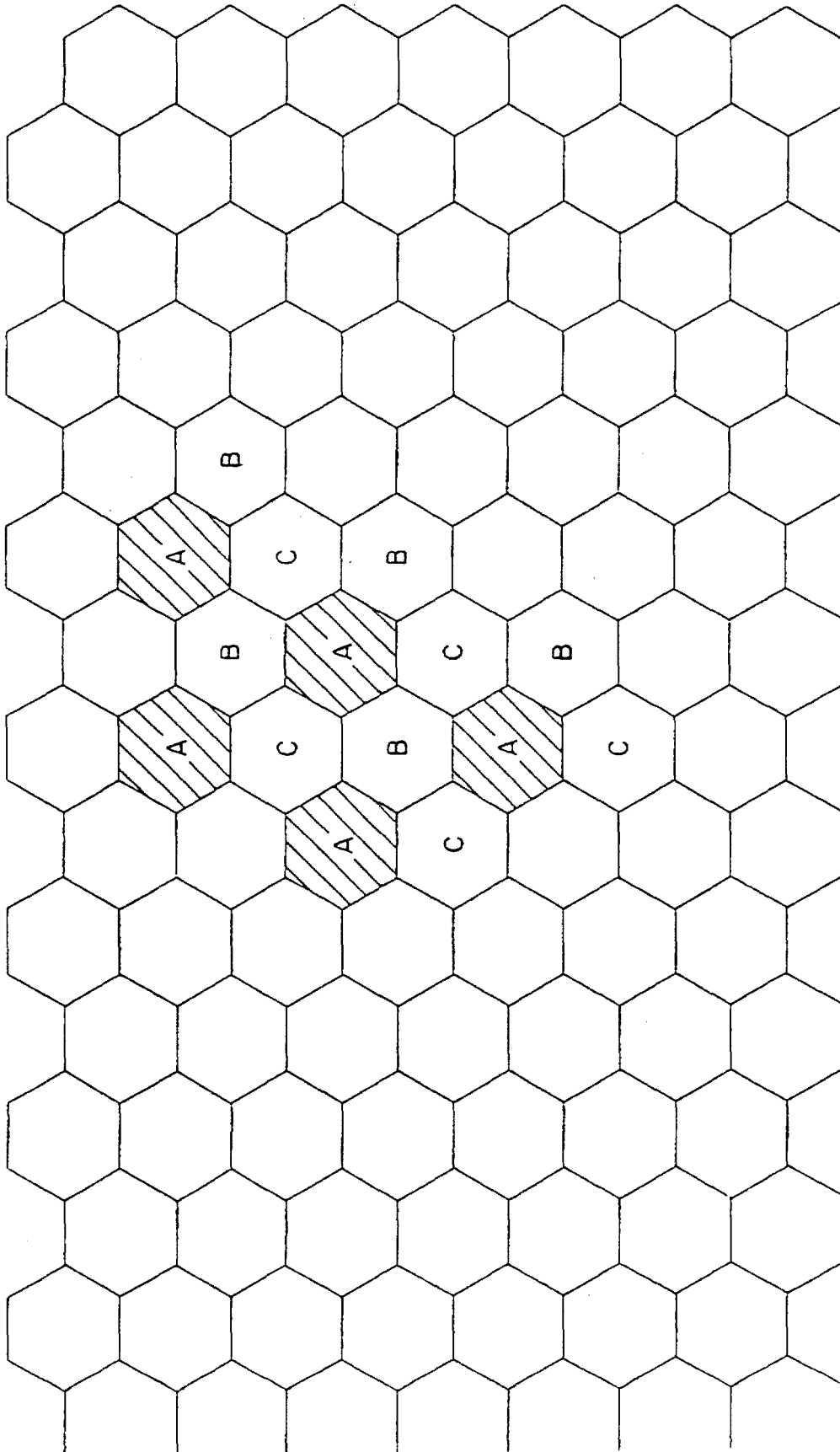


FIG. 66

Fig. 67

	SDMP	MODIFIED ADMP	DDMP
<u>COMMUNICATION RELIABILITY ISSUES</u>			
RESISTANCE TO ADJACENT AREA INTERFERENCE	HIGHEST	MEDIUM	HIGH
IMMUNITY TO WEATHER	HIGH	HIGH	HIGH
EASE OF ADJACENT AREA COVERAGE (IN EVENT OF IDT FAILURE)	MEDIUM	LOW	HIGH
TOPOLOGICAL FLEXIBILITY	LOW	MEDIUM	HIGH
<u>PERFORMANCE ISSUES</u>			
RESPONSE TIME	DETERMINISTIC	DETERMINISTIC	DETERMINISTIC
POLLING EFFICIENCY	0.25	67%	>90%
EAVESDROPPING EFFICIENCY	-	-	-
ALARM REPORTING DELAY	0	0	UP TO 4 SEC.
EASE OF HANDLING DA. TRAFFIC	HIGH	LOW	MEDIUM
<u>COST ISSUES</u>			
HARDWARE DEVELOPMENT COST	LOW	LOW	HIGH
F/W & S/W DEVELOPMENT COST	LOW	HIGH	MEDIUM
HARDWARE FAB COST	LOW	MEDIUM	HIGH
INSTALLATION COST	LOW	LOW	MEDIUM

Fig. 68A



	RND	RND	RND	SCRAM	IRD
IRH	NRR	NRR	NRR	RIQ	RIR

STANDARD CHANNEL

Fig. 68B

	DA	RND	RND	SCRAM	IRD
IRH		NRR	NRR	RIQ	RIR

DA CHANNEL

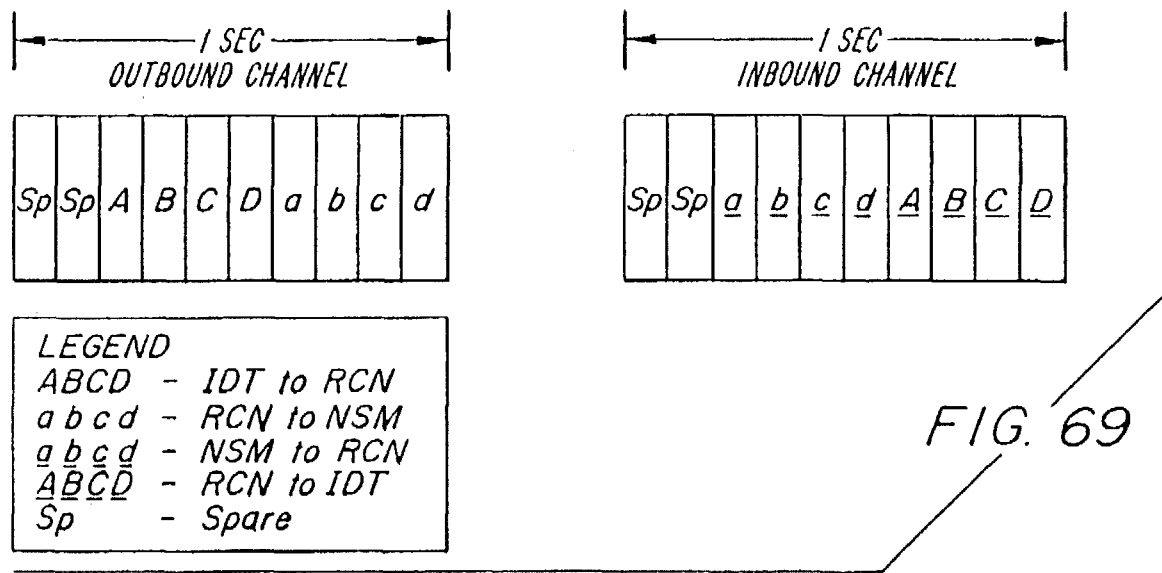
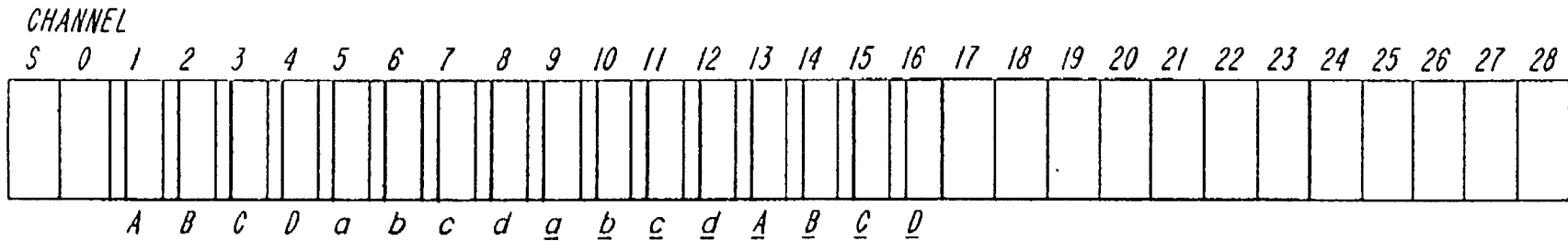


FIG. 70

POLLING METHOD	DA CHANNEL CONFIGURATION	D/A MESSAGES PER HOUR	DA MESSAGES PER DAY	DA RESPONSE TIME	REPORT/CONTROL MESSAGES/DAY
ADMP	4 R CHANNELS/FRAME	240	5260	15 SEC.	120000
ADMP	16 D/A SLOTS/FRAME	120	2880	30 SEC.	140000
SDMP	4 R CHANNELS/FRAME	240	5260	15 SEC.	45000
DDMP	4 R CHANNELS/FRAME	960* 240**	5260	15 SEC.	175000

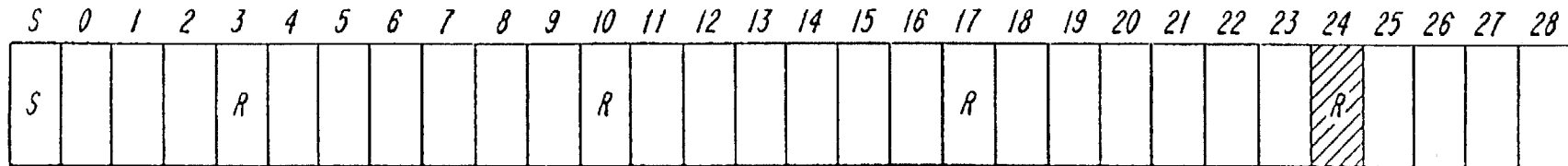
* THEORETICAL
** PRACTICAL



REPORT MESSAGE CAPACITY =
 29 CH/FRAME X 5 SLOTS/CH X 120 FRAMES/HR X .647 POLLING EFF. = 5829 MSG/HR

DA MESSAGE CAPACITY =
 1 MESSAGE/FRAME X 120 FRAMES/HR X 1.0 POLLING EFF. = 120 MSG/HR

FIG. 71



REPORT MESSAGE CAPACITY =
 25 CH/FRAME X 5 SLOTS/CH X 120 FR/HR X .67 POLLING EFF. X .5 EAVESDROPPING EFF. = 5025 MSG/HR = 120,000

DA MESSAGE CAPACITY =
 2 CH/FRAME X 120 FR/HR X 1.0 POLL. EFF. = 240 MSG/HR WITH 15 SECOND RESPONSE TIME

FIG. 72

5,673,252

1

COMMUNICATIONS PROTOCOL FOR REMOTE DATA GENERATING STATIONS

RELATED PATENTS

This application is a continuation of application Ser. No. 08/247,988 filed on Mar. 23, 1994 now abandoned, which is a continuation-in-part patent application of a patent application entitled, HIERARCHIAL RADIO COMMUNICATION NETWORK FOR REMOTE DATA GENERATING STATIONS, having Ser. No. 08/124,495 now abandoned, filing date of Sep. 22, 1993, which was a file wrapper continuation (FWC) patent application of a patent application entitled, RADIO COMMUNICATION NETWORK FOR REMOTE DATA GENERATING STATIONS, having Ser. No. 07/732,183 and filing date Jul. 19, 1991, which is a continuation-in-part patent application of a patent application entitled, RADIO COMMUNICATION NETWORK FOR REMOTE DATA GENERATING STATIONS, having Ser. No. 07/480,573 and filing date of Feb. 15, 1990 and is now U.S. Pat. No. 5,056,107. The benefit of the earlier filing dates of the parent patent applications is claimed pursuant to 35 U.S.C. §120.

BACKGROUND OF THE INVENTION

This invention relates to a protocol for collecting data from remote data generating stations in a communications network, and more particularly a radio based system for sending data from a plurality of network service modules, with each network service module attached to a meter, and communicating through remote cell nodes and through intermediate data terminals, to a central data terminal, i.e., a headend.

DESCRIPTION OF THE RELEVANT ART

Many attempts have been made in recent years to develop an automatic meter reading system for utility meters such as used for electricity, gas and water, which avoids meter reading personnel having to inspect and physically note the meter readings. There are of course many reasons for attempting to develop a system of this type.

Most of the prior art systems have achieved little success. The system which has achieved some success or is most widely used has an automatic meter reading unit mounted on an existing meter at the usage site and includes a relatively small transmitter and receiver unit of very short range. The unit is polled on a regular basis by a travelling reading unit which is carried around the various locations on a suitable vehicle. The travelling reading unit polls each automatic meter reading unit in turn to obtain stored data. This approach is of limited value in that it requires transporting the equipment around the various locations and hence only very infrequent, for example monthly, readings can be made. The approach avoids a meter reader person actually entering the premises to physically inspect the meter which is of itself of some value but only limited value.

Alternative proposals in which reading from a central location is carried out have been made but have achieved little success. One proposal involves an arrangement in which communication is carried out using the power transmission line of the electric utility. Communication is, therefore, carried out along the line and polls each remote reading unit in return. This device has encountered significant technical difficulties.

Another alternative attempted to use the pre-existing telephone lines for communication. The telephone line pro-

2

posal has a significant disadvantage since it must involve a number of other parties, in particular the telephone company, for implementing the system. The utility companies are reluctant to use a system which cannot be entirely controlled and managed by themselves.

A yet further system using radio communication has been developed by Data Beam, which was a subsidiary of Connecticut Natural Gas. This arrangement was developed approximately in 1986 and has subsequently received little attention and it is believed that no installations are presently operative. The system includes a meter reading device mounted on the meter with a transmitting antenna which is separate from the meter reading device. The transmitting antenna is located on the building or other part of the installation site which enables the antenna to transmit over a relatively large distance. The system uses a number of receiving units with each arranged to receive data from a large number of transmitters, in the range 10,000 to 30,000. The transmitters, in order to achieve maximum range, are positioned to some extent directionally or at least on a suitable position of the building to transmit to the intended receiving station. The arrangement leads to using a minimum number of receiving stations for optimum cost efficiency.

The separate transmitter antenna, however, generated significant installation problems due to wiring the antenna through the building to the transmitter and receiver. The anticipated high level of power used for transmitting involved very expensive battery systems or very expensive wiring. The proposal to reduce the excessive cost was to share the transmission unit with several utilities serving the building so that the cost of the transmitter could be spread, for example, between three utilities supplied to the building. Such installation requires separate utility companies to cooperate in the installation. While this might be highly desirable, such cooperation is difficult to achieve on a practical basis.

In order to avoid timing problems, the meter reading units were arranged to communicate on a random time basis. However, the very large number, up to 30,000 of meter reading units reporting to a single receiving station, leads to a very high number of possible collisions between the randomly transmitted signals. The system, therefore, as proposed, with daily or more often reporting signals could lose as many as 20% to 50% of the signals transmitted due to collisions or interference which leads to a very low efficiency data communication. The use of transmitters at the meter reading units which are of maximum power requires a larger interference protection radius between systems using the same allocated frequency.

An alternative radio transmission network is known as ALOHA. ALOHA has a number of broadcasting stations communicate with a single receiving station, with the broadcasting stations transmitting at random intervals. In the ALOHA system, collisions occur so that messages are lost. The solution to this problem is to monitor the retransmission of the information from the receiving station so that each broadcasting station is aware when its transmission has been lost. Each broadcasting station is then programmed to retransmit the lost information after a predetermined generally pseudorandom period of time. The ALOHA system requires retransmission of the information from the receiving station to take place substantially immediately and requires each broadcasting station to also have a receiving capability.

Cellular telephone networks are implemented on a wide scale. Cellular systems, however, use and allocate different

5,673,252

3

frequencies to different remote stations. While this is acceptable in a high margin use for voice communications, the costs and complications cannot be accepted in the relatively lower margin use for remote station monitoring. The technology of cellular telephones leads to the perception in the art that devices of this type must use different frequency networks.

While theoretically automatic meter reading is highly desirable, it is, of course, highly price sensitive and hence it is most important for any system to be adopted for the price per unit of particularly the large number of meter reading units to be kept to a minimum. The high cost of high power transmission devices, receiving devices and battery systems generally leads to a per unit cost which is unacceptably high.

SUMMARY OF THE INVENTION

A general object of the invention is a communications network for communicating data from a plurality of network service modules to a central data terminal.

Another object of the invention is a communications network which is suitable for an automatic meter reading system.

A further object of the invention is a communications network for collecting data from remote data generating stations that is simple and economical to install and maintain.

A still further object of the invention is a communications network for collecting data from network service modules that is spectrum efficient, and has inherent communication redundancy to enhance reliability and reduce operating costs.

An additional object of the invention is an open architecture communication network which accommodates new technology, and allows the network operator to serve an arbitrarily large contiguous or non-contiguous geographic area.

According to the invention, as embodied and broadly described herein, a method is provided for communicating over a wide area communications network between a central data terminal (CDT), a plurality of intermediate data terminals (IDT), a plurality of remote cell nodes (RCN), and a plurality of network service modules (NSM). The method uses a plurality of frames with each frame having a plurality of channels. During each frame, an intermediate data terminal transmits an IDT-synchronization signal to the plurality of remote cell nodes, using a first channel of the frame. The intermediate data terminal also transmits a first polling signal, synchronized to the IDT-synchronization signal, to the plurality of remote cell nodes, using a second channel of the frame.

Upon receipt of the IDT-synchronization signal, the plurality of remote cell nodes synchronize an RCN-timing circuit to the IDT-synchronization signal. The plurality of remote cell nodes then transmit an RCN-synchronization signal, synchronized to the IDT-synchronization signal, using a fourth channel of the frame.

The RCN-synchronization signal is received by at least one network service module. Network service modules receiving the RCN-synchronization signal synchronize an NSM timing circuit to the RCN-synchronization signal. Once synchronized, the network service module transmits, using radio waves, an NSM-packet signal to at least one remote cell node, using a fifth channel of the frame. This transmission from the network service module to the remote cell node can occur at a time which is randomly or pseu-

4

dorandomly selected within a predetermined time period. Alternatively, the network service module can transmit in response to a command signal received from a remote cell node, using radio waves, requesting the NSM-packet signal. The command signal from the remote cell node can also be used to transmit command information from the intermediate data terminal and/or the central data terminal to the network service module. This command information can include a request for an immediate meter reading or other real-time response from the network service module.

In addition to transmitting data, either randomly or in response to a command signal from a particular remote cell node, the NSM-packet signal can also be used to convey alarm conditions from the network service module to the remote cell node. Such alarm conditions can include loss of electrical connection, tilting of the network service module indicative of tampering, or other unusual condition. These alarm conditions can be transmitted on a real-time basis using a real-time channel of the frame. Upon receipt of an alarm condition from the network service module, the remote cell node transmits the alarm condition to the intermediate data terminal; the intermediate data terminal transmits the alarm condition to the central data terminal; the central data terminal processes the alarm condition and responds with appropriate direction back to the network service module using the command signal.

The NSM-packet signal is received by at least one remote cell node which stores the NSM-packet signal. Each remote cell node receives a multiplicity of NSM-packet signals from a multiplicity of network service modules. The multiplicity of network service modules is a subset of the plurality of network service modules. Each remote cell node stores the NSM-packet signals received from the multiplicity of network service modules. Upon receipt of the first polling signal, sent by the intermediate data terminal using the second channel of the frame, the remote cell node transmits the stored NSM-packet signals as an RCN-packet signal, using a third channel of the frame.

The RCN-packet signal is received by the intermediate data terminal on the third channel of the frame. Each intermediate data terminal receives a multiplicity of RCN-packet signals from a multiplicity of remote cell nodes. The multiplicity of RCN-packet signals are then stored by the intermediate data terminal. Upon receipt of a second polling signal, sent by the central data terminal using a sixth channel of the frame, the intermediate data terminal transmits the stored RCN-packet signals as an IDT-packet signal, using a seventh channel of the frame. The IDT-packet signal is received by the central data terminal on the seventh channel of the frame.

Alternatively, the invented method as embodied and broadly described herein, may be effected without the plurality of intermediate data terminals, in which case the central data terminal assumes the roles and functions that would otherwise be provided by the intermediate data terminals.

The wide area communications network, as broadly described herein, collects NSM data generated by a plurality of physical devices located within a geographical area. The physical devices may be, for example, a utility meter as used for electricity, gas or water. Each network service module is coupled to a respective physical device.

The network service module (NSM) includes NSM-receiver means, NSM-transmitter means, and NSM-processor means, NSM-memory means and an antenna. The NSM-receiver means, which is optional, receives a com-

5,673,252

5

mand signal at a first carrier frequency or a second carrier frequency. In a preferred mode of operation, the NSM-receiver means receives the command signal on the first carrier frequency for spectrum efficiency. The wide area communications network can operate using only a single carrier frequency, i.e., the first carrier frequency. The command signal allows the oscillator of the NSM-transmitting means to lock onto the frequency of the remote cell node, correcting for drift. Signalling data also may be sent from the remote cell node to the network service module using the command signal.

The NSM-processor means arranges data from the physical device into packets of data, transfers the data to the NSM-memory means, and uses the received command signal for adjusting the first carrier frequency of the NSM transmitter. The NSM data may include meter readings, time of use and other information or status from a plurality of sensors. The NSM-processor means, for all network service modules throughout a geographical area, can be programmed to read all the corresponding utility meters or other devices being serviced by the network service modules. The NSM-processor means also can be programmed to read peak consumption at predetermined intervals, such as every 15 minutes, throughout a time period, such as a day. The NSM-memory means stores NSM data from the physical device. The NSM-processor means can be programmed to track and store maximum and minimum sensor readings or levels throughout the time period, such as a day.

The NSM-transmitter means transmits at the first carrier frequency the respective NSM data from the physical device as an NSM-packet signal. The NSM-packet signal is transmitted at a time which is randomly or pseudorandomly selected within a predetermined time period, i.e., using a one-way-random-access protocol, by the NSM-processor means. The NSM-transmitter includes a synthesizer or equivalent circuitry for controlling its transmitter carrier frequency. The NSM-transmitter means is connected to the antenna for transmitting multi-directionally the NSM-packet signals.

A plurality of remote cell nodes are located within the geographical area and are spaced approximately uniformly and such that each network service modeled is within a range of several remote cell nodes, and so that each remote cell node can receive NSM-packet signals from a multiplicity of network service modules. The remote cell nodes preferably are spaced such that signals from each of the network service modules can be received by at least two remote cell nodes. Each remote cell node (RCN) includes RCN-transmitter means, RCN-receiver means, RCN-memory means, RCN-processor means, and an antenna. The RCN-transmitter means transmits at the first carrier frequency or the second carrier frequency, the command signal with signalling data. Transmitting a command signal from the RCN-transmitter means is optional, and is used only if the NSM-receiver means is used at the network service module as previously discussed.

The RCN-receiver means receives at the first carrier frequency a multiplicity of NSM-packet signals transmitted from a multiplicity of network service modules. Each of the NSM-packet signals typically are received at different points in time, since they were transmitted at a time which was randomly or pseudorandomly selected within the predetermined time period. The multiplicity of network service modules typically is a subset of the plurality of network service modules. The RCN-receiver means also receives polling signals from the intermediate data terminal, and listens or eavesdrops on neighboring remote cell nodes when they are polled by the intermediate data terminal.

6

The RCN-memory means stores the received multiplicity of NSM-packet signals. The RCN-processor means collates the NSM-packet signals received from the network service modules, identifies duplicates of NSM-packet signals and deletes the duplicate NSM-packet signals. When a polling signal is sent from an intermediate data terminal, the RCN-transmitter means transmits at the first carrier frequency the stored multiplicity of NSM-packet signals as an RCN-packet signal.

When a first remote cell node is polled with a first polling signal by the intermediate data terminal, neighboring remote cell nodes receive the RCN-packet signal transmitted by the first remote cell node. Upon receiving an acknowledgment signal from the intermediate data terminal, at the neighboring remote cell nodes, the respective RCN-processor means deletes from the respective RCN-memory means messages, i.e., NSM-packet signals, received from the network service modules that have the same message identification number as messages transmitted in the RCN-packet signal from the first remote cell node to the intermediate data terminal.

The plurality of intermediate data terminals are located within the geographic area and are spaced to form a grid overlaying the geographic area. Each intermediate data terminal includes IDT-transmitter means, IDT-memory means, IDT-processor means and IDT-receiver means. The IDT-transmitter means includes a synthesizer or equivalent circuitry for controlling the carrier frequency, and allowing the IDT-transmitter means to change carrier frequency. The IDT-transmitter means transmits preferably at the first carrier frequency, or the second carrier frequency, the first polling signal using a first polling-access protocol to the plurality of remote cell nodes. When the first polling signal is received by a remote cell node, that remote cell nodes responds by sending the RCN-packet signal to the intermediate data terminal which sent the polling signal. If the intermediate data terminal successfully receives the RCN-packet-signal, then the IDT-transmitter means sends an acknowledgment signal to the remote cell node. Each intermediate data terminal receives a multiplicity of RCN-packet signals from a multiplicity of remote cell nodes. The multiplicity of remote cell nodes typically is a subset of the plurality of remote cell nodes.

The IDT-receiver means receives the RCN-packet signal transmitted at the first carrier frequency from the remote cell node which was polled. Thus, after polling a multiplicity of remote cell nodes, the IDT-receiver means has received a multiplicity of RCN-packet signals.

The IDT-memory means stores the received RCN-packet signals. The IDT-processor means collates the NSM-packet signals embedded in the RCN-packet signals received from the multiplicity of remote cell nodes, identifies duplicates of NSM-packet signals and deletes the duplicate NSM-packet signals, i.e., messages from network service modules that have the same message identification number. In response to a second polling signal from a central data terminal, the IDT-transmitter means transmits the stored multiplicity of received RCN-packet signals as an IDT-packet signal to the central data terminal.

While not required by the current invention as presently embodied, the intermediate data terminals may also eavesdrop on neighboring intermediate data terminals in the same manner as was described for a given remote cell node eavesdropping on neighboring remote cell nodes. Such intermediate data terminal eavesdropping would serve as an additional means of identifying duplicate NSM data and eliminating such data before sending the non-duplicate data on to the central data terminal.

5,673,252

7

The central data terminal (CDT) includes CDT-transmitter means, CDT-receiver means, CDT-processor means and CDT-memory means. The CDT-transmitter means transmits sequentially the second polling signal using a second polling access protocol to each of the intermediate data terminals. The CDT-receiver means receives a plurality of IDT-packet signals. The central data terminal, intermediate data terminals and the remote cell nodes may be coupled through radio channels, telephone channels, fiber optic channels, cable channels, or other communications medium. The CDT-processor means decodes the plurality of IDT-packet signals as a plurality of NSM data. The CDT-processor means also identifies duplicates of NSM data and deletes the duplicate NSM data. The CDT-memory means stores the NSM data in a data base.

Additional objects and advantages of the invention are set forth in part in the description which follows, and in part are obvious from the description, or may be learned by practice of the invention. The invention disclosed may be adapted for use in any application requiring measurement of the use of a given resource through the use of a meter or other measuring device. The objects and advantages of the invention also may be realized and attained by means of the instrumentalities and combinations particularly pointed out in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and constitute a part of the specification, illustrate preferred embodiments of the invention, and together with the description serve to explain the principles of the invention.

FIG. 1 illustrates the hierarchial communications network topology;

FIG. 2 is a network service module block diagram;

FIG. 3 is a representative NSM-data packet;

FIG. 4 is a listing or representative applications supported by the communications network;

FIG. 5 is a schematic diagram of a network service module;

FIG. 6 shows a front elevation view of an electricity utility meter with a detection unit;

FIG. 7 shows a bottom plan view of the electricity utility meter;

FIG. 8 is an illustration of a typical printout of information obtained by the network service module;

FIG. 9 is a remote cell node block diagram;

FIG. 10 is an intermediate data terminal block diagram;

FIG. 11 is a central data terminal block diagram;

FIG. 12 shows the configuration of the communications network for serving widely separated geographic areas;

FIG. 13 illustrates a typical communications network with gradual growth in the number of areas served;

FIG. 14 illustrates a two-way frame structure for the wide area communications network;

FIG. 15 shows examples of subchannel structure;

FIG. 16 illustrates a general slot structure;

FIG. 17 provides a data slot channel description;

FIG. 18 illustrates the synchronization channel slots;

FIG. 19 illustrates the IRS slot in the synchronization channel;

FIG. 20 illustrates the IIS subchannel;

FIG. 21 illustrates the field sequence used to fill the RNS slot in the synchronization channel;

8

FIG. 22 illustrates the final portion of the RNS slot in the synchronization channel;

FIG. 23 illustrates the RNH slot in the synchronization channel;

FIG. 24 illustrates various forms of redundancy;

FIG. 25 illustrates the HDLC data link frame format;

FIG. 26 illustrates the IRD data link packet structure;

FIG. 27 illustrates the RIR data link packet structure;

FIG. 28 illustrates the IPH data link packet structure with 24 bits in the slot unused;

FIG. 29 illustrates the NRR data link packet structure;

FIG. 30 equals the RIQ data packet structure with 64 bits in the slot unused;

FIG. 31 illustrates the RND broadcast to class address data link packet structure;

FIG. 32 illustrates the RND broadcast to individual address and reverse poll data link packet structure;

FIG. 33 illustrates the RNC broadcast special application control data link packet structure;

FIG. 34 shows interactions with network and physical layers;

FIG. 35 illustrates node identifications;

FIG. 36 is an example tier address specification and selected/non-selected network service modules;

FIG. 37 depicts common fields of broadcast messages;

FIG. 38 is an example of time of use table and schedule broadcast to class address;

FIG. 39A and 39B is an example of a service reconnect broadcast to individual address;

FIG. 40 illustrates delivery of reverse poll messages to network service modules;

FIG. 41 shows an NRR network message structure in the context of a data link packet;

FIG. 42 illustrates the RND broadcast to class address message format in the context of a data link packet;

FIG. 43 illustrates the RND broadcast to individual address and reverse poll network message format in the context of a data link packet;

FIG. 44 illustrates the network message format used to distribute CAT entries, in the context of a data link packet;

FIG. 45 illustrates the format of a subchannel designator;

FIG. 46 illustrates the RIR network message format used to relay NSM messages, in the context of a data link packet;

FIG. 47 illustrates the RIR network message subfields comprising the data link control field;

FIG. 48 illustrates the subfields comprising remote cell node status fields;

FIG. 49 illustrates a maximum number of NSM messages per RIR;

FIG. 50 illustrates the IRH network message format in the context of a data link packet;

FIG. 51 illustrates the subfields comprising the SAC field;

FIG. 52 illustrates the RID network message format for delivering NSM broadcast messages to remote cell nodes;

FIG. 53 illustrates the subfields comprising various IRD fields;

FIG. 54 illustrates the IRD network message format for delivering NSM reverse poll messages to remote cell nodes;

FIG. 55 illustrates the subfields comprising "parms", field of IRD message of FIG.

FIG. 56 illustrates the RIQ message format used to request service from the intermediate data terminal, in the context of a data link packet;

5,673,252

9

FIG. 57 illustrates a summary of message priorities; and
 FIGS. 58A and 58B illustrate a preliminary data-flow diagram for the central data terminal network controller.

FIG. 59 illustrates a command message format that is compatible with a specific protocol whose general format.

FIG. 60 illustrates service traffic for a single neighborhood network;

FIG. 61 example of neighborhood network traffic representing roughly 16% of theoretical network capacity;

FIG. 62 illustrates space division multiplexing showing wide separation of concurrently polled areas;

FIG. 63 illustrates amplitude division multiplexing showing concurrent polling zones;

FIG. 64 illustrates directional multiplexing in which corresponding quadrants of all neighborhood networks are polled concurrently;

FIG. 65 illustrates polarization multiplexing in which alternate zones operate on different polarization, with areas in the corners of each zone may have interference;

FIG. 66 illustrates of SDMP using hexagonal cells;

FIG. 67 illustrates a comparison of IDT polling protocols;

FIG. 68 shows a comparison of standard channel slot assignment with a revised slot assignment for a DA channel;

FIG. 69 shows configuration of real-time channels which are employed in pairs to support deterministic communication to the NSM level and in the case of directional multiplexing, the four quadrants of the neighborhood are covered sequentially;

FIG. 70 shows performance comparison of different channel/frame structures for DA applications;

FIG. 71 illustrates frame configuration using real time channels for DA functions; and

FIG. 72 illustrates one possible frame configuration using allocated D/A slots.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Reference now is made in detail to the present preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings, wherein like reference numerals indicate like elements throughout the several views.

A wide area communications network communicates data from a plurality of network service modules to a central data terminal. The wide area communications network collects NSM data generated by a plurality of physical devices located within a geographical area. The wide area communications network, as illustratively shown in FIG. 1, is a layered network having a hierarchical communications topology comprising a plurality of network service modules 110, a plurality of remote cell nodes 112, a plurality of intermediate data terminals 114, and a central data terminal 120. They physical devices may be, for example, a utility meter as used for electricity, gas or water.

The central data terminal controls network operation. Intelligence exists at a all layers of the network, thereby easing the workload of the central data terminal. The intelligence attributed to each module is a function of the application of that module.

Network Service Module

Information is acquired at the lowest level of the wide area communications network of FIG. 1, and the network

10

service module 110 performs the data acquisition functions. Network service modules 110 include meter service modules for electricity, gas and water, a service disconnect module, a load management module, an alarm monitoring module, or any other module that can be used with the wide area communications network. The network service modules may be used in other applications such as vending machines, pay telephones, etc., where collecting remote data is desirable.

The network service modules 110 are linked to the wide area communications network via high frequency radio channels, typically in the 928 MHz–952 MHz band, as well as related frequencies in the 902 MHz–912 MHz and 918 MHz–928 MHz bands. Radio channels in these bands are the preferred communications medium because use of radio communications eliminates the need for physical connections to the network service modules which drastically reduces installation costs compared to other communication media such as telephone, cable networks and power line carriers. Also, operation in the high frequency bands permits the use of small antennas so the retrofitting standard watt hour meters is simplified. Radio communication channels in other bands may work equally as well, however.

In the exemplary arrangement shown in FIG. 2, the network service module (NSM) 110 includes NSM-receiver means, NSM-transmitter means, NSM-processor means, NSM-memory means and an NSM antenna 322. The NSM-transmitter means and the NSM-receiver means are coupled to the NSM antenna 322. The NSM-processor means is coupled to the NSM-transmitter means, NSM-receiver means, NSM-memory means and the physical device. The physical device is shown as basic sensors 320 and other sensors 322, and application control interface 324. The network service module also includes an AC power supply 310 and back-up battery power 312.

The NSM-receiver means is embodied as a NSM receiver 316, and is optional. If an NSM receiver 316 is included with the network service module, then the NSM receiver 316 can be used for receiving an RCN-synchronization signal and/or a command signal, which includes signalling data. The RCN-synchronization signal and/or the command signal can be transmitted at either a first carrier frequency or a second carrier frequency. Normally the first carrier frequency is used by the NSM-transmitter means for transmitting to a remote cell node. In a preferred embodiment, the NSM receiver 316 receives the RCN-synchronization signal and/or the command signal on the first carrier frequency for spectrum efficiency. Thus, the wide area communications network can operate using only a single carrier frequency, i.e., the first carrier frequency. The RCN-synchronization signal can provide a time reference for updating a local clock, and serve as a frequency reference to the network service module. Signalling data, such as manage service disconnect or control loads, also may be sent from the remote cell node to the network service module using the command signal. While the network service modules could be polled by the command signal, in general, such polling is not required and preferably not used with the present invention. The RCN-synchronization signal may be included as part of the command signal or a separate signal from the command signal.

The NSM-processor means, which is embodied as an NSM controller 314, arranges data from the physical device into packets of data, and transfers the data to the NSM-memory means which is embodied as an NSM memory 315. The term NSM data is defined to include data from the physical device. The NSM controller 314 may be a micro-

5,673,252

11

processor or equivalent circuit for performing the required functions. The NSM controller 314 uses the received RCN-synchronization signal and/or command signal for adjusting the first carrier frequency of the NSM transmitter. The NSM data may include meter readings, time of use and other information or status from a plurality of sensors. The NSM controller 314, for each network service module throughout a geographical area, can be programmed to read all the corresponding utility meters or other devices being serviced by the network service module, respectively. The NSM controller 314 can be programmed to read peak consumption at predetermined intervals, such as every 15 minutes, throughout a time period, such as a day. The NSM controller 314 also can be programmed to track and store maximum and minimum sensor readings or levels throughout the time period, such as a day.

The NSM memory 315 stores NSM, data from the physical device. NSM data may include meter reading data and time of use (TOU) and other information or status from a plurality of sensors. The NSM memory 315 may be random access memory (RAM) or any type of magnetic media other memory storage devices known in the art. The NSM controller 314 uses the received RCN-synchronization signal and/or command signal for adjusting the first carrier frequency of the NSM transmitter 318.

The NSM-transmitter means is embodied as an NSM transmitter 318. The NSM transmitter 318 transmits at a first carrier frequency, using radio waves, the respective NSM data from the physical device in brief message packets called NSM-packet signals. The NSM-packet signal might have a time duration of 100 milliseconds, although other time durations can be used to meet particular system requirements. The NSM-packet signal transmitted by the NSM transmitter 318 follows a generic or fixed format, and a representative message packet is illustrated in FIG. 3. Included in the message is: preamble; opening frame; message type; message identification; service module type; message number; service module address; data field; error detection; and closing frame.

The NSM transmitter 318 is connected to an NSM antenna 322 for transmitting multi-directionally the NSM-packet signals. The NSM transmitter 318 includes a synthesizer, crystal oscillator or equivalent circuitry for controlling its transmitter carrier frequency and schedule.

The NSM-packet signal is transmitted at a time which is randomly or pseudorandomly selected within a predetermined time period, i.e., using a one-way-random-access protocol, by the NSM-processor means. Alternatively, the NSM-packet signal may be transmitted in response to a poll received as part of a command signal from a remote cell node requesting the NSM-packet signal. In order to simplify network operation and reduce costs, the wide area communications network as embodied herein does not poll individual network service modules. Rather, each network service module reports autonomously at a rate appropriate for the application being supported. Routine reports are therefore transmitted randomly or pseudorandomly at fixed average intervals, while alarm signals are transmitted immediately following detection of alarm conditions. Alarm signals may be transmitted several times with random delays. This avoids interference among alarm messages if many alarms occur simultaneously, as in an area-wide power outage.

As an alternative arrangement, the network service module may be programmed to transmit three different types of messages at different intervals. The first type of message can relate to the accumulated usage information. The second

12

type of message can relate to an alarm condition which is basically transmitted immediately. The alarm conditions that occur might relate to a tamper action or to the absence of electrical voltage indicative of a power failure. The third type of information which may be transmitted less frequently can relate to the housekeeping information.

After preparing the packet of data for transmission, the controller 314 is arranged to hold the data packet for a random period of time. This random period can be calculated using various randomizing techniques including, for example, a pseudo-random calculation based upon the rotation of the metering disk at any particular instant. In this way each of the network service modules is arranged to transmit at a random time. The controller 314 is arranged so that the transmission does not occur within a particular predetermined quiet time so that the network service modules are not allowed to transmit during this quiet time. This quiet time could be set as one hour in every eight hour period. In this way, after an eight hour period has elapsed, each of the network service modules would transmit at a random time during the subsequent seven hours followed by one hour of quiet time.

Network capacity or throughput is limited by the probability of message collisions at each remote cell node 112. Because all network service modules 110 share a single carrier channel and transmit at random times, several network service modules 110 within a range of a particular remote cell node 112 may transmit simultaneously, with NSM-packet signals colliding at the remote cell node 112. If the received signal levels were comparable, then the overlapping messages mutually interfere, causing receive errors and both messages being lost. However, if one signal were substantially stronger than the other, then the stronger signal is successfully received. Moreover, since both signals are received by at least two and preferably four of the remote cell nodes, the probability of both messages being received is fairly high unless the network service modules are in close spatial proximity. During an interval T, each NSM transmitter within a geographical area surrounding a single remote cell node sends a single randomly timed message of duration M to several potential remote cell node receive stations.

N=no. of transmitter/cell

M=message duration (seconds)

T=message interval

P_c =probability of collision

P_r =probability of no collision

Once any Transmitter, T_i , starts transmitting, the probability that another particular transmitter, T_j , completes or starts another transmission is

$$\frac{2M}{T}$$

The probability that there will be no collision is

$$1 - \frac{2M}{T}$$

If there were N-1 other transmitters, then the probability of no collision, P_r , is given by

$$P_r = \left(1 - \frac{2M}{T} \right)^{N-1}$$

5,673,252

13

For large N

$$P_s = \left(1 - \frac{2M}{T}\right)^N$$

For a given Transmitter, T_s , the probability of a collision occurring during the interval T is

$$P_c = 1 - P_s = 1 - \left(1 - \frac{2M}{T}\right)^N$$

The probability of collisions occurring on An successive tries is

$$P_{cm} = (P_c)^{An}$$

For M=0.3 Sec T=8 hrs.=28.8×10³ secs.

N	Ps	Pc1	Pc2	Pc3
100	.9979	.0021	4 × 10 ⁻⁶	8 × 10 ⁻⁹
200	.9958	.0042	1.6 × 10 ⁻⁵	6.4 × 10 ⁻⁸
500	.9896	.0104	10 ⁻⁴	10 ⁻⁶
1,000	.9794	.0206	4 × 10 ⁻⁴	8 × 10 ⁻⁶
2,000	.9591	.041	1.6 × 10 ⁻³	6.8 × 10 ⁻⁵
5,000	.9010	.099	9.8 × 10 ⁻³	9.7 × 10 ⁻⁴
10,000	.811	.189	3.5 × 10 ⁻²	6.7 × 10 ⁻³

From the viewpoint of a remote cell node, the number of transmitters, N_T , whose signal level exceeds the receiver noise level and can, therefore, be received reliably depends on:

- the density of transmitters;
- transmit power level;
- propagation path loss;
- background noise.

Propagation path loss is highly variable due to attenuation, reflection, refraction and scattering phenomena which are a function of terrain, building structures, and antenna location. Some of these parameters can even vary on a diurnal and seasonal basis.

In estimating network performance however, the simple message collision model is not completely accurate because:

- random noise bursts from various sources can obscure messages which do not collide;
- some colliding message signals will be of such sufficiently different amplitude that the stronger signal will still be received correctly.

A statistical model can be developed to provide data by which a determination can be made of the best location and number of remote cell nodes for a particular geographical location. Thus, the model can include data relating to house density, the N-value defined above, and also relating to the attenuation of the signal and the location and presence of trees.

FIG. 4 is an illustrative listing of applications supported by the network service module within the wide area communications network. The following is a detailed discussion of the electricity meter application.

Network Service Module with an Electricity Meter

A network service module 110 schematically is shown in FIG. 5 and is mounted in a suitable housing 211 illustrated

14

in FIG. 6 and 7 with the housing including suitable mounting arrangement for attachment of the housing into the interior of a conventional electricity meter 212. Each network service module is coupled to a respective physical device. In FIG. 6, the physical device is an electricity meter 212.

Referring to FIGS. 5, 6 and 7 the electricity meter 212 includes an outer casing embodied as a cover 213 which is generally transparent. Within the casing is provided the meter system which includes a disk 214 which rotates about a vertical axis and is driven at a rate dependent upon the current drawn to the facility. The number of turns of the disk 214 are counted by a counting system including mechanical dials 215. The meter is of conventional construction and various different designs are well known in the art.

An antenna 217 is mounted on a bracket 216 carried on the housing inside the cover 213. The antenna as shown is arc-shaped extending around the periphery of the front face. Other antenna configurations are possible.

As illustrated in FIG. 6, the antenna 217 of each of the network service modules is mounted within the cover of the meter. Thus the NSM antenna 217 is mounted on the support structure itself of the network service module 110. This enables the network service module 110 to be manufactured relatively cheaply as an integral device which can be installed simply in one action. However, this provides an NSM antenna 217 which can transmit only relatively short distances. In addition, the power level is maintained at a relatively low value of the order of 10–100 milliwatts, the energy for which can be provided by a smaller battery system which is relatively inexpensive. An NSM antenna 217 of this type transmitting at the above power level would have a range of the order of one to two kilometers.

The network service module 110 is in a sealed housing 211 which prevents tapering with the sensors, microprocessor and memory located within the housing.

Turning now to FIG. 5, the network service module optionally may include a detection device which uses the microprocessor 220 which has associated therewith a storage memory 221. An essential sensor is for meter reading, for measuring the amount of electricity, amount of water or amount of gas consumed. Such a sensor alleviates having a meter reader person, by allowing the system to automatically report the amount of usage of the physical device.

Any number of sensors may be provided for detection of tampering events with the network service module of the present invention, and the sensors may be adapted for electricity, gas, water or other applications. For the most part, information reported by the various sensors would be considered low data rate. The wide area communications network supports distributed automation functions including basic meter reading, time of use meter reading, service connect and disconnect operations, alarm reporting, theft of service reporting, load research, residential load control, commercial and industrial load curtailment, and distributed supervisory control and data acquisition (SCADA). Furthermore, the wide area communications network is readily expandable to support new applications as they are developed.

While the emphasis, by way of example, is automatic meter reading and on measuring time of use of an electricity meter, other functions such as 15-minute peak consumption recording, line power monitoring, i.e., outage and restoration, tamper sensing and timekeeping are supported.

The following is a representative listing of possible sensors that may be used with the network service module of the present invention. Each sensor is optional, and to a person

5,673,252

15

skilled in the art, variants may be added to the network service module of the present invention. For example, FIG. 5 illustratively shows a temperature sensor 227 and a battery level sensor 228; however, each sensor 227, 228 may be substituted by or may be in addition to other possible sensors from the following representative listing of sensors.

- (a) A tilt sensor 222 detects movement of the housing through an angle greater than a predetermined angle so that once the device is installed, indication can be made if the device is removed or if the meter is removed from its normal orientation.
- (b) A electric field sensor 223 detects the presence of an electric field. Unless there is power failure, the electric field sensor should continue to detect the presence of an electric field unless the meter is removed from the system.
- (c) An acoustic sensor 224 detects sound. The sounds detected are transmitted through a filter 225 which is arranged to filter by analog or digital techniques the sound signal so as to allow to pass through only those sounds which have been determined by previous experimentation to relate to cutting or drilling action, particularly on the cover.
- (d) A magnetic sensor 226 detects the presence of a magnetic field. A magnetic field is generated by the coils driving the disk so than magnetic fields should always be present unless the meter has been by-passed or removed. As is well known, the rate of rotation of the disk is dependent upon the magnetic field and, therefore, this rate of rotation can be varied by changing the magnetic field by applying a permanent or electromagnet in the area of the meter to vary the magnetic field. The magnetic sensor 226 is, therefore, responsive to variations in the magnetic field greater than a predetermined magnitude so as to indicate that an attempt has been made to vary the magnetic field adjacent the disk to slow down the rotation of the disk.
- (e) A temperature sensor 227 detects heat so that the temperature associated with a particular time period can be recorded. A battery level sensor is indicated at 228. The sensors 226, 227 and 228 communicate information through an analog digital converter 328 to the microprocessor 220. The information from sensors 227 and 228 can be communicated to provide "house-keeping" status of the operation of the unit. The temperature sensor 227 can be omitted if required and this information replaced by information gained from a public weather information source. In some cases the meter is located inside the building and hence the temperature remains substantially constant whereas the outside temperature is well known to vary consumption quite dramatically.
- (f) A consumption sensor comprises a direct consumption monitor 229 which can be of a very simple construction since it is not intended to act as an accurate measure of the consumption of the electricity used. The direct consumption monitor can, therefore, simply be a device which detects the value of the magnetic field generated, on the assumption that this value is proportional to the current drawn. The direct consumption value obtained can then be completed with a measurement of the consumption as recorded by the rotation of the disk 214. In the event that the direct consumption monitor provides a sum of the consumption over a time period which is different from the consumption measured by rotation of the disk 214 by an amount greater than a

16

predetermined proportion, then the direct consumption monitor 229 can be used to provide a tamper signal. This would be indicative, for example, of a mechanical tag applied to the disk to reduce recorded consumption.

- (g) A forward/reverse sensor 230, discussed in more detail hereinafter, detects reverse rotation of the disk 214 and provides an input to the microprocessor upon detection of such an event.
- (h) A cover sensor 231 is used to detect the continual presence of the cover 213. The cover sensor comprises a light emitting diode (LED) 232 which generates a light beam which is then reflected to a photo diode 233. The absence of the reflected beam at the photo diode 233 is detected and transmitted as a tamper signal to the microprocessor 220. The reflected beam is generated by a reflective strip 234 applied on the inside surface of the cover adjacent the diode 232 as shown in FIG. 6.

The above sensors thus act to detect various tampering events so that the presence of such a tampering event can be recorded in the storage memory 221 under the control of the microprocessor 220.

The microprocessor 220 also includes a clock signal generator 335 so that the microprocessor 220 can create a plurality of time periods arranged sequentially and each of a predetermined length. In the example of the present invention shown, the time periods are eight hours in length and the microprocessor 220 is arranged to record in each eight hour period the presence of a tamper event from one or more of the tamper signals.

As shown in FIG. 8, the series of predetermined time periods is recorded with the series allocated against specific dates and each eight hour period within the day having a separate recording location within the storage memory 221. One such series is shown in FIG. 8, where a number of tampering events 236 are indicated. The print-out thus indicates when any tampering event 236 has occurred and in addition then identifies which type of tampering event has taken place.

The rotation of the disk 214 also is detected to accurately record the number of rotations of the disk both in a forward and in a reverse direction. In FIG. 8, a table 237 shows in graphical form the amount of rotation of a disk recorded in eight hour periods as previously described. For one period of time the disk is shown to have rotated in a reverse direction 238. Whenever the disk rotates in a reverse direction, the reverse rotation subtracts from the number of turns counted on the conventional recording system 215, shown in FIG. 6.

As shown in FIGS. 6 and 7, detection of the rotation of the disk is carried out by the provision of a dark segment 239 formed on the undersurface of the disk, leaving the remainder of the disk as a reflective or white material. The detection system thus provides a pair of light emitting diodes 240, 241 which are positioned on the housing so as to direct light onto the underside of the disk. The light emitting diodes 240, 241 are angularly spaced around the disk. The diodes are associated with the photo diodes 242, 243 which receive light when the disk is positioned so that the light from the associated light emitting diode 240, 241 falls upon the reflective part of the disk and that light is cut off when the dark part of the disk reaches the requisite location. Basically, therefore, one of the pairs of light emitting diodes 240, 241 or photo diodes 242, 243 is used to detect the passage of the dark segment which is, of course, one rotation of the disk 214. The direction of rotation is then detected by checking with the other of the pairs as the dark segment reaches the first of the pairs as to whether the second pair is also seeing the dark segment or whether it is seeing the reflective

5,673,252

17

material. Provided the sensors are properly spaced in relation to the dimension of the segment, therefore, this indicates the direction which the disk rotated to reach the position which is detected by the first pair of diodes.

In order to conserve energy, the sensors are primarily in a sampling mode using an adaptive sensing rate algorithm. In one example the dark or non-reflective segment is 108° of arc and there is provided a 50° displacement between the sensors. In a practical example of a conventional meter, the maximum rotation rate is of the order of 2 rps. A basic sample interval can be selected at 125 m/sec. to ensure at least one dark sample is obtained from the dark segment. In operation, only the first pair of sensors is sampled continuously. When a dark response is observed, a second confirming sample is obtained and the sample rate increased to 16 pps. As soon as a light segment of the disk is sensed, the second sensor is sampled. If the second sensor still saw the dark segment, then clockwise rotation is confirmed; if a light segment were observed, then counter-clockwise rotation is indicated.

At slower speeds, the algorithm results in a sample rate of 8 pps for 70% of a rotation and 16 pps for 30% of a rotation for the first pair of sensors plus two samples for direction sensing for the second pair. For annual average consumption of 12,000 kwh, the disk rotates approximately 1.6 million times.

In order to sense the presence of stray light which could interfere with measurements, the photo diode output is sampled immediately before and immediately after the light emitting diode (LED) is activated. If light is sensed with the LED off, stray light is indicated and an alarm may be initiated after a confirming test. The latter may include a test of other sensors such as the optical communication port sensor discussed hereinafter.

As shown in FIG. 5, communication from the meter reading unit is carried out by radio transmission from the microprocessor 220 through a modulation device 250 which connects to the antenna 322. The transmission of the signal is carried out under the control of the microprocessor 220. Modulation carried out by the modulation device 250 can be of a suitable type including, for example, phase modulation using amplitude shift keying (ASK), phase shift keying (PSK) such as binary PSK (BPSK), frequency modulation using frequency shift keying (FSK), such as, for example, binary FSK, or spread spectrum modulation. This allows the system to be used without the allocation of a dedicated frequency so that the signal appears merely as noise to receivers which do not have access to the decoding algorithm by which the signal can be recovered from the different frequencies on which it is transmitted.

Remote Cell Nodes

A plurality of remote cell nodes 112 in FIG. 1 is located within the geographical area and is spaced approximately uniformly and such that each network service module 110 is within a range of several remote cell nodes 112 to provide overlapping coverage. The remote cell nodes 112 typically might be spaced at 0.5 mile intervals on utility poles or light standards. Each remote cell node 112 provides coverage over a limited area much like the cell in a cellular telephone network. Remote cell nodes 112 preferably are spaced to provide overlapping coverage, so that on an average, each NSM-packet signal transmitted by a network service module 110 is received by three or four remote cell nodes 112, even in the presence of temporary fading. As a consequence, erection of a tall building near a network service module 110 has little or no effect on message reception, nor does the

18

failure of a remote cell node 112 result in loss of NSM-packet signals or NSM data.

As illustratively shown in FIG. 9, each remote cell node (RCN) 112 of FIG. 1 includes RCN-transmitter means, RCN-receiver means, RCN-memory means, RCN-processor means and an RCN antenna 422. The RCN-transmitter means, RCN-receiver means, RCN-memory means and RCN-processor means may be embodied as an RCN transmitter 418, RCN receiver 416, RCN memory 415 and RCN processor 414, respectively. The RCN transmitter 418 and the RCN receiver 416 are coupled to the RCN antenna 422. The RCN processor 414 is coupled to the RCN transmitter 418, RCN receiver 416, and RCN memory 415.

The RCN transmitter 418, under the control of the RCN processor 414, transmits an RCN-synchronization signal and/or a command signal using radio waves at the first carrier frequency or the second carrier frequency. The choice of frequency depends on which frequency is being used for the NSM receiver 316 at each of the plurality of network service modules 110. Transmitting an RCN-synchronization signal and/or a command signal from the RCN transmitter is optional, and is used if the NSM receiver 316 is used at the network service module 110. The command signal can include signalling data being sent to the network service module 110. The signalling data may require the network service module 110 to transmit status or other data; set reporting time period, e.g., from an eight hour period to a four hour period; and any other command, control or "housekeeping" jobs as required.

The RCN receiver 416 receives at the first carrier frequency a multiplicity of NSM-packet signals transmitted from a multiplicity of network service modules 110 by radio waves. Each of the multiplicity of NSM-packet signals typically is received at a different point in time, since they are transmitted at a time which is randomly or pseudorandomly selected within the predetermined time period. The multiplicity of network service modules 110 usually is a subset of plurality of network service modules 110. Received NSM-packet signals are time stamped by the RCN processor 414 and temporarily stored in the RCN memory 415 before being transmitted to the next higher network level. The RCN receiver 416 also receives polling signals from the intermediate data terminal 114, and listens or eavesdrops on neighboring remote cell nodes when they are polled by the intermediate data terminal 114.

The RCN processor 414 collates the NSM-packet signals received from the network service modules, identifies duplicates of NSM-packet signals and deletes the duplicate NSM-packet signals. The RCN processor 414 controls the RCN transmitter 418 and RCN receiver 416. The RCN memory 415 stores the received multiplicity of NSM-packet signals. Thus each remote cell node 112 receives, decodes and stores in RCN memory 415 each of these NSM-packet signals as received from the network service modules 110.

The remote cell node 112 comprises simply a suitable resistant casing which can be mounted upon a building, lamp standard or utility pole at a suitable location in the district concerned. The remote cell node 112 can be battery powered, and have a simple omni-directional antenna as an integral part of the housing or supported thereon.

Information accumulated at remote cell nodes 112 periodically is forwarded via a polled radio communications link to a higher level network node, as illustrated in FIG. 1, termed an intermediate data terminal 114. The communications link may alternatively be by cable or other communications channel. The intermediate data terminals 114 are

5,673,252

19

spaced typically at four mile intervals and can be conveniently cited at substations, providing coverage for up to 100 cells. Remote cell nodes also receive timing information and command signals from intermediate data terminals.

When a polling signal is sent from an intermediate data terminal 114, the RCN transmitter 418 transmits at the first carrier frequency the stored multiplicity of NSM-packet signals as an RCN-packet signal to the intermediate data terminal 114.

When a first remote cell node is polled with a first polling signal by the intermediate data terminal, neighboring remote cell nodes 112 receive the RCN-packet signal transmitted by the first remote cell node. Upon receiving an acknowledgment signal from the intermediate data terminal that polled the first remote cell node, at the neighboring remote cell nodes 112 the respective RCN processor deletes from the respective RCN memory messages from the network service modules that have the same message identification number as messages transmitted in the RCN-packet signal from the first remote cell node to the intermediate data terminal. The message identification number is illustrated in a typical NSM-data packet in FIG. 3.

FIG. 1 illustrates a plurality of the network service modules 110. The network service modules 110 are set out in a pattern across the ground. This pattern is dependent upon the locations of the utility usage which generally do not have any particular pattern and which vary significantly in density from location to location.

The remote cell nodes 112 are arranged in an array with the spacing between the remote cell nodes 112 relative to the network service modules 110 such that each network service module 110 can transmit to at least two and preferably four of the remote cell nodes 112. Thus, the remote cell nodes 112 are provided in significantly larger numbers than is absolutely necessary for the signals from each network service module 110 to be received by a respective one of the remote cell nodes 112. The remote cell nodes 110 theoretically receive high levels of duplicate information. In a normal residential situation, locating the remote cell nodes 112 so that each network service module 110 can be received by four such remote cell nodes 112 would lead to an array in which each remote cell node 112 would be responsive to approximately 1,000 of the network service modules 110.

Each of the network service modules 110 is arranged to calculate an accumulated value of utility usage for a set period of time which in the example shown is eight hours. Subsequent to the eight hour period, the NSM controller 314 prepares to transmit the information in a packet of data as an NSM-packet signal. The packet of data includes:

- (a) The total of usage during the set period, e.g., eight hours.
- (b) The accumulated total usage stored in the NSM memory 315 to date. The transmission of this information ensures that even if a message is lost, resulting in the total for one of the time periods not being communicated to the central data terminal, the central data terminal 120 can recalculate the amount in the missing time periods from the updated accumulated total.
- (c) Some or all of the tamper signals defined above.
- (d) The time of transmission.
- (e) A message number so that the messages are numbered sequentially. In this way, again, the remote cell node 112 can determine whether a message has been lost or whether the information received is merely a duplicate message from a duplicate one of the receiving stations.

20

(f) Housekeeping information concerning the status of the network service module 110, for example, the temperature and the battery level indicator sensor values.

When information is received at the remote cell node 112, the RCN processor 414 acts to store the information received in the RCN memory 415 and then to analyze the information. The first step in the analysis is to extract from the received messages the identification code relating to the respective network service module 110. The information relating to that network service module 110 is introduced into an RCN memory register relating to that network service module 110 to update the information already stored.

One technique for avoiding transmission of duplicate information from the remote cell nodes 112 to the intermediate data terminal 114 requires that each remote cell node 112 monitor the transmissions of the other remote cell nodes 112. When the signals are monitored, the information transmitted is compared with information stored in the monitoring remote cell node 112 and if any redundant information were found in the memory of the monitoring remote cell node 112, then the redundant information is canceled. Using this technique, when very high levels of redundancy are used, the time for transmission from the remote cell node 112 to the intermediate data terminal is not significantly increased.

In addition to the periodic transmission of the usage data, each network service module 110 can be programmed to transmit an alarm signal upon detection of the removal of the electric voltage or excessive tilting of the network service module. The transmission of the alarm signal can be delayed by a short random period of time so that if the loss of the voltage were due to a power outage covering a number of locations, then all signals are not received at the same time. The remote cell nodes 112 and intermediate data terminals 114 also can be programmed to retransmit such alarm signals immediately this way the central data terminal 120 has immediate information concerning any power outages, including the area concerned. This can, of course, enable more rapid repair functions to be initiated.

In addition to automatic alarm signal transmission, the central data terminal or the intermediate data terminals can send a request for transmission of data to a particular network service module over a real-time channel. Upon receiving such a request, the network service module responds with a current reading of power usage, alarm condition, or other, as data requested. This real-time channel enables the central data terminal to gather up-to-the-minute data rather than having to wait for the network service module's next scheduled transmission. This real-time channel can also be used to send a power cut-off, or other, command from the central data terminal to specific network service modules, with nearly instantaneous results if necessary.

Furthermore, the remote cell nodes 112 can be arranged to transmit control signals for operating equipment within the premises in which the network service module 110 is located. The remote cell nodes 112 are necessarily arranged in a suitable array to transmit such information so that the information can be received in each of the premises concerned using relatively low transmission power and using the equipment already provided for the meter reading system. This transmission capability can be used to control, for example, radio-controlled switches within the premises of relatively high power equipment for load shedding at peak periods. In similar arrangements, the network service module 110 may include a receiving facility to enable detection of signals transmitted by the remote cell nodes 112. In one

5,673,252

21

example, these signals may relate to synchronizing signals so that each of the network service modules 110 is exactly synchronized in time with the remote cell node 112 and/or intermediate data terminal 114 and central data terminal 120. This exact synchronization can be used to accurately detect usage during specific time periods, enabling the utility to charge different rates for usage during different time periods in order to encourage use at non-peak times, again for load shedding purposes.

The attenuation of a radio signal is proportional to the inverse of the distance from the source to the power N . In free space N is equal to 2. In more practical examples where buildings, trees and other geographical obstructions interfere, the power N generally lies between 4.0 and 5.0. This interference, therefore, significantly reduces the distance over which the signal from the network service module can be monitored. Thus, the number of network service modules which can be monitored by a single remote cell node is significantly reduced. Furthermore, the large N rapidly reduces the signal strength after a predetermined distance so that while a network service module can be effectively monitored at a certain distance, the signal strength rapidly falls off beyond that distance. This enables the cells defined by each remote cell node 112 to be relatively specific in size and for the degree of overlap of the cells to be controlled to practical levels without wide statistical variations.

An advantage of the present system is that network service modules, which are located at a position which is geographically very disadvantageous for transmission to the closest remote cell node, may be monitored by a different one of the remote cell nodes. Thus, in conventional systems some of the network service modules may not be monitored at all in view of some particular geographical problem. In the present invention this possibility is significantly reduced by the fact that the network service module concerned is likely to be in a position to be monitored by a larger number of the remote cell nodes so that the geographical problem probably does not apply to all of the remote cell nodes.

The increased density of remote cell nodes permits the network service modules to operate with an integral NSM antenna which can be formed as part of the meter reading unit housed within the conventional electric utility meter. In this way the network service module can be totally self-contained within the meter housing, thus enabling installation to be completed within a very short period of time, avoiding customer dissatisfaction caused by wiring problems, and reducing the possibility of damage to a separately mounted NSM antenna. In addition, this arrangement significantly reduces the cost of the network service module to a level which makes it economically viable to install the system.

The present invention can employ a system in which the network service modules are permitted to transmit only during a predetermined time period so that an open time period is available for communication on the same frequency between the intermediate data terminal and the remote cell node without any interference from the network service modules. This level of communication can be carried out using a polling system from the intermediate data terminals to each of the remote cell nodes, in turn, preferably including a directional transmission system at the intermediate data terminal. This system allows optimization of the remote cell node density to meet cost/performance criteria in different deployment scenarios.

The present invention, by recognizing the non-volatile nature of the information source and the acceptability of missing an occasional update through transmission errors or

22

collisions enables the implementation of data collection networks of greater simplicity and at lower cost than is possible with established communication network approaches involving two-way communication. The present invention, therefore, provides a radio communication network which can be employed to acquire data from a large number of remote meter monitoring devices dispatched over a wide area using very low power transmitters in conjunction with an array of remote cell nodes all operating on a single radio communication channel or frequency.

Intermediate Data Terminal

The plurality of intermediate data terminals 114 are located within the geographic area and are spaced to form a grid overlaying the geographic area. The intermediate data terminals typically are spaced to cover large geographic areas. Intermediate data terminals preferably are spaced to provide overlapping coverage, so that on an average, an RCN-packet signal transmitted from a remote cell node is received by two or more intermediate data terminals.

As illustratively shown in FIG. 10 each intermediate data terminal includes first IDT-transmitter means, second IDT-transmitter means, IDT-memory means, IDT-processor means, first IDT-receiver means, second IDT-receiver means and an IDT antenna. The first IDT-transmitter means, second IDT-transmitter means, IDT-memory means, IDT-processor means, first IDT receiver means and second IDT-receiver means may be embodied as a first IDT transmitter 518, second IDT transmitter 519, IDT memory 515, IDT processor 514, first IDT receiver 521 and second IDT receiver 522, respectively. The first IDT transmitter 518 and the first IDT receiver 521 are coupled to the IDT antenna 522. The IDT processor 514 is coupled to the first IDT transmitter 518 and second IDT transmitter 519, and the first IDT receiver 521 and second IDT receiver 522. The second IDT transmitter 519 and the second IDT receiver 522 may be embodied as a device such as a modem 523.

The first IDT transmitter 518 under the control of the IDT processor 514, includes a synthesizer or equivalent circuitry for controlling the carrier frequency, and allowing the first IDT transmitter 518 to change carrier frequency. The first IDT transmitter 518 transmits preferably at the first carrier frequency, or the second carrier frequency, the first polling signal using a first polling-access protocol to the plurality of remote cell nodes. When the first polling signal is received by a remote cell node, that remote cell node responds by sending the RCN-packet signal to the intermediate data terminal which sent the first polling signal. If the intermediate data terminal successfully receives the RCN-packet signal, then the first IDT transmitter 518 sends an acknowledgment signal to the remote cell node. Upon receiving the acknowledgment signal, the RCN processor 414 at the remote cell node deletes, from the RCN memory 415, the data sent in the RCN-packet signal to the intermediate data terminal.

The transmitted signal may be by radio waves over a freespace channel, or using a high frequency signal over a cable or other channel. Thus, the communications channel between remote cell nodes and intermediate data terminals may be free space, cable or a combination thereof, or other equivalent channels.

Intermediate data terminals also communicate an IDT-synchronization signal for conveying timing information and command signals to remote cell nodes. Remote cell nodes serving important SCADA functions can be polled more frequently by an intermediate data terminal to reduce network response time.

5,673,252

23

The first IDT receiver 521 receives the RCN-packet signal transmitted at the first carrier frequency from the remote cell node which was polled. Thus, after sequentially polling a multiplicity of remote cell nodes 112, the first IDT receiver 521 has received sequentially in time a multiplicity of RCN-packet signals. The multiplicity of RCN-packet signals usually is a subset of the plurality of RCN-packet signals.

The IDT memory 515 stores the received RCN-packet signals. The IDT processor 514 collates the NSM-packet signals embedded in the RCN-packet signals received from the multiplicity of remote cell nodes, identifies duplicates of NSM-packet signals and deletes the duplicate NSM-packet signals, i.e., messages from network service modules that have the same message identification number.

In response to a second polling signal from a central data terminal 120, the second IDT transmitter 519 transmits the stored multiplicity of RCN-packet signals as an IDT-packet signal to the central data terminal 120. The second IDT transmitter 519 and second IDT receiver 522 may be embodied as a modem 523 or other device for communicating information over a communications medium 525 linking the intermediate data terminal via a telephone line or other communications channel with the central data terminal.

The intermediate data terminals may include one or more directional antennas 522. During the quiet time, the intermediate data terminal is arranged to direct the antenna 522 or antennas to each of the remote cell nodes in turn and to transmit to the respective remote cell node the first polling signal, calling for the remote cell node to transmit the stored information from the RCN memory 415. Use of more than one antenna can allow communication with more than one remote cell node at a time. The remote cell node is required, therefore, merely to transmit the information upon request in a collated package which is transmitted to the intermediate data terminal and collected for analysis.

In an alternative embodiment of the invention, the invented method may be effected without the plurality of intermediate data terminals, in which case the central data terminal assumes the roles and functions that would otherwise be provided by intermediate data terminals.

Central Data Terminal

At the upper level of the hierarchy is a central data terminal 120 which acts as a network control center and data consolidation point. The central data terminal 120 controls basic network operation, allowing the central data terminal to make global decisions regarding network organization. The central data terminal's purpose is to integrate information from a variety of network nodes into a coherent form which may be forwarded to different utility operating groups for specific applications. In addition to linking regional data terminals, the central data terminal is connected to critical SCADA sites, some of which may be co-located with intermediate data terminals at sub-stations. At this level, there are relatively few communication links, so those required can be selected to optimize cost, speed and reliability. The transmission between the central data terminal 120 and the plurality of intermediate data terminals 114 is carried out using a communications medium 525 such as telephone lines, T1 carriers, fiber optic channels, coaxial cable channels, microwave channels, or satellite links.

As illustratively shown in FIG. 11, the central data terminal (CDT) includes CDT-transmitter means, CDT-receiver means, CDT-processor means and CDT-memory means. The CDT-transmitter means, CDT-receiver means,

24

CDT-processor means and CDT-memory means may be embodied as a CDT transmitter 618, CDT receiver 616, CDT processor 614 and CDT memory 615, respectively. The CDT transmitter 618 and CDT receiver 616 are coupled to the communications medium 525. The CDT processor 614 is coupled to the CDT transmitter 618, CDT receiver 616 and CDT memory 615. The CDT transmitter 618 and CDT receiver 616 may be a modem 625 or other device suitable for communicating information over the communications medium 525 between the central data terminal 120 and each intermediate data terminal 114.

The CDT transmitter 618 transmits the second polling signal sequentially in time, using a second polling access protocol, to the plurality of intermediate data terminals. The CDT receiver 616 receives a plurality of IDT-packet signals. The CDT processor 614 decodes the plurality of IDT-packet signals as a plurality of NSM data. The CDT processor 614 also identifies duplicates of NSM data and deletes the duplicate NSM data. The CDT memory 615 stores the NSM data in a data base. The NSM data is outputted, analyzed or processed as desired.

Utility Overview

The performance of the network is in large part determined by the performance of the network service module 110 to remote cell node 112 link, which is defined by the network service module message loss rate. The network architecture is designed to minimize the network service module message loss rate, which is defined as the fraction of transmitted network service module messages which are not received by the remote cell nodes. The two issues that affect the message loss rate are:

1. relatively large and varying pathloss which is caused by the nature of the urban propagation environment; and
2. simultaneous message transmissions, or collisions, which are a problem for any multiple-access system.

The issue of large and varying pathloss is resolved through the use of:

1. transmit power adjustment;
2. path redundancy, controlled by the remote cell node grid spacing; and
3. multiple transmissions per day.

The collision issue is resolved using:

1. path redundancy, controlled by the remote cell node grid spacing;
2. multiple transmission per day;
3. partitioning of traffic according to priority; and
4. capture effect.

Remote cell node spacing can be selected to control the path redundancy, thus leading to an adjustable level of performance. Notice that path redundancy and multiple transmission per day are used to resolve both issues, and thus are principal features of the wide area communications network. The effect of collisions is minimal, so the probability of receiving a packet any time during the day is maintained at exceptionally high levels.

The link budget contains all of the gains and losses between the network service module power amplifier and the remote cell node receiver, and is used to calculate the maximum pathloss which can be allowed on any link. The minimum receivable signal at the remote cell node is estimated as 31 115 dBm, which is equal to the sum of the noise floor and the carrier to noise level which is required in order to receive the message, e.g., 10 dB.

Every network service module has many remote cell nodes within receiving range, which increases the reliability

5,673,252

25

of packet reception. When a network service module transmits, the transmission has the potential to be received by many remote cell nodes. Some of the remote cell nodes are in shadow fading zones and do not receive the signal whereas others have an increased signal due to shadowing.

Even though some of the remote cell nodes are quite far from the network service module, and thus the average pathloss is above the maximum allowed limit, receiving the network service module transmission is possible if the signal level fluctuations, shadowing, multipathing, etc., contributed enough to the signal level. Similarly, some remote cell nodes which are close to the network service module do not hear the network service module because signal variations have decreased the signal network level by a significant amount. The unexpected loss of network service module transmission is anticipated to be offset by fortuitous gains as described.

In addition to short-term variations in signal reception, long-term effects also impact the success of transmission. During the life of the system, the urban landscape changes due to building construction and demolition, and foliage growth. These changes in landscape affect the network service module-remote cell node links, causing some remote cell nodes to no longer receive the network service module transmissions while new remote cell nodes begin to receive those same network service module transmissions. For each link that is no longer available, a new link is expected to become operational.

The hierarchical design of the wide area communications network allows the customer to service an arbitrarily large contiguous or non-contiguous geographic area, as shown in FIG. 12, containing many applications and a large number of end points.

FIG. 12 illustrates the configuration of the wide area communications network for serving widely separated geographic areas. This includes the provision of a wide area communications network for serving widely separated geographic areas, as well as isolated smaller communities via satellite, fibre optic, microwave or other back bone network. Due to the unique nature of the wide area communications network's single channel and micro cellular scattering propagation concept, the wide area communications network is ideal for many traditionally difficult environments as it is immune to traditional radio problems such as fading, nulls, multipath, and lack of line of sight typical of mountainous or high density urban settings. The hierarchical design of the wide area communications network allows non-contiguous areas to be serviced over a wide geographic area. Separate areas have their own intermediate data terminal communicating with the central data terminal. Data from non-contiguous areas would be transferred at the central data terminal.

The wide area communications network supports a broad range of monitoring, verifiable control, and fast response transaction applications. A number of these application needs are and continue to be identified by utilities. Due to the standardized network interface protocol and message packet configuration, the wide area communications network is able to readily augment its service offerings in either new hardware or software. The wide area communications network offers not only specialized network service modules for electric, gas and water meters but also provides a series of generic modules with industry standard input/output interfaces for contact closure, voltage or current sensing. This allows a variety of vendors to incorporate a wide area communications network communication interface into their own products, be they fuses, alarms, temperature sensors, vending machines, etc.

26

The wide area communications network can provide a single integrated data channel for other utility operational applications. Some of these applications are hardware oriented but many are application software oriented. They involve the generation of new value-added information reports or services. Although some are primarily for use by the utility, many could be offered for sale to the customer thus creating a new revenue stream for the utility.

The wide area communications network can readily and cost effectively expand to support new hardware and application software growth scenarios. The wide area communications network can be implemented in those regions of the user's service territory and for those services which are most needed on an implementation plan which is not affected by geographic distribution.

The wide area communications network can support the expansion of SCADA due to its highly reliable wireless communication capabilities. Many utilities would like to add instrumental monitoring points to their SCADA, but the wiring costs and difficulties often associated with these points prohibits SCADA growth at a sub-station or other site. Generic network service modules could be used to solve these problems.

The key issues related to expansion are:

1. the size and arrangement of the geographic area;
2. the number of end points which can be serviced; and
3. the ease with which the number of applications can be increased.

As the number of end points increases, either due to an increase in the number of applications in a geographic area or due to an increase in the size of the geographic area being serviced, the network traffic increases. The amount of additional traffic created depends on the type of application being added. Traffic increases in the wide area communications network are dealt with by hardware expansion at the central data terminal and by installation of additional intermediate data terminals in the new area. FIG. 13 illustrates a typical communications network with gradual growth in the number of areas served.

As the number of end points increases, another issue of concern is the identification of the message source. A wide area communications network provides over one trillion serial numbers for each type of service module, which allows unique module identification over the life of the system.

As the number of applications increases, the amount of traffic from a given square mile is assumed to also increase. Simulations to the present time have indicated that more than 20,000 end points can be serviced per square mile, with this maximum number depending on the details of remote cell node deployment, house density and message reporting frequency. A dense urban area with 35 ft. by 100 ft. lots contains approximately 5000 homes per square mile.

Centralized control of the wide area communications network is achieved by allowing the central data terminal to have access to network status data, which it uses to make decisions regarding network optimization. These decisions are downloaded to the intermediate data terminals and remote cell nodes as required.

Centralized traffic control is achieved at the remote cell node and intermediate data terminal levels by using priority tables, message storage instructions and alarm storage instructions. The structure of the priority tables is described as follows.

In each message than is transferred through the system, there is a set of identification tags stating the message type and the source. The priority tables in the remote cell nodes

5,673,252

27

and intermediate data terminals contain a listing of all identification tags in the system; these tables are first installed at the time of deployment, but can be updated from the central data terminal as required. If, during the network operational period, there is a need to change message priorities, this change can then be effectuated with minimal impact on the network traffic.

Control of the alarm traffic within the network requires another table because alarm reporting generates higher traffic levels for a short period of time. This bursty traffic generation can lead to congestion problems, and so an alarm instruction table allows the central data terminal to clear alarm messages out of remote cell node and intermediate data terminal buffers at the end of the alarm. This priority table also allows the utility to tailor the alarm traffic delay suit its particular needs.

Both the priority tables and the alarm instructions are used by the message storage instruction module to properly manage traffic on the network. The message storage instructions maintain the message queue, ensure that response times are within specification, and transmit performance data to the central data terminal to be used for network control.

The network service modules transmit messages to the remote cell nodes, which then use the tables discussed above to organize the message queue. All messages reach the application switch with the specified delay. The central data terminal downloads data to the three control modules and tables as required.

Allocation of Bandwidth to Applications

Many issues should be considered when deciding how the limited available communications bandwidth is divided up and allocated to the various uses required by the wide area communications network. The design of networks should balance operational and performance objectives with available resources. The wide area communications network meets objectives at several levels of abstraction, including:

- low cost network service module design;
- long life for battery powered network service modules;
- high volume, but slow and steady, traffic on network; service module to remote cell node and remote cell node to intermediate data terminal links;
- extra capacity on network service module to remote cell node link to account for contention access;
- multiple copies of NSM messages relayed on remote cell node to intermediate data terminal links;
- low volume traffic on remote cell node to network service module link;
- wide range of delivery time requirements on remote cell node to network service module link; and
- ability to adapt to support future, yet unknown, services.

Bandwidth allocation affects protocol design criteria. Frames are subdivided into equal-sized channels, and channels are partitioned into slots associated with various links. Protocol design parameters include link slot size, channel size, number of channels per frame, and number of frames per cycle. Periods of system quiet time can easily be achieved by simply not assigning a channel or subchannel to any use. This quiet time can be useful in diagnosing communication problems or locating sources of interference.

Application services are granted access to link slots within specified channels of the network service module to remote cell node and remote cell node to network service module links. Access may be exclusive or shared with other

28

services. The number of channels should be large enough to permit a reasonable number of services to co-exist without necessitating shared channel assignment. Total channel capacity on a particular link assigned to an application service can range from several channels to a single subchannel which is a $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$ or $\frac{1}{16}$ fraction of a channel, thus allowing flexible dynamic apportioning of a link to services. Furthermore, channel capacity is allocated in a wide range of units, from several whole channels down to a fraction, $\frac{1}{16}$, of a channel, in a way that is nearly optimal for both extremes. The smallest unit of channel allocation is one channel of one frame each cycle. However, if the number of channels per frame and frames per cycle is large, then the designator used to identify a subchannel requires many bits. Frame size should be small enough to ensure that successive slots of a given channel occur frequently enough such that a service using that channel, which needs to deliver a message within a given time interval, is able to do so. Services requiring larger amounts of bandwidth can be given a few whole channels; those requiring a small amount can be given a single subchannel of appropriate size. The smallest subchannel should correspond to a known service with low bandwidth needs. Beyond that, channel size should correspond to a known service with high volume, such that one or a small number of channels satisfies the service's requirements. In either case, a network service module need only deal with a small number of channel designators to orient itself. Allocation by channel has the added benefit of postponing or experimenting with the options of either having services share a large block of channels or giving each service exclusive access to its own channels.

Each link in the wide area communications network is allocated sufficient bandwidth to support the application services. Again, the channel concept can be used to gain a degree of flexibility. A link may be defined to exist, or have slots, in only certain particular channels. Therefore, all channels need not have the same slot structure. Slot order within a channel may also be constrained, e.g., an IDT-RCN slot containing a poll should come before an RCN-IDT slot for the response, and additionally, a delay is required from the end of one slot to the start of the next to allow the RCN time to formulate its response, or to switch the receiver off and turn the transmitter on. To reduce network service module complexity, however, remote cell node to network service module channels accessible to network service modules should have a common remote cell node to network service module slot structure. Basic slot size and structure are fixed at design time, and careful analysis can avoid problems. Dynamically resizing and/or repositioning slots within time channels is also possible by broadcasting appropriate parameters, but it is not worth the added complexity.

Additionally, many hardware design criteria impose constraints on the design parameters identified above, particularly link slot size. To assist in determining optimal values for the above parameters, it is useful to estimate traffic volume on the various links, as well as message delivery time requirements.

Physical Layer

The wide area communications network is a hierarchical network employing synchronous slotted two-way radio communications. Access to a single RF channel by all nodes is governed by a time division multiplexing (TDM) scheme. The physical layer is responsible for this scheme.

Throughout this disclosure, the following conventions have been employed to simplify the designation of particular links. Link names comprise three letters. The first letter

5,673,252

29

designates the source node, the second letter designates the destination node, and the third letter designates the link type. Source and destination nodes are limited to intermediate data terminal (I), remote cell node (R), and network service module (N). Link types may be report (R), download (D), synchronization (S), header (H), request (Q), and special control (C). Using this convention, the remote cell node to network service module request link, for example, would be designated RNQ; the intermediate data terminal to remote cell node header link would be designated IRH, and so on.

FIG. 14 summarizes, by way of example, the TDM strategy. The primary unit of time division is a one second channel. The system has 30 such channels, numbered S, 0, 1, . . . , 28. A 30 second interval, called a frame, is thus divided into 30 one second channels. The one second channels are divided into slots, and each of the various links has its own predefined slot within each channel to use. Therefore, each of the various links may be considered as having 30 channels, although some of the channels may not exist for some of the links. In fact, the synchronization (S) channel is used entirely by the physical layer for node synchronization, and is unavailable for use on all the defined links. These links include:

RND—commands from remote cell node to network service module;

NRR—reports from network service module to remote cell node;

IHR—intermediate data terminal polling remote cell node, or announcing a download;

IRD—intermediate data terminal downloading to remote cell node;

RIR—remote cell node responding to intermediate data terminal's poll;

RIQ—remote cell node requesting intermediate data terminal to poll it; and

RNC remote cell node broadcasts special application control (SAC) to network service modules.

The slot structure shown in the FIG. 14 could apply to all channels from 0 to 28, or some of the channels might be defined by other structures. Nodes using the constituent links should be aware of such structures; in this regard, network service modules are aware of only the one structure shown, however, links between remote cell nodes and intermediate data terminals may evolve alternative structures at the expense of increased complexity. A channel may contain a group of three NRR or RND slots, but not both, and although such slot designations can be reassigned, the selected usage is static between channel assignment reconfigurations. Slots carry no explicit indication of their intended usage, rather the channel allocation table (CAT) reflects this in its entries. For example, if a slot belongs to the RND link, then a CAT entry exists telling some type of network service module to listen to that channel, and remote cell nodes also are told to use that channel to transmit certain types of messages to network service modules. The slot shown to be on either an IRD or RIR link also can be one of two at any given time, but usage is under direct dynamic control of the intermediate data terminal which indicates this in the previous IRH slot.

Although CAT distribution is a network layer function, the physical layer should know which channel to use for any receive/transmit operation the channel performs. The network service modules are permitted to transmit their reports in any of a set of full channels, and listen to specific subchannels for downstream commands. Different types of network service modules, or those belonging to different

30

utilities sharing a common network, could easily co-exist and be assigned exclusive channels. A network service module's designated channels would not be permanently assigned, but a CAT entry would be broadcast on a dedicated subchannel so that network service modules could be kept informed of dynamic reconfigurations of channel assignments. Upon deployment, a network service module would be told where to look for its CAT entry. A CAT entry contains a network service module type, e.g., 8 bits, a bitmap of the allowed upstream channels the network service module may use, e.g., 28 bits, and two subchannels designated for downstream commands, e.g., 12 bits each. A multi-service network service module may need separate CAT entries for each type of service. Battery-powered network service modules, and those requiring fast response time, could be assigned a CAT distribution subchannel in such a way that they need only receive one packet to obtain their entry. CAT entries for line powered network service modules could be placed in a circulating list, sorted by network service module type and with some indication of the length of the list, in a subchannel shared by several types of network service modules. CAT entries would be obtained by receiving successive packets from the subchannel until the required entry is found.

Note that each slot can contain one data link packet, and that all such packets are prefixed by a fixed length preamble which is used by the physical layer to recognize the start of the packet. In addition, adjacent slots are separated by a guard time to allow for timing inconsistency across the network. This relaxes the need for network service modules to be perfectly synchronized. A packet can begin early or late, within limits, and still be correctly located and received.

TDM Subchannel Plan

A time domain channel may be subdivided into 16 subchannels allowing one to allocate small portions of total bandwidth, approximately 0.2%, to applications with low bandwidth requirements.

The purpose of cycles is to be able to subdivide a channel into smaller subchannels. Specifically, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$ or $\frac{1}{16}$ fractions-of-a-channel-size subchannels may be defined. For example, a $\frac{1}{2}$ channel of channel three includes the channel three slot in every second frame, or alternatively, channel three in only those frames with an even frame number. This is illustrated in FIG. 15, where the dark frames are the ones which contain the specified subchannel (XXX0). FIG. 15 also illustrates $\frac{1}{4}$, $\frac{1}{8}$ and $\frac{1}{16}$ size subchannels, the partitioning of a single channel into several unequal sized subchannels, and the two RND{0} slots out of a cycle's 480 one second channel periods, which comprise the subchannel RND{0}.3.X000.

For example, RND{0}.3.XX10 denotes a $\frac{1}{4}$ sized subchannel of channel three on the RND{0} link. The "XX10" specifies both the size of the subchannel and the exact subset of frames, from each cycle, which belong to that subchannel. Basically "XX10" is a binary number specifying that a frame whose number, in binary, matches the 0's and 1's represents a frame which contains a slot belonging to the subchannel. In this example, frames 2, 6, 10 and 14 contain slots belonging to subchannel RND{0}.3.XX10. Note that this subchannel specification scheme does not exclude the possibility of a subchannel with non-uniformly spaced slots, such as RND{0}.3.00XX, or of overlapping subchannels, such as RND{0}.3.XXX0 and RND{0}.3.X010. Situations may exist in which both of these properties are desired. The following convention is adopted: All subchannels with which network service modules deal are of the form: XXXY,

5,673,252

31

XXYY, XYYY or YYYY, where the Y's represent 0's or 1's. Furthermore, only channels zero to seven are used to create subchannels visible to network service modules.

The three downstream RND slots can represent distinct remote cell node to network service module links, while the three upstream NRR slots per channel are simply part of a single link. Network service modules permitted to transmit in a given channel can do so in any of the three NRR slots.

Periods of system quiet time can easily be achieved by simply not assigning a channel or subchannel to any use.

Physical Layer Data Packets and Slots

The physical layer uses a common conceptual model for time slots, where each slot carries a single data link packet of certain fixed or maximum size.

The wide area communications network physical layer is defined to provide slotted packet delivery service to the data link layer. This is an extension of the usual scope of the physical layer, which provides a mechanism for transmitting and receiving streams of binary values from which the data link layer constructs its packet structure. The time division multiplexing is the jurisdiction of the physical layer, and since the notion of finite sized, labeled, slots are visible to the data link layer, a packet delivery service is an obvious consequence. Additionally, network nodes generally do not continually monitor the radio channel, and the network nodes generally do not have sufficiently accurate local clocks to remain in constant synchronization with the TDM structure. The physical layer identifies slot boundaries by transmitting and recognizing certain signal patterns prefixed to the data link layer packets. The exact nature of such preamble fields depends on the performance of transceiver hardware.

Data channel slots are characterized by the four fields shown in FIG. 16. The leading guard field of eight milliseconds separates consecutive slots and permits remote cell nodes to wake up from power conserving states and enter either a transmit or receive state. The guard field also gives network nodes time to reconfigure their transceivers between slots. The preamble is used to establish slot and bit synchronization. The data link layer packet carries an arbitrary sequence of bits formatted as determined by the data link layer. Network nodes attempting to receive the packet in a slot should begin listening a few bits away from the start of the preamble, therefore problems are unlikely should packet data contents look like the preamble. A short trailing guard field allows for network service module time drift within a frame, permits the data link layer packet to be completely transmitted and received before the start of the next slot, and lets a transmitter hold the final data bit an extra half to a full bit time to prevent sampling receivers from experiencing a glitch on the last sample of the last bit. Data link layer packets are either of fixed length, or have an explicit length field, therefore the trailing guard is not needed to act as a "stop" field.

FIG. 17 describes the slot characteristics of the seven different types of data channel slots. Different bit rates are used on the various links, and the values shown in the figure reflect this. A data link packet's opening flag field may also be used by the physical layer to recognize the start of the packet.

Services Provided to Data Link Layer

The physical layer provides data transfer and time reference services to higher layers. The physical layer at the

32

network service module provides a number of services to the data link and network layers. These services include: obtaining network synchronization, as explicitly requested by a higher layer; maintaining current time, between synchronizations, and returning it upon request; checking for and, if present, receiving a packet of specified length, or receiving bytes until told otherwise, from a specified channel or subchannel; locating the start of a specified slot within a specified channel and transmitting a specified preamble followed by a series of bytes of a data link packet passed to it; and detecting abnormal conditions and aborting its action with an appropriate status return code upon detection of such abnormal conditions.

In addition to these network service module services, the physical layer at the remote cell nodes also provides additional services. These services include returning operational status flags and counters upon request; returning a received signal as a digitized analog waveform; obtaining the IRS slots from the intermediate data terminal for every frame; generating the RNS and RNH slots to the network service modules for every frame; listening, at least for so long as required to conclude no packet is present, to all network service module to remote cell node slots in a prespecified set of channels, but not necessarily all channels, and making the ensuing packets available to higher layers; and receiving and passing on to a higher layer, all packets arriving in the IRH slot.

The physical layer at the intermediate data terminal also provides additional services. These services include returning operational status flags and counters upon request; returning a received signal as a digitized analog waveform; generating the IRS slot every frame; and listening to the RIQ slot in every channel, which has an RIQ slot, and passing on any received packets to higher layers.

Should alternative slot structures be defined for some channels, the remote cell node and intermediate data terminal physical layers should be aware of these and should be able to, generally: (a) receive or transmit in a particular slot on demand, (b) continuously listen to specified slots, and (c) transmit some fixed, or simply derived sequence, repeatedly in specified slots. This capability would thus support remote cell node-to-remote cell node and intermediate data terminal-to-intermediate data terminal interactions, except, perhaps for intermediate data terminal synchronization. It is not required that the remote cell node and intermediate data terminal physical layers be field reprogrammable in this regard, but rather that the above mentioned capabilities at some future time become part of their required operation.

High Level Physical Layer Synchronization Philosophy—IRS Slot

The synchronization (S) channel, illustrated in FIG. 18, is dedicated to the physical layer, and is used to achieve frequency, slot and time synchronization across the network. Intermediate data terminals maintain accurate frequency and time references and distribute reference signals to remote cell nodes, which in turn relay these to network service modules.

IDT-RCN Synchronization

Intermediate data terminals have accurate radio frequency and time clocks to which the rest of the network synchronizes. The primary vehicle for synchronization distribution is the synchronization (S) channel. Remote cell nodes receive synchronization messages from the IRS slot transmitted by the intermediate data terminals. Remote cell nodes

5,673,252

33

in turn transmit synchronization messages to network service modules in the RNS and RNH slots.

The S channel IRS slot, illustrated in FIG. 19, is transmitted by intermediate data terminals to remote cell nodes and contains the following fields:

time synchronization and frequency reference patterns, 112 bits;

flag preceding the data packet, 8 bits;

a data packet containing time and frame information, 40 bits total, including 8 spare;

day-of-week, 0..6, 3 bits;

hour, 0..23, 5 bits;

minutes divided by 4, 0..14, 4 bits;

frame number, 0..15, 4 bits,—note that frame number also provides the least significant 2 bits of minutes, and indicates seconds as being either :00 or :30;

flags for today and tomorrow, indicating whether day light savings time is in effect; and CRC, 16 bits.

Every 30 seconds, intermediate data terminals perform an internal synchronization procedure which, since the synchronization procedure involves the use of the intermediate data terminals' RF receiver and transmitter, can be performed during RND/NRR slots. Ideally the synchronization procedure should occur just before the IRS slot in the S channel and, therefore the synchronization procedure is scheduled to occur during the first half of channel 28. Although time information could be delivered to remote cell nodes in some other fashion, since the frame number is needed and must also be protected, i.e., with a CRC, having at least as many data bits in the packet as there are bits in the CRC is not a drawback.

Remote cell nodes are able to search for and recognize the time synchronization and frequency reference patterns, to obtain frequency, frame and time synchronization. A remote cell node which is totally disoriented does not transmit; however, once in synchronization the remote cell node may be able to operate properly even if an occasional one or two IRS slots are missed. Special attention is given to ensure that the remote cell nodes synchronize their clocks with the IRS slot.

IDT-IDT Synchronization

Maintaining synchronization among intermediate data terminals is a significant problem. The framing structure initially provides a special subchannel dedicated to support this endeavor. In the case of paired RF channels, intermediate data terminals can synchronize using the master RF channel while remote cell nodes and network service modules use the other RF channel. Subchannel IIS.0.0000, i.e., a $\frac{1}{16}$ subchannel of channel 0 on the IDT-to-IDT link, illustrated in FIG. 20, can be used to synchronize all intermediate data terminals relative a designated master intermediate data terminal. Another slot or another subchannel may be required if some intermediate data terminals have to relay this synchronization on to those intermediate data terminals which cannot hear the master intermediate data terminal. The IIS slot is similar to the IRS slot.

NSM Synchronization—RNS Slot

The RNS slot of the S channel is intended to be a large, easily recognized start of frame marker which network service modules can find without much effort. Once located, the RNS slot provides both a radio frequency reference and subsecond time synchronization to the start of frame.

The RNS slot is used for two purposes. The first is to provide a frame synchronization pattern which is easy to

34

find, easy to recognize, and not easily confused by normal data packets. The second is to provide a frequency reference. To achieve the first objective, the slot is as large as possible and is filled, illustrated in FIGS. 21–22, with a repeated series of 46 of the following 18-bit RNS fields:

a synchronization pattern (01111111110) which is an easy pattern to recognize and doubles as a frequency reference, e.g., 12 bits;

a count of the number of RNS fields remaining up to the RNH slot (6); and a final field just before the RNH slot, containing:

22 1's used to verify frequency sync; and

a 4-bit RNS closing flag, or RNH opening flag.

The RNS slot fields need not be CRC protected. The synchronization is fixed, and successive count fields are sequential values and are inherently redundant.

Network service modules obtain synchronization from the RNS slot under two different circumstances. The first arises when the network service module needs to re-synchronize before receiving or transmitting, but while the network service module is still reasonably in synchronization with the network. In this case, the network service module pops up more-or-less where the RNS slot is located, and begins its search from there. The expectation is that the RNS slot is located within the first or second probe. The second circumstance when synchronization is necessary is when the network service module is totally lost and needs to find synchronization. In this circumstance, the network service module begins its search at an arbitrary point in time. Regardless of the circumstances, once the network service module begins its search for the RNS slot, the network service module follows the same steps, specifically: (a) the network service module listens, or probes, for a while to see if the network service module hears the synchronization pattern; (b) if no, the network service module turns off the network service module receiver, waits and then tries again; (c) when the pattern is found, the network service module scans the pattern and receives the count to the start of RNH field, with the count being in units of RNS fields; (d) the network service module may then either check the next count field to confirm synchronization, or may simply delay until the start of the RNH field; and (e) receive the RNH field.

Time Distribution to NSM-RNH Slot

The RNH slot in the S channel, illustrated in FIG. 23, is used by the remote cell nodes to deliver frame identification and current time of day to the network service modules. The RNH slot contains time, frame and protocol information passed from the remote cell node to the network service modules, and includes:

day-of-week, 0..6, 3 bits;

hour, 0..23, 5 bits;

minutes divided by 4, 0..14, 4 bits—note that frame number also provides the least significant 2 bits of minutes, and indicates seconds as being either :00 or :30;

frame number, 0..15, 4 bits;

day-light savings time flags, today, tomorrow, 2 bits total;

active CAT ID, 0.3, 2 bits,—four channel allocation tables may be defined at any one time, but only one can be active at any given time. If network service modules can record their entries from more than one of these tables, then the wide area communications network can instantly, or at least on a frame boundary, switch to

5,673,252

35

another table that may be set up to handle certain exceptional circumstances which may appear quickly and last for a short while; these circumstances may require a sudden redistribution of bandwidth to allow an impending surge of messages, e.g., alarms, to get through. Not all network service modules need to maintain four CAT entries; for those which can only handle one, when the system shifts to another table, that network service module is not allowed to transmit or receive. Alternatively, there may be some commonality between the tables such that the "dumber" network service modules go on as usual; or, for example, channels 0-15 may only have one CAT entry while channels 15-28 may have dual entries, with the "dumber" network service modules being limited to using only channels 0-15.

CAT version sequence number, 0..15, 4 bits,—this field tells the network service module when the channel assignments have changed and a new CAT entry is obtained;

spare bits reserved for future definition, e.g., 8 bits; and CRC, e.g., 16 bits.

The fields included in the RNH slot have been chosen to minimize total network service module receive time. Since the network service module must periodically look for frame and frequency synchronization, it might as well pick up time information at the same time rather than going to a separate subchannel to pick it up. The CAT version number field allows the network service module to obtain its CAT entry only when necessary, rather than periodically picking it up.

Data Link Layer Definition

The intermediate data terminal-remote cell node data link is more conventional than the remote cell node-network service module link. The data link layer is typically concerned with the encapsulating of network layer messages into packets, and the reliable transfer of packets across a physical link with the necessary synchronization, error and flow control. Data link protocols are generally concerned with the use of a single physical link directly connecting two network nodes.

RCN-NSM Link

The wide area communications network data link layer does not require or support many of the capabilities found in more conventional communications networks. The link is connectionless. Alternatively, one may consider all connections as being fixed and permanent. Either way, a requirement does not exist for opening/closing connections dynamically. Network service modules are not polled, since they transmit autonomously. Message delivery is not guaranteed, although most are transmitted more than once to improve the chance of success. Individual packets are not acknowledged, although higher layers may obtain direct or indirect confirmation of reception. The data link layer does not ensure received packets are provided to the higher layers in the same order as transmitted, although higher layer messages are either self-contained or explicitly numbered if broken up into multiple packets. Flow control generally does not exist between individual network service modules and remote cell nodes. Network service modules normally only generate a handful of messages per day, and receive messages even less often.

The data link layer does perform data link address recognition, including global broadcasts. It also provides error control by including, in all packets, a CRC check field

36

that is verified upon reception. Packets failing CRC verification are usually counted and discarded. Remote cell nodes also provide the option of capturing a packet as a digitized analog signal.

IDT-RCN Link

The intermediate data terminal—remote cell node link is a more conventional master/slave polled access link or, in HDLC parlance, an unbalanced normal response mode. As master, the intermediate data terminal is responsible for protocol integrity; initiating all dialogues, determining when retransmission, in either direction, is necessary, and taking corrective actions when the dialogue gets out of synchronization. Data link address recognition and CRC error detection are conventional. Packets sequence numbers; acknowledgements, by returning the sequence number of the last packet successfully received; and packet retransmission are employed. Separate sequence numbers are used for individually addressed and broadcast streams of packets. Flow control is exercised inasmuch as each packet must be acknowledged before the next one can be sent, except for intermediate data terminal to remote cell node broadcasts and RIQs.

Since confirming that all remote cell nodes have received a broadcast can take a relative long period of time, several packets may be sent before requiring acknowledgement; this may be by sliding window, selective reject ARQ protocol. The global nature of such broadcasts requires that the CDT be responsible for coordinating the IDTs to ensure that all transmit in unison, and for determining when retransmission is necessary.

A remote cell node can asynchronously send a packet on the RIQ link, like HDLC's unbalanced asynchronous response mode or ARM, requesting that the remote cell node be polled.

Data Link Layer Definition

Several terms are used to define the data link layer and are defined herein.

Flow Control: Flow control is a technique for assuring that a transmitting station does not overwhelm a receiving station with data.

Stop-and-Wait Flow Control: Stop-and-wait flow control is a technique that prevents a source node from transmitting another packet until the destination node has acknowledged the current packet.

Sliding-Window Flow Control: Under sliding-window flow control, each packet is labeled with a k-bit sequence number, and up to $n=2^k-1$ packets may be transmitted before the source node must stop and wait for acknowledgement. The destination node acknowledges one or more packets by sending to the source node a packet indicating the sequence number the destination node expects to receive next.

Error Control: Error control refers to the mechanisms for detecting and correcting errors that occur in the transmission of packets. Packets may be lost when interfering circumstances affect the destination node to the extent that the destination node is not aware that a packet has been transmitted. A damaged packet is one which is received but has bits in error. These mechanisms are referred to as automatic repeat request (ARQ), and the following error control techniques are commonly employed:

Error Detection: typically a CRC is used.

Positive Acknowledgement: receiver sends back an ACK for successfully received, error free packets.

5,673,252

37

Retransmission after timeout: source node retransmits a packet that has not been acknowledged after a pre-determined amount of time.

Negative Acknowledgement: destination node sends back a NAK for packets in which an error is detected; on broadcast links, node can only NAK when next successfully received packet indicates that one or more intervening ones were not received.

Stop-and-Wait ARQ: Stop-and-wait ARQ is based on stop-and-wait flow control. The destination node sends back ACK or NAK for each packet received, and the source node must use timeout in case either original packet or acknowledgement of the original packet is lost. Packets are sequence numbered, typically 1-bit minimum, in case the source retransmits due to lost ACK. When this technique is adapted to sliding-window flow control, the technique is referred to as continuous ARQ.

Go-back-N ARQ: Go-back-N ARQ is a continuous ARQ variant based on sliding-window flow control. If multiple packets are transmitted, and one is lost, i.e., source node times out waiting for ACK/NAK, or is damaged, i.e., destination node sends back a NAK, then that packet and all those which came after it are retransmitted. If an ACK is lost or damaged, but a subsequent ACK is sent before the source node times out, then the later ACK is cumulative and no retransmissions are required. Packets must be sequence numbered, and up to $n=2^k-1$ ACKs may be outstanding.

Selective-reject ARQ: Using selective-reject ARQ, the only frames retransmitted are those that receive a NAK or which time out. Both source and destination nodes are more complex as they must be able to store and handle packets out of sequence. The destination node uses sequence numbers no determine if a packet has been lost, and can go on receiving subsequent packets. A window size of $n \leq 2^k-1$ is required to ensure there is no overlap between source and destination windows.

Attaining Reliable Data Transport Over the RCN-NSM Links

A data link with contention multiple access and packet error detection, but never any acknowledgements, relies on other techniques to attain a high probability of message delivery.

Packet radio transmissions occasionally are subject to loss or corruption due to noise, interference or collisions. Network nodes which fail their built-in self-test stop radio transmission. The system performs error control at several levels of abstraction.

Error Detection and Correction

Received signals match particular physical synchronization patterns which prefix the message before being considered as packets, thus filtering out noise and some corrupted messages. Bit patterns used to create the frame/channel structure also are selected to prevent misinterpretation, either due to noise or because some part of a legitimate message looks the same. In general, a two level scheme may be employed where two patterns, separated by a fixed time interval, match. All packets are further protected by encoding them using a cyclic code, 16-bit CRC, which affords a degree of information redundancy. Although not required, an error correcting code can be used to recover from certain types of error, e.g., to achieve single error correction with a Hamming code. Single error correction could also be achieved with just the CRC, at considerable computational cost, using a brute force search method.

Path Redundancy and Collision Avoidance

Another form of redundancy exists in the system, namely multipath redundancy. On the NRR link, several remote cell

38

nodes potentially may be able to hear the same network service module's transmission, and since different propagation paths are taken, and in the case of collisions, different capture ratios will exist, some of the remote cell nodes may successfully receive the message. On the RND link, several remote cell nodes are transmitting simultaneously, thus a temporarily blocked path between a particular network service module and remote cell node need not prevent the network service module from hearing the message from another remote cell node. In the case of packet collisions on contention access links, e.g., NRR and RIQ, random temporal distribution algorithms tend to spread messages over time to help avoid such collisions, and the stronger of the two colliding messages may in fact be received correctly due to the capture effect.

Message Redundancy and Self-sufficiency

The application layer for AMR services employs message redundancy as another measure to help ensure the reliable transport of data. FIG. 24 illustrates various forms of redundancy. Most data are reported several times over an extended time period, and are reported in a cumulative fashion to further reduce the impact of any lost messages. Downstream commands to network service modules are also repeated multiple times. These commands are generally absolute in nature; commands which "toggle" a network service module parameter are avoided.

General Data Link Packet Structure

The design of data link packets used on the links within the wide area communications network is loosely modeled on the HDLC standard. A primary consideration is efficient usage of the relatively low data rate channels, i.e., small time slots.

FIG. 25 depicts the structure of a well-known standard data link packet, the HDLC frame format. Using this structure, flag fields demark the start and end of the packet, and implicitly define the length of the variable length information field. The source or destination address is specified in an expanding, in byte units, address field. An 8-bit, or 16-bit, if larger sequence numbers are desired, control field identifies the type of the packet, e.g., information, supervisory or unnumbered, and contains send and receive sequence numbers. Several kinds of supervisory functions are defined to manage flow control and request packet retransmissions. The unnumbered type packets are used to negotiate operational modes and carry other sorts of data link related status. The structure is designed for a bi-directional point-to-point connection where all packets are acknowledged and where a station may have multiple transmitted packets with acknowledgements outstanding.

Each of the various wide area communications network links has a data link packet structure which is loosely modeled on the HDLC format, but optimized to the special needs and purposes of the particular link. When designing data link packets generally, a physical layer preamble precedes the packet; this preamble is required to recognize the start of a slot. The bit values of a received preamble may or may not be made available, and packets end at the final bit of the CRC. However, the physical layer requires that a transmitter transmit at least one bit having an arbitrary value after the final bit of the CRC. A flag pattern marks the start of a data link packet. This flag pattern is considered part of the data link packet, but the physical layer also uses the flag pattern to recognize the start of the slot and then makes this field available to the data link layer.

When designing data link packets of a specific type, packets except for IRD and RIR are a fixed length, and

5,673,252

39

packets sizes are byte multiples, except RND and RNC. An IRD link packet is special in that it is always paired with the previous IRH packet, as shown in FIG. 26. RND link packets are special in that their structure depends on CAT subchannel assignment. Additionally, each link has associated with it an intrinsic form of addressing: NRR, RIR and RIQ packets contain source node address fields; IRH packets contain destination node address fields which may be broadcast address; IRD packets implicitly refer to the same destination address as the prior IRH packet; RND addressing depends on subchannel use designated by the CAT; and RNC is implicitly a broadcast-only link. Finally, the data link layer shares a control field with the network layer, all packets are protected by a 16-bit CRC check field, and cardinal numbers are transmitted most significant byte and, bit within byte, first. FIGS. 27-30 illustrate RIR data link packet structure, IRH data link packet structure, NRR data link packet structure, and RIQ data link packet structure, respectively.

Data Link Packet Structure—RND and BNC Links

The RND link is used to deliver commands and operating tables to network service modules. Various methods of delivery and addressing are employed, and each has its own data link packet structure. The CAT designates what types of application messages can be carried in a particular subchannel, and indirectly specifies the data link packet format; all packets sent in that subchannel must have the same method of delivery, and hence the same structure.

The wide area communications network data link layer uses three delivery mechanisms for messages on the RND link. The first, broadcast to class address, is used for messages which are supposed to be received by all network service modules belonging to a particular class, usually based on network service module type. FIG. 31 illustrates an RND broadcast to class address data link packet structure. The second, broadcast to individual address, is used for messages intended for one individual network service module; the message is periodically transmitted and the network service module is expected to eventually hear the message. The third, reverse poll, is also used for messages intended for one individual network service module, but a message is not transmitted until the network service module sends a message explicitly requesting the reverse poll. FIG. 32 illustrates an RED broadcast to individual address and reverse poll data link packet structure. One delivery mechanism is used on any particular subchannel, and a network service module listening to that subchannel knows which mechanism is being used.

Generally the data link layer address on a data link packet also is considered to be the network layer address for the network message carried in the data link packet. However, this does not preclude the network layer from having an explicit network layer address subfield of the network message field as well. In fact, tiered addressing, for load management network service modules, is supported in this fashion, with the data link layer employing broadcast to class address for these messages, and the network layer having additional address fields.

RNC Link

The RNC and RIQ links share a common slot in every data channel. The intermediate data terminal can issue a special application control command, in an IRH packet, which is echoed by the remote cell nodes in the immediately following RNC/RIQ slot. FIG. 33 illustrates an RNC broadcast special application control data link packet structure.

Operational Measurements

Data link layer modules keep account of normal and exception data link events. These statistics are used for system tuning and fault diagnosis.

40

Each layer of the communications network, at each node, collects a statistical summary of network operation. These quantities help the network manager decide if the system is operating properly, determine values for tunable parameters, and locate the cause of failures. Statistics kept include:

- number of packets transmitted
- number of packet retransmissions
- number of packets received with valid CRCs
- number of packets with CRC errors
- number of protocol violations

Additionally, higher layers can ask the data link layer to relay packets with CRC errors or packets introduced with a bad preamble along with CRC validated packets, in decoded binary form, or for any packet as a digitized analog signal.

Interactions with Network and Physical Layers

The data link layer 97 of FIG. 34 uses the services of the physical layer 98 while providing services to the network layer 96. The physical layer 98 services used include waiting for the beginning of a particular slot of a particular channel or subchannel; selecting frequency; transmitting a stream of bytes; receiving a stream of raw decoded data bytes; receiving a stream of digitized samples of the analog signal, along with decoded values; aborting an operation in progress; obtaining an accurate value for current time; obtaining physical layer status and operational measurements; and explicitly requesting that the physical layer synchronize with network.

Services provided to the network layer 96 include encapsulating a network message in a packet and transmitting the network message in a particular slot; receiving packets from pre-specified slots, verifying CRCs, extracting and buffering the network message fields, and returning them upon request; returning current time; and returning status and operational measurements, including those from the physical layer 98.

Overview of the Network Layer

The network layer in the OSI model has the responsibility for message delivery from source to destination across a single network, possibly passing through one or more intermediate network nodes. The network layer 96, illustrated in FIG. 34, of the wide area communications network performs several functions not found in conventional communication networks.

The purpose of the network layer 96 is to provide a generalized communication mechanism to support application layer requirements. Control of the entire network is considered to reside in the network layer. As claimed by the present invention, the network layer 96 encompasses end-to-end transport issues associated with OSI transport and network layers. The network layer 96 isolates higher layers from the data transmission and switching technologies used to establish, maintain and terminate end-to-end connections between systems, which may include a number of data link connections in tandem if the systems are not directly connected. The network layer 96 provides a message delivery service to higher layers and also carries application layer messages.

The wide area communications network is designed to primarily support connectionless data communication on a hierarchical network architecture. With this design goal in mind, the network layer functions may be considered to provide:

5,673,252

41

node addressing
 routing
 strategy for polling remote cell nodes by intermediate data terminals
 packet format, related to network control
 packet priority—message delay related
 channel allocation table (CAT)—message throughput related
 upstream message (report) delivery
 downstream message, e.g., command delivery
 broadcast to class or tier address
 broadcast to individual address
 reverse poll
 eavesdropping, a form of redundant message reduction
 network control, downloads to remote cell nodes
 operational measurements

Since message sizes are quite small, in order to maximize the amount of information which is carried, network, data link, and in some cases even application layer message fields are sometimes shared across layers, rather than including a strict encapsulation of higher layer messages within lower layer packets/frames. Indeed, at times it appears that lower layer fields are visible to higher layer entities.

Addressing Within the Wide Area Communications Network

Network nodes within the wide area communications network have unique network addresses, but each type of node has a separate address space. A node's identification (ID) serves as the node's network, data link and physical layer address. In some circumstances, a node may have a different application layer identification known only at the central data terminal, or the node may have a second class or tier address to which the node responds.

Individual Node Addresses

Addresses can be considered data link addresses relative to the links used, i.e., NSM addresses on NRR/RND links, and RCN addresses on IRD/RIR. links, but may also be considered as network addresses as there are no individual physical point-to-point links between nodes. Nodes are uniquely identified with binary integer addresses. FIG. 35 summarizes the address space for each node type. FIG. 36 is an example of tier address specification and selected/non-selected network service modules. Address space size is considerably larger than any network conceivable. Network service module addresses need only be unique within each network service module type, thus if an 8-bit field specifies NSM type, then the NSM address space is potentially 256 times that shown in FIG. 35.

Upstream messages are implicitly marked for delivery to the central data terminal, whereupon they are redirected to appropriate application processes on the basis of content, i.e. NSM and message type, primarily. Generally, downstream messages contain only destination addresses, upstream messages only source addresses. Remote cell nodes accept messages from network service modules that they hear. When polled, remote cell nodes relay upstream messages to anonymous intermediate data terminals, but intermediate data terminals hearing a remote cell node's response, or an RIQ, first identify the source remote cell node as being one under their control. Each intermediate data terminal may also have another address associated with the IDT-CDT network.

42

Broadcast Addressing

Several variations of broadcast addressing are employed. Due to the broadcast nature of the radio medium, messages are broadcast, even if there is only one intended recipient. For remote cell nodes, and intermediate data terminals should they exist on a multicast medium, the all 1's address is reserved to indicate a broadcast to all. The set of remote cell nodes to which such broadcasts apply depends on which intermediate data terminals transmit the message. If all intermediate data terminals broadcast the message simultaneously, then all remote cell nodes have the opportunity to hear the message. If only a single intermediate data terminal transmits, then the message may be considered to be directed to only those remote cell nodes within hearing range of that intermediate data terminal. The contents of a message may be defined so as to further restrict the message to a particular group of remote cell nodes.

For network service modules, broadcast addressing arises in several different circumstances, and is handled differently for each. Some information is intended for all network service modules, and is identified only by the slot/subchannel the information occupies; no address is specified. Some information is intended for only one type of network service module. The information may be identified either by subchannel only or by using the network service module type as the address. Some information is intended for only one network service module, and network service module type and address are required; alternatively, type may be omitted if implied by subchannel. Finally, some information is intended only for a subset, or tier, of the network service modules of a particular type. In this case, all network service modules which recognize a tiered address have, in addition to their normal ID, a 24-bit tier address assigned to them. A tiered address, on a transmitted packet, includes two parts, the first is a 24-bit pattern and the second is a 24-bit mask selecting which of the pattern bits must match corresponding bits of a network service module's assigned tier address for that network service module to be addressed. The type of network service module addressing employed is predetermined for each type of message delivered, and network service modules know this. Within a given subchannel, only one NSM addressing mechanism is used.

Routing Within the Wide Area Communications Network

Message routing in a hierarchical network is straightforward considering that all routes are direct. But the multicast nature of downstream delivery introduces another level of network wide synchronization, namely that between the intermediate data terminals or that between the remote cell nodes. Reverse poll delivery to network service modules is a type of network traffic which requires the making of significant routing decisions.

In the upstream direction, network service modules transmit to no remote cell nodes in particular, and any remote cell nodes hearing the message relays it to any intermediate data terminal which polls it, and the intermediate data terminal will relay it to the central data terminal when polled. In this regard, the only decision to be made is which intermediate data terminal should poll any given remote cell node.

In the downstream direction, messages destined for an individual intermediate data terminal or remote cell node have a direct path from the central data terminal, assuming each remote cell node is assigned to a single intermediate data terminal. Broadcast messages directed towards remote cell nodes would be transmitted in unison by all intermediate

5,673,252

43

data terminals. One fundamental concept permeating the entire design is that downstream communication is performed mostly by several network nodes transmitting the same information in unison. One consequence of this involves intermediate data terminals broadcasting to remote cell nodes. Since the central data terminal-to-intermediate data terminal link most likely is not a multicast medium, the central data terminal is not able to talk to all intermediate data terminals simultaneously. Therefore, as the central data terminal talks to each intermediate data terminal separately, the central data terminal schedules the intermediate data terminals to begin their broadcasts to remote cell nodes at a common future time. Intermediate data terminals similarly schedule remote cell nodes, since all remote cell nodes transmit in unison most messages destined for network service modules.

The opportunity to make any real routing decisions comes with reverse poll delivered messages. Such messages are downloaded to only those remote cell nodes, say up to four, likely to hear the reverse poll of the target network service module. Compared to having all messages go to all remote cell nodes, remote cell node memory requirements are reduced and message throughput is increased when network service module reverse poll messages are delivered to individual remote cell nodes, but at the expense of performing this delivery to each candidate relay remote cell node separately. The routing table, at the central data terminal, is also an expense, in that for each network service module it becomes necessary to record which remote cell nodes normally hear the routing table.

The complete route includes:

Central data terminal—source of delivery

Intermediate data terminal(s)—which are in charge of the remote cell nodes best able to execute final delivery

Remote cell node(s)—which normally hear the target network service module, and are chosen to perform final delivery

Network service module—target of delivery

For the route to be effective, some sort of reciprocity should exist between a network service module and the remote cell nodes which hear it, as a remote cell node can deliver a reverse poll message only if it hears the network service module's reverse poll.

One possible alternative to individualized delivery of reverse poll messages to remote cell nodes is to broadcast all messages to all remote cell nodes. A second alternative would require that each remote cell node keep a list of the network service modules the remote cell node normally hears, and although all reverse poll messages would be broadcast to all remote cell nodes, only those remote cell nodes which hear from a target network service module would store the message for delivery later. For low traffic systems, all messages could be broadcast to all remote cell nodes.

Still another approach might be to correlate network service module visibility to intermediate data terminals which poll those remote cell nodes which hear the network service module. In this way, the central data terminal would route the reverse poll messages only to the intermediate data terminal(s) associated with the target network service module, and only the remote cell nodes which can hear that intermediate data terminal would get the NSM message. It may now be beneficial to consider letting the remote cell nodes know which intermediate data terminal they are talking to so that those remote cell nodes which can hear an intermediate data terminal, but which are not polled by the

44

intermediate data terminal, know not to act on the download. The argument against telling remote cell nodes which intermediate data terminal is polling, is that if an intermediate data terminal fails, other intermediate data terminals can take over its remote cell nodes without the remote cell nodes being aware, other than perhaps noting a sequence number discontinuity. However, intermediate data terminals could be set up to "impersonate" the failed intermediate data terminal. The best apparent compromise is to have intermediate data terminals give their IDs to remote cell nodes under their charge so that downloads could be targeted to only those RCNs belonging to a particular intermediate data terminal. The intermediate data terminal ID would only be used to initially configure or re-configure a remote cell node and for specific types of downloads, such as reverse poll messages.

RCN Polling Strategy Used by IDTs

The remote cell node polling strategy used by the intermediate data terminals includes the order in which an intermediate data terminal polls its remote cell nodes, the way the intermediate data terminal reacts to RIQs, the manner in which downloads and polls for status are interspersed with polls for messages, and the coordination of adjacent intermediate data terminals so that only one uses the RF channel at a time.

The intermediate data terminal polls its remote cell nodes according to a set of rules designed to guarantee minimum polling rate for individual remote cell nodes; dynamically adjust attention proportional to need; respond to RIQs in a timely manner; provide appropriate attention to μ RTUs; apportion the half-duplex links (IRD & RIR) appropriately; and periodically collect remote cell node (μ RTU) status reports. To expand upon the need to dynamically adjust attention proportional to need, if one remote cell node has 20 messages and another has 200, clearly the latter should be polled more frequently, until both remote cell nodes have similar buffer sizes. Similar concerns arise in the context of high priority messages.

An objective of the system is to be able to have intermediate data terminals in different parts of the network poll their remote cell nodes concurrently. If intermediate data terminals are near each other, or if a remote cell node is within range of multiple intermediate data terminals, then the polling strategy also provides for methods which avoid concurrent polls/responses from interfering with each other. One approach is to allow intermediate data terminals access to the IR link in only centrally assigned channels. Another approach involves the use of directional antennae at the intermediate data terminal, with auto-configuration providing information as to which antenna to use for each remote cell node. A third approach gives the central data terminal complete control over all intermediate data terminal operations. All approaches have ramifications on the polling strategy employed.

Alternative approaches to the remote cell node polling problem include having the central data terminal construct and deliver polling schedules to all intermediate data terminals. Under this alternative, intermediate data terminals are not permitted to deviate from the schedule, but inform the central data terminal of all pertinent polling status, including RIQs, and the central data terminal periodically adjusts the polling schedules accordingly. The polling schedules are designed to accommodate incremental updates, and hence avoid wholesale redistribution for every little change.

A second alternative requires that the central data terminal instruct intermediate data terminals as to which channels

5,673,252

45

each can use and when, avoiding co-channel interference both during the intermediate data terminals' polls/downloads and during the RCNs' responses. Under this alternative, the intermediate data terminals make all other decisions as to which remote cell node to poll and when to poll that remote cell node. Simultaneous broadcasts from all intermediate data terminals to all remote cell nodes would still be under direct control of the central data terminal.

Delivery of Broadcast Messages to NSMs

Under a broadcast message delivery strategy, remote cell nodes repeatedly transmit lists of messages, and network service modules periodically listen to see if any of those messages are for them. This method is used when messages are delivered to multiple network service modules at the same time, or when messages are delivered with minimal delay. FIG. 37 lists common fields of broadcast messages.

Broadcast to Class Address

Broadcasting to class addresses delivers the same information to a large group of network service modules. FIG. 38 illustrates an example of time of use timetable and schedule broadcast to class address. The method is used both for small-sized commands requiring rapid reception, and large amounts of relatively static or slowly changing information; as an example of the latter, network service modules may take up to a week to acquire updated information. Delivery delay is a function of the circulation period and how often network service modules listen to the remote cell node.

FIG. 37 lists the kinds of fields that comprise such messages. The actual message formats may vary from one message type to another, and network service modules know the structures of all message types that they intend to receive. Remote cell nodes repeatedly circulate a series of such messages in consecutive slots of a subchannel designated for that purpose. Generally, each message applies to only one type of network service module, and so one unique subchannel is assigned to each network service module type and only messages pertaining to the same network service module type appear in that subchannel. In this case, the same message type codes may be used for different types of messages appearing in different subchannels. This does not preclude having a single subchannel carry messages for multiple network service module types, however the message type codes used must be unambiguous, or the network service module type is included as part of the message. The message type field itself may be omitted if it is the only type carried in the subchannel. The number of different subchannels required is anticipated to be small, and the number of messages in each list also is anticipated to be small. Small sized subchannels can be used.

Any particular message may have additional content fields that further restrict the delivery to a specific subset of recipient network service modules. The network service modules of the specified type receive such messages, but only those satisfying the selection criteria actually accept delivery and act upon the message. In this fashion, tiered addressing is a specialization of the general category of broadcast to class addressing.

Broadcast to Individual Address

Broadcasting to individual addresses delivers single packet messages to individual network service modules, and for accomplishing this with minimal delay, i.e., under 15 minutes 90% of the time. This is another specialization of the general broadcast delivery mechanism. All messages

46

contain the address, and possibly the network service module type, of a single destination network service module. The list of messages circulated in a subchannel may be sorted by address to permit a network service module to exit early from a futile search. The number of subchannels required is expected to be small, but lists may be quite long so larger size subchannels may be necessary. A particular message would only circulate for a limited length of time, long enough to give a dedicated listener several opportunities to hear it. To achieve minimal delivery delay, target network service modules must listen often, or perhaps continuously, to the subchannel. FIG. 39A and 39B illustrates an example of service reconnect broadcast to individual address.

Delivery of NSM Broadcast Messages to RCNS

Batches of messages destined for network service modules via broadcast delivery are delivered to all remote cell nodes along with parameters specifying when to begin and how long to transmit the messages so that all remote cell nodes operate in a coordinated fashion.

Distribution of network service module broadcast messages to remote cell nodes should achieve the following objectives:

- 1) the broadcast delivery mechanism should be capable of delivering some messages with relatively short delay from time of original dispatch at the central data terminal;
- 2) messages have a finite life time during which they are transmitted a certain number of times;
- 3) message life times range from minutes to weeks;
- 4) remote cell nodes circulate lists of messages;
- 5) messages may be added to or removed from the list, or new entries may replace current ones, at arbitrary times; while it is possible to have all messages expire at the same time, this is not required and, since the new list may have many messages in common with the old one, it would be desirable to avoid downloading the entire list for every change;
- 6) all remote cell nodes should broadcast the same messages at the same time and, when the set of messages changes, having some remote cell nodes broadcast the old set while others broadcast the new set must be avoided.

Although the fifth objective may be satisfied by allowing lists to grow and shrink in size as required, circulating the list as often as possible, such an approach would make attaining the sixth objective quite difficult. One way to satisfy the sixth objective is illustrated as Algorithm A. Algorithm A takes a variable length of time, perhaps minutes, to confirm that all remote cell nodes have received the changes to the network service module broadcast list. If fixed length lists are tolerable, then the simpler distribution strategy shown as Algorithm B can be used. A fixed number of slots are used to circulate the list of messages; slots can be empty. Each network service module message is transmitted by the remote cell nodes only within an explicitly specified time interval. The central data terminal is responsible for keeping track of when slots become available for new messages. The central data terminal may choose to divide the list into equal sized blocks, batch up network service module messages, and then dispatch them one block at a time at regular intervals. For example, considering that there are six RND slots per minute in a full channel, a list of 48 messages can be transmitted in eight minutes. If the list is divided into four blocks of twelve messages, and each block has a 32-minute life span, the central data terminal

5,673,252

47

could stagger the start of each block by eight minutes so that it feeds the network twelve new messages every eight minutes, but all messages still are circulated for the full 32 minutes. At this rate, the central data terminal can transfer 2,160 messages over the course of an entire day. However, keeping messages sorted by destination address would be difficult, if even possible. Message delivery rates and life spans can be varied dynamically by the central data terminal, but changes to list size requires the same degree of coordination as Algorithm A. Maximum list length is constrained by remote cell node memory size.

Algorithm A

- i. intermediate data terminals broadcast changes to all remote cell nodes, perhaps multiple times, along with switch-over time
- ii. remote cell nodes hearing the broadcast continue transmitting the old list until switch-over time
- iii. intermediate data terminals confirm that each RCN has heard the broadcast; re-deliver if necessary
- iv. remote cell nodes still transmitting after switch-over time stop immediately upon hearing broadcast
- v. once all remote cell nodes confirm receipt, as determined by central data terminal, intermediate data terminals broadcast switch-over command
- vi. if a remote cell node hears the switch-over command before the switch-over time, it switches from old to new list at the switch-over time without any interruption of delivery
- vii. if a remote cell node does not hear the switch-over command by the time of switch-over, it stops transmitting list at the switch-over time
- viii. if a remote cell node only hears the switch-over command after the switch-over time, it starts transmitting the new list, but must begin at the point in the list where it would have been if it had begun at the switch-over time
- ix. intermediate data terminals confirm that all remote cell nodes have heard the switch-over command; re-deliver if necessary.

Algorithm B

- i. intermediate data terminals broadcast to all remote cell nodes the NSM message list, or portion thereof, along with start of broadcast and total life times; start time follows or coincides with end of life of previous contents
- ii. remote cell nodes hearing the broadcast begin and end transmission at specified times
- iii. remote cell nodes not hearing the broadcast go silent at end of current list life
- iv. intermediate data terminals confirm that each remote cell node has heard the broadcast; re-deliver if necessary
- v. if remote cell node only hears broadcast after start time, it begins transmitting the new list at the point in the list where it would have been if it had started on time.

In general, since all remote cell nodes must broadcast in unison, some or all of the following parameters will accompany each NSM message handed to the remote cell node for final delivery:

- subchannel in which to deliver message;
- when to transmit for the first time, required by both algorithms;

48

message expiration time, for Algorithm B—a special code could be defined to indicate that the message be circulated indefinitely, e.g., TOU timetables, however this can only work if some mechanism similar to Algorithm A exists to eventually replace such messages;

position in list, Algorithm B;

ordering criteria, Algorithm A—messages could be sorted by NSM address.

Delivery of Reverse Poll Messages to NSMs

The reverse poll method is intended for the delivery of messages for which fast delivery is not required. It is designed to conserve energy at both the network service module and the remote cell node, while using channel bandwidth more efficiently than the broadcast to individual address delivery mechanism. The circulation of long lists of addressable messages is avoided at the expense of storing messages at the remote cell node end delaying delivery until the network service module polls the remote cell node, which may be hours or even days. In general, when an network service module transmits a report, the network service module has the option of asking the remote cell node if there are any messages for the network service module.

Delivery of reverse poll messages is illustrated in FIG. 40. Each network service module which can potentially receive a command via reverse poll delivery has a polling schedule, e.g., once every ten reports, so that periodically one of the network service modules report packets requests a reverse poll. If a remote cell node which receives this report has a command for that network service module, based on NSM type and address, then the remote cell node transmits the command at a mutually agreed upon future time. The network service module listens at that time. Both the network service module and the remote cell node compute the delivery time using a common hashing function which can be based on NSM address, time of the original report, slot used for the original report, and/or an arbitrary field in the report. The delay until delivery should be long enough to permit a battery-powered network service module to recharge; for line-powered network service modules, the minimal delay could be shorter. The delay is a relatively easy to compute function, and the command is delivered in a specific subchannel. The network service module knows which subchannel based on its CAT entry, while the remote cell node is explicitly told by the central data terminal; the delivery subchannels are bundled with the commands downloaded to the remote cell node. In the event of a hashing collision, which occurs when two or more reverse poll messages happen to be scheduled for delivery in the same slot, either one message can be picked and delivered, or one message can be delivered in the selected slot and the second one can be delivered in the next successive slot in the same subchannel. Collisions of an order greater than two are expected to be extremely rare. A network service module hearing someone else's command could then listen in the next slot. In either case, priority, i.e., whether to be the first or the only delivery, should be given to battery-powered network service modules.

Possible alternatives and/or additions to the above scheme include removing a specific message from the remote cell node's list once the remote cell node has delivered that message n times; removing any messages a remote cell node may have for a specific network service module if the remote cell node has non been contacted by that network service module in any way for m hours; giving every message an expiration time regardless of the number of times in has been

5,673,252

49

transmitted or whether there has been contact from the network service module; giving messages that have been delivered at least once eligibility for early removal under the direction of the central data terminal; including the amount of space left in the remote cell node's message buffer in the status report provided to the central data terminal; allowing the remote cell node to store in its buffer only one message per network service module, resulting in new messages superseding old messages; allowing the remote cell node to store in its buffer multiple messages per network service module, resulting in the remote cell node delivering the next message each time the network service module does a reverse poll while also indicating to the network service module whether the remote cell node has more undelivered messages for the same network service module; including a network service module's acknowledgement of receipt of a message in the network service module's next report; and allowing the central data terminal to specify additional criteria which is satisfied by the upstream messages containing a network service module's reverse poll before the corresponding downstream message will be delivered.

Delivery of Network Service Module Reverse Poll Messages to Remote Cell Nodes

Messages slated for reverse poll delivery to network service modules are downloaded to remote cell nodes most likely to perform successful delivery, along with parameters specifying delivery subchannel and message disposition under various circumstances.

NSM reverse poll messages can be distributed to remote cell nodes in one of at least two ways:

- α) broadcast all messages to all remote cell nodes.
- β) deliver only those messages supposed to be handled by a remote cell node to each remote cell node individually.

Several factors are considered in evaluating the above alternatives. These include the length of time any given message remains in a remote cell node's memory, the average rate for messages actually delivered by any one remote cell node, the size of the remote cell node's message buffer, the average system-wide message delivery rate, and the amount of intermediate data terminal-remote cell node traffic required to distribute the messages to the remote cell nodes.

Evaluation of the first factor, how long a given message stays in a remote cell node's memory, is as follows. If a network service module performs a reverse poll on average every a hours, then an average delay of $a/2$ hours exists from the time the remote cell node gets a message until the first opportunity to deliver the message. Furthermore, if the remote cell node delivers the message more than one time before discarding the message, then the message stays in the remote cell node's memory for a total of $(n-1)a$ hours, with the delay for the last delivery being on the average $a/2$ hours. For $a=8$ and $n=3$, the average storage life of a reverse poll message is 16 hours.

The second factor, average rate for messages actually delivered by any one remote cell node, can be evaluated as follows: If a network of r remote cell nodes has N messages to deliver per day, and k remote cell nodes hear each network service module, then on average each remote cell node delivers kN/r messages. For $r=2500$, $N=5000$ and $k=4$, each remote cell node actually will deliver around 8 messages per day.

The third factor, size of the remote cell node's message buffer, varies greatly between the alpha and beta alterna-

50

tives. Under the alpha, if the network has to deliver N messages per 24 hour day, and each message is stored for $(n-1)a$ hours, then in the best case a buffer of size $M=N(n-1)a/24$ is required. For the amounts reached in the discussion of the first and second factors, this works out to be the value 3334. Under the beta, a buffer size two or three times larger than the average found in the discussion of the second factor should suffice to handle the messages actually delivered by a remote cell node.

The fourth factor, the average system-wide message delivery rate, also varies greatly between the alpha and beta alternatives. Under the alpha, if the maximum number of reverse poll messages a remote cell node can store, M , is fixed then the equation from the discussion of the third factor can be used to determine a maximum value for N and hence the maximum average system-wide delivery rate. For $M=1000$, a rate of 1500 messages per day is obtained. Under the beta, if the formula from the discussion of the second factor represents a half (or third) of M , solving for N with $M=1000$ yields a rate of 312,500 (or 208,333) messages per day.

Finally, the fifth factor, intermediate data terminal-remote cell node traffic required, can be evaluated for the alpha alternative as follows: In the best case, each of the N messages must be broadcast only once to the remote cell nodes—several message may be blocked together in a single download. Once the remote cell nodes' buffers become full, the central data terminal explicitly indicates which old messages to overwrite, since remote cell nodes have no way of knowing which messages were delivered by the others. The beta alternative, by contrast, can be evaluated as follows: Each message is delivered to each of the k remote cell nodes slated to perform final delivery, nominally taking k times longer than broadcasting to all. If several, distant, intermediate data terminals can poll/download to their own remote cell nodes concurrently, then several different network service module messages can be distributed at the same time. Generally, because the k remote cell nodes chosen to deliver a particular message must all be "close" to the target network service module, the remote cell nodes are likely to be under the control of the same innermediate data terminal, and instead of sending each network service module message to each remote cell node separately, k remote cell node addresses can be attached to the network service module message and the network service module message need be sent only once. These two techniques may combine so that this alternative may actually take less time to distribute the N messages than simply broadcasting.

Summarizing the evaluation of these factors indicates that alternative β is superior for the following reasons. RCN buffer requirements are much smaller and higher system-wide message throughput can be achieved; RCN message buffer size is the limiting factor for the alpha alternative where the buffer would have to be almost as large as the number of messages desired to be delivered system-wide in a day. From the remote cell node's perspective, the distribution process is simpler because the remote cell node does not have to be told to discard/replace anything already in the RCN buffer. The distribution process is more complex for the central data terminal because the central data terminal has to decide to which remote cell nodes the central data terminal should send each message. Finally, if several intermediate data terminals can transmit different packets at the same time distribution bandwidth requirements may even be less than broadcasting to all remote cell nodes.

An intermediate data terminal, or perhaps several concurrently, broadcasts a download containing a list of

5,673,252

51

remote cell node IDs associated with each NSM message. All remote cell nodes hearing this broadcast receive the NSM message, but only those identified in the download actually store the NSM message. A message is removed from a remote cell node's buffer after it has been delivered the required number of times or if its specified expiration time has passed. Depending on message and packet sizes, two or three such NSM messages may be blocked in a single download, however, unless NSM messages at the central data terminal or intermediate data terminal are held until all such blocks are filled, NSM messages are likely distributed one-at-a-time as they become available.

The following parameter items accompany NSM messages to the remote cell node, but are not delivered to the network service module:

- subchannel used for final delivery;
- indication of what algorithm, e.g., hashing, to use to determine the delivery slot;
- number of times to deliver message before discarding it;
- time before message expiration in the event of non-delivery; up to 31 hours, or days;
- hashing collision priority, i.e., messages for battery-powered network service modules get higher priority;
- in the event of a hashing collision, whether chaining is permitted or not;
- additional criteria the corresponding upstream message with reverse poll bit set must satisfy before delivery can occur, such as matching a service disconnect password.

Remote Cell Node Contention Access Request for Intermediate Data Terminal's Attention

Remote cell nodes can use the RIQ link to transmit a request for service from the intermediate data terminal. All remote cell nodes contend for access to this link.

Under certain conditions, a remote cell node may decide that the remote cell node needs to be polled by the intermediate data terminal sooner than the intermediate data terminal's next normally scheduled time. The RIQ link is a contention access link which can be used by any remote cell node to transmit an RIQ message requesting attention from the intermediate data terminal. The RIQ message identifies the remote cell node and contains an indication of why the request is being issued, in the form of remote cell node status fields. The two main reasons why a remote cell node might transmit an RIQ are because its message buffers are getting full, or because it has high priority messages (alarms) to relay. With adequate remote cell node polling, the former should not occur very often, and the latter is also not expected often either, except that during a large area power outage many remote cell nodes have high priority alarms to relay, and activity may be considered on the RIQ link. A field could be added to the IRH message which would allow intermediate data terminals to broadcast special instructions regarding the use of the RIQ slot during power outages, perhaps directing remote cell nodes to increase the time delay, lower the priority of power outage messages, or discard such messages. In the case of filling buffers, the threshold should be tuned so that the intermediate data terminal's reaction to the RIQ need not be very fast. In the case of high priority messages, delay in responding to the RIQ is dependent on the requirements of the application originating the message.

Once the intermediate data terminal receives an RIQ message, the intermediate data terminal decides whether or not the remote cell node should be serviced by an early poll,

52

or if the next regularly scheduled poll will be soon enough. Once the decision to perform an early poll is made, the intermediate data terminal determines the next available opportunity to do so, given that actions of all intermediate data terminals must be coordinated. Clearly, until the remote cell node is polled, the remote cell node has no assurance that its RIQ was even heard by the intermediate data terminal, and after a suitable timeout waiting to be polled, the remote cell node re-issues the after a randomized back-off interval in case the first RIQ collided with an RIQ message from another remote cell node.

If polling cycles turn out to be short, or if polling is constrained to a very rigidly coordinated sequence, as may be necessary to avoid inter-intermediate data terminal and/or inter-remote cell node interference, then the concept of a remote cell node requesting "immediate" attention may not be viable, and the RIQ link may be ineffectual.

Message Redundancy Reduction

The existence of multiple paths from one network service module to several remote cell nodes markedly enhances the chance that any particular NSM message will be heard by at least one remote cell node, but at the expense of increased traffic arising from relaying multiple copies of the same message when several remote cell nodes successfully receive it. Remote cell nodes eavesdropping on each other's uploads is a technique which may be employed to reduce redundant messages. Other techniques considered are also documented.

Message redundancy is a fundamental feature of the wide area communications network contributing to the system's ability to achieve a high probability of reception of NSM messages. However, relaying multiple copies of the same message, heard by multiple remote cell nodes, is undesirable, and may not even be possible if a large number of remote cell nodes hear each network service module. The number of remote cell nodes which receive a given NSM message, formerly called overlap, is estimated to be three or four, but could be higher.

To emphasize the need for redundancy reduction, consider an intermediate data terminal servicing 25 remote cell nodes, polling each remote cell node every 50 seconds (this is optimistic), and collecting five NSM messages per poll, resulting in a maximum out-flow of six msgs/min. One estimate of remote cell node in-flow (dependent on a number of assumptions) is 9,000 messages/day, or 6.25 messages/min. Perfect redundancy reduction would mean that the remote cell node would be required to relay only one or none of these.

With eavesdropping, remote cell nodes listen to the reports neighboring remote cell nodes transmit to the intermediate data terminal, and discard NSM messages that a neighbor has already relayed. Eavesdropping remote cell nodes need not confirm reception at the intermediate data terminal as that is the responsibility of the polled remote cell node. RIR messages are structured so that the eavesdropping remote cell node need not receive the entire report to determine if it has any of the reported NSM messages, thus reducing the energy required to perform this function. NSM messages are uniquely identified by: nsmtyp, nsmadr, msgtype, and msgno fields. The message priority is useful in localizing the search to the appropriate queue.

While redundancy reduction is needed, message redundancy reduction does have an undesirable side-effect, namely the destruction of information allowing the central data terminal to determine which remote cell nodes hear a

particular network service module. However, just the random variation of which remote cell nodes relay messages from a particular network service module may be sufficient to identify the remote cell nodes. Alternatively, designating a particular message type, such as CSTAT which is transmitted, once per day, as being non-discarded is another way to identify which remote cell nodes hear the network service module. However, more detailed analysis of hearing patterns would require that eavesdropping be temporarily disabled. Remote cell nodes would keep a count of the number of NSM messages discarded due to eavesdropping.

Performance of any redundancy reduction technique may be evaluated based on several criteria. These criteria include amount of reduction achieved; cost in terms of electrical energy, RAM, and communication overhead; sensitivity to level of redundancy, which is a function of propagation conditions; computational complexity and network management required to support the endeavor; and the risk of completely discarding an NSM message.

Eavesdropping is expected to eliminate at least half of the redundant messages, require 10–20% of the remote cell node's power budget, need no additional RAM, require a significant amount of computation to determine neighbor tables for each RCN, and require a small amount of communications to distribute these tables. Alternatively, eavesdropping can be carried out on the basis of signal strength of the reporting remote cell node alone. The technique is applicable to any degree of redundancy. There is no risk of completely discarding an NSM message, but there is the potential for being unfair or unbalanced in that some remote cell nodes may relay more messages than others.

Alternatives to eavesdropping which would also provide message redundancy reduction include edge-gather and partial poll, wider-gather and partial poll, table based message acceptance, random discard, random discard based on signal strength, and signal strength with exception list. Each will be discussed in the paragraphs that follow.

Edge-Gather and Partial Poll

Edge-gather and partial poll requires a remote cell node to poll one or more edge neighbors, search for redundant messages in its local buffer, and then wait to be polled by the intermediate data terminal. An edge neighbor is one that could not be polled directly by the intermediate data terminal. Some remote cell nodes would poll neighbors, others would not. This method has an impact on polling strategy as well.

Wider-Gather and Partial Poll

Wider-gather and partial poll requires remote cell nodes to poll their neighbors in some manner. The messages from 25 remote cell nodes may end up in only four remote cell nodes. This method has impact on polling strategy as well.

Table Based Message Acceptance

Using this alternative, if an NSM address is in a remote cell node's table, then the remote cell node stores it. About three remote cell nodes would have to have an network service module's address on their lists.

Random Discard

Under random discard, some percentage of NSM messages, with the exception of alarms and CSTAT, is discarded on a random basis. Perhaps half of the messages could just be discarded. If six remote cell nodes hear an NSM message, then there is a 98.4% probability of still getting the message through to the central data terminal.

Random Discard Based on Signal Strength

When employing random discard based on signal strength, if the received signal strength exceeds some threshold, the remote cell node keeps the message; if the

received signal strength is in a grey area, then the remote cell node discards the message on a random basis.

Signal Strength with Exception List

Using signal strength with exception list, if the received signal strength exceeds some threshold, the remote cell node keeps the message. For those NSM messages which have low signal strength at all remote cell nodes hearing them, the ID of the broadcasting network service module would be downloaded to several remote cell nodes, and any of those remote cell nodes hearing that network service module would keep the message regardless of signal strength.

General Network Message Structure

Network messages are encapsulated within data link packets and the address and control fields are generally common to both layers. The network layer distinguishes between messages which carry, using various delivery mechanisms, application layer messages, and those messages used to control the operation of the network itself. The network layer of the wide area communications network message structure is not based on any particular standard.

Different sets of network messages are defined for each type of network link. A message may contain an explicit message type subfield of the control field, or the message type may be implicitly determined by the channel or sub-channel over which it is transmitted, with that channel only carrying a single type of message. If present, a message type field may in fact be inherited from the application layer message, just as address and control fields, which are defined to belong to the data link layer, may be considered as being inherited from the network layer.

Network messages used to transport application layer messages also contain control information instructing nodes how or when to perform final delivery. The network layer does not know how to interpret the content of such application layer messages. Other network messages used to control network operation are fully defined within the network layer and carry no application message.

As an application message is relayed from node to node by the network layer, different forms of network messages may be used on each successive link, fields may be rearranged, and fields containing message attributes may be attached to or removed from the original message.

Discussion now turns to the description of network messages in terms of the content of the "control" and "network message" fields of corresponding data link packets for each link.

Network Service Module Report Messages

An NRR message (NRRM) contains a network service module application layer report and is encapsulated in a data link packet. Subfields of the control field exist for application message type, and for application specific use such as sequence numbering, which is not used for ARQ. The "network message" field is identical to the application message. FIG. 41 illustrates an NRR network message structure in the context of a data link packet.

Control Fields

Description:

msgtype—application message type. The network layer does not interpret msgtype, but uses this field to derive message priority and, along with msgno, to identify individual messages for the purpose of eavesdropping.
msgno—message sequence number. Increments modulo 16 with each message transmitted by a network service

5,673,252

55

module, independent of msgtype. Used by the network layer to help identify and count lost messages and, along with msgtype, for eavesdropping; msgno is not used for message acknowledgement or retransmission.

revpoll—reverse poll. Considered within the jurisdiction of data link layer and used to request delivery of reverse poll commands, if there are any.

protocol status—can be used to report simple indication of command reception, perhaps one bit each set if the network service module has recently successfully received a broadcast (class or individual) or reverse poll command—used by head-end to estimate delivery delays. Reverse poll indicator could be used as an ACK to free up RCN command buffer space. Indicators for broadcasts could persist a fixed time or number of messages; indicator for reverse poll could persist until next reverse poll request.

priority—message priority as defined in the later section entitled, "Summary of Message Priorities."

Network Message Field

This field contains application report data (RDATA) as defined for msgtype within NSM type. It should be noted that the data link and network layers are closely related, sharing the address and control fields in common. When a remote cell node relays a network service module's report in an RIR message to an intermediate data terminal, it is broken into two parts, NRRMa and NRRMb, to facilitate eavesdropping.

Messages for Network Service Modules—RND

An RND message contains a network service module application layer command and is encapsulated in a data link packet. Subfields of the control field exist for application message (command) type, and for application specific use such as sequence numbering; which is not used for ARQ. The "network message" field is identical to the application command, except for CAT distribution.

Broadcast to Class Address Messages

FIG. 42 illustrates an RND broadcast to class network message format in the context of a data link packet. A complete message includes: optional address (nsmtyp—Data Link packet field), message (coffend) type, application specific control subfield and an application message. The address may only be omitted, and "network message" field enlarged by 8 bits, if the delivery subchannel is dedicated to a single network service module type. Each application is responsible for any message sequencing performed.

Delivery to Individually Addressed Network Service Modules

Delivery to individually addressed network service modules is identical in principle to the broadcast to class address except the nsmadr portion of the data link address field must be present, resulting in a "network message" field which is 32 bits smaller. This message structure is used both for broadcast to individual address and for reverse poll delivery mechanisms. FIG. 43 illustrates an RND broadcast to individual address and reverse poll network message format in the context of a data link packet.

CAT Distribution

CAT distribution is just a special case of broadcast to class address, where the "network message" field contains only network layer, i.e., no application, data. CAT distribution is in a dedicated subchannel, so msgtyp is omitted. FIG. 44 illustrates network message format used to distribute CAT entries, in the context of a data link packet. The CAT distribution message fields include:

56

nsmtyp—NSM type.

appltyp—application subtype within NSM type; permits delivery of multiple CAT entries to a single network service module, each intended for a specific functional subprocess of the network service module, e.g., TOU or load survey; while the invention as disclosed gives each NSM type a single CAT entry, applcyp is included now to provide for future extension.

CAT ID—permits rapid switching from one CAT to another.

CAT version—CAT version number. This is used to identify old vs. new versions of the CAT.

NRR channel bitmap—bitmap with one bit corresponding to each of the channels 1 through 28 in which the network service module is allowed to transmit its report on the NRR link. Any NRR slot with the specified channels may be used.

RND broadcast subchannel—identifies the channel, subchannel, slot and subchannel size the NSM is supposed to listen to for broadcast to class address commands directed towards it, or for broadcast to individual address commands for individual network service modules. The network service module knows whether the network service module is supposed to expect class or individual address command delivery.

RND reverse poll subchannel—as above, except for reverse poll commands for individual network service modules.

Due to the relatively small RND packet size, the network layer structure of messages going down the RND link is highly application dependent, allowing optimization of the use of available bits. FIG. 45 illustrates the format of a subchannel designator.

Remote Cell Node Report Messages

An intermediate data terminal can poll a remote cell node to have the remote cell node either relay NSM messages, or report various kinds of internal status. NSM messages are rearranged in an RIR report to facilitate the eavesdropping method of redundancy control.

Remote cell nodes respond to polls for messages by sending a block of up to five messages at a time. These RCN report messages are structured to minimize receiver energy of neighboring remote cell nodes who are using eavesdropping to perform message redundancy control. The fields from the NSM message which uniquely identify it are placed first in the RCN report, followed by an intermediate CRC. Eavesdropping remote cell nodes can stop listening once they receive this CRC. The rest of the NSM message content comes after that. Remote cell node transmit energy is further minimized by making these report messages variable length. The maximum number of NSM messages which fit in the report depends on how many additional tag fields are requested by the intermediate data terminal, and the report message size varies because an integer number of tagged NSM messages may be smaller than the maximum size of the network message field of an RIR data link packet. Remote cell nodes which only have fewer than this number of NSM messages to relay, transmit a shorter report message. FIGS. 46–49 illustrate RIR network message format used to relay NSM messages in the context of a data link packet, RIR network message subfields comprising the data link control field, subfields comprising the RCN status field, and the maximum number of NSM messages per RIR, respectively.

As shown in FIG. 46, RCN reports carrying NSM messages use the RIR data link packet structure in a specialized fashion:

5,673,252

57

control—field of data link packet
 msgtype—also implies a certain type of item contained in the message (8 bits)
 nitems—the number of items ($0 \leq n \leq 15$)
 seqref—poll/response reference number (4 bits)
 seqbcst—broadcast sequence number (incorporates selective ACK/NAK) (4)
 seqind—message sequence number (per individual RCN) (4)
 length—length in bytes of network message field of data link packet
 NSM msg IDs
 NRRMa₁ . . . n—list of n NSM message IDs
 intermediate CRC—from start of data link packet
 RCN status
 msgs.s—indication of number of NSM messages in RCN buffer (4)
 alms.s—indication of number of NSM alarm messages in RCN buffer (4)
 RCNST—RCN sensor status (8)
 msgfmt—NSM message format (indicates tags attached) (4)
 priority—highest actual priority of NSM messages contained (4)
 NSM msg contents (msgfmt indicates which tags are present)
 NRRMb₁ . . . n—list of n NSM msgs corresponding to IDs above, each with attached tags;
 ptag—NSM message priority (4 bits)
 vtag—NSM message value (4)
 dtag—indication of date of message reception (day of week only) (3)
 etag—indicates message received with CRC error (1)
 ttag—indication of time of message reception (16)
 ftag—measured frequency of NSM message (8)
 stag—measured signal strength of NSM message (8)
 crc—original CRC received with NSM message (16)
 ctag—measured correlation coefficient on preamble (8)
 mtag—measured figure of merit on preamble (?)

RCN Report Messages—RCN Status

An intermediate data terminal can poll a remote cell node for the remote cell node to either relay NSM messages, or for the remote cell node to report various kinds of internal status. Status report messages contain remote cell node internal status information.

The remote cell node status report is expected to closely follow the form of the one-way RIST message. New fields which may be introduced for new operational measurements, whether at the physical, data link or network layers, include:

- NSM command buffer space available;
- number of reverse poll messages delivered;
- number of NSM messages discarded due to wide area power outage;
- number of NSM messages discarded due to eavesdropping (per neighbor or otherwise);
- number of packets discarded due to below threshold preamble correlation.

RCN Poll Messages

An intermediate data terminal can use the IRH slot to either poll remote cell nodes for messages or to announce a

58

subsequent download. The IRH therefore directly indicates whether the RIR/IRD slot is to be used as an RIR slot or as an IRD slot. The IRH also can be used to deliver special application control commands to remote cell nodes, which are immediately relayed to network service modules.

Intermediate data terminals control the dialogue with remote cell nodes by either polling individual remote cell nodes for specific information, or by downloading to one or all remote cell nodes at a time. Intermediate data terminals indicate the desired operation in an IRH message, and the following shared RIR/IRD slot is used appropriately for either the remote cell node's response to poll, or the intermediate data terminal's download. The IPH message is carried in an IRH/RIQ data link packet, and contains the following fields:

- slottype—indicates usage of RIR/IRD slot: RIR, IRD, or unused (3 bits)
- seqno—sequence number; interpretation depends on context, one of:
 - seqref—poll/response reference number if polling an individual remote cell node
 - seqind—sequence number if downloading to an individual remote cell node
 - seqbcst—sequence number if broadcasting to all remote cell nodes
- msgtype—type of data being polled for or downloaded
- index—array index when downloading/uploading a portion of a large table
- special application control (SAC)—includes load control SCRAM command
- SAC enable—indicates whether remote cell nodes are to relay SAC field

FIG. 50 illustrates an IRH network message format in the context of a data link packet. The SAC field may be used to deliver special application control commands to participating network service modules. When a remote cell node receives an IPH with a SAC command enabled, the remote cell node relays the command in the following RIQ slot, overriding any RIQ it might be attempting to transmit. Network service modules capable of receiving such special application commands must continuously monitor the RIQ link. If the SAC enable bit is clear, then the SAC field is omitted. FIG. 51 illustrates the subfields comprising the SAC field.

Polls, responses to polls, and downloads to individual remote cell nodes all operate under a stop-and-wait ARQ strategy, hence a single bit sequence number field for error control is sufficient. However, a 4-bit field is used, with the provision that successive messages need not have consecutive, modulo 16, sequence numbers. Any sequence number differing from the last one transmitted represents a new message which is acknowledged with the same sequence number value before the next message can be sent.

For broadcast, due to the long delay to acquire acknowledgements, a sliding window flow control strategy is proposed, using the selective reject ARQ technique for error control. A 4-bit sequence number allows up to eight outstanding messages to be unambiguously ACKed or NAKed selectively. An ACK-n acknowledges all outstanding messages with sequence numbers less than n, up to eight, and says nothing about message n, while a NAK-n also acknowledges messages less than n but explicitly requests a retransmission of message with sequence number n.

RCN Download Messages—Broadcast to NSM Delivery Messages

Download information from an intermediate data terminal to a remote cell node generally consists of either messages

5,673,252

59

to be relayed onto network service modules or instructions regarding the operation of the remote cell node itself. The delivery of messages to network service modules can be characterized as being performed either by coordinated simultaneous broadcast by all remote cell nodes or by the reverse poll mechanism.

FIG. 52 illustrates an IRD network message form for delivery network service module broadcast messages to remote cell nodes. FIG. 53 illustrates the subfields comprising various IRD fields. A list of messages is associated with a particular subchannel. Different subchannel may each have their own lists. A mechanism is required which permits the intermediate data terminal to change the lengths of message lists associated with subchannels. This has to be coordinated across all remote cell nodes. The main objective is to avoid having different remote cell nodes broadcasting different messages at the same time. To avoid this, an IRD message with a new list length could be sent for each subchannel in use, and all remote cell nodes could be required to acknowledge this new list length before it goes into effect. Remote cell nodes coming on line for the first time either after installation or after a RESET also need to be synchronized with the rest of the remote cell nodes already in operation.

The delivery of NSM broadcast messages to remote cell nodes was discussed earlier and described the general mechanism for delivering NSM messages to remote cell nodes for subsequent broadcast delivery. Regardless of the method of addressing used, which may be broadcast to all, some or one network service module, each NSM message is delivered to the remote cell nodes and subsequently transmitted to the network service module in a common manner. Remote cell nodes maintain a fixed length list of NSM messages to be delivered on a given subchannel, and repeatedly transmit the entire list of messages. Parameters describe when and where the NSM message is to be broadcast, specifically:

subchan—delivery subchannel

initialSlot—day, cycle and frame number of first list element for first transmission

lifetime—number of times, frames, message is broadcast before discarding

position—position in list occupied by NSM message

The central data terminal, from which all NSM messages originate, specifies that at most one message be delivered in any given slot. This requires that the initial slot of a new message in a list position does not occur before the lifetime of the message it supersedes has expired.

Remote cell nodes are expected to maintain a list of NSM commands in slot order so that when the required slot turns up, the message is ready for transmission. Generally, the intermediate data terminals deliver new commands for network service modules in advance of the old ones expiring, so remote cell nodes are able to store the new commands until needed.

Intermediate data terminals may use Algorithm A to configure the length of the message list associated with any subchannel. The central data terminal should ensure that the switch-over time coincides with the start of the message list. Remote cell nodes stop transmitting messages past the end of a shortened list, and remain silent for list positions which are "empty".

Under this arrangement, remote cell nodes need not be aware of CAT assignments for individual NSM types. However, there are serious implications involved in switching from one CAT ID to another, such that it may be necessary to declare that only upstream traffic may be subject to multiple CATs.

60

This mechanism for message delivery is intended for a regular scheduled message delivery pattern, and does not allow for preemptive replacement of actively circulating messages. If desired, "holes" may be left in the list so that urgent messages can be inserted at arbitrary times. However, all messages currently under consideration can be suitably scheduled and delivered using this mechanism.

Keeping a list of messages sorted, when messages are directly placed a particular positions, may be a challenge, but may potentially be solved by replacing the entire list at once. Otherwise, unordered lists may have to be tolerated.

RCN Download Messages—NSM reverse Poll Delivery Messages

Distribution to remote cell nodes of reverse poll delivery messages for network service modules requires a different format IRD than for broadcast delivery NSM messages. FIG. 54 illustrates an IRD network message form for delivering NSM reverse poll messages to remote cell nodes. FIG. 55 illustrates the subfield, comprising "parms" field of IRD message in FIG. 54.

NSM reverse poll deliverable messages are broadcast to all remote cell nodes within hearing of a single intermediate data terminal, but the IDs of only those four remote cell nodes slated to perform final delivery are attached to each NSM message. The global RCN address can also be used, if desired. Parameters are used to describe additional delivery details, not passed on to network service modules, specifically:

subchan—delivery subchannel

repetitions—number of times message is delivered before discarding

lifetime—time before message expiration in the event of non-delivery

hashParms—hashing algorithm, priority, and chaining option

criteria—optional criteria NSM reverse poll message must satisfy

Because these IRDs are generally directed to only remote cell nodes controlled by a single intermediate data terminal, another form of remote cell node addressing may be more efficient than using broadcast to all remote cell nodes. For example, the rcnadr field of the IRD could be composed of eight 1's followed by the intermediate data terminal's 16-bit ID. This would result in the selection of all remote cell nodes polled by that intermediate data terminal. Widely separated intermediate data terminals may be able to broadcast to their own sets of remote cell nodes simultaneously. A separate sliding window selective reject ARQ sequence number would be required for each remote cell node.

The criteria field must match the first 16-bits of the NSM message with the reverse poll bit set in order for the command to be delivered. This mechanism is primarily intended as an added measure of security when requesting service disconnection, i.e., when the criteria is an encoded password.

RCN Download Messages—Other Messages

Other IRD message types carry network control information to remote cell nodes. Intermediate data terminals download to remote cell nodes, either globally or individually addressed, various other kinds of control information. This information can include a list of neighboring remote cell nodes, remote cell node CAT entries, priorities assigned to NSM messages by msgtype, and other operational commands and parameters.

5,673,252

61**RIO Message**

A remote cell node transmits an RIQ message to request service from the intermediate data terminal. The content of this message basically identifies the remote cell node and provides an indication of the reason for the request. FIG. 56 illustrates the RIQ network message format used to request service from the intermediate data terminal in the context of a data packet.

The fields of an RIQ are a subset of the RIR network message. There are two main reasons why an RCN might transmit an RIQ. First, the remote cell node's message buffers are getting full, and second, the remote cell node has high priority messages, e.g. alarms, to relay. If necessary, the intermediate data terminal can deduce which is the case by looking at the msgs.s and alms.s subfields of the remote cell node status field.

RNC Message

Whenever a remote cell node receives an IRH network message which contains a special application control (SAC) command to relay to network service modules, the remote cell node does so in the immediately following RNC/RIQ slot. The remote cell node simply relays the SAC without interpretation.

The RNC slot is intended for very occasional use to deliver a very limited amount of command information to network service modules with a very short delay. Network service modules are expected to be listening to every RNC slot. The remote cell node simply takes the SAC field from the IRH, surrounds it with an opening flag and CRC, and transmits.

Summary of Message Priorities

Messages from all levels of the network have an associated priority, with higher priority messages being transferred before those with lower priority. Consequently, higher priority messages will cross the network with less delay than those with lower priority. In general, priority level is encoded as a 4-bit integer with 0 representing the lowest priority and 15 representing the highest priority. There are two priority levels for messages transmitted by network service modules: low (0) and high (12); the latter corresponds only to electric network service module power outage alarms. NSM messages do not necessarily carry their actual priority in the priority field. Rather, actual priority is a function of the priority field as shown in FIG. 57. The actual numeric priority levels assigned to "NSM-low" and "NSM-high" are parameters under central data terminal control. NSM messages may be tagged with absolute priorities, in the range 1 . . . 14, but this is intended for only special circumstances.

As NSM messages are received by the remote cell node, they are appended to the end of the appropriate priority queue. Message priorities are partitioned into two groups, normal messages and alarms, for the purpose of determining when a remote cell node should transmit an RIQ. The priority level defining these two groups is a parameter under central data terminal control.

Actual priority is an explicit field of all messages originating from the remote cell node level on up the network. This allows μ RTUs and RCN or IDT status messages to be assigned priorities if appropriate, and ensures a common criteria governing which messages are to be relayed first. In the case of RIR messages relaying NSM reports, the RIR's priority is that of the highest actual priority NSM message contained.

62

Priorization of downstream traffic arises only as a side-effect of the polling strategy, and command delivery. Generally, it is conceived that an application layer process on the central data terminal decides when to issue a download, and the intermediate data terminals and remote cell nodes relay the messages as soon as they receive the messages, or at explicitly scheduled times.

CDT Network Control Tasks

The network layer controller resides in the central data terminal, and is concerned with network layer control issues including fault management, performance management, operating tables, configuration management, downstream scheduling, and the process of developing a specification.

Fault Management

Faults may occur either at the node level, i.e., improper transmit frequency, low battery voltage, etc., or at the network level, i.e., non-functioning polling system, downloads that consistently do not work. The intent of the fault management function is to offer fault identification and recovery functions to the system while being transparent to the network users. A detailed listing of fault definitions is required before fault management can be designed into the wide area communications network. In order to perform fault management, the proper parameters need to be measured. The steps in the fault management procedure are fault recognition, fault isolation, system or node reconfiguration, and fault recovery.

Performance Management

Even if the network or nodes are not faulty, the performance of the wide area communications network may not meet the specifications for a variety of reasons, such as improper or untuned deployment, untuned operating tables, or improper network algorithms for the type of environment in which the system is operating. The purpose of the performance management system is to allow the central data terminal to correct the performance of the network while maintaining user transparency. Involvement of the user in performance management details can result in improvement in the routines.

Three key functions of the performance management system are to describe the performance of the system in a few calculable parameters, compare the performance of the system with the specified performance limits, and initiate corrective action in those cases where performance is out of specification. Calculable parameters can include eavesdropping efficiency, polling efficiency, average message delay, and download efficiency. Specified performance limits may also be in the form of the calculable parameters, and set upper and lower bounds to the performance parameters.

Network performance is managed through changes to the control cables. These tables are controlled by the central data terminal and downloaded to the target node or nodes.

Operating Tables

Operating tables that can and should be constructed at the central data terminal, and which are related to performance management, include polling tables for the central data terminal; polling tables for the intermediate data terminal; neighbor tables for the remote cell nodes, if eavesdropping is used; priority tables for the intermediate data terminals and the upload direction of the remote cell nodes; and IRLMAP for the intermediate data terminals.

Configuration Management

The configuration listing for the system is a listing of all nodes and their locations and present operating status. The

5,673,252

63

configuration management section can also include tags to indicate the type of messages currently being received and the probability of receiving messages of the various types. Downstream Scheduling at the Central Data Terminal, Intermediate Data Terminal and Remote cell nodes

Within the central data terminal, a download scheduler downloads messages into the network at a rate and at times dictated both by message priority in the download direction and the need for efficient usage of the intermediate data terminal-central data terminal links. For messages targeted to network service modules or remote cell nodes, the network controller assigns control bytes which designate the subchannel or channel in which the message is to be transmitted and the start/stop/other control information which is required before the destination node can execute its transmission. The intermediate data terminal and remote cell node then download the messages within the specified sub-channel or channels. The intermediate data terminal and remote cell node do not make decisions regarding the type of channel or sub-channel which gets a certain message; even in the case of reverse poll, the remote cell node chooses the exact time of the download, but the type of sub-channel used is dictated by the central data terminal. Upstream scheduling is taken care of by the polling schedule, which is either designed at the central data terminal or which is partially or fully dynamic at the intermediate data terminal.

Specification Process

Before beginning the specification of these management tasks it is necessary to provide a more detailed description of the tasks required in each management system; a listing of all network data which is available to the central data terminal; and an assumption about the data items stored in the databases. The listing of network data available to the central data terminal would be a subset of the data dictionary, and would include only those data items transmitted up the system. These network data items include much of the operational measurements which are performed at each layer. Using these documents, a specification could be written in which specific operations of RF available data items can be outlined.

The network layer controller will require a fault management database, a performance management database and a configuration management database. These databases may be integrated into one or may be separate, depending on future design decisions.

Network Layer Control Tables

For the network layer, control tables are required for the RF network to operate properly. The network service module, remote cell node and the intermediate data terminal operate from these tables, and use them to dictate the type of operations they will perform. Each node contains both control tables and operating tables.

IDT Control Tables

The IDT network control tables include Received Message Priority, Delivery, Generator, IDT-RCN Link Map, Download Table, and Channel Allocation Table. There may be two copies of every table, one that is currently being used and one that is currently being downloaded for future use.

RXPRI—Received Message Priority (Upstream)

For each RCN message which is received, the intermediate data terminal checks the message for priority by using

64

the RXPRI. The priority of the message is used to place the message in memory using the memory management routine, and generate an ICQ if requested. Higher priority messages should be easily accessible to the intermediate data terminal, in order that the intermediate data terminal can quickly forward the messages to the central data terminal. An ICQ is an intermediate data terminal to central data terminal request for poll. The use of an ICQ will depend on the type of intermediate data terminal-central data terminal network that is being used.

Delivery

This table is downloaded into the intermediate data terminal from the central data terminal, and is common to all intermediate data terminals in the network. This table must include μ RTU message priorities as well.

Generator

This table is generated by the central data terminal.

IRLMAP—IDT-RCN Link Map

All entries in the CATTBL which are not quiet can be used by the IRLMAP. This map is indexed using frame and channel numbers, and dictates the operation to be performed in this frame and channel. Possible operations include polling; downloading by node addressable methods, to a single remote cell node; and downloading by broadcast, to all remote cell nodes. If polling were allowed, then the intermediate data terminal uses the polling table to specify which poll will occur in this channel. If downloading were allowed, then the intermediate data terminal uses the download table to specify which download should occur in this channel. During quiet times, the intermediate data terminal does not perform any transmit/receive functions on the RF channel. This map should remain constant for long periods of time.

DWNTBL—Download Table

For each frame and channel type, the download table contains the messages that are being downloaded at the present time. A given message may be downloaded in only one frame and channel, for example, in which case it would be sent once every eight minutes.

CATTBL—Channel Allocation Table

The Channel Allocation Table for the intermediate data terminals specifies which IRL channels are to be used and which ones should contain quiet time.

RCN Control Tables

The RCN control tables at the network layer include received message priority, neighbor table, download table, channel allocation table, operating node, and CONFIG. There may be two copies of every table, one that is currently being used and one that is currently being downloaded for future use.

RXPRI—Received Message Priority

The received message priority table is the same table as the IDT.RXPRI except that it does not have μ RTU messages given.

NTBL—Neighbor Table

The neighbor table is initially constructed at the central data terminal based on geographical distance. The neighbor

5,673,252

65

table can also be constructed at the remote cell node itself over the first few days of installation based on signal strength. The signal strengths from up to eight other remote cell nodes are maintained using running averages from the start of network operation.

DWNTBL—Download Table

The messages to be downloaded by the remote cell node to the network service module are listed here, along with the control information which is passed down from the central data terminal. The control information is destroyed when the message is destroyed.

CATTBL—Channel Allocation Table

The channel allocation table for the remote cell nodes specifies

OPMOD—Operating Mode

The operating mode of the remote cell node is set by the central data terminal using basic download methods.

CONFIG

The configuration of an remote cell node is downloaded to the remote cell node by the central data terminal.

NSM Control Tables

The network service module interfaces to the network through the use of the CATTBL. This is the only control table in the network service module.

CATTBL

Each network service module receives a CAT table which is specific to its operation, according to nsmtyp. The use of the CAT table is described elsewhere within this section.

Operational Measurements and Tables

Both the remote cell nodes and the intermediate data terminals perform operational measurements to allow the central data terminal to have some visibility into the network.

IDT Operational Measurements and Tables

The network layer at the intermediate data terminal measures polling and downstream delivery performance. Measurements taken include MSGHD.PLL, MSGHD.RIQ, MSGHD.UIQ, RCNSS, NPOLLA, and NSPLL.

MSGHD.PLL

This measurement represents the number of messages heard since power-up. This value rolls over and it is up to the central data terminal to request the MSGHD field often enough to maintain consistency. This field is incremented once for every ten messages that are received by the intermediate data terminal from polls. It is used to provide an indication of the average number of messages per poll.

MSGHD.RIO

This measurement is the number of remote cell node messages heard in one RIQ slot. This value rolls over and this buffer is incremented once for every message received from a remote cell node in the RIQ slot.

MSGHD.UIO

This measurement is the number of uRTU messages heard in one RIQ slot. This value rolls over and this buffer is

66

incremented once for every message received from a uRTU in the RIQ slot.

RCNSS

The intermediate data terminal should obtain one RSSI value each time the intermediate data terminal receives an RIR message. This RSSI value is inserted into the appropriate RCN address location of RCNSS, using a running average technique. The averaging should be taken over 256 measurements, after which the signal strength can be zeroed. The central data terminal should request the remote cell node signal strength data as needed for configuration evaluation during the central data terminal's performance management routine.

NPOLLA

This measurement is the number of poll attempts. This table records the number of poll attempts per remote cell node since power-up. All values roll-over. This data is used by the central data terminal's performance management routine, and should be picked up every few hours.

NSPLL

This measurement is the number of successful polls. This table records the number of poll attempts which resulted in the correct remote cell node responding. This data should be picked up by the central data terminal's performance management routine every few hours.

RCN Operational Measurements and Tables

The network layer at the remote cell node produces tables that monitor remote cell node operation in the upstream and downstream direction. These tables include MSGHD, MSGDIS, NPOLL, NACKs, and Average Delay for High-Priority Messages.

NSPLL

This table measures the number of messages heard since power-up and rolls over at the top.

NSPLL

This table measures the number of messages discarded due to eavesdropping since power-up and rolls over at the top.

NSPLL

This table measures the number of polls made to a specific remote cell node. Every time a remote cell node is polled, this is incremented. If the remote cell node does not respond with an RIR, NPOLL is still incremented.

NSPLL

This table measures the number of NACKed RIR messages. In those cases in which the remote cell node cannot hear an intermediate data terminal for whatever reason, the remote cell node will not get the poll or the NACK. These conditions must be counted by the intermediate data terminal.

Average Delay for High-Priority Messages from Central Data Terminal to Remote Cell Node Delivery

These numbers are used to monitor the high-priority delivery system of the wide area communications network. The values that are measured aid in identifying areas of

5,673,252

67

improvements for the system, and aid in fault and performance analysis. This value is maintained as a running average. For every download into a remote cell node, whether specific or broadcast, the download looks at the timetag and stores the difference between the entry time and the remote cell node reception time.

NSM Operational Measurements and Table

The network service module measures the downstream delivery quality. This is done by counting the number of successful receptions it obtains.

RNHSUC

After getting frequency synchronization, the network service module attempts to receive RNH. Every attempt is either successful or not successful. The number of unsuccessful RNH attempts and the total number of RNH attempts will both come up through the network; each should be one byte or "nibble". As there are only 16 attempts between roll-over it must come up about every two days. Both nibbles are forced to roll-over at the same time as soon as the number of attempts reaches 16.

RXSUC

Other than receiving RNH, the network service module may sometimes attempt to receive messages which are scheduled on its CAT table. Every receive attempt and every unsuccessful attempt will be counted using one nibble each and will be sent up to the central data terminal. This takes one byte. Both nibbles are forced to roll-over at the same time as soon as the number of attempts reaches 16.

RVPLLSUC

For all messages which must be received using reverse poll, the network service module counts the number of reverse poll attempts and the number of unsuccessful attempts. A reverse poll attempt is defined as any time the network service module does a reverse poll routine. An unsuccessful attempt is defined as a message in the slot which is for somebody else. Both nibbles are zeroed at the same time as soon as the number of attempts reaches 16.

Services Provided to the Application Layer

The philosophy adopted regarding the network-application layer interface is that application processes, especially at the central data terminal, receive and present application messages, addressed to network service modules, to the network layer, which is responsible for scheduling and coordinating actual delivery.

The primary locations at which a significant interaction between the network and application layers exist are at the network service module and at the central data terminal. The former has limited capabilities, and in reality it is likely that the data link, network and application layers will be tightly intertwined. At the central data terminal, upstream message flow is relatively straightforward. Messages arriving at the central data terminal pass through a message routing process which directs them to awaiting pre-registered application processes.

Downstream messages emanating from application processes on the central data terminal are another matter. The network layer insulates application processes from the details of scheduling and coordinating message delivery. The application process simply passes the message it wants delivered, perhaps along with instructions such as how many

68

times to broadcast the message or an expiration time, to the network layer. The network layer decides how and when to perform delivery, perhaps patching up several related messages and then beginning a coordinated delivery to intermediate data terminals and subsequently to remote cell nodes and finally to network service modules.

Network to Application Interface

The interface between the network layer and the application layer only exists in the network service module, μ RTU and central data terminal. The interface at each node is exactly the same. The philosophy used in the layered approach to systems' design is that the messages at each layer are not modified by the layers below, but are delivered in pristine form.

Downstream Direction—Central Data Terminal to μ RTU and Central Data Terminal to Network Service Module

When the application layer passes the network layer a message, the destination of the packet appears at the front of the message, and is used to route the message to the appropriate node. Selection of which slot to transfer the message in is performed by the network layer.

When the message arrives at the destination, the network layer passes up the same packet it received at the application layer. If the network layer were in charge of packetizing a given message due to length or security reasons, it is the job of the network layer at the other end to recompile the proper message before passing it to the application layer at the destination node.

Structural differences do not exist between downloads to the μ RTU and network service module.

Upstream Direction— μ RTU to Central Data Terminal and Network Service Module to Central Data Terminal

The application layer responds to a message according to the protocol in operation for that message type. The same message which is inserted into the μ RTU network layer surfaces at the central data terminal application-network interface.

All tags, whether physical, data link or network, are stripped off at the network controller so that only the correct message appears to the application layer.

Network Layer Databases

The network layer contains databases for configuration control, performance management, and fault management. According to the network layer database access philosophy, the network layer database is accessed by the network layer controller and the network operator. The common users on the network do not have access to this database and cannot read or write to the database. All of the data which common users want is located elsewhere. The network operator participates in the installation entry, but does not have access to write to some of the data items, such as node status, node performance, or time of installation.

Configuration Control Database Items

The configuration control database has a potential size of ten million customers (PG&E); if 41 bytes are used per customer+an extra 9 bytes for overhead, the result is 10M*50=500 Mbytes.

Performance Management Database Items

The performance management database includes entries designed for supporting performance management at both the node and network level. Node performance data includes battery level and other relevant data. Network performance data includes the number of messages heard from each network service module, the remote cell nodes which are

5,673,252

69

receiving messages from each network service module, the average of the last specified number of frequency tags from that network service module, average of the last specified number of signal strength tags from the network service module and the average of the last specified number of time tag errors. Ten million CMEs requires about 20,000 remote cell nodes and about 800 intermediate data terminals.

Fault Management Database Items

The fault management database includes a listing of those nodes which are currently in fault conditions. The insertion of nodes into this database is controlled by the fault identification modules. The ability of the network to respond rapidly to fault conditions and events is the duty of the fault response modules, which recalculate configurations for the intermediate data terminals and remote cell nodes. The fault management database is, therefore, quite small and will usually have no entries.

Database Access Times and Network Layer Processor Sizing

For each message received by the network layer at the central data terminal from the lower layers, some database insertion operation needs to be done. For each received message, a search is assumed to be performed to find the CME address, and that about 15 update operations are performed, using about 20 instructions per update operation. Therefore, assuming a binary search of IM addresses takes about 20 jumps=20*4 ops.=80 ops, about 80 jump ops.+ 15*20=380 network layer operations per message. Assuming a 10 MHz clock, each message requires 380*0.2 μsec.= 0.76 msec.

Network Layer Design—Network Controller Data Flow Diagram

A preliminary data-flow diagram for the central data terminal network controller is shown in FIG. 58A and FIG. 58B. The interaction of the modules is shown, along with a brief description of the software modules required.

Distribution Automation Channel

Distribution automation (DA) services, involving remote control and polled data acquisition, require two-way communication with faster response and greater reliability than automatic meter reading services.

Basic data collection and control services such as meter reading and direct load control can function effectively with indirect acknowledgements and, with the exception of SCRAM commands, message delivery delays of minutes to hours. This is not the case for distribution automation (DA) applications where a human operator is involved. For distribution automation, response to commands should be rapid and deterministic. Generally 10 second response times are the maximum desirable but 15 to 30 seconds might be acceptable for some applications which are routine but infrequent, e.g. capacitor bank switching once or twice per day. Moreover, the probability of lost messages must be lower for distribution automation applications. The ability to co-ordinate IDT transmissions is reduced because of the need for short transmission delays.

FIG. 59 illustrates a command message format that is compatible with a specific protocol.

FIG. 60 illustrates a possible message traffic scenario associated with three different services within a single neighborhood network. In general, the daily DA traffic is substantially lower than basic service traffic, but this could change for short time periods during power outages where peak distribution automation traffic could be substantial. For this reason it is desirable to dynamically reallocate band-

70

width to distribution automation on a frame by frame basis, or allocate enough channels to accommodate daily traffic within a one hour period.

An isolated neighborhood network, for example, has a theoretical traffic capacity of 17,400 messages per hour, but this capacity is reduced by imperfect eavesdropping to roughly half that value. A further reduction in capacity may arise from IDT polling algorithms which minimize interference between adjacent neighborhoods.

The traffic capacity of an isolated neighborhood network is limited by the IRD/RIR slot which can carry about five NSM report messages in each active channel, of which there can be up to 29 per frame. This gives an hourly message capacity of:

$$I = 5 \frac{\text{messages}}{\text{channel}} \times 29 \frac{\text{channels}}{\text{frame}} \times 120 \frac{\text{frames}}{\text{hour}} = 17,400 \text{ messages/hr.}$$

or 417,600 messages per day.

If the eavesdropping efficiency, ϵ_e , were 57% and if polling efficiency, ϵ_p , were 100% then the non-redundant message capacity is roughly 10,000 messages/hour, or 240,000 messages/day.

This capacity in round numbers of 10,000 messages per hour represents a maximum capacity which is reduced by practical considerations related to polling efficiency. FIG. 61 illustrates, as an example, a reasonable level of traffic associated with a neighborhood network which represents roughly 16% of theoretical capacity. The following section discusses practical network capacity of 45,000 to 175,000 messages per day depending on polling protocols and allocation of bandwidth for distribution automation applications.

Network Polling Protocols for DA/DSM

The requirements of distribution automation impose additional criteria for selection of network polling protocols.

FIGS. 62–66 illustrates four different IDT polling protocols, each designed to prevent interference between different messages delivered to, or received from, remote cell nodes and network service modules along neighborhood boundaries where path loss to two intermediate data terminals may be roughly equal. Each protocol employs some method of multiplexing transmissions so as to preclude mutual interference.

Space Division Multiplexing Protocol (SDMP), illustrated in FIG. 62, avoids interference by partitioning large networks into four groups of neighborhood networks designated A, B, C, D. Groups are arranged as shown in FIGS. 62 and 66 so that no member of a single group is adjacent to any other members of the same group, and polling by intermediate data terminals takes place in only one group at a time. Since group members are separated by at least one neighborhood network, roughly two miles, the possibility of message interference along neighborhood boundaries is very small. A major advantage of SDMP is that no co-ordination is required between intermediate data terminals and downstream addressing is very simple. In the event of an IDT failure, adjacent neighborhood intermediate data terminals can poll parts of the area through overlapping coverage, but some co-ordination of intermediate data terminals within adjacent groups may be required to avoid interference in the center of the failed area. However, despite its advantages of simplicity and minimal co-ordination by the central data terminal, space division multiplexing has the disadvantage of low efficiency since only one quarter of the network is active at a time.

5,673,252

71

Amplitude Division Multiplexing Protocol (ADMP) improves the efficiency of space division multiplexing by managing IDT-RCN communications so that adjacent intermediate data terminals can talk to remote cell nodes concurrently. This is achieved by partitioning the network as shown in FIG. 63 into four groups of neighborhood networks with the central zones of each area designated as concurrent polling zones. Within these zones signal strengths from the central IDT transmissions exceed those from adjacent intermediate data terminals by an amount greater than the capture ratio of RCN receivers. Providing the remote cell nodes respond with equal power, the received signal strength at the intermediate data terminals also exceeds the capture ratio so that independent communication can take place within these zones. The areas outside the zones of concurrency (P) are polled using SDMP so that efficiency is reduced, but the concurrency zones may encompass 21 of 25 remote cell nodes in a typical neighborhood network leaving only four to be polled by the SDMP. This gives an overall polling efficiency of 0.67.

Directional Division Multiplexing Protocol (DDMP) further improves the efficiency of polling. In one of its simpler embodiments DDMP employs a directional antenna on every intermediate data terminal, providing coverage in 90° increments so that polling can be accomplished sequentially in four quadrants of a neighborhood network. FIG. 64 illustrates the sequential coverage pattern A, B, C, D which is executed synchronously by all intermediate data terminals. Since coverage areas are widely separated, interference is minimal even with all intermediate data terminals polling together. The efficiency of this protocol can approach 100% providing corresponding quadrants have comparable levels of traffic because the dwell time must be the same for each quadrant.

Polarization Division Multiplexing Protocol (PDMP), as illustrated in FIG. 5, operates in a manner similar to ADMP but with adjacent intermediate data terminals operating concurrently with opposite antenna polarizations. The corner areas of each IDT coverage zone, however, may experience interference so some type of SDMP may be required. Moreover, remote cell nodes also require more expensive polarized antennas, and two types have to be stocked.

Comparison of Polling Protocols

There are significant differences in cost, performance, and communication reliability among the three different polling protocols.

FIG. 67 compares polling protocols on specific issues associated with cost, performance, and reliability.

Communication reliability involves several issues such as interference from adjacent intermediate data terminals which can occur along neighborhood network boundaries, immunity to weather conditions which might affect antenna patterns, ability to provide backup coverage in the event of an IDT failure and topological flexibility which characterizes ability to add a new intermediate data terminal within an existing network to improve coverage in a problem area. DDMP rates high in all categories although SDMP provides the highest resistance to interference from adjacent intermediate data terminals and can overcome shortcomings in topological flexibility by reducing polling efficiency from 25% to 20%.

The protocols can provide deterministic response times which are essential for distribution automation applications. Overall, however, the performance winner is DDMP which exhibits the highest polling efficiency despite possible delays in alarm reporting.

72

Cost involves four components: hardware, installation, maintenance and development, the later being closely related to complexity.

A preferred choice is SDMP which has the simplest hardware in terms of IDT antenna, and IDT/RCN memory, and also has the simplest firmware. Although ADMP has only marginally higher hardware cost because of extra memory required for polling tables and polling algorithms, its complexity is greater than SDMP, requiring more research and development effort for both intermediate data terminal and central data terminal. DDMP is substantially higher in cost because of the need for a steerable antenna array which could increase the cost of an intermediate data terminal by several hundred dollars. Moreover, the need to carefully observe intermediate data terminal orientation during installation will increase costs and maintenance costs.

Communication Channel Configuration for Distribution Automation

Distribution automation requires two-way communication with faster response and greater reliability than automatic meter reading services. To meet this requirement channels will have to be specially configured for distribution automation services.

The basic communication granularity is the 100 millisecond message slot. In order to maintain compliance with the single radio channel compatibility goal, each hop in the hierarchy will require one slot. Thus, if an acknowledged message were sent to the NSM level, then at least four slots are required.

IDT to RCN—1 slot@ 2 kb/s

RCN to NSM—1 or 2 slots@ 1 kb/s

NSM to RCN—1 slot@ 2 kb/s

RCN to IDT—1 slot@ 2 kb/s

Possibly a double slot on the RCN to NSM link might be required to provide broader compatibility with some existing utility protocols such as the PG&E protocol working to the NSM level. However, network service modules supporting distribution automation functions are limited in functionality and therefore typically required no more than two or three bytes of information in the data field, which fit within a single slot.

Because of more stringent requirements on distribution automation message loss rates, and response times, the directional or space division methods of communication provides the simplest, fastest and most reliable approach for IDT and RCN communication, as well as for NSM to RCN communication. Intermediate data terminals are partitioned into four non-adjacent groups and communication only occurs within one group at a time which avoids any need for IDT co-ordination by the central data terminal. However, because only one group at a time communicates, a total of 16 slots are allocated within a frame to provide a single independent "DA channel" for each of a intermediate data terminal groups. The maximum response time (T_R) is equal to

$$T_R = \frac{16}{\#DA \text{ slots/frame}} \times 30 \text{ seconds}$$

At least two approaches to supporting fast response communication are compatible with the two-way frame structure. One is to allocate a single slot (say NRR1) within each channel for distribution automation as shown in FIG. 68. This would provide a response time of

5,673,252

73

$$\frac{16}{29} \times 30 = 16.55 \text{ seconds}$$

for each command message with a capacity of 217 messages 5
per hour.

A second approach is to designate a new type of channel 5
called a real-time channel or R channel comprising ten slots
allocated as shown in FIG. 69. At least two channels would
be required to support distribution automation and they 10
could be organized as shown with two spare slots/channel.
The spare slots could be employed as contention access slots
for DA information gathering or they could be used to
indicate the type of channel so that other channel configu-
rations could be supported. 15

Channel Frame Structure for Distribution Automation

Of the two fast-response communication approaches, 20
allocated DA slots in designated channels; or special DA
channels called R-channels, the R-channel approach appears
to be sufficiently flexible in that variable amounts of band-
width can be allocated to DA functions through the CAT
distribution process with minimal impact on existing NSM
firmware. Adding new channel configurations independent 25
of old ones at the NSM level is possible whereas changing
slot assignments in future would require additional NSM
firmware to support both slot and channel assignments.

FIG. 71 shows a possible frame configuration employing 30
the modified amplitude multiplexing (ADMP) plus two real
time channel pairs to provide a neighborhood network
capacity of 120,000 report/control messages per day plus
120 DA messages/hr (2880/day) with a maximum response
time of 15 seconds. This exceeds the projected requirements
of 150 DA control messages/day for a neighborhood net- 35
work.

When Directional Multiplexing (DDMP) is employed, 2R
channel pairs per frame would have to be allocated to give
15 second response time. Since each of the four ABCD slots 40
can be used to send a message within a neighborhood, one
to each quadrant, the theoretical DA message capacity is
four times greater with DDMP than with ADMP, but if a
human operator were involved, the practical limit would be
one message per R channel pair, i.e. $\frac{1}{15}$ seconds. However, 45
DDMP would provide greater capacity for more automated
control strategies. FIG. 70 compares different approaches.

FIG. 72 shows a frame configuration employing allocated
D/A slots with modified ADMP to provide a capacity of 50
2880 DA messages/day with 30 second response time.
Response times below 16 seconds would require allocation
of more than one slot per channel which would reduce NSM
to RCN report capacity.

It will be apparent to those skilled in the art that various
modifications can be made to the communications network 55
for collecting data from remote data generating stations of
the instant invention without departing from the scope or
spirit of the invention, and it is intended that the present
invention cover modifications and variations of the commu-
nications network provided they come within the scope of 60
the appended claims and their equivalents.

We claim:

1. A method for communicating between an intermediate
data terminal (IDT), a plurality of remote cell nodes (RCN),
and a plurality of network service modules (NSM), using a 65
plurality of frames with each frame having a plurality of
channels, comprising the steps, during each frame, of:

74

transmitting, in a first channel of the frame, from the
intermediate data terminal to the plurality of remote
cell nodes, an IDT-synchronization signal;

transmitting, synchronized to the IDT-synchronization
signal in a second channel of the frame, from the
intermediate data terminal to a first remote cell node, a
first polling signal;

receiving, at each of the plurality of remote cell nodes, the
IDT-synchronization signal;

synchronizing, at each of the plurality of remote cell
nodes, responsive to receiving the IDT-synchronization
signal, an RCN-timing circuit to the IDT-
synchronization signal;

receiving, at the first remote cell node, the first polling
signal;

transmitting, synchronized to the IDT-synchronization
signal in a third channel of the frame, from the first
remote cell node to the intermediate data terminal,
responsive to the first polling signal, an RCN-packet
signal;

transmitting, using radio waves, in a fourth channel of the
frame, from each of the plurality of remote cell nodes
to the plurality of network service modules, responsive
to receiving and synchronizing to the IDT-
synchronization signal, an RCN-synchronization signal;

receiving, at each of the plurality of network service
modules, the RCN-synchronization signal;

synchronizing, at each of the plurality of network service
modules, responsive to receiving the RCN-
synchronization signal, a NSM-timing circuit to the
RCN-synchronization signal;

collecting, at a first network service module, data from a
first physical device; and

transmitting, using radio waves, responsive to receiving
and synchronizing to the RCN-synchronization signal
synchronized to the IDT-synchronization signal, in a
fifth channel of the frame, from the first network
service module to at least one of the plurality of remote
cell nodes, a first NSM-packet signal containing the
data from the first physical device.

2. The method as set forth in claim 1, further including the
steps of:

transmitting, synchronized to the IDT-synchronization
signal, in a sixth channel of the frame, from the first
remote cell node to a second network service module,
a command signal; and

transmitting, using radio waves, synchronized to the
RCN-synchronization signal and responsive to the
command signal, in a seventh channel of the frame,
from the second network service module to the first
remote cell node, a second NSM-packet signal.

3. A method for communicating between an intermediate
data terminal (IDT), a plurality of remote cell nodes (RCN),
and a plurality of network service modules (NSM), using a
plurality of frames with each frame having a plurality of
channels, comprising the steps, during each frame, of:

transmitting, in a first channel of the frame, from the
intermediate data terminal to the plurality of remote
cell nodes, an IDT-synchronization signal;

receiving, at each of the plurality of remote cell nodes, the
IDT-synchronization signal;

synchronizing, at each of the plurality of remote cell
nodes, responsive to receiving the IDT-synchronization
signal, an RCN-timing circuit to the IDT-
synchronization signal;

5,673,252

75

transmitting, using radio waves, in a fourth channel of the frame, from each of the plurality of remote cell nodes to the plurality of network service modules, responsive to receiving and synchronizing to the IDT-synchronization signal, an RCN-synchronization signal;

receiving, at each of the plurality of network service modules, the RCN-synchronization signal;

synchronizing, at each of the plurality of network service modules, responsive to receiving the RCN-synchronization signal, an NSM-timing circuit to the RCN-synchronization signal;

collecting, at a first network service module, data from a first physical device;

transmitting, using radio waves, responsive to receiving and synchronizing to the RCN-synchronization signal synchronized to the IDT-synchronization signal, in a fifth channel of the frame, from the first network service module to a first remote cell node, a first NSM-packet signal containing the data from the first physical device;

receiving, at the first remote cell node, the NSM-packet signal;

storing, at the first remote cell node, the NSM-packet signal;

transmitting, synchronized to the IDT-synchronization signal, in a second channel of the frame, from the intermediate data terminal to the first remote cell node, a first polling signal;

receiving, at the first remote cell node, the first polling signal; and

transmitting, synchronized to the IDT-synchronization signal and responsive to the first polling signal, in a third channel of the frame, from the first remote cell node to the intermediate data terminal, a plurality of stored NSM-packet signals as an RCN-packet signal.

4. The method as set forth in claim 3, further including the steps of:

transmitting, using radio waves, synchronized to the IDT-synchronization signal, in a sixth channel of the frame, from the first remote cell node to a second network service module, a command signal; and

transmitting, using radio waves, synchronized to the RCN-synchronization signal and responsive to the command signal, in a seventh channel of the frame, from the second network service module to the first remote cell node, a second NSM-packet signal.

5. A method for communicating between an intermediate data terminal (IDT), a plurality of remote cell nodes (RCN), and a plurality of network service modules (NSM), using a plurality of frames with each frame having a plurality of channels, comprising the steps, during each frame, of:

transmitting, in a first channel of the frame, from the intermediate data terminal to the plurality of remote cell nodes, an IDT-synchronization signal;

receiving, at each of the plurality of remote cell nodes, the IDT-synchronization signal;

synchronizing, at each of the plurality of remote cell nodes, responsive to receiving the IDT-synchronization signal, an RCN-timing circuit to the IDT-synchronization signal;

transmitting, using radio waves, in a fourth channel of the frame, from each of the plurality of remote cell nodes to the plurality of network service modules, responsive

76

to receiving and synchronizing to the IDT-synchronization signal, an RCN-synchronization signal;

receiving, at each of the plurality of network service modules, the RCN-synchronization signal;

synchronizing, at each of the plurality of network service modules, responsive to receiving the RCN-synchronization signal, an NSM-timing circuit to the RCN-synchronization signal;

collecting, at a first network service module, data from a first physical device;

transmitting, using radio waves, at a pseudorandom time within a predetermined time period, responsive to receiving and synchronizing to the RCN-synchronization signal synchronized to the IDT-synchronization signal, in a fifth channel of the frame, from the first network service module, an NSM-packet signal containing the data from the first physical device, to a multiplicity of remote cell nodes, said multiplicity of remote cell nodes being a subset of the plurality of remote cell nodes and said multiplicity of remote cell nodes including a first remote cell node;

receiving, at the multiplicity of remote cell nodes, the NSM-packet signal;

storing, at the multiplicity of remote cell nodes, the NSM-packet signal;

transmitting, synchronized to the IDT-synchronization signal in a second channel of the frame, from the intermediate data terminal to the first remote cell node, a first polling signal;

receiving, at the first remote cell node, the first polling signal; and

transmitting, synchronized to the IDT-synchronization signal, in a third channel of the frame, from the first remote cell node to the intermediate data terminal, responsive to the first polling signal, a first plurality of NSM-packet signals as a first RCN-packet signal.

6. The method as set forth in claim 5, further including the steps of:

receiving, at the intermediate data terminal, the first RCN-packet signal; and

transmitting, in a sixth channel of the frame, from the intermediate data terminal to the first remote cell node, responsive to receiving the first RCN-packet signal, an acknowledgement signal.

7. The method as set forth in claim 5, further including the steps of:

transmitting, in a real-time request channel of the frame, from the intermediate data terminal to the first network service module, an IDT-request for a set of requested data;

receiving, at the first network service module, the IDT-request; and

transmitting, responsive to the IDT-request and in real-time, from the first network service module to the intermediate data terminal, the set of requested data.

8. The method as set forth in claim 6, further including the steps of:

eavesdropping, by a second remote cell node, said second remote cell node being within the multiplicity of remote cell nodes, to the first polling signal;

eavesdropping, by the second remote cell node, to the transmission of the first plurality of NSM-packet signals;

5,673,252

77

comparing, by the second remote cell node, the first plurality of NSM-packet signals to a second plurality of NSM-packet signals, said second plurality of NSM-packet signals being stored by the second remote cell node in a memory; and

identifying a third plurality of NSM-packet signals common to both the first plurality of NSM-packet signals and the second plurality of NSM-packet signals.

9. The method as set forth in claim 8, further including the step of:

deleting the third plurality of NSM-packet signals from the memory.

10. The method as set forth in claim 8, further including the step of:

overwriting the third plurality of NSM-packet signals.

11. The method as set forth in claim 8, further including the step of:

transmitting, synchronized to the IDT-synchronization signal in a seventh channel of the frame, from the intermediate data terminal to the second remote cell node, a second polling signal;

receiving, at the second remote cell node, the second polling signal; and

transmitting, synchronized to the IDT-synchronization signal, in a eighth channel of the frame, from the second remote cell node to the intermediate data terminal, responsive to the second polling signal, the second plurality of stored NSM-packet signals, but not including the third plurality of NSM-packet signals, as a second RCN-packet signal.

12. The method as set forth in claim 11, further including the steps of:

receiving, at the intermediate data terminal, the second RCN-packet signal; and

transmitting, in a ninth channel of the frame, from the intermediate data terminal to the second remote cell node, responsive to receiving the second RCN-packet signal, an acknowledgement signal.

13. The method as set forth in claim 5, further including the steps of:

receiving, at the intermediate data terminal from a second remote cell node, a second plurality of NSM-packet signals as a second RCN-packet signal;

comparing, at the intermediate data terminal, the second plurality of NSM-packet signals to the first plurality of NSM-packet signals; and

identifying a third plurality of NSM-packet signals common to both the first plurality of NSM-packet signals and the second plurality of NSM-packet signals.

14. The method as set forth in claim 13, further including the step of:

deleting the third plurality of NSM-packet signals.

15. The method as set forth in claim 13, further including the step of:

overwriting the third plurality of NSM-packet signals.

16. The method as set forth in claim 7, further including the steps of:

detecting, at the first network service module, an alarm condition; and

transmitting, responsive to detecting the alarm condition, in the real-time request channel of the frame, from the first network service module to the intermediate data terminal, the alarm condition.

17. A method for communicating between a central data terminal (CDT), a plurality of intermediate data terminals

78

(IDT), a plurality of remote cell nodes (RCN), and a plurality of network service modules (NSM), using a plurality of frames with each frame having a plurality of channels, comprising the steps, during each frame, of:

5 transmitting, in a first channel of the frame, from the plurality of intermediate data terminals to the plurality of remote cell nodes, an IDT-synchronization signal;

receiving, at each of the plurality of remote cell nodes, the IDT-synchronization signal;

10 synchronizing, at each of the plurality of remote cell nodes, responsive to receiving the IDT-synchronization signal, an RCN-timing circuit to the IDT-synchronization signal;

15 transmitting, using radio waves, in a fourth channel of the frame, from each of the plurality of remote cell nodes to the plurality of network service modules, responsive to receiving and synchronizing to the IDT-synchronization signal, an RCN-synchronization signal;

20 receiving, at each of the plurality of network service modules, the RCN-synchronization signal;

synchronizing, at each of the plurality of network service modules, responsive to receiving the RCN-synchronization signal, an NSM-timing circuit to the RCN-synchronization signal;

collecting, at a first network service module, data from a first physical device;

25 transmitting, using radio waves, at a pseudorandom time within a predetermined time period, responsive to receiving and synchronizing to the RCN-synchronization signal synchronized to the IDT-synchronization signal, in a fifth channel of the frame, from the first network service module, an NSM-packet signal containing the data from the first physical device, to a multiplicity of remote cell nodes, said multiplicity of remote cell nodes being a subset of the plurality of remote cell nodes and said multiplicity of remote cell nodes including a first remote cell node;

30 receiving, at the multiplicity of remote cell nodes, the NSM-packet signal;

storing, at the multiplicity of remote cell nodes, the NSM-packet signal;

35 transmitting, synchronized to the IDT-synchronization signal, in a second channel of the frame, from a first intermediate data terminal to the first remote cell node, a first polling signal;

40 receiving, at the first remote cell node, the first polling signal;

45 transmitting, synchronized to the IDT-synchronization signal, in a third channel of the frame, from the first remote cell node to the first intermediate data terminal, responsive to the first polling signal, a first plurality of NSM-packet signals as a first RCN-packet signal;

50 transmitting, from the central data terminal to the first intermediate data terminal, in a sixth channel of the frame, a second polling signal;

receiving, at the first intermediate data terminal, the second polling signal; and

55 transmitting, in a seventh channel of the frame, from the first intermediate data terminal to the central data terminal, responsive to the second polling signal, a first plurality of RCN-packet signals as a first IDT-packet signal.

60 18. The method as set forth in claim 17, further including the steps of:

5,673,252

79

receiving, at the central data terminal, the first IDT-packet signal; and

transmitting, from the central data terminal to the first intermediate data terminal, responsive to receiving the first IDT-packet signal, an acknowledgement signal.

19. The method as set forth in claim 17, further including the step of:

transmitting, in a real-time request channel of the frame, from the central data terminal to the first network service module, a CDT-request for a set of requested data;

receiving, at the first network service module, the CDT-request; and

transmitting, responsive to the CDT-request and in real-time, from the first network service module to the central data terminal, the set of requested data.

20. The method as set forth in claim 17, further including the steps of:

eavesdropping, by a second remote cell node, to the first polling signal;

eavesdropping, by the second remote cell node, to the transmission of the first plurality of NSM-packet signals;

comparing, by the second remote cell node, the first plurality of NSM-packet signals to a second plurality of NSM-packet signals, said second plurality of NSM-packet signals being stored by the second remote cell node in a memory; and

identifying a third plurality of NSM-packet signals common to both the first plurality of NSM-packet signals and the second plurality of NSM-packet signals.

21. The method as set forth in claim 20, further including the step of:

deleting the third plurality of NSM-packet signals from the memory.

22. The method as set forth in claim 20, further including the step of:

overwriting the third plurality of NSM-packet signals.

23. The method as set forth in claim 20, further including the step of:

transmitting, synchronized to the IDT-synchronization signal in an eighth channel of the frame, from the first intermediate data terminal to the second remote cell node, a third polling signal;

receiving, at the second remote cell node, the third polling signal; and

transmitting, synchronized to the IDT-synchronization signal, in a ninth channel of the frame, from the second remote cell node to the first intermediate data terminal, responsive to the third polling signal, the second plurality of stored NSM-packet signals, but not including the third plurality of NSM-packet signals, as a second RCN-packet signal.

24. The method as set forth in claim 17, further including the steps of:

eavesdropping, by a second intermediate data terminal, to the second polling signal;

eavesdropping, by the second intermediate data terminal, to the transmission of the first plurality of RCN-packet signals;

comparing, by the second intermediate data terminal, the first plurality of RCN-packet signals to a second plurality of RCN-packet signals, said second plurality of RCN-packet signals being stored by the second intermediate data terminal in a memory; and

80

identifying a third plurality of RCN-packet signals common to both the first plurality of RCN-packet signals and the second plurality of RCN-packet signals.

25. The method as set forth in claim 24, further including the step of:

deleting the third plurality of RCN-packet signals from the memory.

26. The method as set forth in claim 24, further including the step of:

overwriting the third plurality of RCN-packet signals.

27. The method as set forth in claim 24, further including the step of:

transmitting, synchronized to the IDT-synchronization signal in an eighth channel of the frame, from the central data terminal to the second intermediate data terminal, a third polling signal;

receiving, at the second intermediate data terminal, the third polling signal; and

transmitting, synchronized to the IDT-synchronization signal, in a ninth channel of the frame, from the second intermediate data terminal to the central data terminal, responsive to the third polling signal, the second plurality of stored RCN-packet signals, but not including the third plurality of RCN-packet signals, as a second IDT-packet signal.

28. The method as set forth in claim 17, further including the steps of:

transmitting, from the central data terminal to a second intermediate data terminal, in an eighth channel of the frame, a third polling signal;

receiving, at the second intermediate data terminal, the third polling signal; and

transmitting, in a ninth channel of the frame, from the second intermediate data terminal to the central data terminal, responsive to the third polling signal, a second plurality of RCN-packet signals as a second IDT-packet signal.

29. The method as set forth in claim 28, further including the steps of:

receiving, at the central data terminal, the second IDT-packet signal; and

transmitting, from the central data terminal to the second intermediate data terminal, responsive to receiving the second IDT-packet signal, an acknowledgement signal.

30. The method as set forth in claim 28, further including the steps of:

receiving, at the central data terminal from the second intermediate data terminal, the second plurality of RCN-packet signals;

comparing, at the central data terminal, the second plurality of RCN-packet signals to the first plurality of RCN-packet signals; and

identifying a third plurality of RCN-packet signals common to both the first plurality of RCN-packet signals and the second plurality of RCN-packet signals.

31. The method as set forth in claim 30, further including the step of:

deleting the third plurality of RCN-packet signals.

32. The method as set forth in claim 30, further including the step of:

overwriting the third plurality of RCN-packet signals.

33. The method as set forth in claim 19, further including the steps of:

detecting, at the first network service module, an alarm condition; and

5,673,252

81

transmitting, responsive to detecting the alarm condition, in the real-time request channel of the frame, from the first network service module to the central data terminal, the alarm condition.

34. A method for communicating between a central data terminal (CDT), a plurality of remote cell nodes (RCN), and a plurality of network service modules (NSM), using a plurality of frames with each frame having a plurality of channels, comprising the steps, during each frame, of:

transmitting, in a first channel of the frame, from the central data terminal to the plurality of remote cell nodes, a CDT-synchronization signal;

receiving, at each of the plurality of remote cell nodes, the CDT-synchronization signal;

synchronizing, at each of the plurality of remote cell nodes, responsive to receiving the CDT-synchronization signal, an RCN-timing circuit to the CDT-synchronization signal;

transmitting, using radio waves, in a fourth channel of the frame, from each of the plurality of remote cell nodes to the plurality of network service modules, responsive to receiving and synchronizing to the CDT-synchronization signal, an RCN-synchronization signal;

receiving, at each of the plurality of network service modules, the RCN-synchronization signal;

synchronizing, at each of the plurality of network service modules, responsive to receiving the RCN-synchronization signal, an NSM-timing circuit to the RCN-synchronization signal;

collecting, at a first network service module, data from a first physical device;

transmitting, using radio waves, at a pseudorandom time within a predetermined time period, responsive to receiving and synchronizing to the RCN-synchronization signal synchronized to the CDT-synchronization signal, in a fifth channel of the frame, from the first network service module, an NSM-packet signal containing the data from the first physical device, to a multiplicity of remote cell nodes, said multiplicity of remote cell nodes being a subset of the plurality of remote cell nodes and said multiplicity of remote cell nodes including a first remote cell node;

receiving, at the multiplicity of remote cell nodes, the NSM-packet signal;

storing, at the multiplicity of remote cell nodes, the NSM-packet signal;

transmitting, synchronized to the CDT-synchronization signal in a second channel of the frame, from the central data terminal to the first remote cell node, a first polling signal;

receiving, at the first remote cell node, the first polling signal; and

transmitting, synchronized to the CDT-synchronization signal, in a third channel of the frame, from the first remote cell node to the central data terminal, responsive to the first polling signal, a first plurality of NSM-packet signals as a first RCN-packet signal.

35. The method as set forth in claim 34, further including the steps of:

receiving, at the central data terminal, the first RCN-packet signal; and

transmitting, in a sixth channel of the frame, from the central data terminal to the first remote cell node,

82

responsive to receiving the first RCN-packet signal, an acknowledgement signal.

36. The method as set forth in claim 34, further including the step of:

transmitting, in a real-time request channel of the frame, from the central data terminal to the first network service module, a CDT-request for a set of requested data;

receiving, at the first network service module, the CDT-request; and

transmitting, responsive to the CDT-request and in real-time, from the first network service module to the central data terminal, the set of requested data.

37. The method as set forth in claim 34, further including the steps of:

eavesdropping, by a second remote cell node, said second remote cell node being within the multiplicity of remote cell nodes, to the first polling signal;

eavesdropping, by the second remote cell node, to the transmission of the first plurality of NSM-packet signals;

comparing, by the second remote cell node, the first plurality of NSM-packet signals to a second plurality of NSM-packet signals, said second plurality of NSM-packet signals being stored by the second remote cell node in a memory; and

identifying a third plurality of NSM-packet signals common to both the first plurality of NSM-packet signals and the second plurality of NSM-packet signals.

38. The method as set forth in claim 37, further including the step of:

deleting the third plurality of NSM-packet signals from the memory.

39. The method as set forth in claim 37, further including the step of:

overwriting the third plurality of NSM-packet signals.

40. The method as set forth in claim 37, further including the step of:

transmitting, synchronized to the CDT-synchronization signal in a seventh channel of the frame, from the central data terminal to the second remote cell node, a third polling signal;

receiving, at the second remote cell node, the third polling signal; and

transmitting, synchronized to the CDT-synchronization signal, in a eighth channel of the frame, from the second remote cell node to the central data terminal, responsive to the third polling signal, the second plurality of stored NSM-packet signals, but not including the third plurality of NSM-packet signals, as a second RCN-packet signal.

41. The method as set forth in claim 40, further including the steps of:

receiving, at the central data terminal, the second RCN-packet signal; and

transmitting, from the central data terminal to the second remote cell node, responsive to receiving the second packet signal, an acknowledgement signal.

42. The method as set forth in claim 34, further including the steps of:

receiving, at the central data terminal from a second remote cell node, a second plurality of NSM-packet signals as a second RCN-packet signal;

comparing, at the central data terminal, the second plurality of NSM-packet signals to the first plurality of NSM-packet signals; and

5,673,252

83

identifying a third plurality of NSM-packet signals common to both the first plurality of NSM-packet signals and the second plurality of NSM-packet signals.

43. The method as set forth in claim 42, further including the step of:

deleting the third plurality of NSM-packet signals.

44. The method as set forth in claim 42, further including the step of:

overwriting the third plurality of NSM-packet signals.

45. The method as set forth in claim 36, further including the steps of:

detecting, at the first network service module, an alarm condition; and

transmitting, responsive to detecting the alarm condition, in the real-time request channel of the frame, from the first network service module to the central data terminal, the alarm condition.

46. A method for communicating between an intermediate data terminal (IDT), a plurality of remote cell nodes (RCN), and a plurality of network service modules (NSM), using a plurality of frames with each frame having a plurality of channels, comprising the steps, during each frame, of:

transmitting, in a first channel of the frame, from the intermediate data terminal to the plurality of remote cell nodes, an IDT-synchronization signal;

transmitting, synchronized to the IDT-synchronization signal, in a second channel of the frame, from the intermediate data terminal to a first remote cell node, a first polling signal;

receiving, at each of the plurality of remote cell nodes, the IDT-synchronization signal;

synchronizing, at each of the plurality of remote cell nodes, responsive to receiving the IDT-synchronization signal, an RCN-timing circuit to the IDT-synchronization signal;

transmitting, using radio waves, in a third channel of the frame, from each of the plurality of remote cell nodes to the plurality of network service modules, responsive to receiving and synchronizing to the IDT-synchronization signal, an RCN-synchronization signal;

receiving, at each of the plurality of network service modules, the RCN-synchronization signal;

synchronizing, at each of the plurality of network service modules, responsive to receiving the RCN-synchronization signal, an NSM-timing circuit to the RCN-synchronization signal;

collecting, at the plurality of network service modules, data from a plurality of physical devices, respectively;

transmitting, using radio waves, in a fourth channel of the frame, from the first remote cell node to a first network service module, a command signal;

receiving, at the first network service module, the command signal;

transmitting, using radio waves, synchronized to the RCN-synchronization signal and responsive to the command signal, in a fifth channel of the frame, from the first network service module to the first remote cell node, data as an NSM-packet signal;

receiving, at the first remote cell node, the NSM-packet signal;

storing, at the first remote cell node, the NSM-packet signal;

receiving, at the first remote cell node, the first polling signal; and

84

transmitting, synchronized to the IDT-synchronization signal and responsive to the first polling signal, in a sixth channel of the frame, from the first remote cell node to the intermediate data terminal, a first plurality of stored NSM-packet signals as a first RCN-packet signal.

47. The method as set forth in claim 46, further including the steps of:

receiving, at the intermediate data terminal, the first RCN-packet signal; and

transmitting, in an seventh channel of the frame, from the intermediate data terminal to the first remote cell node, responsive to receiving the first RCN-packet signal, an acknowledgement signal.

48. The method as set forth in claim 46, further including the step of:

transmitting, in a real-time request channel of the frame, from the intermediate data terminal to the first network service module, an IDT-request for a set of requested data;

receiving, at the first network service module, the IDT-request; and

transmitting, responsive to the IDT-request and in real-time, from the first network service module to the intermediate data terminal, the set of requested data.

49. The method as set forth in claim 46, further including the steps of:

eavesdropping, by a second remote cell node, to the first polling signal;

eavesdropping, by the second remote cell node, to the transmission of the first plurality of stored NSM-packet signals;

comparing, by the second remote cell node, the first plurality of stored NSM-packet signals to a second plurality of stored NSM-packet signals, said second plurality of stored NSM-packet signals being stored by the second remote cell node in a memory; and

identifying a third plurality of NSM-packet signals common to both the first plurality of stored NSM-packet signals and the second plurality of stored NSM-packet signals.

50. The method as set forth in claim 49, further including the step of:

deleting the third plurality of NSM-packet signals from the memory.

51. The method as set forth in claim 49, further including the step of:

overwriting the third plurality of NSM-packet signals.

52. The method as set forth in claim 49, further including the step of:

transmitting, synchronized to the IDT-synchronization signal in an eighth channel of the frame, from the intermediate data terminal to the second remote cell node, a second polling signal;

receiving, at the second remote cell node, the second polling signal; and

transmitting, synchronized to the IDT-synchronization signal, in a ninth channel of the frame, from the second remote cell node to the intermediate data terminal, responsive to the second polling signal, the second plurality of stored NSM-packet signals, but not including the third plurality of NSM-packet signals, as a second RCN-packet signal.

53. The method as set forth in claim 52, further including the steps of:

5,673,252

85

receiving, at the intermediate data terminal, the second RCN-packet signal; and

transmitting, from the intermediate data terminal to the second remote cell node, responsive to receiving the second RCN-packet signal, an acknowledgement signal.

54. The method as set forth in claim 48, further including the steps of:

detecting, at the first network service module, an alarm condition; and

transmitting, responsive to detecting the alarm condition, in the real-time request channel of the frame, from the first network service module to the intermediate data terminal, the alarm condition.

55. A method for communicating between a central data terminal (CDT), a plurality of intermediate data terminals (IDT), a plurality of remote cell nodes (RCN), and a plurality of network service modules (NSM), using a plurality of frames with each frame having a plurality of channels, comprising the steps, during each frame, of:

transmitting, in a first channel of the frame, from the plurality of intermediate data terminals to the plurality of remote cell nodes, an IDT-synchronization signal;

transmitting, synchronized to the IDT-synchronization signal, in a second channel of the frame, from a first intermediate data terminal to a first remote cell node, a first polling signal;

receiving, at each of the plurality of remote cell nodes, the IDT-synchronization signal;

synchronizing, at each of the plurality of remote cell nodes, responsive to receiving the IDT-synchronization signal, an RCN-timing circuit to the IDT-synchronization signal;

transmitting, using radio waves, in a third channel of the frame, from each of the plurality of remote cell nodes to the plurality of network service modules, responsive to receiving and synchronizing to the IDT-synchronization signal, an RCN-synchronization signal;

receiving, at each of the plurality of network service modules, the RCN-synchronization signal;

synchronizing, at each of the plurality of network service modules, responsive to receiving the RCN-synchronization signal, an NSM-timing circuit to the RCN-synchronization signal;

collecting, at the plurality of network service modules, data from a plurality of physical devices, respectively;

transmitting, using radio waves, in a fourth channel of the frame, from the first remote cell node to a first network service module, a command signal;

receiving, at the first network service module, the command signal;

transmitting, using radio waves, synchronized to the RCN-synchronization signal and responsive to the command signal, in a fifth channel, from the first network service module to the first remote cell node, data as an NSM-packet signal;

receiving, at the first remote cell node, the NSM-packet signal;

storing, at the first remote cell node, the NSM-packet signal;

receiving, at the first remote cell node, the first polling signal;

transmitting, synchronized to the IDT-synchronization signal and responsive to the first polling signal, in a

86

sixth channel of the frame, from the first remote cell node to the first intermediate data terminal, a first plurality of stored NSM-packet signals as a first RCN-packet signal;

transmitting, in a seventh channel of the frame, from the central data terminal to the first intermediate data terminal, a second polling signal;

receiving, at the first intermediate data terminal, the second polling signal; and

transmitting, responsive to the second polling signal, in an eighth channel of the frame, from the first intermediate data terminal to the central data terminal, a first plurality of RCN-packet signals as a first IDT-packet signal.

56. The method as set forth in claim 55, further including the steps of:

receiving, at the first intermediate data terminal, the first RCN-packet signal; and

transmitting, from the first intermediate data terminal to the first remote cell node, responsive to receiving the first RCN-packet signal, an acknowledgement signal.

57. The method as set forth in claim 55, further including the step of:

transmitting, in a real-time request channel of the frame, from the central data terminal to the first network service module, a CDT-request for a set of requested data;

receiving, at the first network service module, the CDT-request; and

transmitting, responsive to the CDT-request and in real-time, from the first network service module to the central data terminal, the set of requested data.

58. The method as set forth in claim 55, further including the steps of:

receiving, at the central data terminal, the first IDT-packet signal; and

transmitting, from the central data terminal to the first intermediate data terminal, responsive to receiving the first IDT-packet signal, an acknowledgement signal.

59. The method as set forth in claim 55, further including the steps of:

eavesdropping, by a second remote cell node, to the first polling signal;

eavesdropping, by the second remote cell node, to the transmission of the first plurality of stored NSM-packet signals;

comparing, by the second remote cell node, the first plurality of stored NSM-packet signals to a second plurality of stored NSM-packet signals, said second plurality of stored NSM-packet signals being stored by the second remote cell node in a memory; and

identifying a third plurality of NSM-packet signals common to both the first plurality of stored NSM-packet signals and the second plurality of stored NSM-packet signals.

60. The method as set forth in claim 59, further including the step of:

deleting the third plurality of NSM-packet signals from the memory.

61. The method as set forth in claim 59, further including the step of:

overwriting the third plurality of NSM-packet signals.

62. The method as set forth in claim 59, further including the step of:

5,673,252

87

transmitting, synchronized to the IDT-synchronization signal in a ninth channel of the frame, from the first intermediate data terminal to the second remote cell node, a third polling signal;

receiving, at the second remote cell node, the third polling signal; and

transmitting, synchronized to the IDT-synchronization signal, in a tenth channel of the frame, from the second remote cell node to the first intermediate data terminal, responsive to the third polling signal, the second plurality of stored NSM-packet signals, but not including the third plurality of NSM-packet signals, as a second RCN-packet signal.

63. The method as set forth in claim 62, further including the steps of:

receiving, at the first intermediate data terminal, the second RCN-packet signal; and

transmitting, from the first intermediate data terminal to the second remote cell node, responsive to receiving the second RCN-packet signal, an acknowledgement signal.

64. The method as set forth in claim 55, further including the steps of:

transmitting, from the central data terminal to a second intermediate data terminal, in a ninth channel of the frame, a third polling signal;

receiving, at the second intermediate data terminal, the third polling signal; and

transmitting, in a tenth channel of the frame, from the second intermediate data terminal to the central data terminal, responsive to the third polling signal, a second plurality of RCN-packet signals as a second IDT-packet signal.

65. The method as set forth in claim 64, further including the steps of:

receiving at the central data terminal, the second IDT-packet signal; and

transmitting, from the central data terminal to the second intermediate data terminal, responsive to receiving the second IDT-packet signal, an acknowledgement signal.

66. The method as set forth in claim 64, further including the steps of:

receiving, at the central data terminal from the second intermediate data terminal, the second plurality of RCN-packet signals;

comparing, at the central data terminal, the second plurality of RCN-packet signals to the first plurality of RCN-packet signals; and

identifying a third plurality of RCN-packet signals common to both the first plurality of RCN-packet signals and the second plurality of RCN-packet signals.

67. The method as set forth in claim 66, further including the step of:

deleting the third plurality of RCN-packet signals.

68. The method as set forth in claim 66, further including the step of:

overwriting the third plurality of RCN-packet signals.

69. The method as set forth in claim 55, further including the steps of:

eavesdropping, by a second intermediate data terminal, to the second polling signal;

eavesdropping, by the second intermediate data terminal, to the transmission of the first plurality of RCN-packet signals;

88

comparing, by the second intermediate data terminal, the first plurality of RCN-packet signals to a second plurality of RCN-packet signals, said second plurality of RCN-packet signals being stored by the second intermediate data terminal in a memory; and

identifying a third plurality of RCN-packet signals common to both the first plurality of RCN-packet signals and the second plurality of RCN-packet signals.

70. The method as set forth in claim 69, further including the step of:

deleting the third plurality of RCN-packet signals from the memory.

71. The method as set forth in claim 69, further including the step of:

overwriting the third plurality of RCN-packet signals.

72. The method as set forth in claim 69, further including the step of:

transmitting, synchronized to the IDT-synchronization signal in a ninth channel of the frame, from the central data terminal to the second intermediate data terminal, a third polling signal;

receiving, at the second intermediate data terminal, the third polling signal; and

transmitting, synchronized to the IDT-synchronization signal, in a tenth channel of the frame, from the second intermediate data terminal to the central data terminal, responsive to the third polling signal, the second plurality of stored RCN-packet signals, but not including the third plurality of RCN-packet signals, as a second IDT-packet signal.

73. The method as set forth in claim 57, further including the steps of:

detecting, at the first network service module, an alarm condition; and

transmitting, responsive to detecting the alarm condition, in the real-time request channel of the frame, from the first network service module to the central data terminal, the alarm condition.

74. A method for communicating between a central data terminal (CDT), a plurality of remote cell nodes (RCN), and a plurality of network service modules (NSM), using a plurality of frames with each frame having a plurality of channels, comprising the steps, during each frame, of:

transmitting, in a first channel of the frame, from the central data terminal to the plurality of remote cell nodes, a CDT-synchronization signal;

transmitting, synchronized to the CDT-synchronization signal, in a second channel of the frame, from the central data terminal to a first remote cell node, a first polling signal;

receiving, at each of the plurality of remote cell nodes, the CDT-synchronization signal;

synchronizing, at each of the plurality of remote cell nodes, responsive to receiving the CDT-synchronization signal, an RCN-timing circuit to the CDT-synchronization signal;

transmitting, using radio waves, in a third channel of the frame, from each of the plurality of remote cell nodes to the plurality of network service modules, responsive to receiving and synchronizing to the CDT-synchronization signal, an RCN-synchronization signal;

receiving, at each of the plurality of network service modules, the ECN-synchronization signal;

synchronizing, at each of the plurality of network service modules, responsive to receiving the RCN-

5,673,252

89

synchronization signal, an NSM-timing circuit to the RCN-synchronization signal;

collecting, at the plurality of network service modules, data from a plurality of physical devices, respectively;

transmitting, using radio waves, in a fourth channel of the frame, from the first remote cell node to a first network service module, a command signal;

receiving, at the first network service module, the command signal;

transmitting, using radio waves, synchronized to the RCN-synchronization signal and responsive to the command signal, in a fifth channel of the frame, from the first network service module to the first remote cell node, data as an NSM-packet signal;

receiving, at the first remote cell node, the NSM-packet signal;

storing, at the first remote cell node, the NSM-packet signal;

receiving, at the first remote cell node, the first polling signal; and

transmitting, synchronized to the CDT-synchronization signal and responsive to the first polling signal, in a sixth channel of the frame, from the first remote cell node to the central data terminal, a first plurality of stored NSM-packet signals as a first RCN-packet signal.

75. The method as set forth in claim 74, further including the steps of:

receiving, at the central data terminal, the first RCN-packet signal; and

transmitting, in a seventh channel of the frame, from the central data terminal to the first remote cell node, responsive to receiving the first RCN-packet signal, an acknowledgement signal.

76. The method as set forth in claim 74, further including the step of:

transmitting, in a real-time request channel of the frame, from the central data terminal to the first network service module, a CDT-request for a set of requested data;

receiving, at the first network service module, the CDT-request; and

transmitting, responsive to the CDT-request and in real-time, from the first network service module to the central data terminal, the set of requested data.

77. The method as set forth in claim 74, further including the steps of:

eavesdropping, by a second remote cell node, to the first polling signal;

eavesdropping, by the second remote cell node, to the transmission of the first plurality of stored NSM-packet signals;

comparing, by the second remote cell node, the first plurality of stored NSM-packet signals to a second plurality of stored NSM-packet signals, said second plurality of stored NSM-packet signals being stored by the second remote cell node in a memory; and

identifying a third plurality of NSM-packet signals common to both the first plurality of stored NSM-packet signals and the second plurality of stored NSM-packet signals.

90

78. The method as set forth in claim 77, further including the step of:

deleting the third plurality of NSM-packet signals from the memory.

79. The method as set forth in claim 77, further including the step of:

overwriting the third plurality of NSM-packet signals.

80. The method as set forth in claim 77, further including the step of:

transmitting, synchronized to the CDT-synchronization signal in an eighth channel of the frame, from the central data terminal to the second remote cell node, a second polling signal;

receiving, at the second remote cell node, the second polling signal; and

transmitting, synchronized to the CDT-synchronization signal, in a ninth channel of the frame, from the second remote cell node to the central data terminal, responsive to the second polling signal, the second plurality of stored NSM-packet signals, but not including the third plurality of NSM-packet signals, as a second RCN-packet signal.

81. The method as set forth in claim 80, further including the steps of:

receiving, at the central data terminal, the second RCN-packet signal; and

transmitting, from the central data terminal to the second remote cell node, responsive to receiving the second RCN-packet signal, an acknowledgement signal.

82. The method as set forth in claim 74, further including the steps of:

receiving, at the central data terminal, from a second remote cell node, a second plurality of NSM-packet signals as a second RCN-packet signal;

comparing, at the central data terminal, the second plurality of NSM-packet signals to the first plurality of NSM-packet signals; and

identifying a third plurality of NSM-packet signals common to both the first plurality of NSM-packet signals and the second plurality of NSM-packet signals.

83. The method as set forth in claim 82, further including the step of:

deleting the third plurality of NSM-packet signals.

84. The method as set forth in claim 82, further including the step of:

overwriting the third plurality of NSM-packet signals.

85. The method as set forth in claim 76, further including the steps of:

detecting, at the first network service module, an alarm condition; and

transmitting, responsive to detecting the alarm condition, in the real-time request channel of the frame, from the first network service module to the central data terminal, the alarm condition.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 1 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 1, line 6, please delete the word "Mar." and replace with the word "May".

Column 6, line 33, please replace the word "nodes" with the word "node".

Column 7, line 37, please replace the word "or" with the word "of".

Column 8, line 9, please replace the word "IPH" with the word "IRH".

Column 8, line 30, please replace the word "is" with the word "are".

Column 8, line 64, please insert the numeral "54" after the word "FIG.".

Column 9, line 1, please delete the word "and".

Column 9, line 2, please delete the word "illustrates".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 2 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 9, line 4, please replace the "." with a ",".

Column 9, line 6, please replace the "." with a ",".

Column 9, line 9, please insert the words "shows an" after the numeral "61".

Column 9, line 57, please replace the word "They" with the word "The".

Column 9, line 59, please delete the word "a".

Column 10, line 36, please replace the word "a" with the word "an".

Column 11, line 17, please delete the ",".

Column 12, line 11, please replace "pseudo-random" with the word "pseudorandom".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 3 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 12, line 49, please replace the word "Transmitter" with the word "transmitter".

Column 13, line 6, please replace the word "Transmitter" with the word "transmitter".

Column 13, line 13, please replace the word "An" with the word "an".

Column 14, line 10, please replace the word "race" with the word "rate".

Column 14, line 34, please replace the word "tapering" with the word "tampering".

Column 15, line 12, please replace the word "A" with the word "An".

Column 17, line 10, please replace the word "race" with the word "rate".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 4 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 17, line 15, please replace the word "confiming" with the word "confirming".

Column 17, line 28, please replace the word "innerfere" with the word "interfere".

Column 17, line 32, please replace the word "ocher" with the word "other".

Column 17, line 66, please replace the word "nodule" with the word "module".

Column 18, line 38, please insert the word "the" following the words "subset of".

Column 19, line 37, please delete the "-" between the words "the" and "remote".

Column 19, line 51, please replace the word "see" with the word "set".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 5 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 20, line 6, please replace the word "The" with the word "the".

Column 20, line 36, please insert a "." following the word "immediately". Please insert the word "In" preceding the words "this way".

Column 20, line 49, please replace the word "module's" with the word "modules".

Column 20, line 51, please replace the word "co" with the word "to".

Column 21, line 7, please replace the word "races" with the word "rates".

Column 21, line 14, please replace the word "general" with the word "generally".

Column 21, line 61 and 62, please replace the word "innermediate" with the word "intermediate".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 6 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 22, line 31, please replace the word "no" with the word "to".

Column 22, line 49 and 50, please delete the "-" between the words "packet" and "signal".

Column 23, line 27, please replace the word "antennas" with the word "antenna".

Column 23, line 29, please replace the word "call" with the word "cell".

Column 23, line 41, please insert the word "the" between the words "by" and "intermediate".

Column 24, line 62, please replace the word "mode" with the word "node".

Column 24, line 63, please replace "31 115" with "31-115".

Column 25, line 51, please add the word "level" following the word "terminal".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 7 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 26, line 13, please replace the word "chose" with the word "those".

Column 26, line 41, please replace the word "massage" with the word "message".

Column 26, line 53, please replace "5000" with "5,000".

Column 27, line 16, please add the word "to" before the word "suit".

Column 27, line 43, please delete one tab to align the paragraph.

Column 29, lines 13 and 14, please remove the bold highlight from the numerals "0, 1, , 28".

Column 29, line 28, please replace "IHR" with "IRH".

Column 29, line 40, please remove the bold highlight from the numerals "0 to 28".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 8 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 29, line 58, please insert the word "the" following the word "of".

Column 30, line 51, please replace "{0}.3.XX10" with "{0} .3 .XX10".

Column 30, line 52, please replace "{0}.3.X000" with "{0} .3 .X000".

Column 30, line 53, please replace "{0 }" with "{0}". Please add a space between "{0}" and the word "link".

Column 30, line 59, please remove the bold highlight from the numerals "2, 6, 10 and 14".

Column 30, line 60, please replace "{0}.3.XX10" with "{0} .3 .XX10". Please replace the word "than" with the word "that".

Column 30, line 63, please replace "{0}.3.00XX" with "{0} .3 .00XX".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 9 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 30, line 64, please replace "{0}.3.XXX0" with "{0} .3 .XXX0," and replace "{0}.3.X010" with "{0} .3 .X010".

Column 31, line 24, please replace the word "sloes" with the word "slots".

Column 32, line 50, please replace the word "synchronizatin" with the word "synchronization".

Column 35, line 13, please remove the bold highlight from the numerals "0-15".

Column 35, line 14, please remove the bold highlight from the numerals "15-28".

Column 35, line 16, please remove the bold highlight from the numerals "0-15".

Column 35, line 17, please remove the bold highlight from the numerals "0..15".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 10 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 36, line 8, please replace the words
“terminal—remote” with the words “terminal - remote”.

Column 36, line 10, please replace the word “date” with the
word “data”.

Column 36, line 25, please replace the word “relative” with
the word “relatively”.

Column 36, line 41, please highlight in bold the words “Flow
Control”.

Column 36, line 44, please highlight in bold the words “Stop-
and-Wait Flow Control”.

Column 36, line 48, please highlight in bold the words
“Sliding-Window Flow Control”.

Column 36 line 55, please highlight in bold the words “Error
Control”.

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 11 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 36, lines 61 and 62, please highlight in bold the words "automatic repeat request".

Column 36, line 65, please highlight in bold the words "Error Detection".

Column 36, line 66, please highlight in bold the words "Positive Acknowledgement".

Column 37, line 9, please highlight in bold the words "Stop-and-Wait ARQ".

Column 37, line 18, please highlight in bold the words "Go-back-N ARQ".

Column 37, line 27, please highlight in bold the words "Selective-reject ARQ".

Column 37, line 33, please replace the word "no" with the word "to".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 12 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 37, line 34, please replace the symbol " \leq " with the symbol " \leq ".

Column 37, line 37, please replace the word "TranSport" with the word "Transport".

Column 37, line 48, please highlight in bold the words "Error Detection and Correction".

Column 37, line 58, please insert "e.g.," following the words "cyclic code".

Column 37, line 65, please highlight in bold the words "Path Redundancy and Collision Avoidance".

Column 38, line 16, please highlight in bold the words "Message Redundancy and Self-sufficiency".

Column 39, line 15, please replace the word "date" with the word "data".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 13 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 40, line 34, please replace the word "sloes" with the word "slots".

Column 41, line 42, please delete the "." between the words "IRD/RIR" and "links".

Column 42, line 1, please highlight in bold the words "Broadcast Addressing".

Column 44, line 58, please replace the word "no" with the word "to".

Column 44, line 65, please delete the "." between the words "every" and "little".

Column 44, line 66, please delete the "." between the words "requires" and "that".

Column 45, line 5, please replace the word "no" with the word "to" following the words "decisions as". Please replace the word "no" with the word "to" following the words "and when".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 14 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 46, line 12, please replace the word "illustrates" with the word "illustrate".

Column 48, line 21, please replace the word "an" with the word "a".

Column 48, line 65, please replace the word "non" with the word "not".

Column 48, line 67, please replace the word "in" with the word "it".

Column 52, line 9, please insert the word "RIQ" between the words "the" and "after".

Column 53, line 6, please replace the word "transmitred" with the word "transmitted". Please delete the "," following the word "transmitted". Please replace the word "non-discarded" with the word "non-discardable".

Column 53, line 7, please delete the "," between the words "to" and "identify".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. :5,673,252

Page 15 of 31

DATED :September 30, 1997

INVENTOR(S) :Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 53, line 38, please highlight in bold the words "Edge-Gather and Partial Poll".

Column 53, line 44, please replace the word "ochers" with the word "others".

Column 53, line 47, please highlight in bold the words "Wider-Gather and Partial Poll".

Column 53, line 52, please highlight in bold the words "Table Based Message Acceptance".

Column 53, line 55, please replace the word "an" with the word "a".

Column 53, line 57, please highlight in bold the words "Random Discard".

Column 53, line 64, please highlight in bold the words "Random Discard Based on Signal Strength".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 16 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 54, line 3, please highlight in bold the words "Signal Strength with Exception List".

Column 54, line 60, please highlight in bold the words "Control Fields".

Column 54, line 62, please highlight in bold the word "msgtype".

Column 54, line 66, please highlight in bold the word "msgno".

Column 55, line 5, please highlight in bold the word "revpoll".

Column 55, line 8, please highlight in bold the words "protocol status".

Column 55, line 18, please highlight in bold the word "priority".

Column 55, line 20, please highlight in bold the words "Network Message Field".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 17 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 55, line 32, please replace the word "an" with the word "a".

Column 55, line 36, please replace the ";" with a "," following the word "numbering".

Column 55, line 39, please highlight in bold the words "Broadcast to Class Address Messages".

Column 55, line 43, please replace the word "(cofftend)" with the word "(command)".

Column 55, lines 49 and 50, please highlight in bold the words "Delivery to Individually Addressed Network Service Modules".

Column 55, line 60, please highlight in bold the words "CAT Distribution".

Column 56, line 1, please highlight in bold the word "nsmtyp".

Column 56, line 2, please highlight in bold the word "appltyp".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252 Page 18 of 31
DATED : September 30, 1997
INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 56, line 7, please replace the word "applcyp" with the word "appltyp".

Column 56, line 9, please highlight in bold the words "CAT ID".

Column 56, line 11, please highlight in bold the words "CAT version".

Column 56, line 13, please highlight in bold the words "NRR channel bitmap".

Column 56, line 14, please remove the bold highlighting from the numerals "1" and "28".

Column 56, line 18, please highlight in bold the words "RND broadcast subchannel".

Column 56, line 26, please highlight in bold the words "RND reverse poll subchannel".

Column 56, line 60, please replace the word "line" with the word "link".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 19 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 57, line 1, please highlight in bold the word "control".

Column 57, line 4 please replace both " \leq " symbols with the symbol " \leq ".

Column 57, line 10, please highlight in bold the word "length".

Column 57, line 12, please highlight in bold the words "NSM msg IDs".

Column 57, line 14, please highlight in bold the words "intermediate **CRC**".

Column 57, line 15, please highlight in bold the words "RCN status".

Column 57, line 25, please highlight in bold the words "NSM msg contents".

Column 57, line 29, please highlight in bold the word "ptag".

Column 57, line 30, please highlight in bold the word "vtag".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 20 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 57, line 31, please highlight in bold the word "dtag".

Column 57, line 33, please highlight in bold the word "etag".

Column 57, line 34, please highlight in bold the word "ttag".

Column 57, line 35, please highlight in bold the word "ftag".

Column 57, line 36, please highlight in bold the word "stag".

Column 57, line 37, please highlight in bold the word "crc".

Column 57, line 38, please highlight in bold the word "ctag".

Column 57, line 39, please highlight in bold the word "mtag".

Column 58, line 13, please replace the word "IPH" with the word "IRH".

Column 58, line 16, please highlight in bold the word "slottype".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 21 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 58, line 18, please highlight in bold the word "seqno".

Column 58, line 26, please highlight in bold the word "msgtype".

Column 58, line 27, please highlight in bold the word "index".

Column 58, line 29, please highlight in bold the word "special application control".

Column 58, line 31, please highlight in bold the word "SAC enable".

Column 58, line 37, please replace the word "IPH" with the word "IRH".

Column 58, line 49, please replace the word "than" with the word "that".

Column 59, line 7, please replace the words "form an" with the word "format".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 22 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 59, line 37, please highlight in bold the word "subchan".

Column 59, line 38, please highlight in bold the word "initialSlot".

Column 59, line 40, please highlight in bold the word "lifetime".

Column 59, line 42, please highlight in bold the word "position".

Column 60, line 9, please replace the word "an" with the word "at".

Column 60, line 18, please replace the words "form an" with the word "format".

Column 60, line 30, please highlight in bold the word "subchan".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 23 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 60, line 31, please highlight in bold the word "repetitions".

Column 60, line 32, please highlight in bold the word "lifetime".

Column 60, line 35, please highlight in bold the word "hashParms".

Column 60, line 37, please highlight in bold the word "criteria".

Column 61, line 37, please replace the word "Those" with the word "those".

Column 61, line 39, please replace the word "which" with the word "with".

Column 62, line 15, please highlight in bold the words "Fault Management".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 24 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 62, line 21, please replace the word "Transparent" with the word "transparent".

Column 62, line 29, please highlight in bold the words "Performance Management".

Column 63, lines 4 and 5, please highlight in bold the entirety of lines 4 and 5.

Column 63, line 53, please replace the word "These" with the word "these".

Column 65, line 23, please replace the word "an" with the word "a".

Column 65, line 59, please replace the paragraph heading "MSGHD.RIO" with "MSGHD.RIQ".

Column 65, line 65, please replace the paragraph heading "MSGHD.UIO" with "MSGHD.UIQ".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 25 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 66, line 10, please replace the word "She" with the word "the".

Column 66, lines 63 and 64, please highlight in bold the entirety of lines 63 and 64.

Column 67, line 3, please replace the word "an" with the word "a".

Column 68, line 13, please replace the word "than" with the word "that".

Column 68, lines 16 and 17, please highlight in bold the entirety of lines 16 and 17.

Column 68, lines 32 and 33, please highlight in bold the entirety of lines 32 and 33.

Column 68, line 56, please highlight in bold the words "Configuration Control Database Items".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 26 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 68, line 59, please replace the words "customer+an" with the words "customer + an".

Column 68, line 61, please highlight in bold the words "Performance Management Database Items".

Column 69, line 6, please insert a paragraph break preceding the word "Ten".

Column 69, line 8, please highlight in bold the words "Fault Management Database Items".

Column 69, line 18, please highlight in bold the entirety of line 18.

Column 70, lines 22 and 23, please have the number "240,000" placed on one line only.

Column 71, line 7, please replace the word "wish" with the word "with".

Column 71, line 13, please delete the word "is".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 27 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 72, line 46, please replace the word "races" with the word "rates".

Column 72, line 56, please replace the word "a" with the word "an".

Column 73, line 27, please insert the word "the" between the words "in" and "future".

Column 77, line 4, please replace the word "tell" with the word "cell".

Column 77, line 17, please replace the word "step" with the word "steps".

Column 77, line 26, please replace the word "a" with the word "an".

Column 77, line 36, please replace the word "an" with the word "a".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 28 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 79, line 7, please replace the word "step" with the word "steps".

Column 79, line 40, please replace the word "step" with the words "steps".

Column 80, line 12, please replace the word "step" with the word "steps".

Column 81, line 15, please replace the word "etch" with the word "each".

Column 81, line 49, please replace the ":" with a ";".

Column 82, line 4, please replace the word "step" with the word "steps".

Column 82, line 38, please replace the word "step" with the word "steps".

Column 82, line 46, please replace the word "a" with the word "an".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 29 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 82, line 58, please replace the word "packet" with the word "RCN-packet".

Column 84, line 11, please replace the word "an" with the word "a".

Column 84, line 16, please replace the word "step" with the word "steps".

Column 84, line 51, please replace the word "step" with the word "steps".

Column 86, line 24, please replace the word "step" with the word "steps".

Column 86, line 36, please replace the word "an" with the word "at".

Column 86, line 67, please replace the word "step" with the word "steps".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,673,252

Page 30 of 31

DATED : September 30, 1997

INVENTOR(S) : Dennis F. Johnson, Don Marcynuk, Erwin Holowick.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 87, line 16, please replace the word "an" with the word "at".

Column 88, line 17, please replace the word "step" with the word "steps".

Column 88, line 65, please replace the word "ECN-synchronization" with the word "RCN-synchronization".

Column 89, line 31, please replace the word "an" with the word "a".

Column 89, line 36, please replace the word "step" with the word "steps".

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : **5,673,252**

Page 31 of 31

DATED : **September 30, 1997**

INVENTOR(S) : **Dennis F. Johnson, Don Marcynuk, Erwin Holowick.**

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 90, line 9, please replace the word "step" with the word "steps".

Signed and Sealed this
Seventh Day of July, 1998



Attest:

BRUCE LEHMAN

Attesting Officer

Commissioner of Patents and Trademarks



US006044062A

United States Patent [19]
Brownrigg et al.

[11] **Patent Number:** **6,044,062**
 [45] **Date of Patent:** **Mar. 28, 2000**

- [54] **WIRELESS NETWORK SYSTEM AND METHOD FOR PROVIDING SAME**
- [75] Inventors: **Edwin B. Brownrigg**, Roseville;
Thomas W. Wilson, Alameda, both of Calif.
- [73] Assignee: **CommUnique, LLC**, Alameda, Calif.
- [21] Appl. No.: **08/760,895**
- [22] Filed: **Dec. 6, 1996**
- [51] **Int. Cl.⁷** **H04Q 7/38**
- [52] **U.S. Cl.** **370/238; 370/315; 455/445; 455/11.1**
- [58] **Field of Search** 370/310, 315, 370/327, 328, 338, 351, 237, 238, 501, 401, 402, 255, 256, 389; 455/11.1, 445; 340/826, 827, 825.03; 709/238, 239, 240, 241, 242, 243, 244

- [56] **References Cited**
- U.S. PATENT DOCUMENTS
- | | | | |
|-----------|--------|-----------------|-----------|
| 5,282,204 | 1/1994 | Shpancer et al. | 370/350 |
| 5,592,491 | 1/1997 | Dinkins | 455/111.1 |
| 5,757,783 | 5/1998 | Eng et al. | 455/11.1 |
| 5,790,938 | 8/1998 | Talarmo | 455/11.1 |

OTHER PUBLICATIONS

Westcott, Jil et al., "A Distributed Routing Design for a Broadcast Environment," IEEE 1982, pp. 10.4-1-10.4-5.

Kahn, Robert E., "Advances in Packet Radio Technology," IEEE Nov. 1978, vol. 66, No. 11, pp. 1468-1496.

Kahn, Robert E., "The Organization of Computer Resources into a Packet Radio Network," IEEE Jan. 1977, vol. Com-25, No. 1, pp. 169-178.

Frankel, Michael S., "Packet Radios Provide Link for Distributed, Survivable C³ in Post-Attack Scenarios," MSN Jun. 1983.

Lauer, Greg et al., "Communications in the Information Age," pp. 15.1.1-15.1.4, IEEE Globecom '84, 1984.

Westcott, Jil A., "Issues in Distributed Routing for Mobile Packet Radio Network," IEEE 1982, pp. 233-238.

Gower, Neil et al., "Congestion Control Using Pacing in a Packet Radio Network," IEEE 1982, pp. 23.1-1-23.1-6.

MacGregor, William et al., "Multiple Control Stations in Packet Radio Networks," IEEE 1982, pp. 10.3-1-10.3-5.

Shacham, Nachum et al., "Future Directions in Packet Radio Technology," IEEE 1985, pp. 93-98.

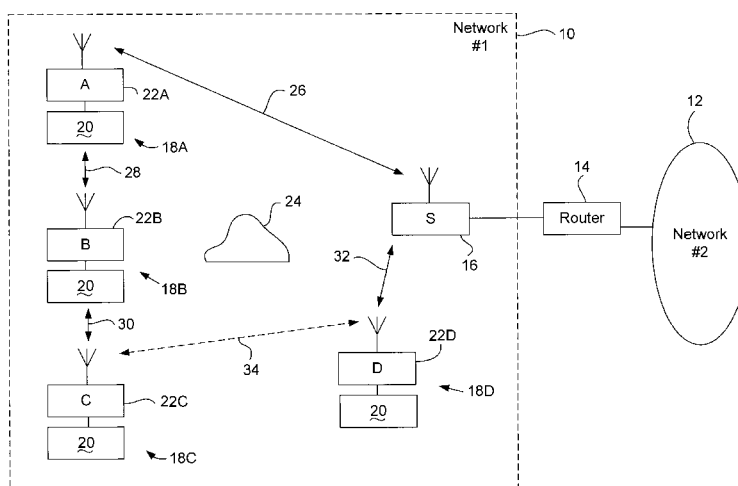
Jubin, John, "Current Packet Radio Network Protocols," IEEE 1985, pp. 86-92.

Primary Examiner—Huy D. Vu
Attorney, Agent, or Firm—Hickman Stephens Coleman & Hughes, LLP

[57] **ABSTRACT**

A wireless network system includes a server having a server controller and a server radio modem, and a number of clients each including a client controller and a client radio modem. The server controller implements a server process that includes the receipt and the transmission of data packets via the radio modem. The client controllers of each of the clients implements a client process that includes the receipt and transmission of data packets via the client radio modem. The client process of each of the clients initiates, selects, and maintains a radio transmission path to the server that is either a direct path to the server, or is an indirect path or "link" to the server through at least one of the remainder of the clients. A method for providing wireless network communication includes providing a server implementing a server process including receiving data packets via a radio modem, sending data packets via the server radio modem, communicating with the network, and performing housekeeping functions, and further includes providing a number of clients, each implementing a client process sending and receiving data packets via a client radio modem, maintaining a send/receive data buffer, and selecting a radio transmission path to the server. The radio transmission path or "link" is either a direct path to the server, or an indirect path to the server through at least one of the remainder of the clients. The process preferably optimizes the link to minimize the number of "hops" to the server.

16 Claims, 42 Drawing Sheets



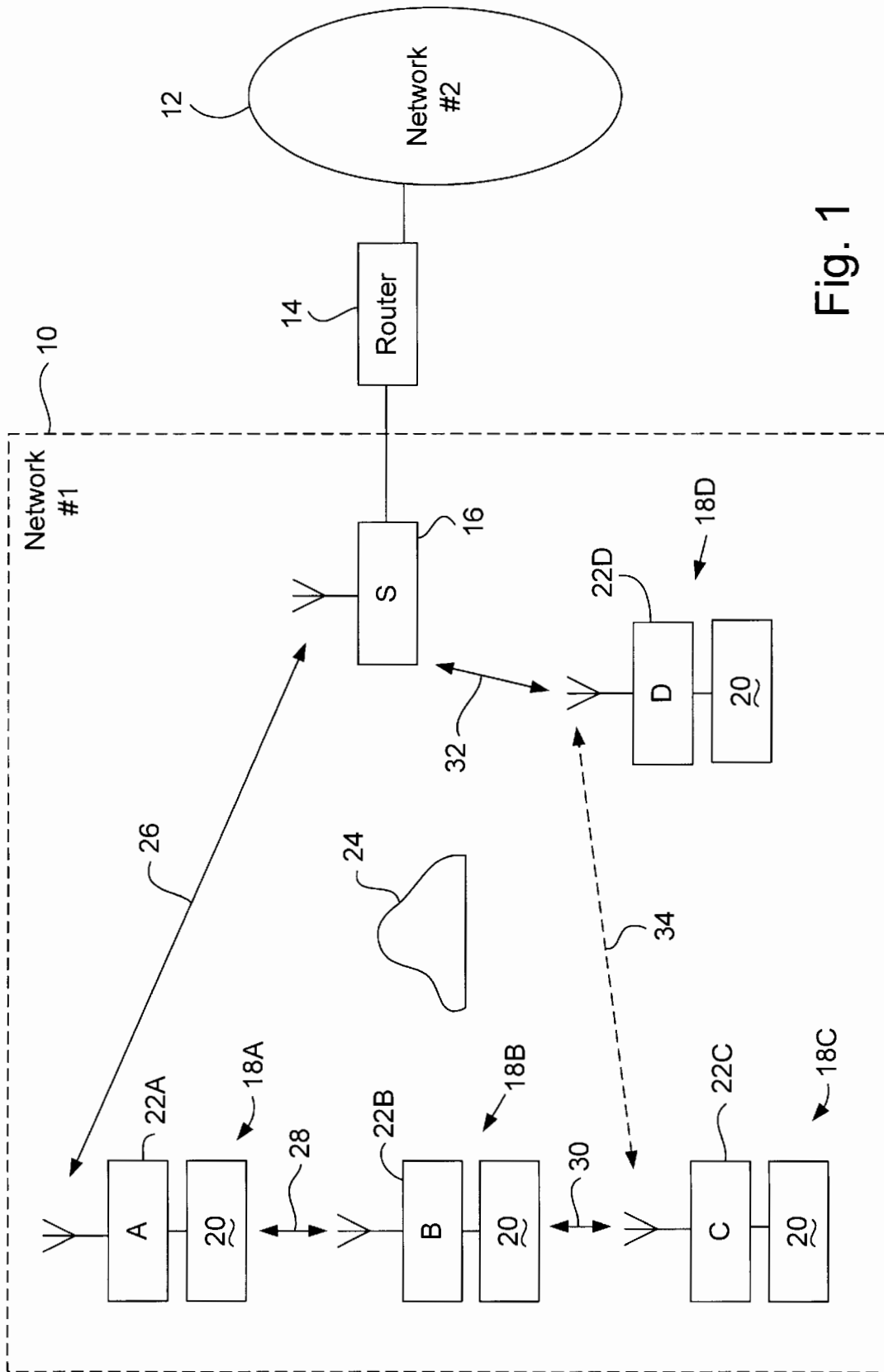
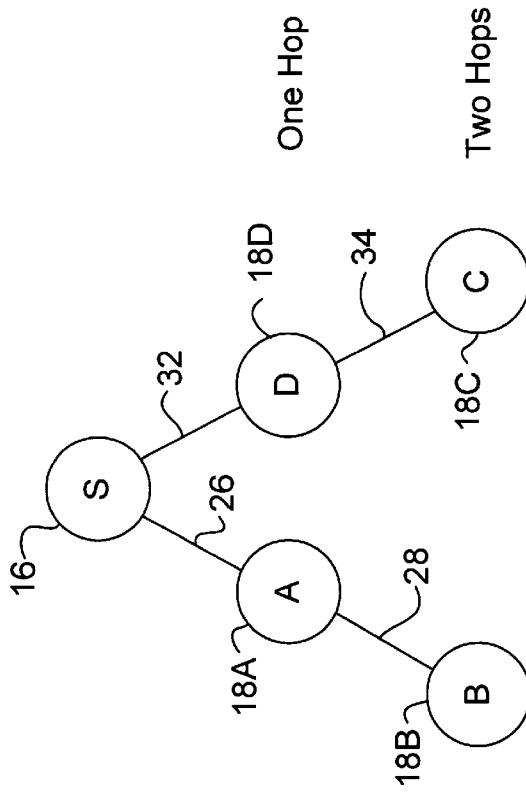


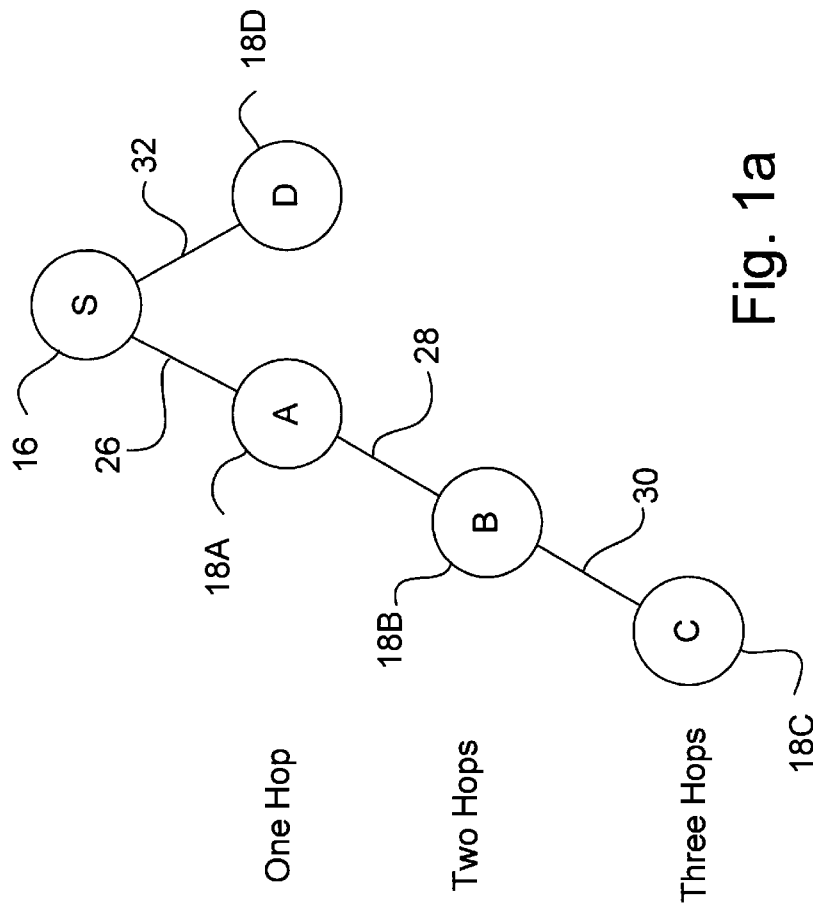
Fig. 1



One Hop

Two Hops

Fig. 1b



One Hop

Two Hops

Three Hops

Fig. 1a

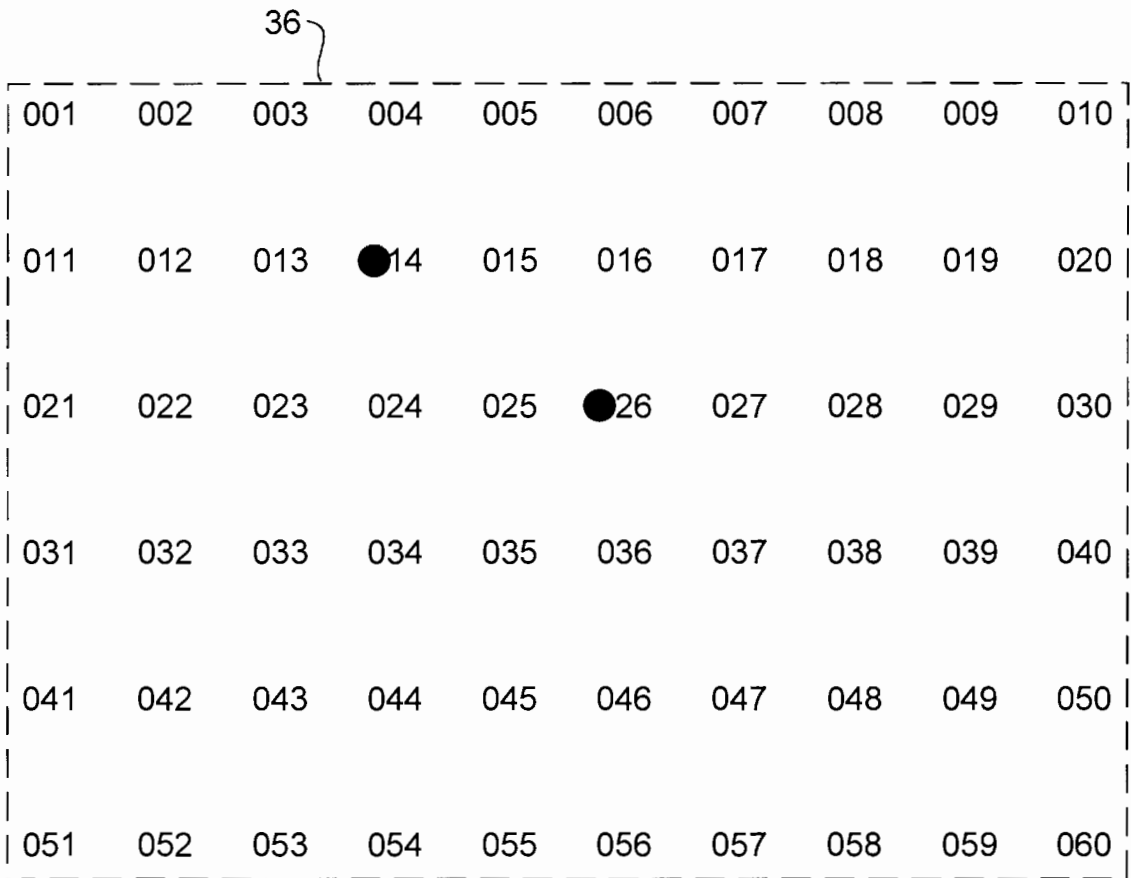
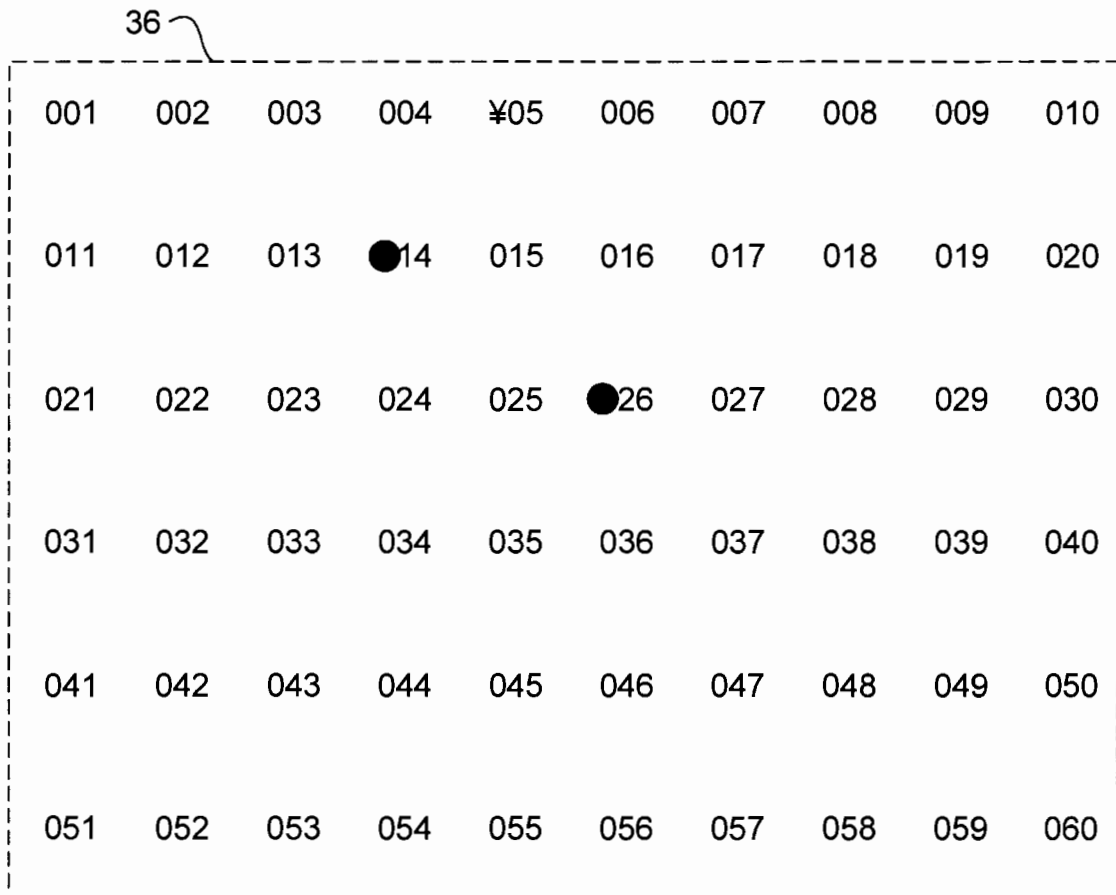


Fig. 2a



There is only one Internet server in range of client 5.

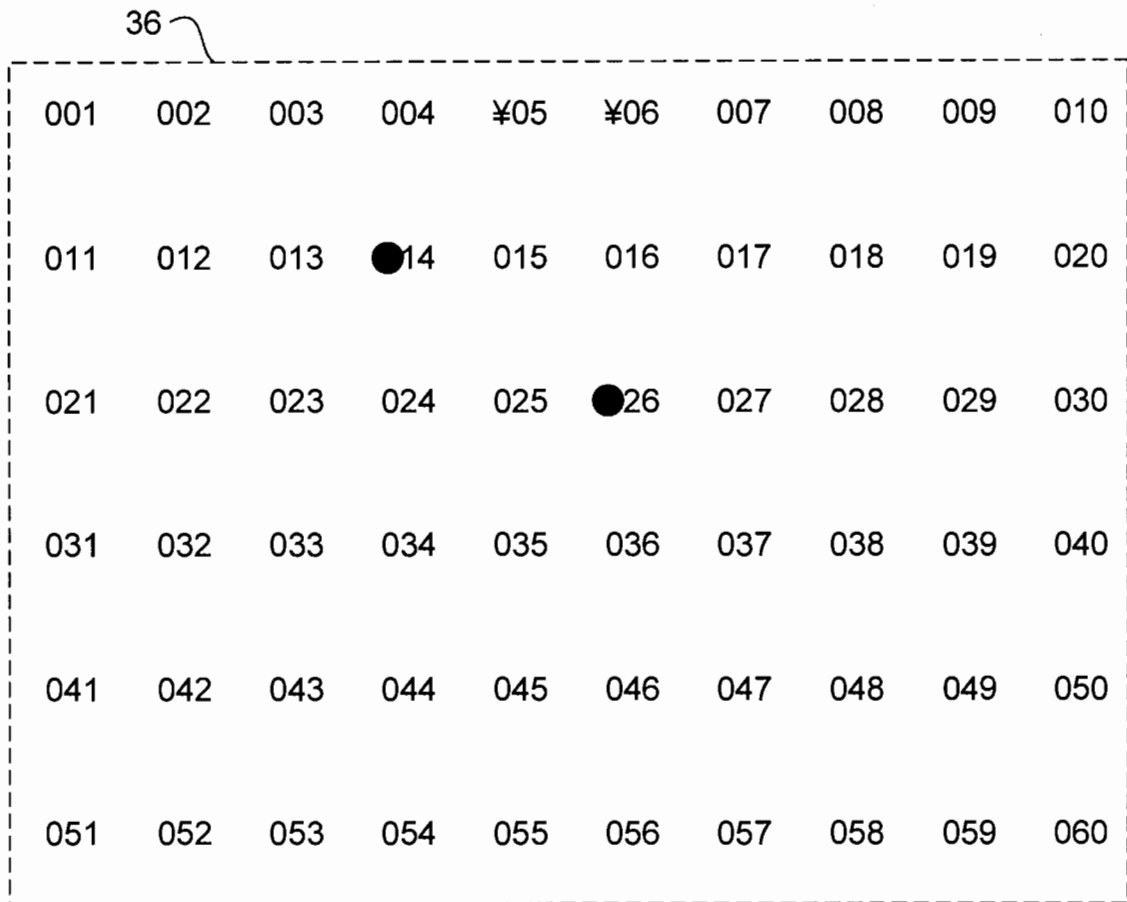
5 will issue an "I am alive" packet seeking a route to the Internet.

Internet server 14 will respond and add user client 5 to its routing table as its left son.

The updated routing table of Internet server 14 is: 14(05).

The route from user client 5 to the Internet is: 05>14.

Fig. 2b



There is only one user client in range of client 6.

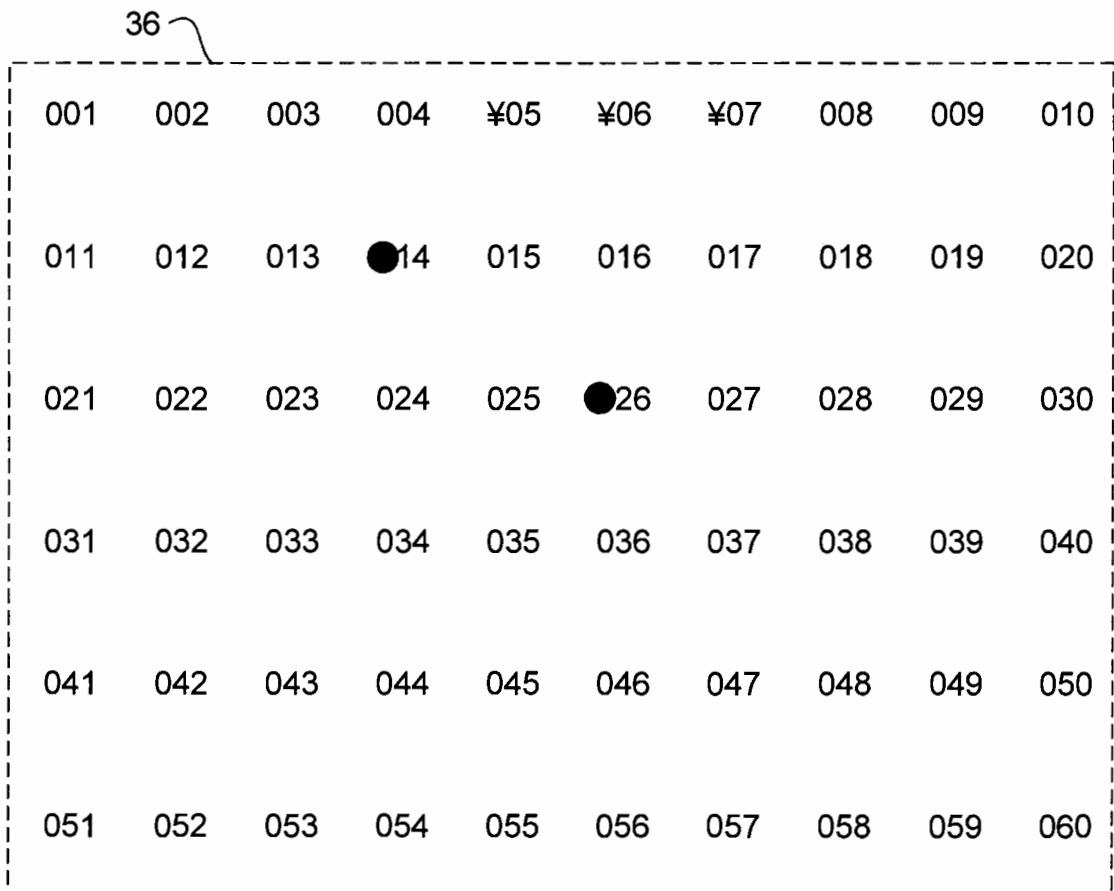
6 will issue an "I am alive" packet seeking a client repeater route to the Internet.

5 will respond and add 6 to its routing table as its left son.

The updated routing table of Internet server 14 is:
14(05(06)).

The route from user client 6 to the Internet is: 06>05>14.

Fig. 2c



There is only one user client in range of client 7.

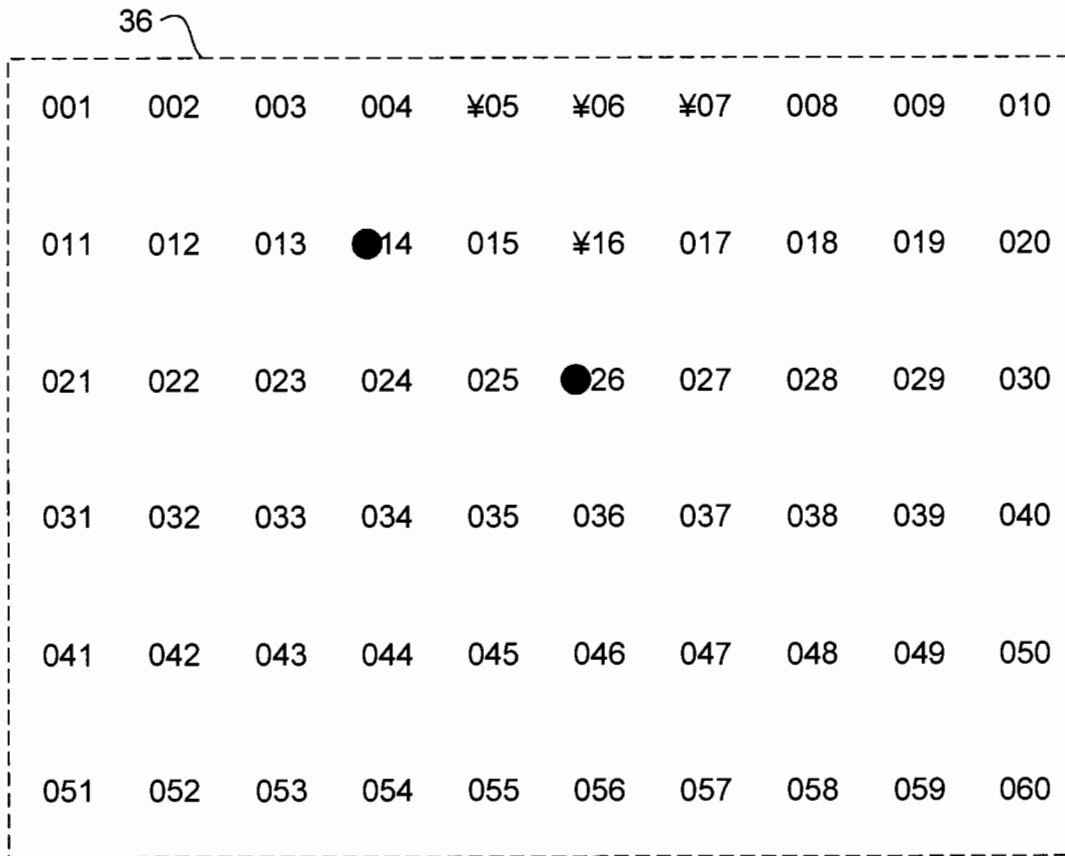
7 will issue an "I am alive" packet seeking a client repeater route to the Internet.

6 will respond and add 7 to its routing table as its left son.

The updated routing table of Internet server 14 is:
14(05(06(07))).

The route from user client 7 to the Internet is: 07>06>05>14.

Fig. 2d



There is only one Internet server in range of client 16.

16 will issue an "I am alive" packet seeking a route to the Internet.

Internet server 26 will respond and add user client 16 to its routing table as its left son.

The updated routing table of Internet server 26 is: 26(16).

The route from user client 16 to the Internet is: 16>26.

Fig. 2e

Server 14 = 14(05(06))

Server 26 = 26(16(07))

Client 05 = 05(06); >14

Client 06 = 06; >05>14

Client 07 = 07; >16>26

Client 16 = 16(07); >26

In a universe of 6 nodes, of which 2 are servers, the average hop distance from a client to an Internet server is 1.5.

Fig. 2f

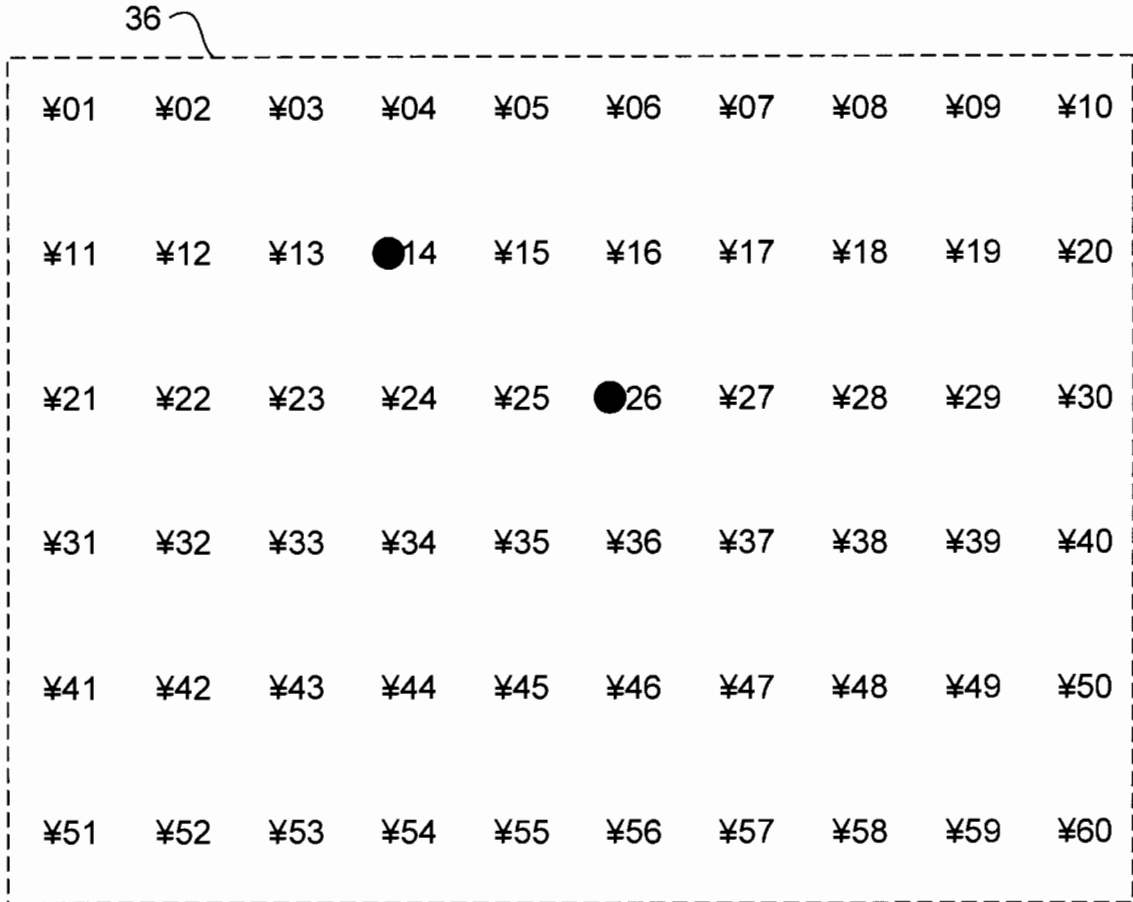


Fig. 2g

U.S. Patent

Mar. 28, 2000

Sheet 10 of 42

6,044,062

Server 14 =
 14(24(34),23(32(41(51),42(52)),33(43)),13(22(31)),05(06),
 04,03(02(11),12(01,21)))

Server 26 =
 26(37(48(59(60),49(50),58),38(39(40)),28(29(30)),47(57)),
 36(46(56)),35(44(53,54),45(55)),27(18(19(20))),25,
 17(08(09(10)),07),16,15)

Client 01 = 01; >12>03>14
 Client 02 = 02(11); >03>14
 Client 03 = 03(02(11),12(01,21)); >14
 Client 04 = 04; >14
 Client 05 = 05(06); >14
 Client 06 = 06; >05>14
 Client 07 = 07; >17>26
 Client 08 = 08(09(10)); >17>26
 Client 09 = 09(10); >08>17>26
 Client 10 = 10; >09>08>17>26
 Client 11 = 11; >02>03>14
 Client 12 = 12(01,21); >03>14
 Client 13 = 13(22(31)); >14
 Client 15 = 15; >26
 Client 16 = 16; >26
 Client 17 = 17(08(09(10)),07); >26
 Client 18 = 18(19(20)); >27>26
 Client 19 = 19(20); >18>27>26
 Client 20 = 20; >19>18>27>26
 Client 21 = 21; >12>03>14
 Client 22 = 22(31); >13>14
 Client 23 = 23(32(41(51),42(52)),33(43)); >14
 Client 24 = 24(34); >14
 Client 25 = 25; >26
 Client 27 = 27(18(19(20))); >26
 Client 28 = 28(29(30));>37>26
 Client 29 = 29(30);>28>37>26
 Client 30 = 30;>29>28>37>26
 Client 31 = 31;>22>13>14

Fig. 2h'

Client 32 = 32(41(51),42(52)); >23>14
Client 33 = 33(43);>23>14
Client 34 = 34;>24>14
Client 35 = 35(44(53,54),45(55));>26
Client 36 = 36(46(56));>26
Client 37 = 37(48(59(60),49(50),58),38(39(40)),
28(29(30)),47(57));>26
Client 38 = 38(39(40));>37>26
Client 39 = 39(40);>38>37>26
Client 40 = 40;>39>38>37>26
Client 41 = 41(51);>32>23>14
Client 42 = 42(52);>32>23>14
Client 43 = 43;>33>23>14
Client 44 = 44(53,54);>35>26
Client 45 = 45(55);>35>26
Client 46 = 46(56);>36>26
Client 47 = 47(57);>37>26
Client 48 = 48(59(60),49(50),58);>37>26
Client 49 = 49(50);>48>37>26
Client 50 = 50;>49>48>37>26
Client 51 = 51;>41>32>23>14
Client 52 = 52;>42>32>23>14
Client 53 = 53;>44>35>26
Client 54 = 54;>44>35>26
Client 55 = 55;>45>35>26
Client 56 = 56;>46>36>26
Client 57 = 57;>47>37>26
Client 58 = 58;>48>37>26
Client 59 = 59(60);>48>37>26
Client 60 = 60;>59>48>37>26

In a universe of 60 nodes, of which 2 are servers, the average hop distance from a client to an Internet server is 2.36206897.

Fig. 2h"

Traversing user client universe

User client, 9, has 5 user client neighbors.

User client, 9, will probe each for the shortest route to the Internet.
9's current route to the Internet is: nonexistent.

9 is now probing 10.
User client, 9, has no Internet server.
9's current route to the Internet is: nonexistent.

9 is now probing 20.
User client, 9, has no Internet server.
9's current route to the Internet is: nonexistent.

9 is now probing 19.
User client, 9, has no Internet server.
9's current route to the Internet is: nonexistent.

9 is now probing 18.
User client, 9, has no Internet server.
9's current route to the Internet is: nonexistent.

9 is now probing 8.
User client 8 will add 9 to its routing table as its left son.

The updated routing table of Internet server 14 is:
14(05(06(07(08(09))))),04,03).

The route from user client 9 to the Internet is:
09>08>07>06>05>14.

Fig. 2i

Traversing user client universe . . .

User client, 29, has 8 user client neighbors.

User client, 29, will probe each for the shortest route to the Internet.
29's current route to the Internet is: nonexistent.

29 is now probing 19.

User client 19 will add 29 to its routing table as its left son.

The updated routing table of Internet server 14 is:
14(24,23,13,05(06(07(08(18(28),09(19(29),10(20)))))),
04,03(12(22,21))).

The route from user client 29 to the Internet is:
29>19>09>08>07>06>05>14.

Fig. 2j

Traversing user client universe

User client, 7, has 5 user client neighbors.

User client, 7, will probe each for the shortest route to the Internet.
7's current route to the Internet is: 07>06>05>14.

7 is now probing 8.

7's current route to the Internet is: 07>06>05>14.

7 is now probing 18.

7's current route to the Internet is: 07>06>05>14.

7 is now probing 17.

User client, 7, has probed its neighbor, user client, 17, and found a shorter path to the Internet.

The old routing table of Internet server, 14, is:

14(24(34(44(54))),23(33(43(53))),13,05(06(07(08(18(28(38(48(58)))))),
09(19(29(39(49(59))))),10(20(30(40(50(60))))))))) ,04,03(02,12(01,
22(32(42(52))),21(31(41(51)))))).

The updated routing table of Internet server, 26, is:

26(37(47(57)),36(46(56)),35(45(55)),27,25,17(07(08(18(28(38
(48(58))))),09(19(29(39(49(59))))),10(20(30(40(50(60))))))))) ,16,15).

The route from user client, 7, to the Internet is: 07>17>26.

7's current route to the Internet is: 07>17>26.

7 is now probing 16.

7's current route to the Internet is: 07>17>26.

7 is now probing 6.

7's final route is 07>17>26.

Fig. 2k

Traversing user client universe

User client, 8, has 5 user client neighbors.

User client, 8, will probe each for the shortest route to the Internet.
8's current route to the Internet is: 08>07>17>26.

8 is now probing 9.
8's current route to the Internet is: 08>07>17>26.

8 is now probing 19.
8's current route to the Internet is: 08>07>17>26.

8 is now probing 18.
8's current route to the Internet is: 08>07>17>26.

8 is now probing 17.
User client, 8, has probed its neighbor, user client, 17, and found a shorter path to the Internet.

The old routing table of Internet server, 26, is:
26(37(47(57)),36(46(56)),35(45(55)),27,25,17(07(08(18(28(38(48(58))))),09(19(29(39(49(59))))),10(20(30(40(50(60))))))))) ,16,15).

The updated routing table of Internet server, 26, is:
26(37(47(57)),36(46(56)),35(45(55)),27,25,17(08(18(28(38(48(58))))),09(19(29(39(49(59))))),10(20(30(40(50(60))))))))) ,07,16,15).

The route from user client, 8, to the Internet is: 08>17>26.
8's current route to the Internet is: 08>17>26.

8 is now probing 7.
8's final route is 08>17>26.

Fig. 2I

Traversing user client universe

User client, 18, has 8 user client neighbors.

User client, 18, will probe each for the shortest route to the Internet.
18's current route to the Internet is: 18>08>17>26.

18 is now probing 8.

18's current route to the Internet is: 18>08>17>26.

18 is now probing 9.

18's current route to the Internet is: 18>08>17>26.

18 is now probing 19.

18's current route to the Internet is: 18>08>17>26.

18 is now probing 29.

18's current route to the Internet is: 18>08>17>26.

18 is now probing 28.

18's current route to the Internet is: 18>08>17>26.

18 is now probing 27.

User client, 18, has probed its neighbor, user client, 27, and found a shorter path to the Internet.

The old routing table of Internet server, 26, is:

26(37(47(57)),36(46(56)),35(45(55)),27,25,17(08(18(28(38(48(58))))),09(19(29(39(49(59))))),10(20(30(40(50(60))))))))) ,07),16,15).

The updated routing table of Internet server, 26, is:

26(37(47(57)),36(46(56)),35(45(55)),27(18(28(38(48(58))))),25,17(08(09(19(29(39(49(59))))),10(20(30(40(50(60))))))))) ,07),16,15).

The route from user client, 18, to the Internet is: 18>27>26.

18's current route to the Internet is: 18>27>26.

18 is now probing

18's final route is 18>27>26.

Fig. 2m

User client, 29, has 8 user client neighbors.

User client, 29, will probe each for the shortest route to the Internet.
29's current route to the Internet is: 29>19>18>27>26.

29 is now probing 19.

29's current route to the Internet is: 29>19>18>27>26.

29 is now probing 20.

29's current route to the Internet is: 29>19>18>27>26.

29 is now probing 30.

29's current route to the Internet is: 29>19>18>27>26.

29 is now probing 40.

29's current route to the Internet is: 29>19>18>27>26.

29 is now probing 39.

29's current route to the Internet is: 29>19>18>27>26.

29 is now probing 38.

29's current route to the Internet is: 29>19>18>27>26.

29 is now probing 28.

User client, 29, has probed its neighbor, user client, 28, and found a shorter path to the Internet.

The old routing table of Internet server, 26, is:

26(37(28(38(48(58))),47(57)),36(46(56)),35(45(55)),27(18(19(20(30(40(50(60))))) ,29(39(49(59))))),25,17(08(09(10)),07),16,15).

The updated routing table of Internet server, 26, is:

26(37(28(29(39(49(59))),38(48(58))),47(57)),36(46(56)),35(45(55)),27(18(19(20(30(40(50(60))))) ,25,17(08(09(10)),07),16,15).

The route from user client, 29, to the Internet is: 29>28>37>26.

29's current route to the Internet is: 29>28>37>26.

29 is now probing 18.

29's final route is 29>28>37>26.

Fig. 2n

Traversing user client universe . . .

User client, 44, has 8 user client neighbors.

User client, 44, will probe each for the shortest route to the Internet.
44's current route to the Internet is: 44>34>24>14.

44 is now probing 34.

44's current route to the Internet is: 44>34>24>14.

44 is now probing 35.

User client, 44, has probed its neighbor, user client, 35, and found a shorter path to the Internet.

The old routing table of Internet server, 14, is:

14(24(34(44(54))),23(32(41(51),42(52)),33(43(53))),13(22(31)),
05(06),04,03(02,12(01(11),21))).

The updated routing table of Internet server, 14, is:

14(24(34),23(32(41(51),42(52)),33(43(53))),13(22(31)),05(06),04,
03(02,12(01(11),21))).

The updated routing table of Internet server, 26, is:

26(37(38(39(40(50(60)),49(59)),48(58)),28(29(30)),47(57)),36(46
(56)),35(44(54),45(55)),27(18(19(20))),25,17(08(09(10)),07),16,15).

The route from user client, 44, to the Internet is: 44>35>26.

44's current route to the Internet is: 44>28>37>26.

44 is now probing . . .

.

.

.

44's final route is 44>35>26.

Fig. 2o

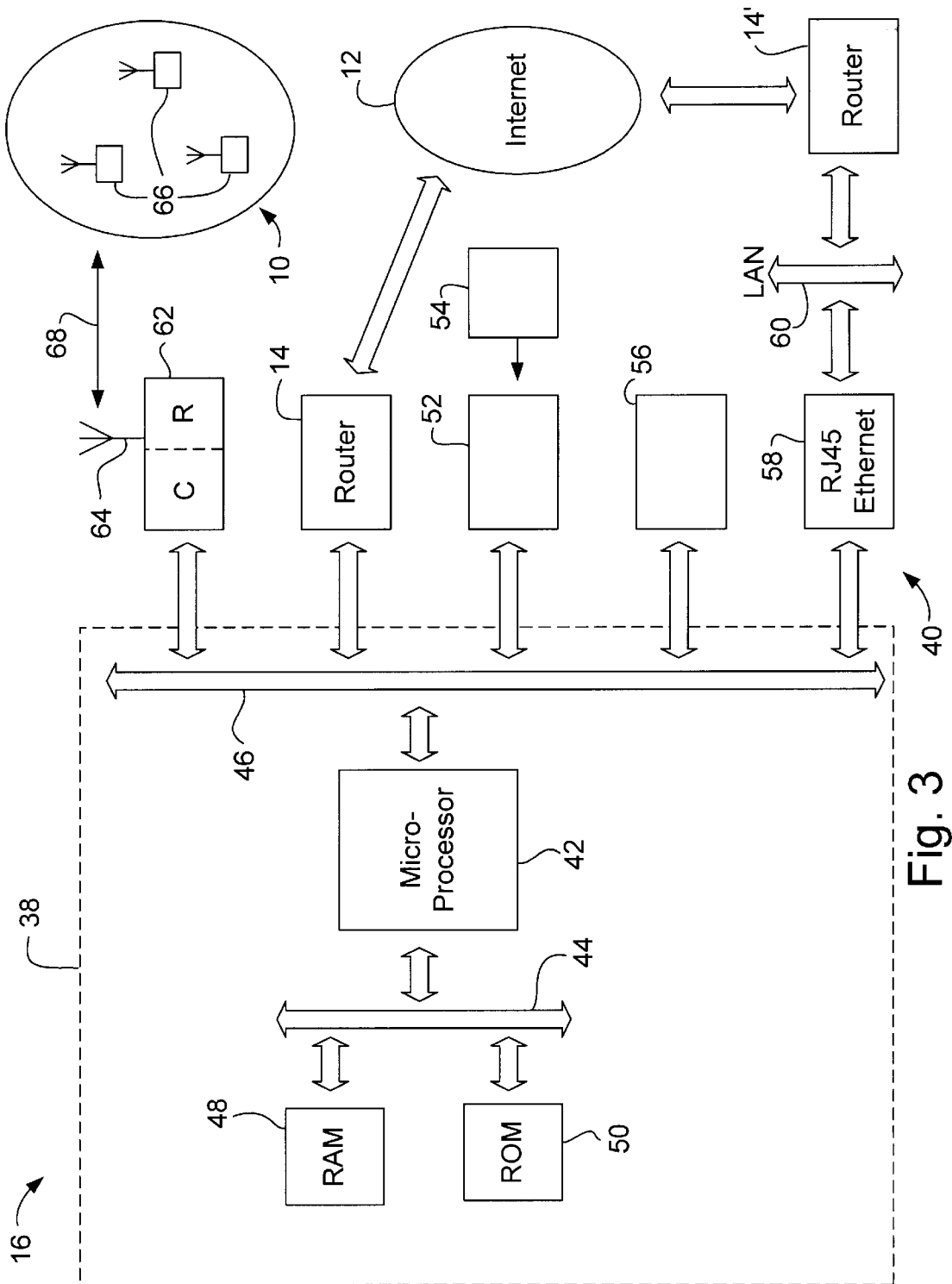


Fig. 3

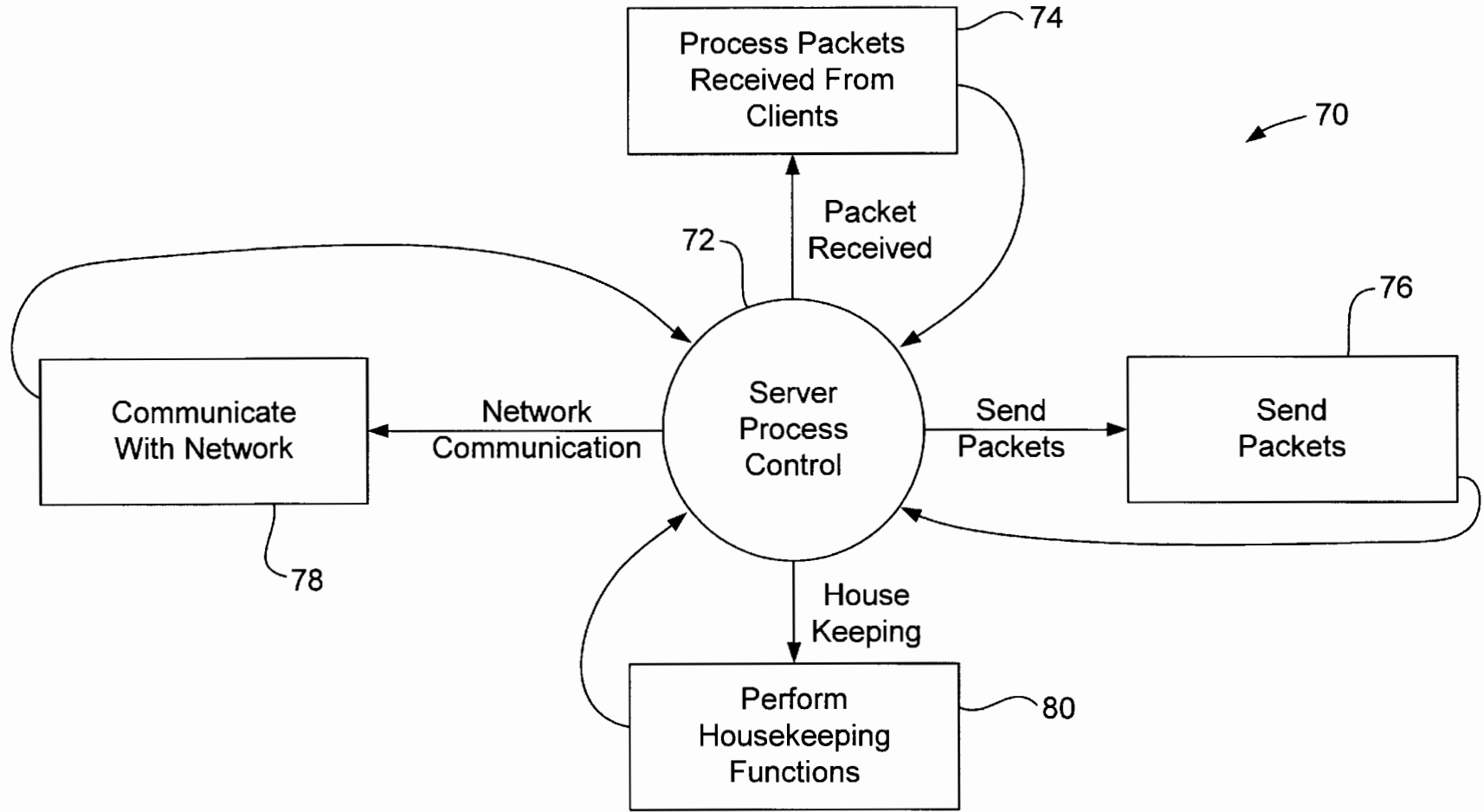


Fig. 4

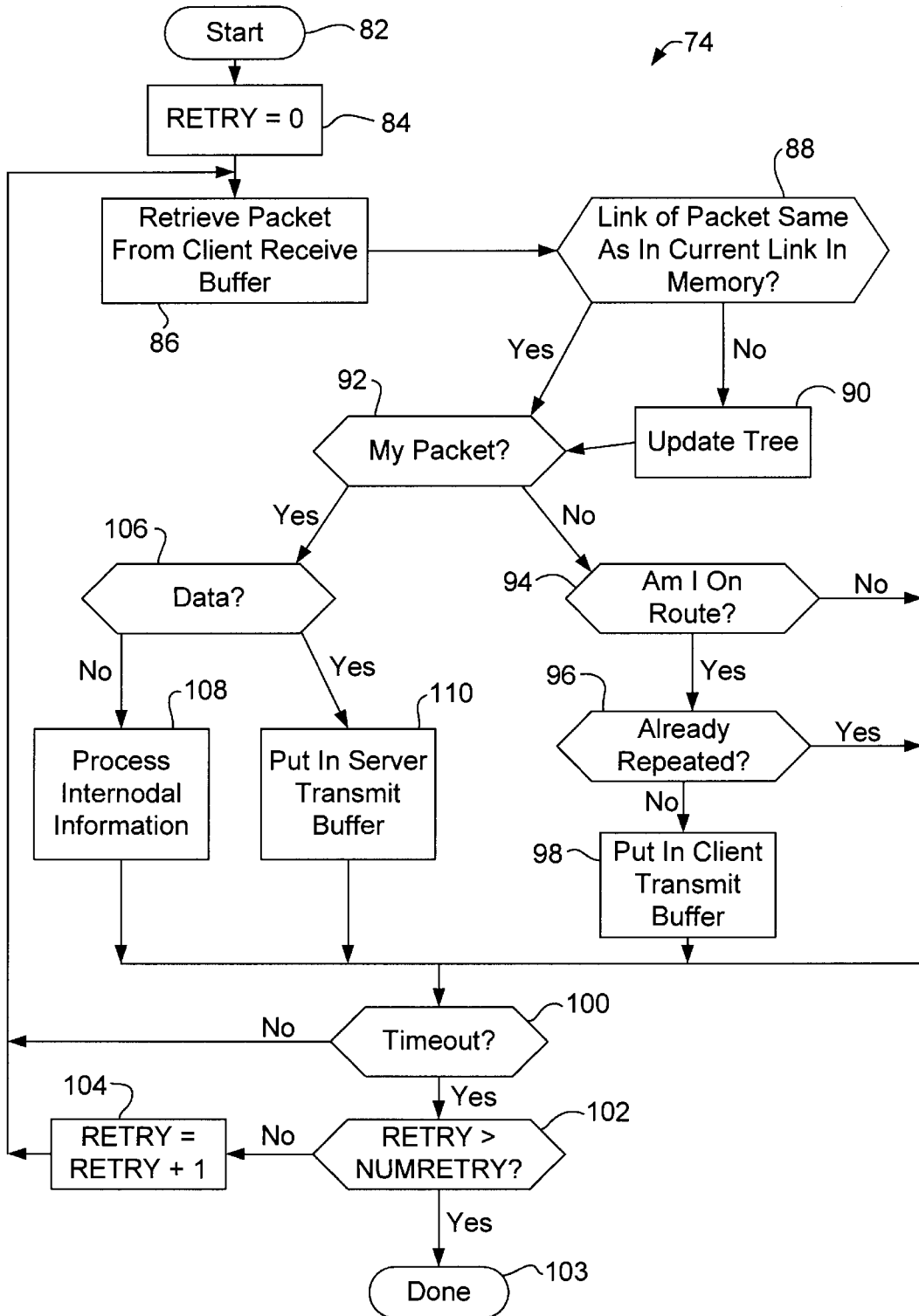


Fig. 5

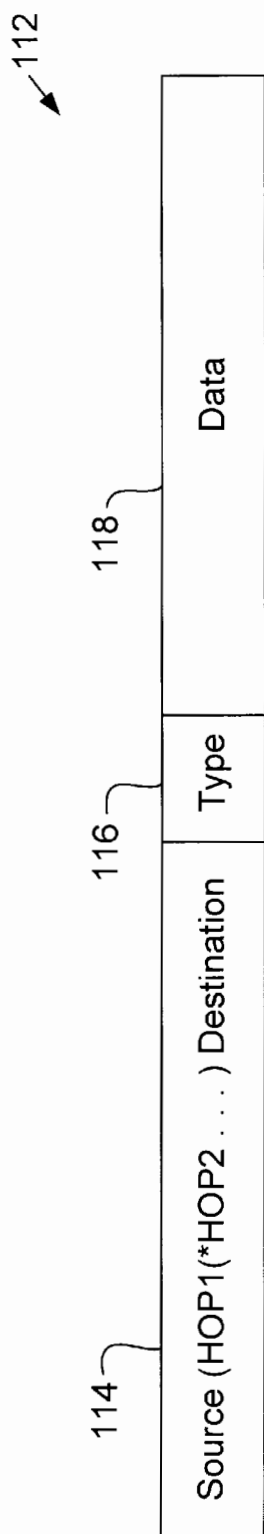


Fig. 5a

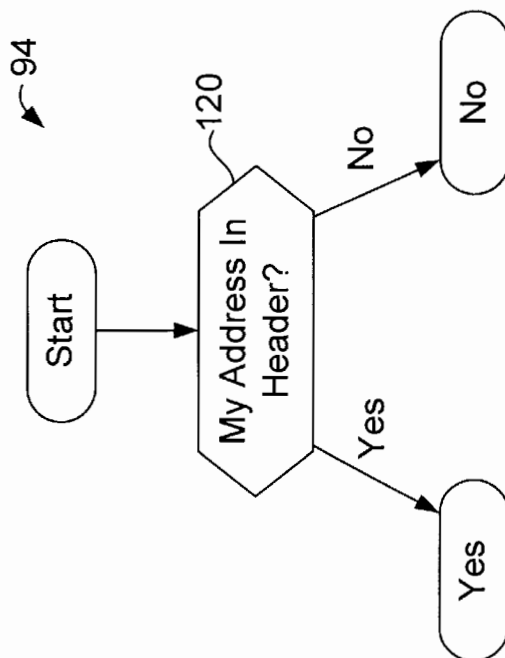


Fig. 5b

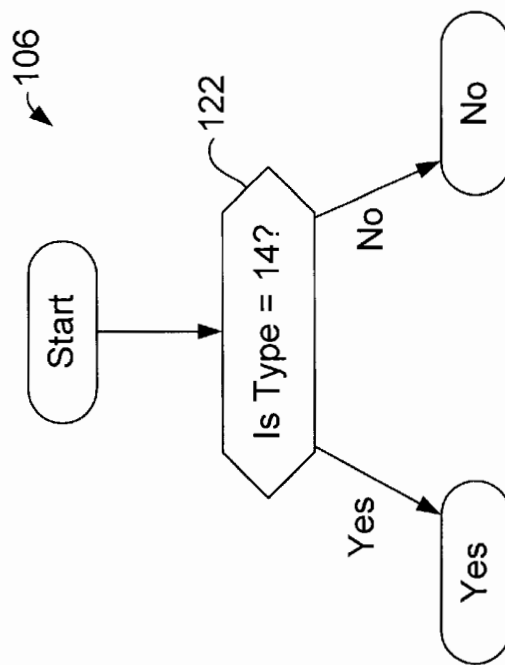


Fig. 5c

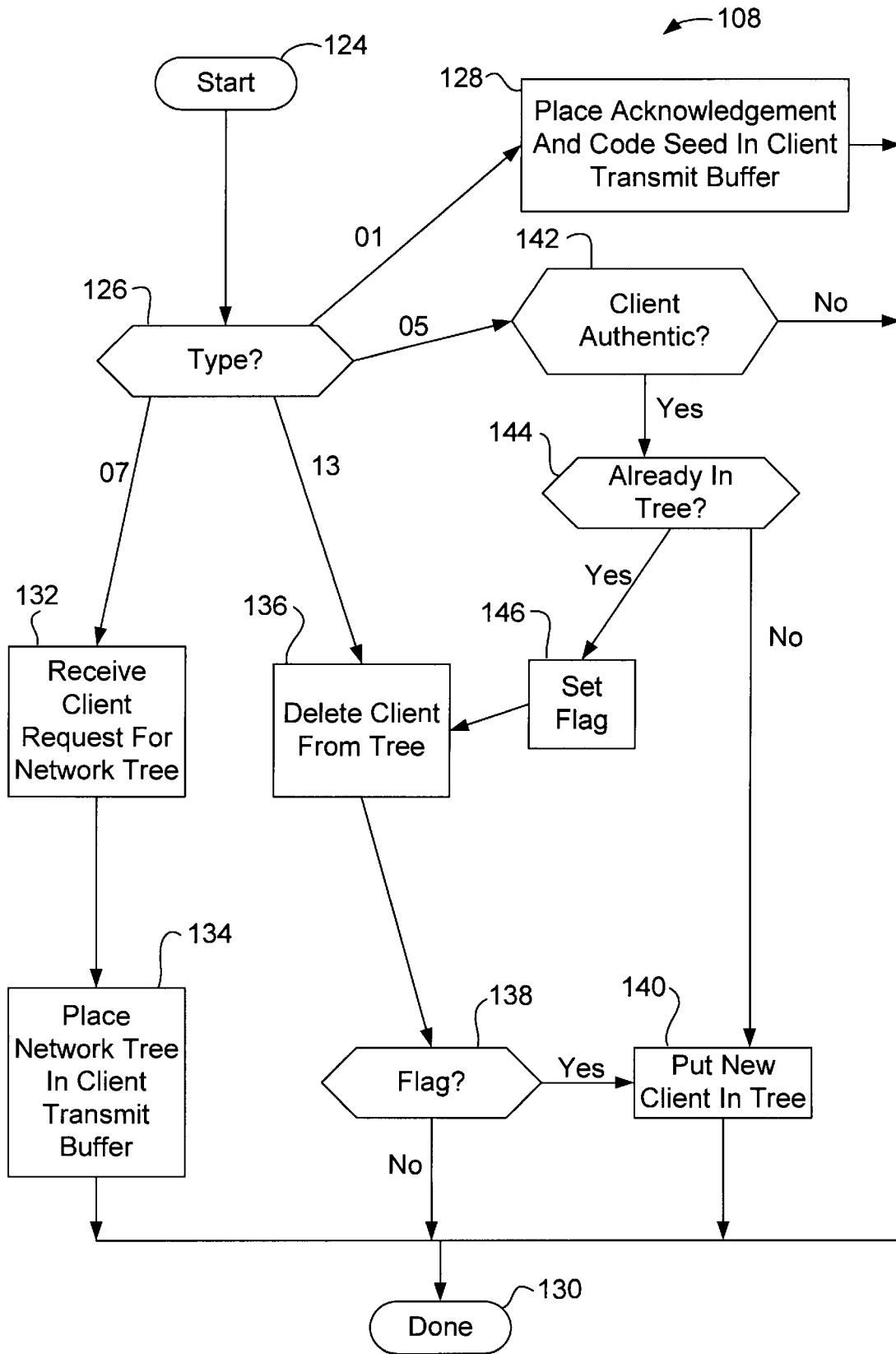


Fig. 6

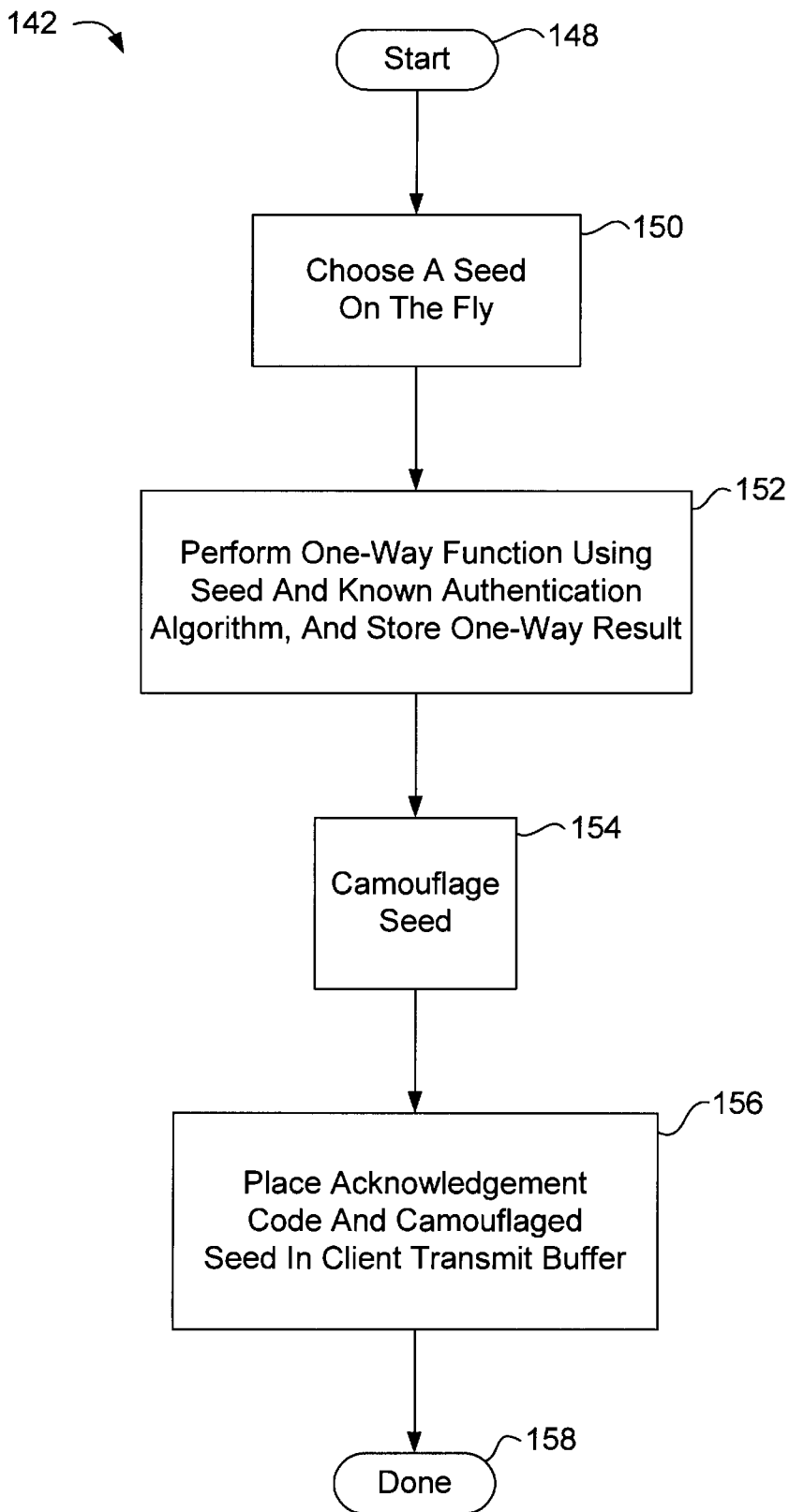


Fig. 6a

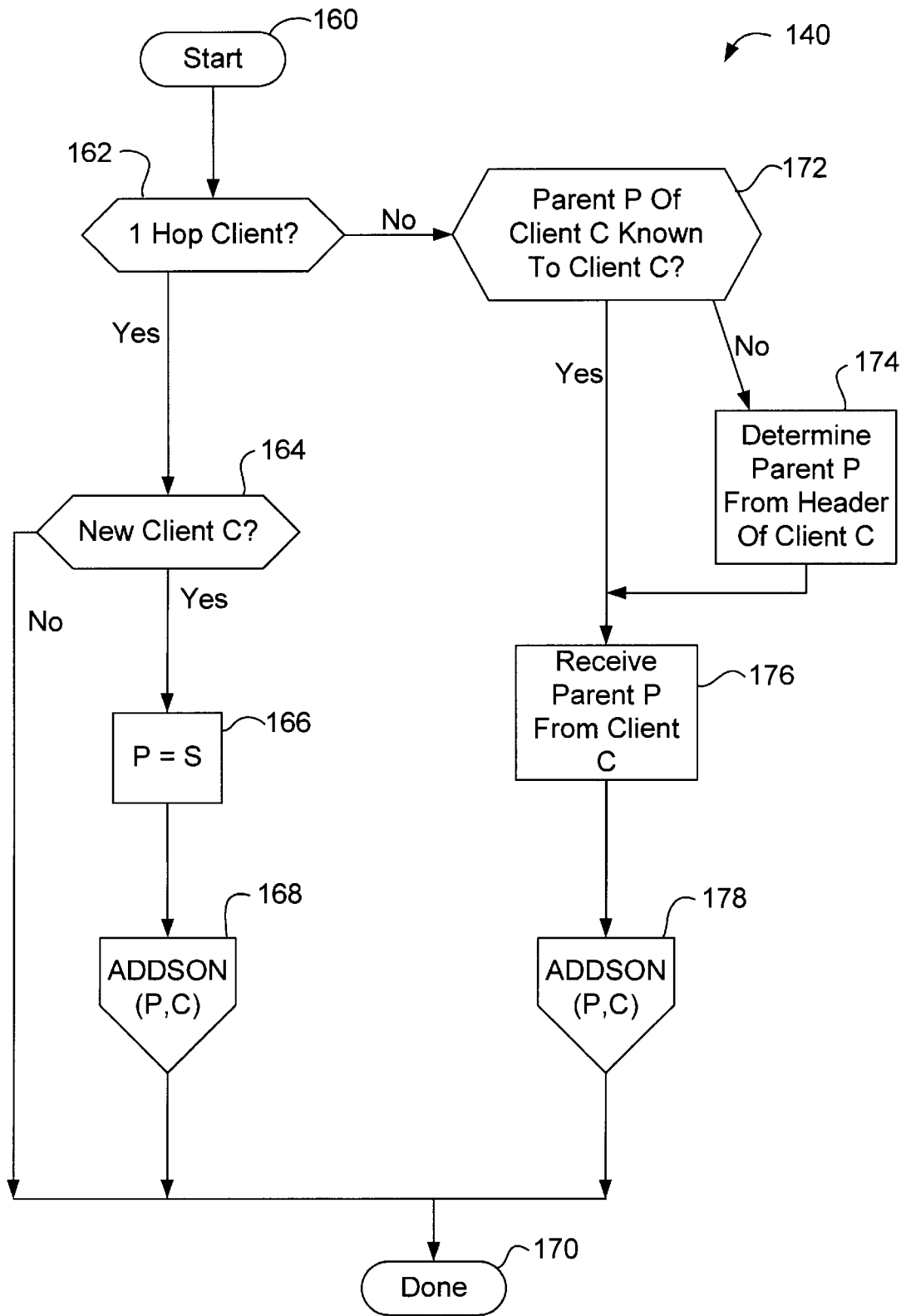


Fig. 6b

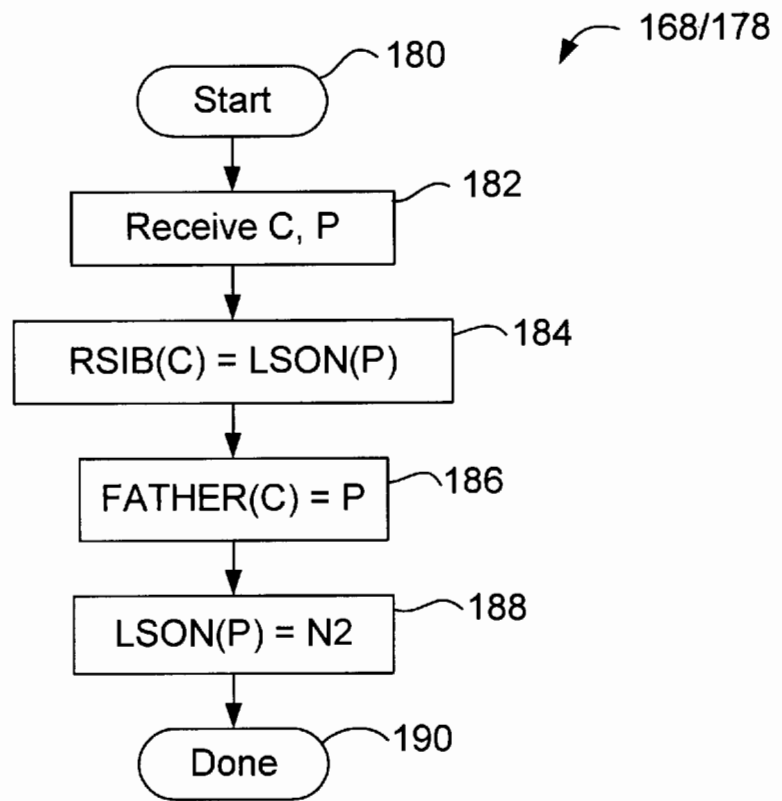


Fig. 7

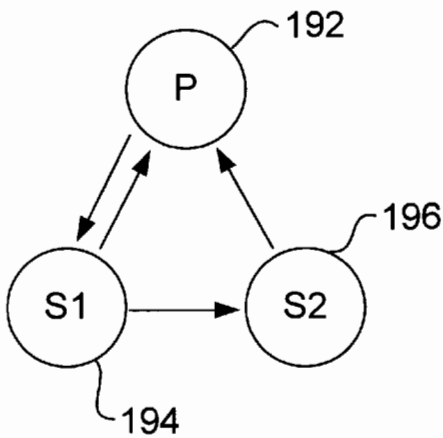


Fig. 7a

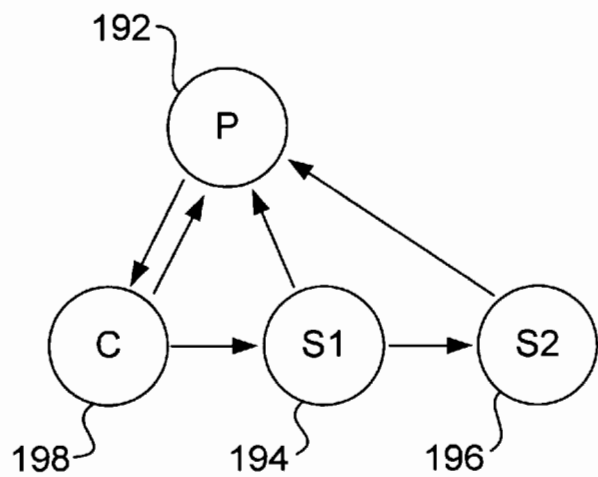


Fig. 7b

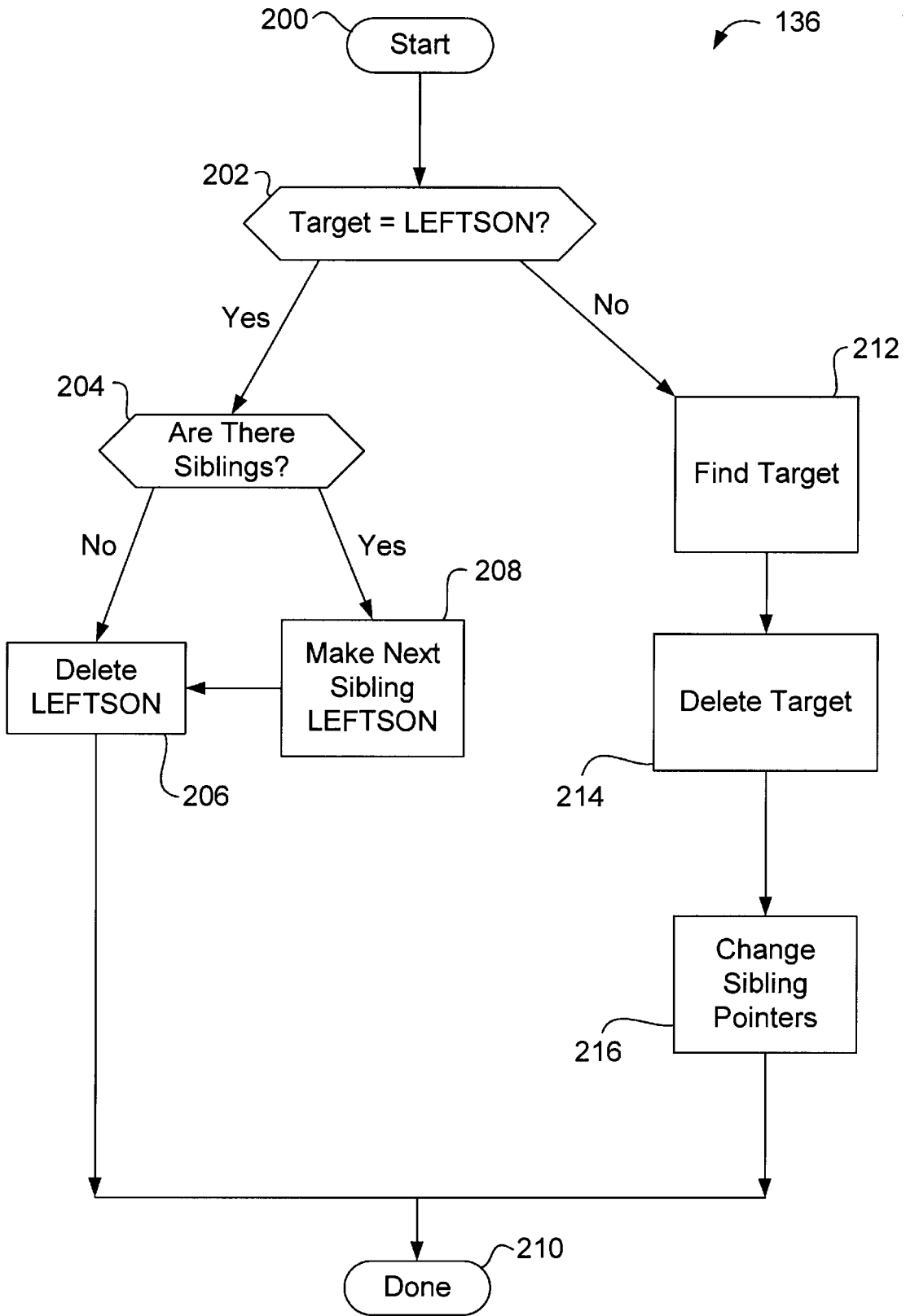


Fig. 8

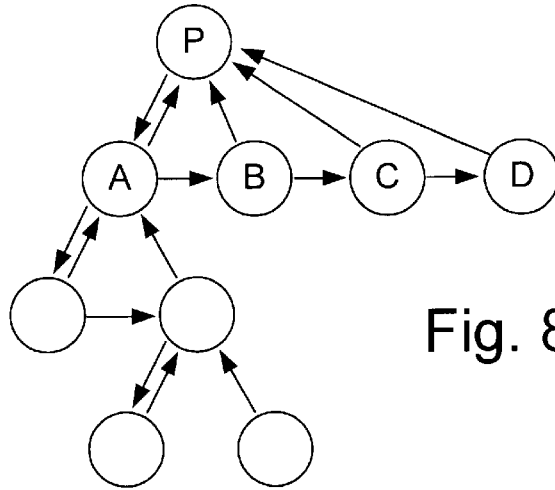


Fig. 8a

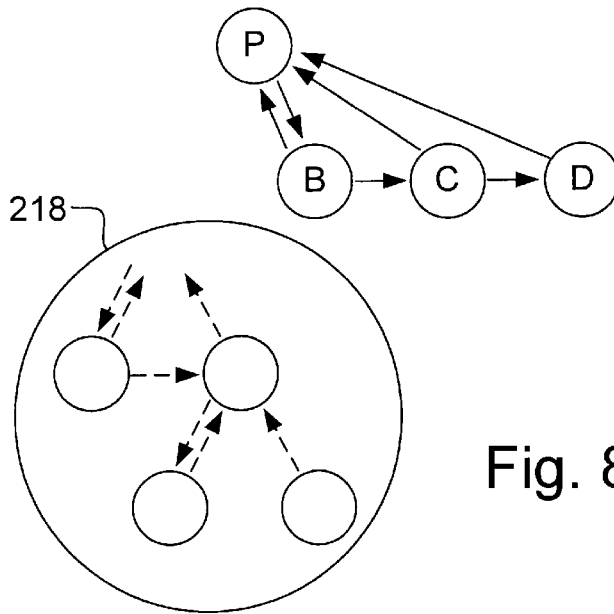


Fig. 8b

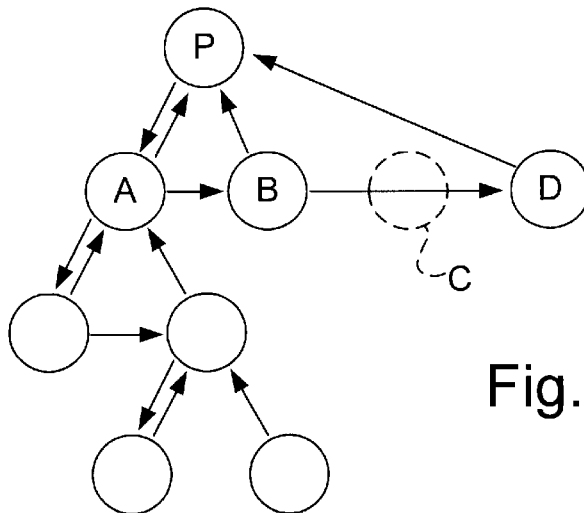


Fig. 8c

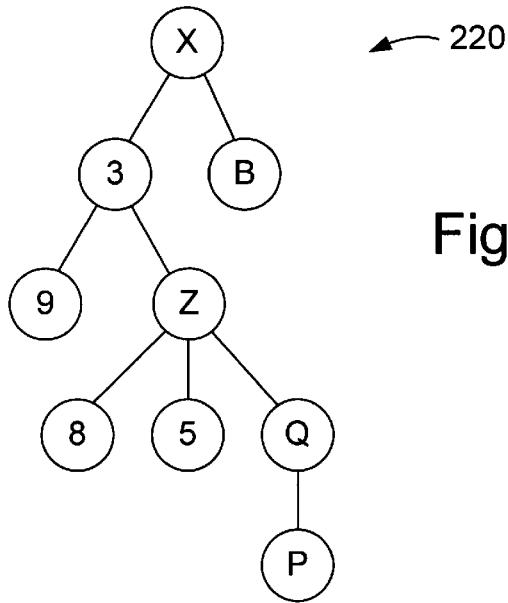


Fig. 9a

X(3(9,Z(8,5,Q(P))),B) ← 222

Fig. 9b

Element	Node Name	Time Stamp	Memory Location Of Node
1	3	1AFG	12FG3
2	P	E013	9AA22
3	X	99F6	.
4	5	B999	.
5	8	B999	.
6	Q	.	.
7	9	.	.
8	B	.	.
9	Z	.	.
.	.	.	.
.	.	.	.
.	.	.	.

Fig. 9c

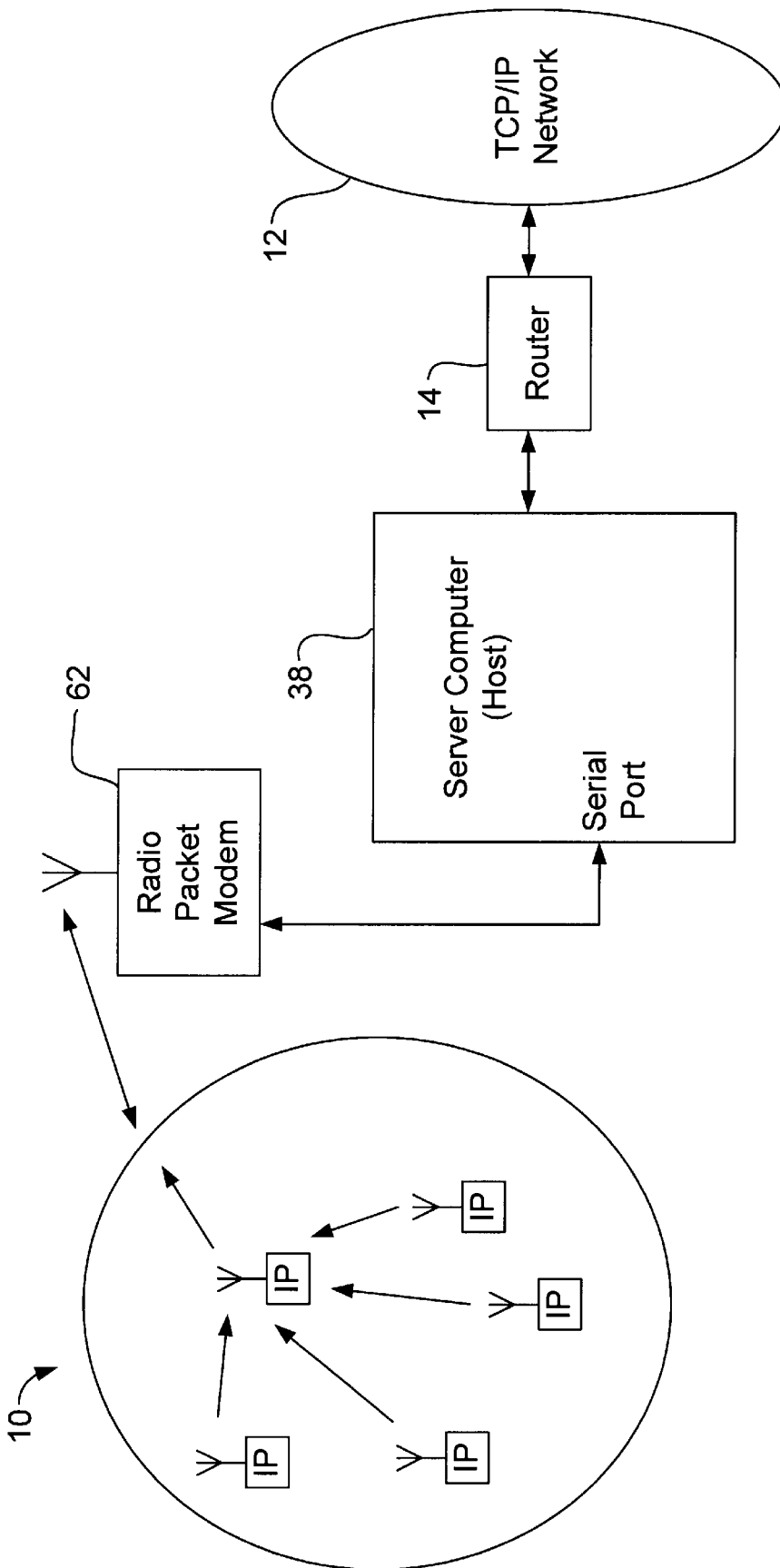


Fig. 10

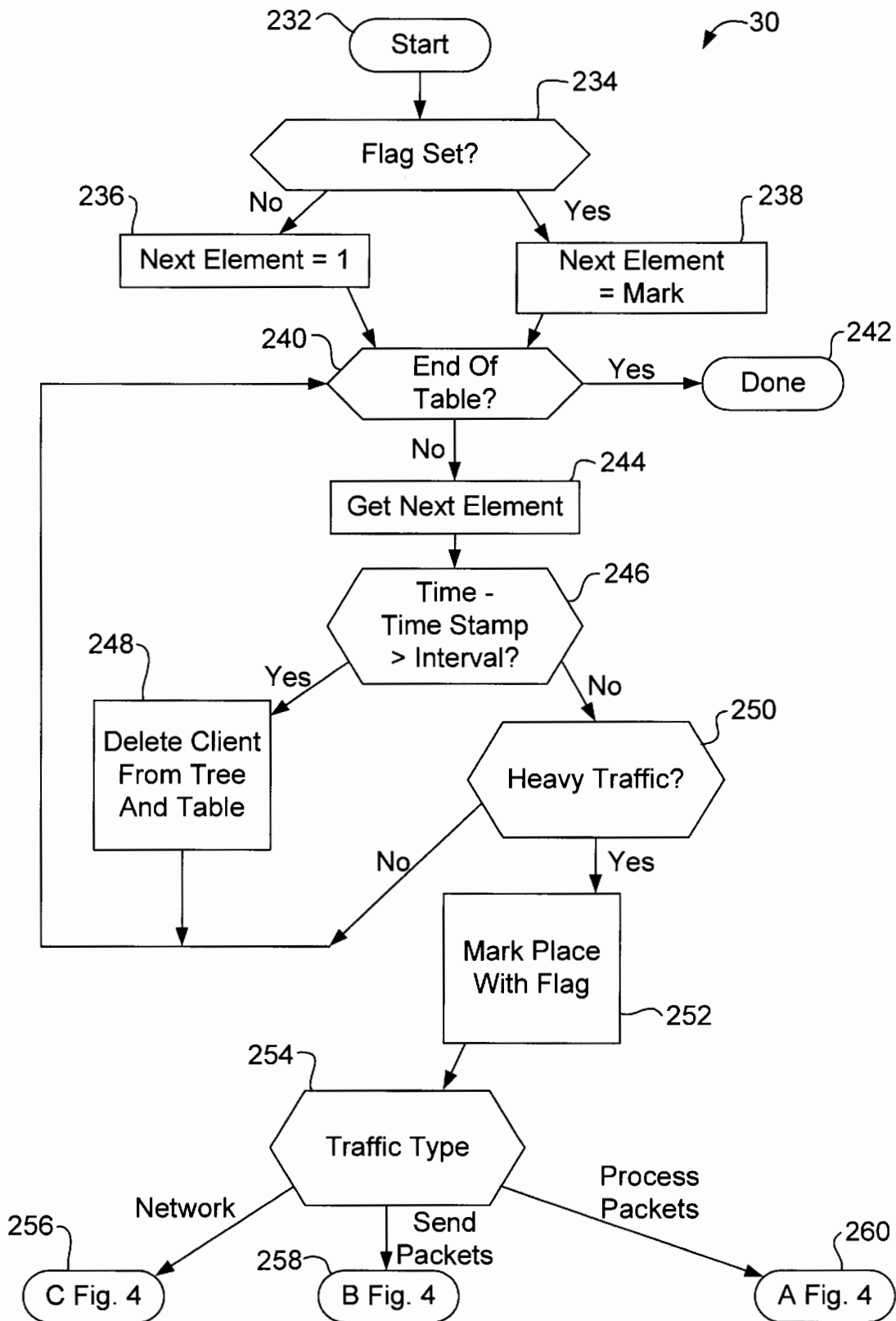


Fig. 11

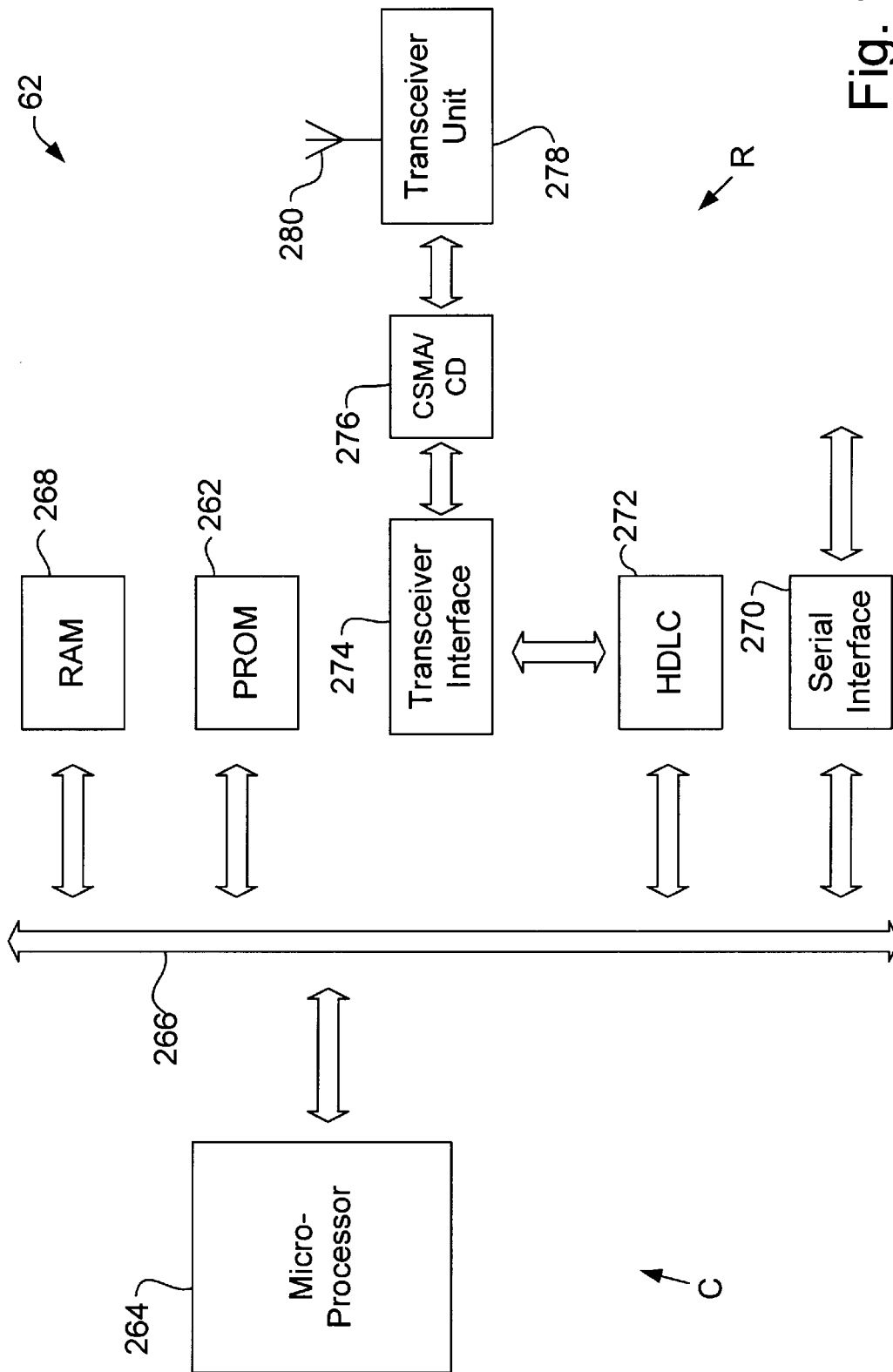


Fig. 12

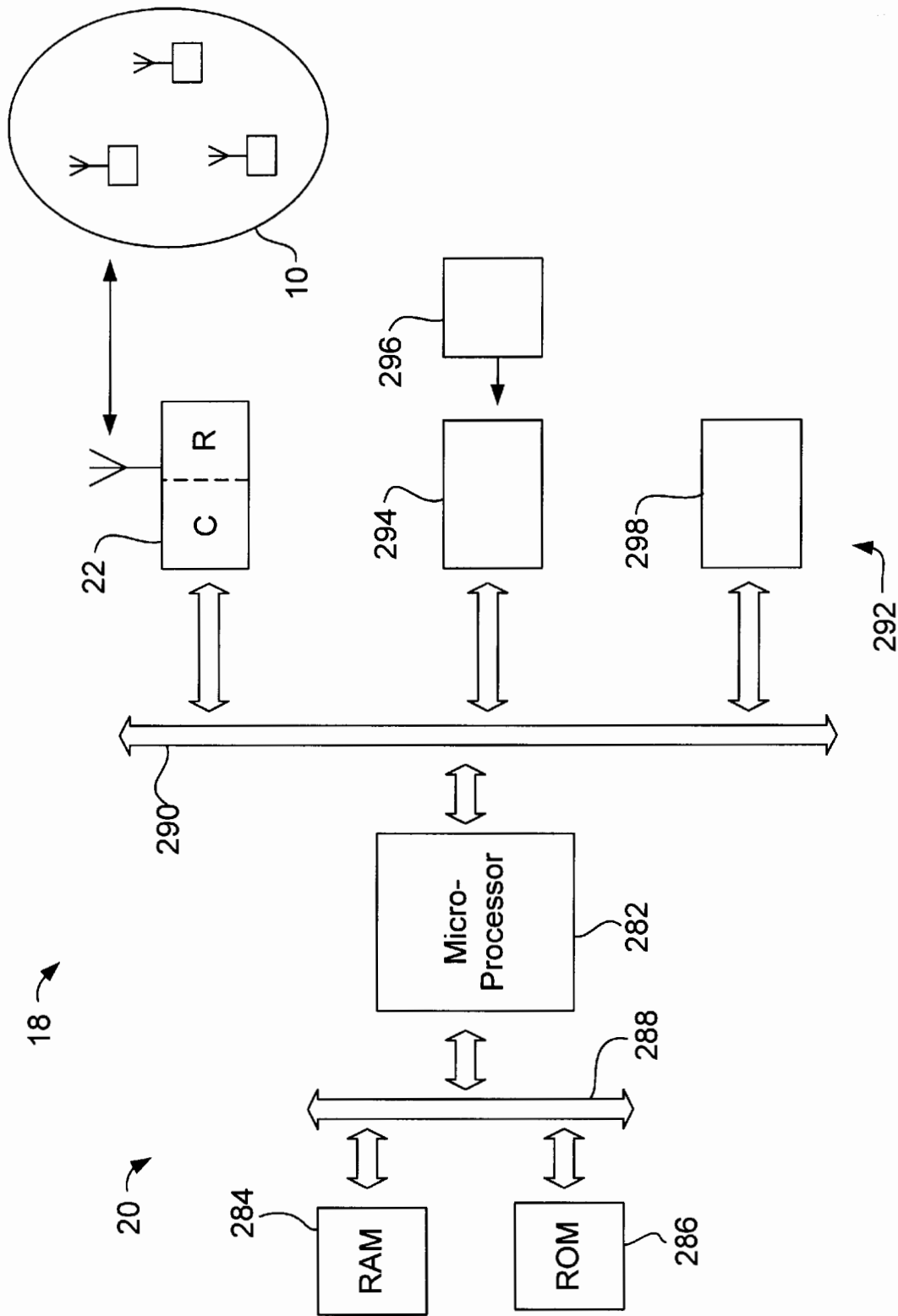


Fig. 13

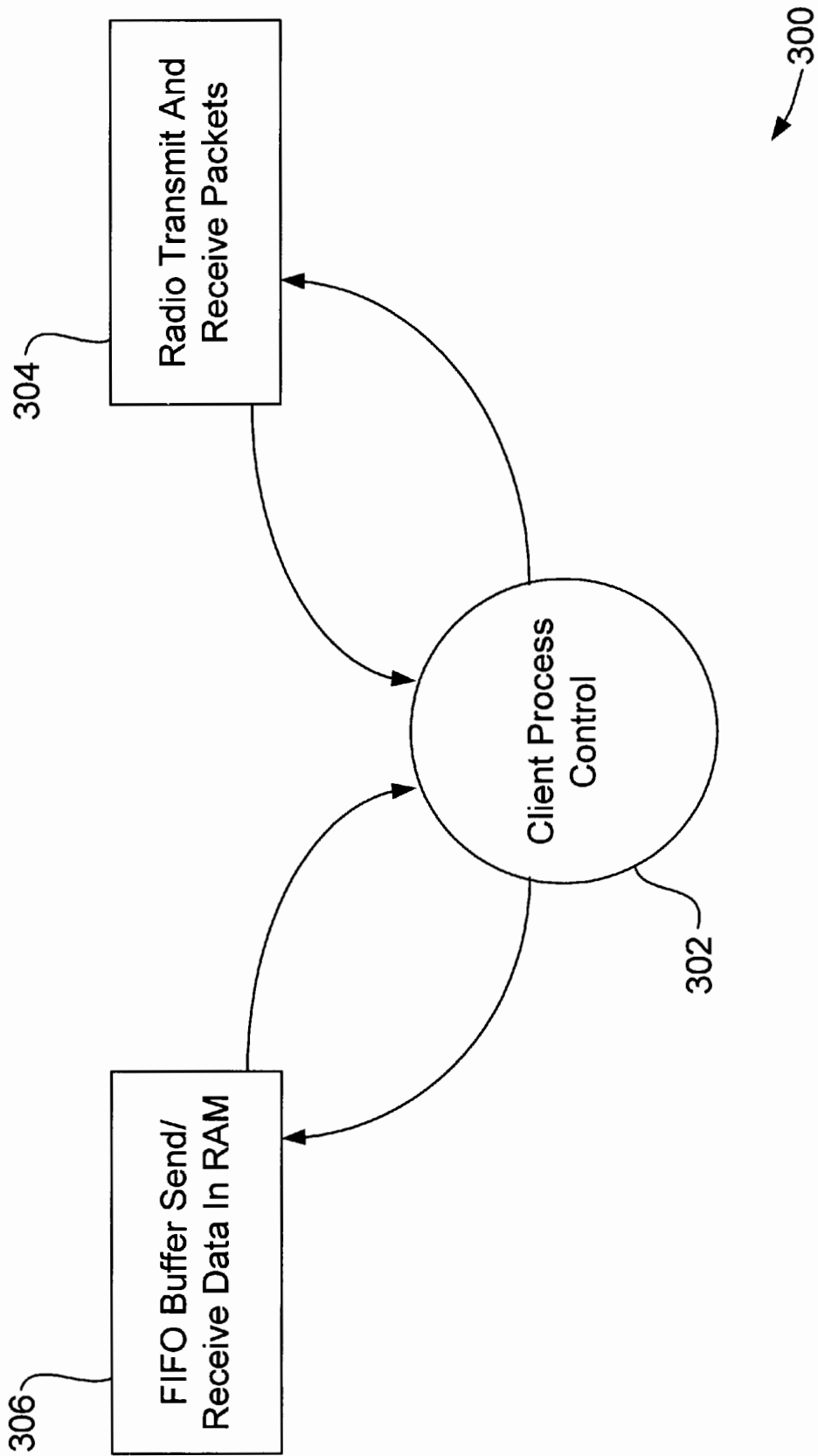


Fig. 14

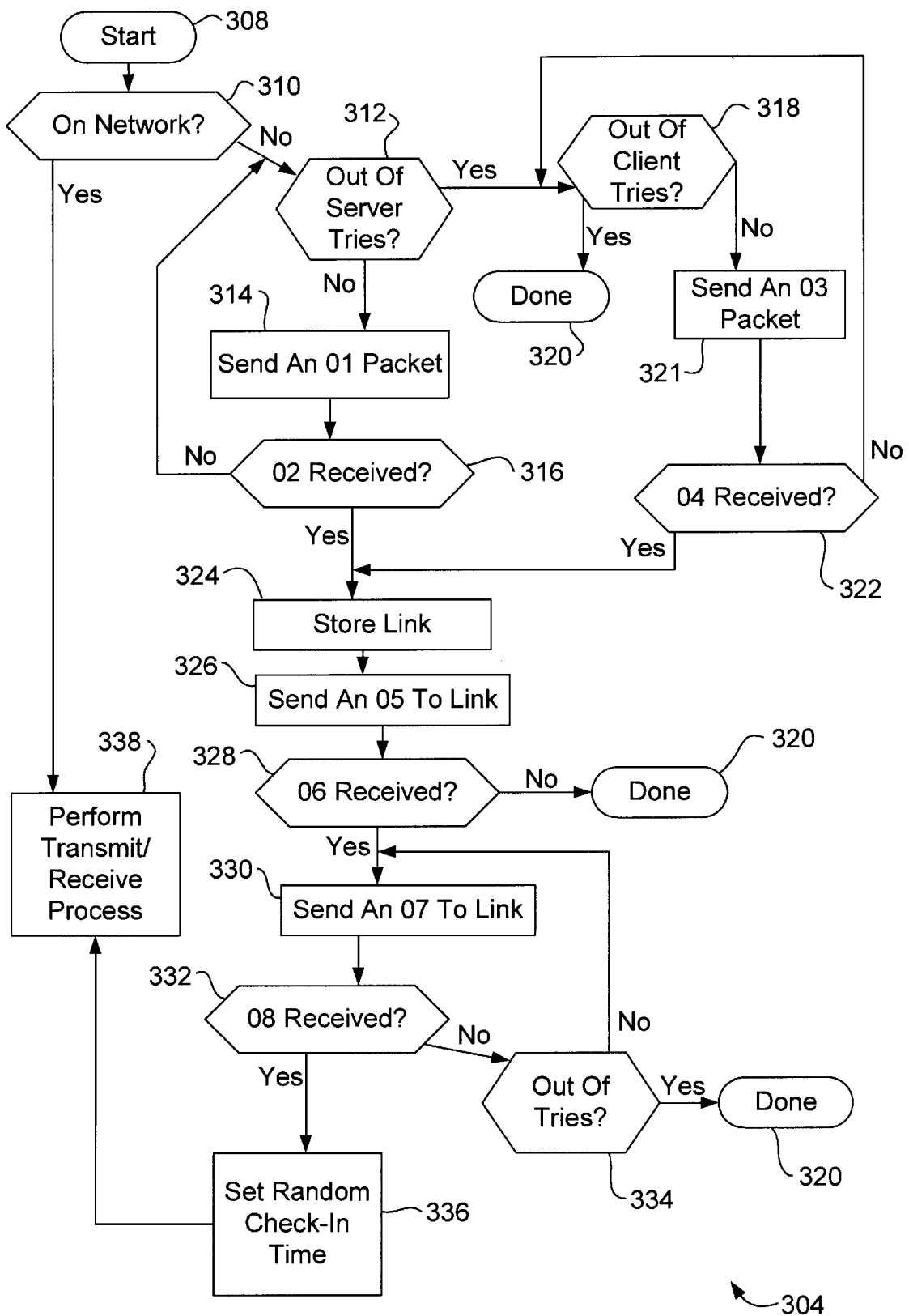


Fig. 15

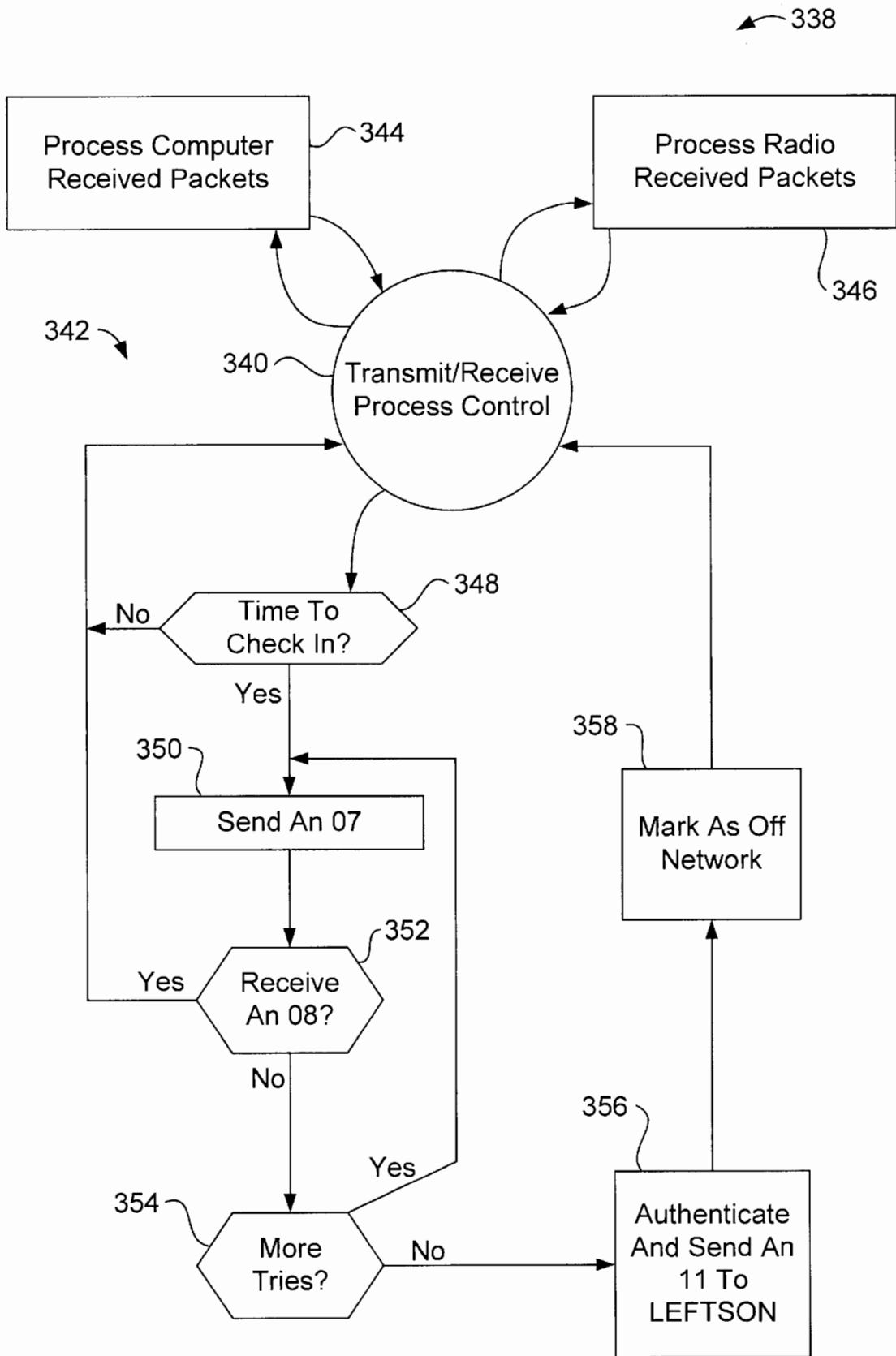


Fig. 16

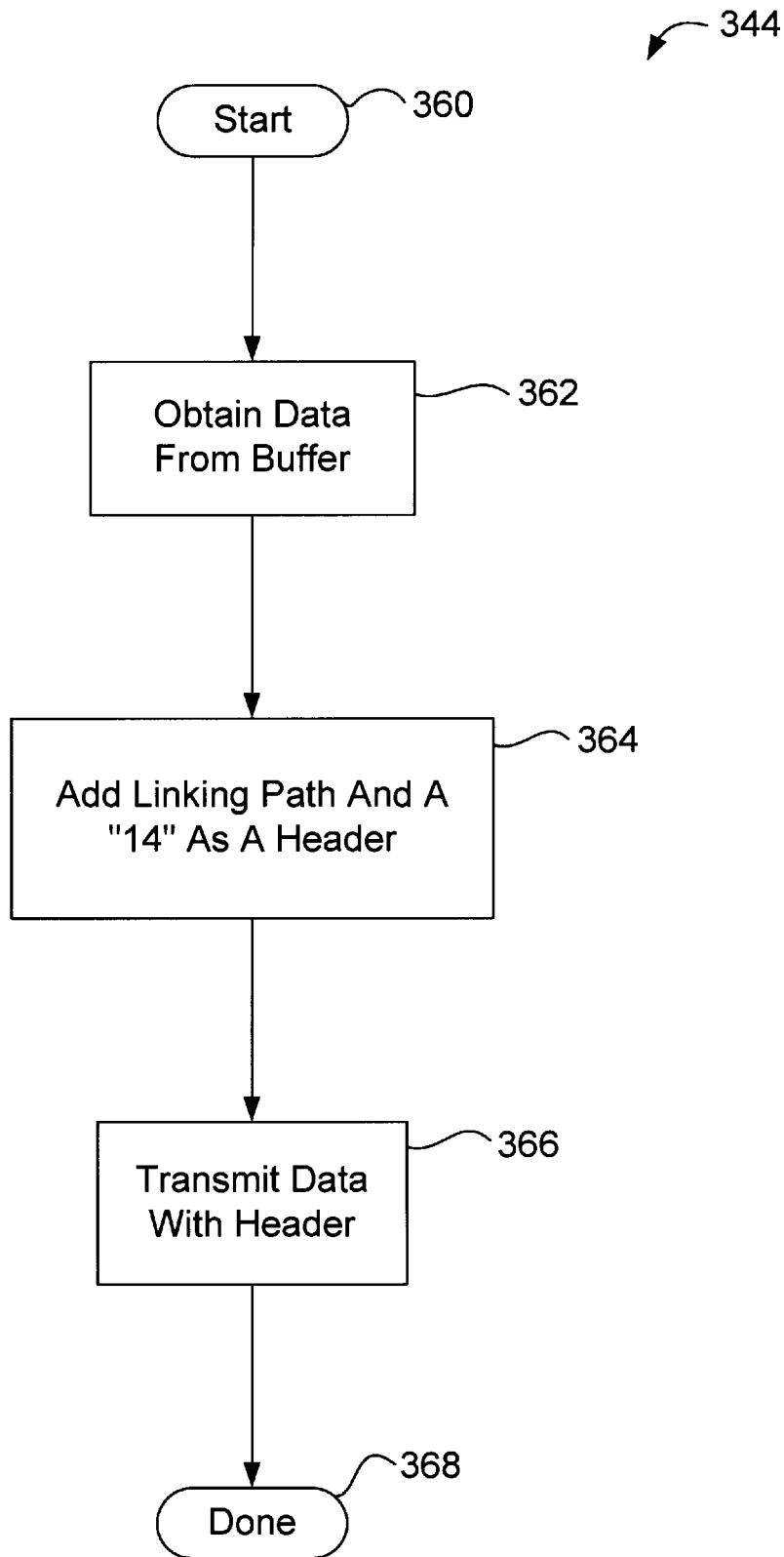


Fig. 17

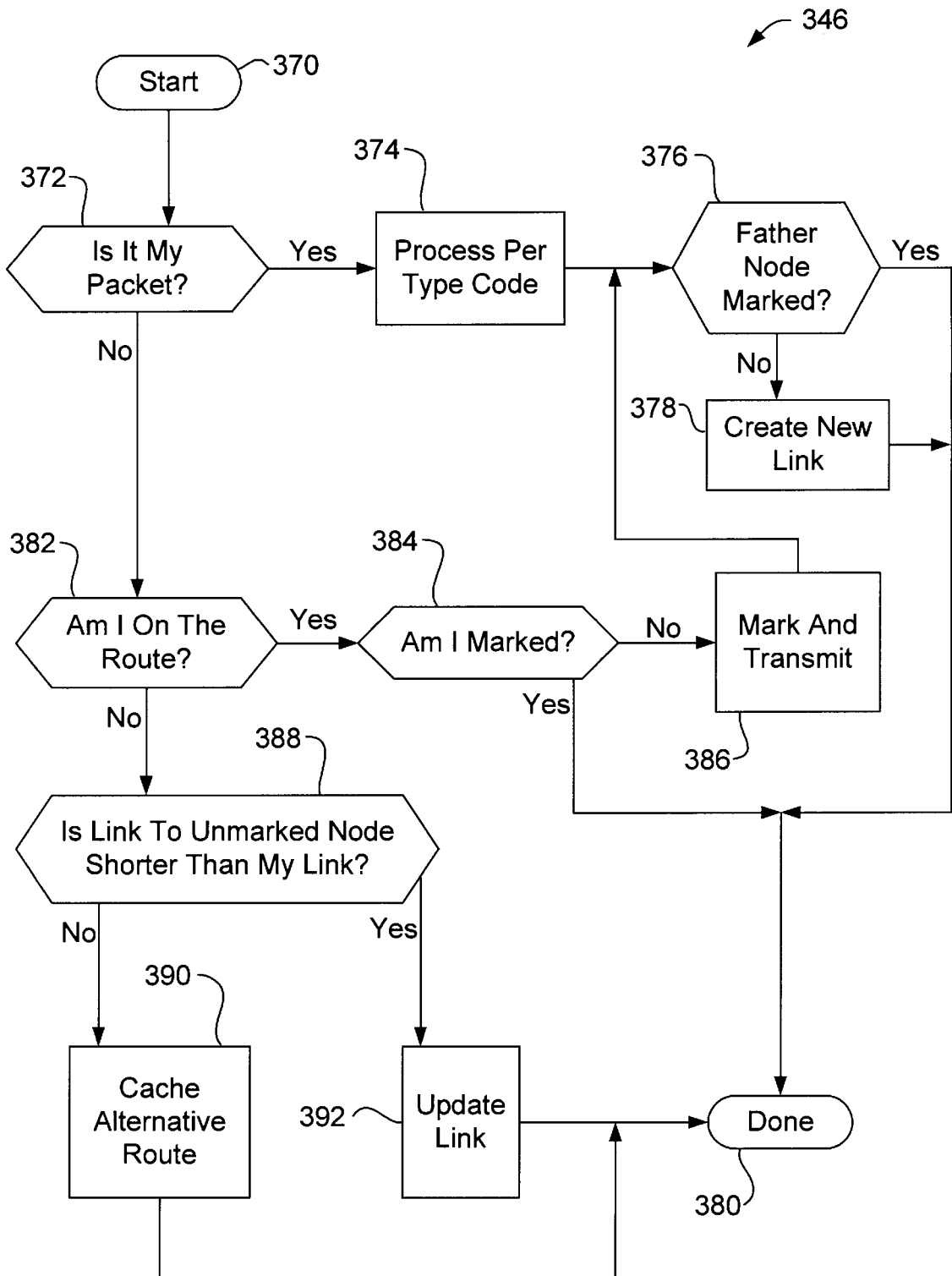


Fig. 18

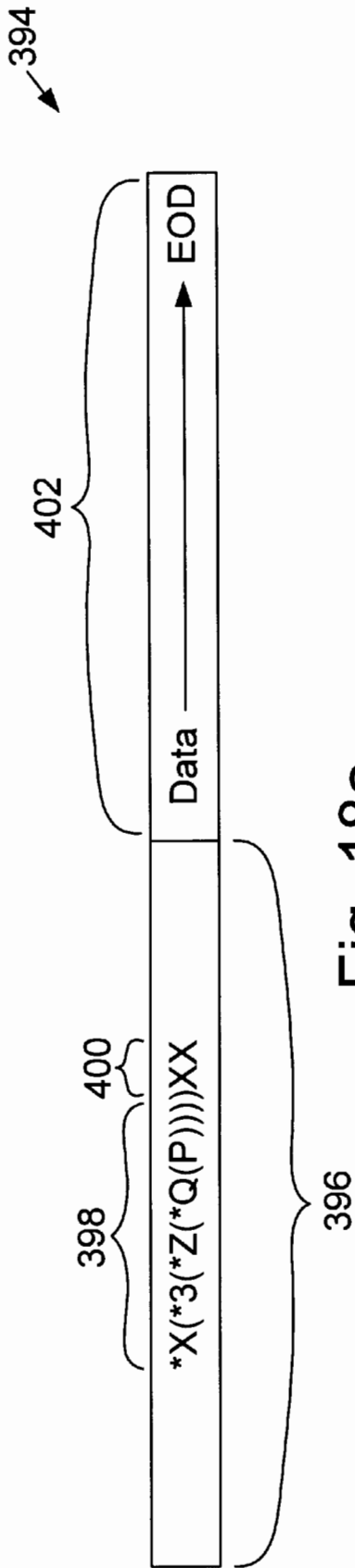


Fig. 18a

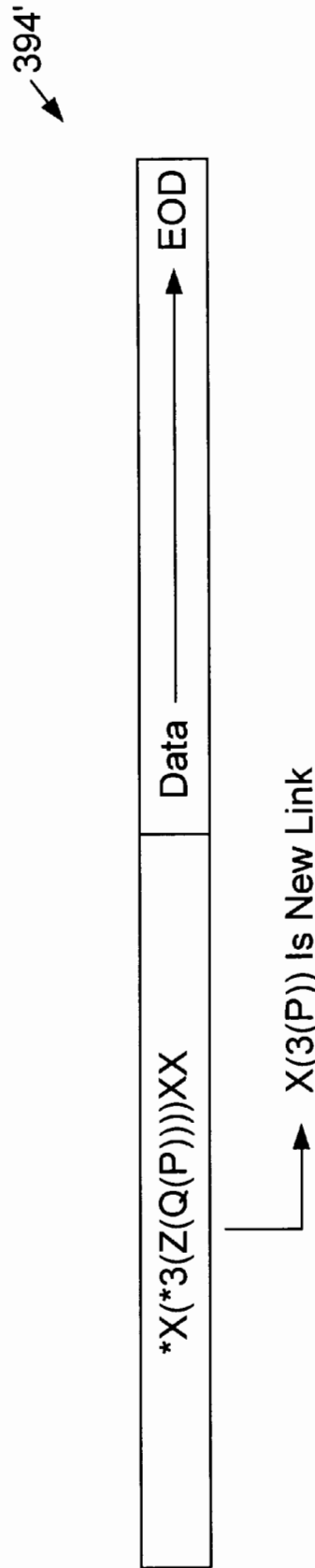


Fig. 18b

U.S. Patent

Mar. 28, 2000

Sheet 40 of 42

6,044,062

404 ↓	406 ↓	408 ↓
Code Received	Server Response	Client Response
01	02 + One-Way Seed	Drop
02	Drop	05 + One-Way Response
03	Drop	04 + Seed Or Null
04	Drop	05 + One-Way Seed
05	If 02 And Authentic Then 06 Else Drop	If 04 And Authentic Then 06 Else Drop
06	Drop	If 05 Then 07 Else Drop
07	08	Drop
08	Drop	Update Tree Or Repeat Data
09	Drop	10
10	Drop	Update Tree Or Repeat Data
11	Drop	Send 11 To LEFTSON With Address of Departer Plus 01 to Reconnect
12	Reserved	Reserved
13	Delete Sender	Drops
14	Send To Network Transmit Buffer	Send To Computer Transmit Buffer
86	Reserved	Reserved

Fig. 19

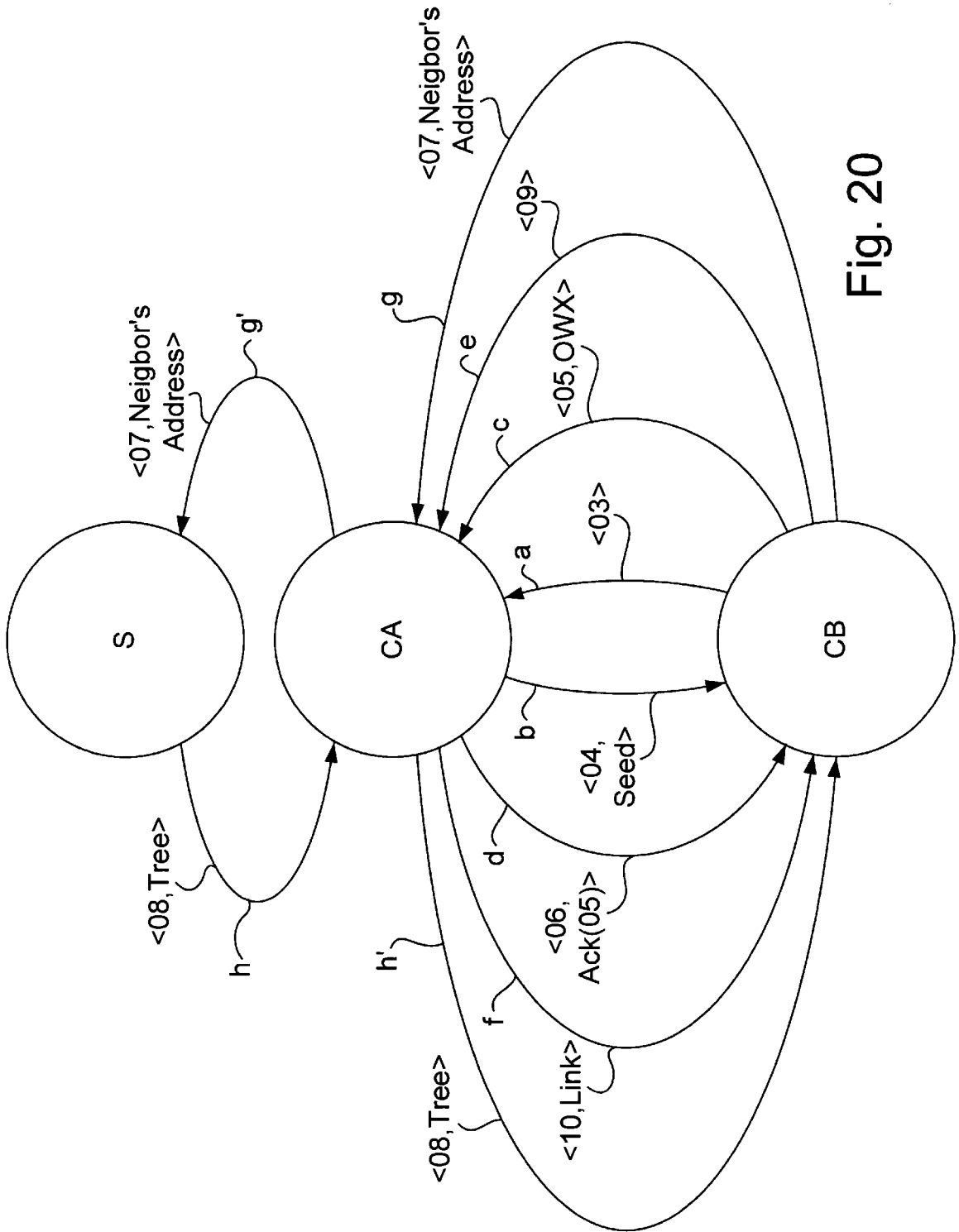


Fig. 20

$X(3(9,Z(8,5,Q(P))),B)$

Fig. 21a

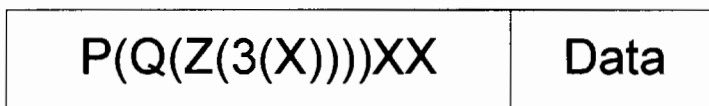
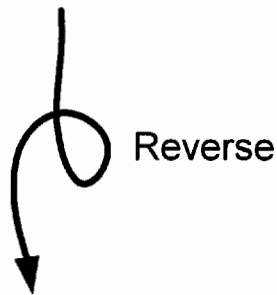


Fig. 21b



$X(3(Z(Q(P))))$

Fig. 21c

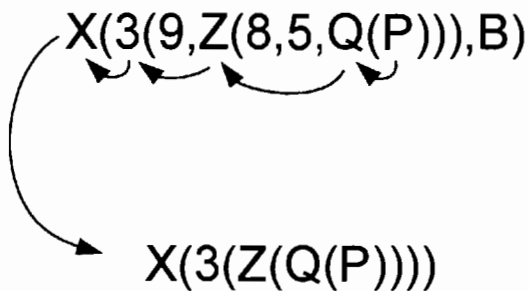


Fig. 21d

6,044,062

1

WIRELESS NETWORK SYSTEM AND METHOD FOR PROVIDING SAME

TECHNICAL FIELD

This invention relates generally to digital computer networks, and more particularly to wireless digital networks for the transmission of digital data packets.

BACKGROUND ART

There are many kinds of networks that can be used to couple computers together for data communication. For example, a simple local area network (LAN), such as a Novell® network or an Appleshare® network, can be used to couple together the personal computers in an office. Often, one or more network “servers” or “hosts” will influence data flow within the network and access to certain network functions such as a central file repository, printer functions, Internet gateways, etc. Other local area networks operate on a peer-to-peer basis without the use of servers.

A wide area network (WAN) is sometimes referred to as a “network of networks.” The Internet is a WAN that has, of late, become extremely popular. The origins of the Internet date back several decades to a government-sponsored military/business/research WAN that was designed to remain operational even in the event of a catastrophic loss of a large portion of the network. To accomplish this goal, robust protocols and systems were developed which allowed a geographically distributed collection of computer systems to be connected by means of a network that would remain operational even if large portions of the network were destroyed.

While the use of the Internet has been prevalent for many years now, its use has been limited by the arcane and often difficult commands required to access the various resources of the network. To address this problem, a protocol known as the “World Wide Web” or “WWW” was developed to provide an easier and more user-friendly interface to the Internet. With the World Wide Web, an entity having a domain name creates a “web page” or simply “page” which can provide information and, to an ever greater extent, some interactivity with the web page.

The Internet is based upon a transmission protocol known as “Transmission Control Protocol/Internet Protocol” (or “TCP/IP” for short), which sends packets of data between a host machine, e.g. a server computer on the Internet, and a client machine, e.g. a user’s personal computer connected to the Internet. The WWW is an Internet interface protocol which is supported by the same TCP/IP transmission protocol. Intranets are private networks based on Internet standards, and have become quite common for managing information and communication within an organization. Intranets, since they subscribe to Internet standards, can use the same web browser and web server software as used on the Internet. Intranets are, in many cases, supplementing or replacing traditional local area network protocols.

Most, if not all, of the data communication links between the various machines of most networks are hard-wired. That is, client machines are typically coupled to a server and to other client machines by wires (such as twisted-pair wires), coaxial cables, fiber optic cables, and the like. In some instances, some of the communication links can be wireless communication links, such as microwave links, radio frequency (r.f.) links, infrared links, etc., but this tends to be rare with most LANs.

The majority of so-called wireless networks use radio modems for data communication, although there are some

2

IR networks available that work over very short distances, such as within a single large room. However, networks spanning larger areas will predominately use radio modems. GRE America, Inc. of Belmont, Calif. sells a number of spread-spectrum modems that can be used for the transmission of digitally encoded information. A number of wireless network services, such as Ricochet® network services (Ricochet is a subsidiary of Metcocom, Inc. of Los Gatos, Calif.) combine a radio modem with a portable personal computer to allow the personal computer to connect to the Internet. The Ricochet system operates by providing a large number of r.f. data transceivers within a given geographic area, that are often attached to telephone poles, and that are coupled to centralized server that serves as a gateway to the Internet.

The assumption made by the Ricochet system designers is that a given radio modem coupled to portable computer will be in radio contact with one, and only one, transceiver of the network. A data “packet” sent by the portable computer via the radio modem will be received by the transceiver and broadcast through the Ricochet network until it reaches a Wide Area Processor or WAP, where it is transmitted by twisted pair over the Internet to a Ricochet server connected to the Internet. Packets destined for a particular personal computer are received by the server of the Ricochet system, and are transmitted from each of the transceivers with the expectation that the radio modem of the destination portable computer will receive the data packets from one of those transceivers.

It should be noted that wireless communication systems such as the Ricochet system exhibit a number of drawbacks. For one, if the radio modem of the personal computer is not within transmission range of one of the transceivers of the Ricochet network, a connection cannot be made to the network. Furthermore, the Ricochet network can create a great deal of “packet duplication” or “pollution” as copies of a particular data packet are multiply repeated, rather than routed. This packet duplication can also occur if a radio modem of a particular personal computer in radio transmission range of two or more transceivers of the Ricochet network. In such an instance, the two or more transceivers can each receive the data packets, and each proliferates copies of the data packet across the Ricochet network. While duplicate packets are ultimately discarded, such duplicate packets increase data congestion in the network and increases the work that must be performed by the server. In addition, since data packets are transmitted from all the transceivers of the Ricochet network, there may be packet duplication at the personal computer if it is in contact with more than one transceiver of the Ricochet network, and the bandwidth available from each transceiver is reduced since each transceiver is transceiving each client-destined data packet on the network. Also, since the data is transmitted to the Internet over twisted pair, there is a 28.8K baud bottleneck in the system, resulting in average system performance of even less than 28.8K baud. It is therefore apparent that prior art wireless networks of the Ricochet network type lack robustness (i.e. the ability to maintain communication with the network under adverse conditions) and exhibit a number of inefficiencies such as data packet proliferation.

Cellular telephone systems operate using a number of transceivers, where each transceiver occupies a “cell.” As a mobile telephone moves from one cell to another, an elaborate and expensive land-based system causes the mobile telephone to be “handed-off” from the cell that it was previously in to the cell that it is entering. As noted, the equipment and system used for the hand-off is expensive

6,044,062

3

and, further, such hand-off sometimes fail, dropping the telephone connection. Furthermore, individual radios at a given cell can handle only one call at a time, which is inadequate for many computer network systems.

Amateur radio (“Ham”) operators have developed a peer-to-peer digital repeater system referred to as the AX.25 protocol. With this protocol, each peer repeats all data packets that it receives, resulting in rapid packet proliferation. In fact, with this protocol, so many packet collisions occur among the peers that the packets may never reach the intended peer.

Lastly, there is abundant reporting in the literature, but it cannot be substantiated, that the U.S. Military has a wireless communication system which allows digital information to be transmitted in a more robust and efficient matter. More specifically, it is suspected that the U.S. Military has a system in which digital data can follow multiple paths to a server that may include one or more clients of the network. However, source code listings, or source code in machine-readable form for these U.S. military systems remains secret and unavailable to the public. Some of the literature pertaining to this U.S. military technology is summarized below.

“Packet Radios Provide Link for Distributed Survivable Command Control Communications in Post-Attack Scenarios”, M. Frankel, *Microwave Systems News* 13:6 (June 1983), pp. 80–108, discusses the SURAN (Survivable Radio Network) project and its relation to overall command and control communications (C³) development.

“Control Using Pacing in a Packet Radio Network”, N. Goweer and J. Jubin, *Proceedings of Milcom 82*, (New York: IEEE Press, 1982), pp. 23.1–23.6, describes a technique for pacing flow control used in the DARPA packet radio project.

“Current Packet Radio Network Protocols”, J. Jubin, *Proceedings of Infocom 85* (New York: IEEE Press, 1985), pp. 86–92, is a systematic review of the various protocols currently used in the DARPA packet radio network. The article includes a discussion of pacing, route calculation, maintenance of route and connectivity tables, acknowledgment schemes, and other mechanisms. The article also provides a discussion on how the various protocols interrelate and reinforce each other.

“The Organization of Computer Resources into a Packet Radio Network”, R. Kahn, *IEEE Transactions on Communications* COM-25:1 (January 1977), pp. 169–178, is a prospectus for the second generation of the DARPA radio project. This led to the development of the DARPA Bay Area Packet Radio experimental work in the mid to late 1970’s.

“Advances in Packet Radio Technology”, R. Kahn, S. Gronemeyer, J. Burchfiel, R. Kunzelman, *Proceedings of the IEEE* 66z:11 (November 1978), pp. 1468–1496 is a survey of packet radio technology in the second generation of the DARPA packet radio project.

“Survivable Protocols for Large Scale Packet Radio Networks”, G. Lauer, J. Westcott, J. Jubin, J. Tornow, *IEEE Global Telecommunications Conference*, 1984, held in Atlanta, Ga., November 1984 (New York: IEEE Press, 1984) p. 468–471, describes the SURAN network, with an emphasis on network organizations and management protocols.

“Multiple Control Stations in Packet Radio Networks”, W. MacGregor, J. Westcott, M. Beeler, *Proceedings of Milcom 82* (New York: IEEE Press, 1982) pp. 10.3–10.3-5, is a transitional paper that describes design considerations involved in converting the DARPA packet radio network from single to multistation operation while eliminating the

4

additional step to a fully hierarchical design. It focuses on the self-organizing techniques that are necessary in the multistation environment.

“Future Directions in Packet Radio Technology”, N. Shacham, J. Turnow, *Proceedings of IEEE Infocom 85* (New York: IEEE Press, 1985), pp. 93–98, discusses new research areas in packet radio, with some references to SURAN developments.

“Issues in Distributed Routing for Mobile Packet Radio Networks”, J. Westcott, *IEEE Global Telecommunications Conference*, 1982 (New York: IEEE Press, 1982), pp. 233–238, studies the issues involved in the DARPA packet radio network, prior to the availability of signal strength sensing from the radio receivers as a hardware capability on which to build. The paper describes issues that must be considered in evaluating the usability of an RF link and gives details of the alternate route mechanism used in the DARPA system to smooth temporary RF propagation problems that appear in a mobile node environment.

“A Distributed Routing Design for a Broadcast Environment”, J. Westcott, J. Jubin, *Proceedings of Milcom 82* (New York: IEEE Press, 1982), pp. 10.4-1–10.4-5, is a detailed study of the problems involved in connectivity and routing table management in stationless packet radio, including a discussion of algorithms proposed for the DARPA packet radio network.

There is, therefore, a great deal of literature describing packet radio systems. The prior art does not disclose, however, a packet-based wireless computer network that is both robust and efficient, wherein each client of the network can be efficiently and effectively in communication with a multiplicity of other clients and servers of the network, greatly multiplying the number of link choices available and, if conditions change, or if a better link to a server becomes known to a client, where the link for a client can be updated and improved.

DISCLOSURE OF THE INVENTION

The present invention includes a wireless network system which is particularly well adapted for connection to a wide area network such as an Intranet or the Internet. The wireless network system includes one or more servers which are coupled to the wide area network, and two or more clients capable of communicating with the server or with each other via radio modems. The communication in the wireless network system preferably takes the form of digital data packets, which are not too dissimilar from the TCP/IP data packets used over the Internet. However, the data packets of the present invention also include data routing information concerning the path or “link” from the source of the packet to the destination of the packet within the wireless network. The data packets also include a code indicating the type of packet being sent.

In operation, a client of the wireless network system of the present invention has either a direct or an indirect path to a server of the wireless network system. When in direct communication with the server, the client is said to be “1 hop” from the server. If the client cannot reliably communicate directly with the server, the client will communicate with a “neighbor” client which has its own path (“link”) to the server. Therefore, a client can communicate with the server along a link that includes one or more other clients. If a client communicates with the server through one other client, it is said to be “2 hops” from the server, if the client communicates to the server through a series of two other clients, it is said to be “3 hops” from the server, etc. The

6,044,062

5

process of the present invention preferably includes an optimization process which minimizes the number of hops from the clients to the servers, on the theory that the fewer the number of hops, the better the performance of the network. Alternatively, the optimization process can also factor in traffic and transmission reliability of the various links to determine the optimal path to the server.

A wireless network system in accordance with the present invention includes at least one server having a server controller and a server radio modem, and a plurality of clients, each including a client controller and a client radio modem. The server controller implements a server process that includes the controlling the server radio modem for the receipt and transmission of data packets from clients of the network. The client controller implements a client process including the transmission and receipt of data packets from the server and from other clients. Preferably, the client process of each of the clients initiates, selects, and maintains a radio transmission path ("link") to the server. As noted previously, this radio transmission path to the server is either a direct path to the server (1 hop) or an indirect path to the server (multi-hop) through one or more other clients. Preferably, the client process of a particular client also constantly searches for improved paths to the server.

A method for providing wireless network communication in accordance with the present invention includes providing a server implementing a server process, and providing a plurality of clients, each client implementing a client process. The server process includes receiving data packets via a server radio modem, sending data packets via the server radio modem, performing a "gateway" function to another network, and performing housekeeping functions. The client process includes the sending and receiving of data packets via a client radio modem, maintaining a send/receive data buffer in digital memory, and selecting links to the server. Again, the client process preferably chooses a "best" link to the server that is either a direct path or an indirect path through one or more other clients.

The server of the present invention provides a gateway between two networks, where at least one of the networks is a wireless network. The gateway function of the server makes any necessary translations in digital packets being sent from one network to the other network. The server includes a radio modem capable of communicating with a first, wireless network of the present invention, a network interface capable of communicating with the second network (which may or may not be wireless and, in fact, is preferably a wired TCP/IP protocol network), and a digital controller coupled to the radio modem and to the network interface. The digital controller passes data packets received from the first network that are destined for the second network to the second network, and passes data packets received from the second network that are destined for the first network to the first network, after performing any necessary translations to the data packets. The digital controller further maintains a map of the links of the first network and provides that map to first network clients on request. By maintaining a map of the first network links, the server is able to properly address packets received from either the first network or the second network to the appropriate client of the first network, and allows the client of the network to maintain and upgrade their data communication paths to the server.

A network client for a wireless communication network of the present invention includes a radio modem capable of communicating with at least one server and at least one additional client, and a digital controller coupled to the radio

6

modem to control the sending and receiving of data packets. The digital controller is further operative to determine an optimal path to at least one server of the wireless network. The optimal path can be either a direct path to the server, or an indirect path to the server through at least one additional client.

The method, apparatus, and systems of the present invention therefore provide a wireless network that is both robust and efficient. Since each client of the network can potentially be in communication with a multiplicity of other clients and servers of the network, there are a great number of link choices available. If conditions change, or if a better link becomes known to a client, the link can be updated and improved.

These and other advantages of the present invention will become apparent upon reading the following detailed descriptions and studying the various figures of the drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a pictorial representation of a wireless network system in accordance with the present invention;

FIG. 1a illustrates a first tree structure of the data communication paths or "links" of the wireless network system of FIG. 1;

FIG. 1b illustrates a second tree structure illustrating optimized or "stabilized" data communication paths for the wireless network system of FIG. 1;

FIGS. 2a-2g, 2h'-2h'', and 2i-2o are used to describe a prototype of the wireless network system of FIG. 1, illustrating both the path connection and path optimization processes of the present invention;

FIG. 3 is a block diagram of a server, router, the first wireless network, and the second network of FIG. 1;

FIG. 4 is a flow diagram of a server process of the present invention operating on the server of FIG. 3;

FIG. 5 is a flow diagram of the "Process Packets Received From Client" step of FIG. 4;

FIG. 5a illustrates a data packet processed by the process illustrated in FIG. 5;

FIG. 5b is a flow diagram illustrating the process "Am I on Route?" of FIG. 5;

FIG. 5c is a flow diagram illustrating the process "Data?" of FIG. 5;

FIG. 6 is a flow diagram illustrating the "Process Inter-nodal Information" process of FIG. 5;

FIG. 6a is a flow diagram illustrating the process "Client Authentic?" of FIG. 6;

FIG. 6b is a flow diagram illustrating the process "Put New Client In Tree" of FIG. 6;

FIG. 7 is a flow diagram illustrating the function "ADDSON(P,C)" of FIG. 6b;

FIGS. 7a and 7b are used to illustrate the operation of the ADDSON function of FIG. 7;

FIG. 8 is a flow diagram illustrating the "Delete Client From Tree" process of FIG. 6;

FIGS. 8a-8c illustrate the process of FIG. 8;

FIGS. 9a-9c illustrate the "Place Network Tree In Client Transmit Buffer" of FIG. 6;

FIG. 10 is a pictorial representation of the "Communicate With Network" process of FIG. 4;

FIG. 11 is a flow diagram of the process "Communicate With Network" of FIG. 4;

6,044,062

7

FIG. 12 is a block diagram of a radio packet modem used in the present invention;

FIG. 13 illustrates a client, such as a client A, B, C, or D of FIG. 1;

FIG. 14 is a flow diagram of a client process running on the client of FIG. 13;

FIG. 15 is a flow diagram of the process "Radio Transmit and Receive Packet" of FIG. 14;

FIG. 16 is a flow diagram of the process "Perform Transmit/Receive Process" of FIG. 15;

FIG. 17 is a flow diagram of the process "Process Computer Receive Packets" of FIG. 16;

FIG. 18 is a flow diagram of the process "Process Radio Received Packets" of FIG. 16;

FIGS. 18A and 18B are used to illustrate the process "Is It My Packet?" of FIG. 18;

FIG. 19 is used to illustrate the "Process Per Type Code" of FIG. 18;

FIG. 20 illustrates an initialization routine of the client process of the present invention; and

FIGS. 21a-21d illustrate the process of FIG. 20.

BEST MODES FOR CARRYING OUT THE INVENTION

FIG. 1 illustrates a wireless network system 10 in accordance with the present invention. The wireless network system 10, which will also be referred to herein as a "first network," is preferably in communication with a second network 12 via a digital communication bridge or router 14. The construction and operation of networks, such as second network 12, and bridges or routers, such as router 14, are well-known to those skilled in the art. In the present invention, it is preferred that the second network operates on the aforementioned TCP/IP protocols, i.e. the second network is the Internet or is a private Intranet. At times, herein, the second network will be referred to as simply the Internet, it being understood that other forms of a second network are also operable with the systems, apparatus, and processes of the present invention. Again, the construction and operation of the Internet and Intranets are well-known to those skilled in the art. Likewise, routers, bridges, and other network devices such as hubs, gateways and Ethernet interfaces are well-known to those skilled in the art, and are available from a variety of sources including Cisco Systems, 3-Com, Farillon, Asante, etc. In general, as a "network interface" will refer to any such device that allows a server of the wireless network system of the present invention to communicate, directly or indirectly, with the second network.

The wireless network system 10 of the present invention includes one or more servers 16, the single example of which is herein labeled S. It should be noted that the server 16 serves as a gateway in that it performs a translation service between the first network and the second network. For example, the data packets on the first network include links and data types that are only applicable to the first network. Therefore, such links and data types are removed from the data packets before they are transmitted to the second network which, as noted previously, preferably operates on a TCP/IP protocol. Conversely, data packets received from the second network are modified to include the links and data types before they are transmitted to the first network. Therefore, the data packets on the first or wireless network can be essentially "packages" or "envelopes" for TCP/IP data packets when they are destined for the Internet

8

or received from the Internet. However, as will be discussed in greater detail subsequently, the data packets of the first network can be of types other than "data" types for TCP/IP formatted data. It should also be noted that while only a single server S is shown in this example that, in most cases, multiple servers, each with their own gateway to the Internet, will be used in the first network.

The wireless network system 10 further includes a number of clients 18, each including a client machine 20 and a radio modem 22. The client machine 20 can be any form of digital processor, including a personal computer (PC), a computer workstation, a personal digital assistant (PDA), etc. In the present invention, the client machine 20 is preferably a personal computer (PC) made to the Microsoft Windows/Intel microprocessor ("Wintel") standard, or to the Apple Macintosh standard. Wintel and Macintosh compatible computers are commercially available from a variety of vendors. Likewise, computer workstations and PDAs are available from a number of vendors. Radio modems, such as the radio modem 22, are further available from a number of vendors. The present invention has been implemented using radio modems produced by GRE America, Inc. which operate on a spread spectrum technology, and which provide good receiver sensitivity and repeater capabilities. These GRE America, Inc. radio modems are commercially available under the Gína trademark and operate in the 2.4 gigahertz or 90 megahertz bands with support for packetized data transmission. The Gína brand radio modems further include error detection and correction, can operate in asynchronous or synchronous modes, and can support data speed from 300 to 64 kbps. Furthermore, the Gína radio modems can operate in a point-to-point or a point-to-multipoint mode.

A server process, to be discussed in greater detail subsequently, is implemented on the server 16, and a client process, also to be discussed in detail subsequently, operates on each of the clients 18. In the present invention, the client process operates, at least in part, on the client machine 20. However, in alternative embodiment of the present invention, the client process can operate on the controller of the radio modem 22 of the client 18.

In wireless network system 10 illustrated in FIG. 1, the client 18A is in "direct" radio communication with the server 16 as indicated by the radio communication link 26. This will be referred to herein as "direct" or "1 hop" or "line-of-sight" connection with the server 16. The client 18B, however, does not have a direct path or "link" to the server 16 due to an obstacle 24, such as a hill, large building, etc. Therefore, the client 18B communicates via a radio link 28 with client 22A which relays the data packets from client 18B to the server 16. A client 18C has a direct line-of-sight to the server 16, but is out of transmission range to the server 16. Therefore, the client 18C transmits its data packet by a radio link 30 to client 18B, from where it is relayed to client 18A via link 28, for eventual relay to the server S via radio link 26.

As noted in FIG. 1, 18D is in direct communication with server 16 via radio communication link 32. If client 18C detects the transmissions of client 18D, it will note that client 18D has less "hops" to the server 16 than does client 18B, and will switch its link from client 18B to client 18D. This process is a part of the "stabilization" or "optimization" process of the network 10.

It will therefore be appreciated that the wireless network system 10 of the present invention is constantly attempting to optimize itself for the "best" data transmission. In the

6,044,062

9

embodiment described herein, this optimization looks solely to the number of hops between the client and the server for the sake of simplicity. However, other factors can also affect the quality of the data transmission. For example, the traffic of data packets through a particular client modem may be large, such that it is better to route the data from neighboring clients through other clients, even though there may be more hops involved with this alternative routing. Also, some radio links may be less robust or may be slower than other links, such that optimization may result in a routing of data around the less robust or slower links, even though it may increase the number of hops to the server 16. Therefore, although the present preferred embodiment looks at only one single factor in its optimization processes, it will be appreciated by those skilled in the art that multiple factors can be used to stabilize or optimize the wireless network system 10 of the present invention.

It should also be noted that the wireless network system 10 of the present invention is quite robust in that it will survive the loss of one or more clients of the system. For example, if the client 18A is lost due, for example, to a power or system failure, the data packets for client 18C can be routed through the client 18D, and the data packets for the client 18B can be routed through clients 18C. Therefore, the wireless network system 10 is highly robust and highly survivable under a number of adverse conditions.

In addition, the present invention permits mobile communication within the wireless network system 10. For example, if the client 18D is a portable computer and is moved around within the wireless network system 10, it will opportunistically change its data communication path as better links become available. For example, if the client 18D is moved close to the client 18B, it may use the client 18B as its link to the server 16. Also, any routing through the client 18D from other clients (such as 18C in this example) will be updated and optimized as the data path for the client 18D changes.

It should be noted that, in general, the network will work the best and will be the most stable if the radio modems and their client/controllers are never turned off. It is therefore desirable to not have an on/off switch on the radio modem, so that clients are always participating in the network traffic distribution. However, even if a radio modem is turned off, the remaining clients will re-route through other clients, as will be discussed subsequently.

In FIGS. 1a and 1B, two “tree” structures are shown illustrating the various links that were discussed, by way of example, with reference to FIG. 1. The tree structure is maintained in the server S, and is transmitted to any client that may request it.

In FIG. 1a, a tree indicates that client 18A is linked to server 16 by a link 26, client 18B is linked by link 28 to client 18A and by link 26 to the server, and client 18C is linked by line 30 to client 18B, by link 28 to client 18A, and by line 26 to the server 16. The client 18D is in direct communication with the server 16 via radio link 32. Therefore, clients 18A and 18D are both “1 hop” away from the server 16, client 18B is “2 hops” away from server 16, and client 18C is “3 hops” away from server 16.

In the scenario where client 18C realizes it has a better connection to server 16 through the client 18D, the link 30 to client 18B is no longer used, and a new radio link 34 to client 18D is established. This is illustrated in FIG. 1b. Now, clients 18A and 18B remain 1 hop clients, clients 18B remains a 2 hop client, but client 18C is upgraded from a 3 hop client to a 2 hop client. Therefore, the data transmission efficiency of the network has been “stabilized” or “optimized.”

10

It should be noted that the term “link” is used to convey both the connection to an adjacent client as well as the entire path from a client to a server. It will therefore be understood that when speaking of a link to an adjacent client, that this also implicitly includes all necessary links from that adjacent client to the server, i.e. a link is the entire path description from a given client to a given server.

FIGS. 2a–2o, an exemplary wireless point-to-multipoint network in accordance with the present invention is prototyped to facilitate a discussion of the theory and operation of the present invention. In FIG. 2a, a network 36 with 60 potential “nodes” is illustrated. As used herein, a “node” can either be a client or a server. The nodes 14 and 16 have been arbitrarily selected as servers for the purpose of this example. The nodes 14 and 16 are marking servers with the large, black dot immediately adjacent to those numerals. For the purpose of this example, it is assumed that a node can only communicate with an immediately adjacent node. Of course, in actual operation, nodes may be able to communicate with more distant nodes than its immediate neighbor nodes.

FIG. FIG. 2b, a first client is designated at node 05 (hereafter “client 5”). For the purposes of this example, the Yen or “¥” symbol is positioned next to the client 5. As noted previously, for the purpose of this example, we will assume that any particular node is only in radio communication range of a node that is adjacent in a horizontal, vertical, or diagonal direction, i.e. is an immediately adjacent “neighbor.” In this instance, client 5 detects that there is a radio contact with node 14, which is a server (hereafter “server 14”). The server 14 and the client 5 will build a routing path or “link” between each other. This is accomplished by client 5 transmitting a “I Am Alive” packet seeking a route to a server. The server 14, being within radio transmission range, will respond and will add the client 5 to its routing table as its “left son.” The meanings of the “routing table” and the “left son” will be described subsequently. The routing table of the server 14 is therefore 14(05), and the route from the client 5 to the server 14 is 05>14. Again, this notation will be discussed in greater detail subsequently.

The network 36 then has a second client 6 added as indicated by the ¥ symbol next to node 6 in FIG. 2c. Second client 6 makes radio contact with client 5 and builds a routing path or “link” to the server 14 through the client 5. Server 14 updates its routing table accordingly. This is accomplished by client 6 issuing an “I Am Alive” packet seeking a client repeater route to a server. Client 5 will respond and add client 6 to its routing table as its left son. The updated routing table of the server 14 is therefore: 14(05(06)). The route from the user client node 6 to the server 14 is: 06>05>14.

In FIG. 2d, a third client 7 is added to the network 36 as indicated by the “¥” symbol next to node 07. Client 7 establishes contact with client 6 and finds a path through clients 6 and 5 to server 14. This is accomplished by client 7 issuing a “I Am Alive” packet seeking a client repeater route to the server 14. Client 6 will respond and add client 7 to its routing table as its left son. The updated routing table of the server 14 is then: 14(05(06(07))). The route from client 7 to the server 14 is: 07>06>05>14.

In FIG. 2e, another client 16 has been added at node 16 as indicated by the “¥” symbol. It should be noted that client 16 can make radio contact with clients 05, 06, and 07. However, client 16 recognizes node 26 as being a server (hereafter “server 26”) and then connects directly to server 26. This is accomplished by client 16 transmitting a “I Am

6,044,062

11

Alive" packet seeking a route to a server. The server 26 will respond and will add client 16 to its routing table as its left son. The updated routing table of server 26 is then: 26(16). The routing from client 16 to the server 26 is 16>26.

In FIG. 2f, a server routing table and a route for each client thus far in the example are illustrated. It should be noted that when client 16 came into existence, a shorter route was created for client 7 to a server, namely via client 16 to server 26. As noted in this figure, client 7 has made the adjustment to connect to server 26, thereby "stabilizing" or "optimizing" the network 26. Also, it should be noted that server 14 has deleted client 7 from its routing table, since client 7 is now using server 26 as its gateway to the Internet. This creates a universe of six nodes, of which are two are servers and of which are four are clients. The average "hop" distance from a client to a server is 1.5 hops. The remainder FIGS. 26g-26o further illustrate these concepts.

In FIG. 26g, the network 36 illustrates an extreme example where 58 clients are connected to the two servers 14 and 26. FIGS. 2h' and 2h" show a fully "stabilized" or "optimized" network where the path or "link" from any client to a server is as short as possible, i.e. where there is few "hops" as possible. It should be noted that the optimization occurs dynamically during operation and without complex algorithms and look-up tables. As will be discussed in greater detail subsequently, the optimization occurs when clients "hear" transmission from other clients that have a better (i.e. shorter) path to a server.

FIG. 2h' shows the network as seen from the point of view of servers 14 and 26 and from the point of views of clients 1-client 31. In FIG. 2h", the network as seen from the point of view of clients 32-60, along with statistics for the overall network, are shown. In brief, in a universe of 60 nodes, of which are two are servers and 58 are clients, the average hop distance from a client to a server is 2.36206897 hops.

In FIG. 2i, the process of adding a new client 9 to the server is illustrated. The first time that client 9 came "alive" (i.e. became operational) it took five tries before node 9 found a client neighbor with the path to the server. The reason that it may take many tries to find a connection path is that multiple neighbors of client 9 are responding to client 09 "I Am Alive" message via CSMA/CD (Carrier Sent Multiple Access/Collision Detection) protocol. The likelihood that any particular neighbor of client 9 will respond first is, essentially, random. Once client 09 hear from a neighbor that it does not have a path to a server, client 9 tells that neighbor not to respond to the next "I Am Alive" announcement from client 9. In consequence, client 9 keeps trying to find a path to the server until it succeeds. However, that path may not be the shortest path. In this example, the client 9 finds a path to the Internet server, resulting in the updating of the routing table for the Internet server 14 as 14(05(06(07(08(09))))),04,03). The route or "link" from client 9 to the server is: 09>08>07>06>05>14.

In FIG. 2j, a client 29 is finding a route to the server via one of its neighbors. It finds a route through client 19, and is added to the routing table of client 19 as its left son. The routing table of server 14 is also updated, and the route from user client 29 to the server is determined. However, this route is not an optimal route in that it includes a greater number of hops than necessary.

In FIG. 2k, the "stabilization" or "optimization" process is illustrated. It was previously noted that the client 29 has a non-optimal path to its server. In order to improve this path, client 29 will receive "help" from its neighbors starting with client 7. Client 7 currently has a route to server 14. Client 7

12

starts randomly probing its neighbors looking for a shorter route to a server. Client 7 finds a shorter route to client 26. Client 7 informs server 14 to drop client 7 from server 14's routing table, and client 7 informs server 26 to add client 7 to its routing table. Since client 29 was "downstream" from client 07, client 29 dynamically becomes switched to a route to server 26.

In FIG. 2l, this process is repeated for client 08. Notably, client 8 shortens its route to server 26 by 1 hop. Client 9 cannot improve its route to server 26.

In FIG. 2m, client 18 shortens its route to server 27 to 2 hops. This is despite the fact that the route through clients 07 and 08 are a relatively efficient 3 hop links.

In FIG. 2n, client 29 is optimizing its path. Client 29 eliminates 18 from its route by "leap frogging" past client 18 with the result of the shortest possible 3 hop route to a server. Ultimately, therefore, client 29 route has improved from a 7 hop path through server 14 to the shortest possible 3 hop path to server 26. This result is dynamically accomplished with the efficiencies of client 7, 8, and 18 also improving, and without the need for complex routing algorithms.

In FIG. 2o, another example of individual dynamic routing is illustrated for client 44. This client node shortens its route from 3 to 2 hops by switching server destinations. Client 44 drops out of the server 14's routing table and gets added to server 26's routing table.

The advantage of prototyping the system as explained in FIGS. 2a-2o is that further optimizations become apparent. For example, if a great deal of network traffic is going through a particular node, it may be desirable to place a "passive repeater" at that node. A passive repeater is not a client, per se, but, rather, is a transceiver that receives and rebroadcasts packets. The passive repeater therefore effectively extends the range of the transmitting clients, and reduces data bottlenecks in the system. A passive repeater is also useful for clients with long links to a server in that it can shorten the link by effectively allowing the clients to skip some intermediate links. The prototyping of the system is also useful in that it shows that placing servers near the center of the network reduces the average link length (i.e. reduces the average number of client hops) in the network.

In FIG. 3, a block diagram of the server 16 of FIG. 1 is illustrated. In this instance, the server 16 includes a computer system 28 and a number of peripherals coupled to the computer system. The computer system 38 can be a personal computer system, a computer workstation, or a custom data processor capable of implementing the processes of the present invention.

By way of example, the computer system 38 includes a microprocessor 44 that is coupled to a memory bus 44 and to an input/output (I/O) bus 46. Typically also coupled to the memory bus 44 are random access memory (RAM) 48 and read only memory (ROM) 50. The RAM 48 is usually volatile (i.e. its contents are lost when power is removed) and is used for temporarily or "scratch pad" memory. The ROM 50 is non-volatile (i.e. its contents are not lost when power is removed), and typically includes the start-up instructions for the computer system 38. A number of peripherals are typically coupled to the I/O bus 46. For example, a removable media drive 52 for a removable media 54 (such as a floppy disk, a Zip® disk, or a C/D ROM) is typically coupled to the I/O bus 46, as is a fixed or hard disk 56. Furthermore, a router 14 or bridge can be used to couple the I/O bus 46 to the Internet 12 as previously described. In addition, an RJ45 Ethernet interface 58 can be used to couple the computer system 38 to a local area network 60

6,044,062

13

and from there to the Internet 12 by a router 14', or the like. Also, a radio modem 62 (including a control section C, a radio section R, and an antenna 64 coupled to the radio section R) can be coupled to the I/O bus 46. The radio modem 62 can communicate with the network 10 including a number of nodes 66 by a wireless transmission or "radio link 68." The assembly of the hardware of the server illustrate in FIG. 3 will be apparent to those skilled in the art.

In FIG. 4, a server process 70 of the present invention is implemented on the server 16. More particularly, the server process 70 can be implemented on computer system 38, within the control section of the radio modem 62, or partially in both of those places. In the present preferred embodiment, the majority of the server process 70 is implemented on the computer system 38. However, it should be noted that the control section C of the radio modem 62 includes a micro-processor and memory and, with proper program instructions, can be made to implement the process 70 of FIG. 4, freeing the personal computer 38 for other tasks.

The server process 70 includes a server process control 72 and four subprocesses. More particularly, the subprocesses include a process 74 which processes received from clients, a process 76 which sends packets, a process 78 which communicates with the network, and a process 80 which performs housekeeping functions. Each of these processes will be discussed in greater detail subsequently.

In FIG. 5, the process "Process Packets Received From Clients" 74 of FIG. 4 is illustrated in greater detail. The process 74 begins at 82, and in a step 84, the variable RETRY is set to 0. Next, a step 86 retrieves a packet from the client receive buffer, and a decision step 88 determines whether the path or "link" of the packet is same as the currently stored link in memory. If not, a step 90 updates the tree. If so, or after the updating of the tree in step 90, a decision step 92 determines whether it is "My Packet?" In other words, step 92 determines whether the packet being received by the server was intended for that server. If not, a decision step 94 determines whether that server is on the route. If that server is on the route, but it is not its packet, a decision step 96 determines whether the packet has already been repeated. If not, the packet is placed in the client transmit buffer. If decision step 94 determines that the server is not on the route, or the packet has already been repeated, or upon the completion of step 98, a decision step 100 looks for time-out. The time-out is provided by the server process control 72 such that the computer hardware resources on which process 70 are implemented can be shared among the four processes. More particularly, in most instances, the computer hardware resources are shared among the subprocesses 74-78 in a "round-robin" fashion well-known to those skilled in the art. However, it should be noted that at times the strict round-robin scheduling is not adhered to, as will be discussed subsequently.

If step 100 determines that a time-out has occurred, the decision step 102 determines whether the retry number RETRY is greater than the number of retries allowed, namely NUMRETRY. In its preferred embodiment, the number of retries RETRY are set at, perhaps, 2 or 3 so that the server does not tie up its resources with endless retries of the process. If RETRY is greater than NUMRETRY, the process is completed as indicated at 103. Otherwise, a step 104 increments RETRY by 1. In the absence of a time-out and in the absence of the number of retries being used up, process control returns to step 86.

If step 92 determines that the packet is for that server, a step 106 determines whether the packet is a data type. If not,

14

a step 108 processes "internodal information." If so, a step 110 places the data in a server transmit buffer. After the completion of steps 108 or 110, process control is returned to step 100 to determine if there is a time-out.

In FIG. 5a, a "data packet" 112 in accordance with the present invention is illustrated. As it will be appreciated by those skilled in the art, a data packet is an associated string of digital information that is transferred and processed as a unit. The data packet 112 of the present invention includes a header 114, a type 116, and data 118. The data 118 can be standard TCP/IP data. The header 114 includes the source address, the address of all hops along the way (i.e. the "link" of the data packet), and the destination address. Hops (i.e. clients and servers) that already have been traversed (i.e. have already forwarded the data packet) are indicated with an asterisk ("*") symbol. The type 116 is, in this implementation, a two digit code indicating the type of the data packet 112, as will be discussed in greater detail subsequently. The data section 118 of the data packet 112 includes the data associated with that packet. In the present invention, the data section is in the range of 128-1024 bytes in length.

In FIGS. 5b and 5c, respectively, the decision steps 94 and 106, respectively are illustrated with respect to the data packet architecture of FIG. 5a. The decision step 94 ("Am I On Route?") of FIG. 5 is simply determined by the process 120 "My Address In the Header?" If yes, the process of FIG. 5 branches to step 96, and if no, the process of FIG. 5 branches to step 100. In FIG. 5c, the decision step 106 "Data?" simplifies to a process 122 "Is the Type Equal to 14?" This is because, in the present invention, a type 14 has been arbitrarily chosen to indicate a data type. If yes, the process of FIG. 5 branches to step 100, and if no, the process of FIG. 5 branches to step 108.

In FIG. 6, the step 108 "Process Internodal Information" of FIG. 5 is explained in greater detail. The process 108 begins at 124 and, in a multi-branch decision step 126, the type of the data packet is determined. If the type is a "01", a step 128 places an acknowledgment and a "code seed" in the client transmit buffer, and the process is completed at 130. Acknowledgments and "code seeds" will be discussed subsequently. If the type is a "07", a step 132 receives the client request for the network tree, and the process places the network tree in the client transmit buffer in a step 134. The process is then completed at 130. If, however, the type is "13", a step 136 deletes the client from the tree and a step 138 determines whether a flag has been set. If not, the process is completed at 130. If, the flag has been set as determined by step 138, a step 140 puts a new client in the tree and the process is then completed at 130.

If decision step 126 determines that the type is "05", a step 142 determines whether the client is authentic. The authentication process, which will be discussed subsequently, keeps unauthorized clients from being added to the network. If the client is not authentic, the process is completed at 130 and the client is not allowed to connect to the server. If step 142 determines that the client is authentic, a step 144 determines whether the client is already in the server tree. If yes, the flag is set in a step 146 and process control is turned over to step 136 to delete the client from the tree. Since the flag has been set, step 138 branches the process control to step 140 and the new client is placed in the tree, after which the process is completed at 130.

The addition and removal of nodes from trees are well known to those skilled in the art. For example, in the book, incorporated herein by reference, *SNOBOL 4: Techniques*

6,044,062

15

and Applications, by Ralph E. Griswald, Department of Computer Science, University of Arizona, Prentiss-Hall, Inc., © 1975, ISBN 0-13-853010-6, algorithms for placing and removing clients from trees are discussed.

FIG. 6a illustrates the process 142 of FIG. 6 in greater detail. More particularly, the process 142 begins at 148 and, in a step 150, a “seed” is chosen on the fly. Next, in a step 152, a “one-way” function is performed using the seed and a known authentication algorithm, and a one-way result is stored. Next, found in step 154, the seed is “camouflaged,” and in a step 156, places an acknowledgment code and the camouflaged seed in the client transmit buffer. The process is then completed at 158.

The purpose of the process 142 is to prevent unauthorized “clients” from accessing the network. For example, hackers can be prevented from accessing the network unless they can crack the authentication process, which is nearly impossible.

Authentication techniques are well known to those skilled in the art. For example, the book, incorporated herein by reference, *Algorithms in SNOBOL 4*, by James F. Gimpel, Bell Telephone Laboratories, John Wiley & Sons, a Wiley Interscience Publication, © 1976 by Bell Telephone Labs, Inc., ISBN 0-471-30213-9, describes authentication techniques using one-way seeds. See, in particular, pp 348–349 with back-references. In brief, a “seed” is chosen “on the fly”, such as by reading the system clock. The one-way function modifies the seed using an algorithm known to both the server and the clients. The one-way result, which in this instance is 4 bytes in length, is stored. The step 154 then “camouflages” the seed by dispersing the 4 bytes among perhaps 26 other bytes prior to transmitting the camouflaged seed. The receiving clients know which of the four bytes to use for their one-way function.

The process 140 “Place New Client In Tree” of FIG. 6 is illustrated in greater detail in FIG. 6d. The process 140 begins at 160 and in a step 162, it is determined whether this is a “1 hop” client. If so, a decision step 164 determines whether it is a new client C. If so, the variable P is set to S and the function “ADDSO” with the variables (P, C) is evoked. S, of course, is the server or root of the tree. If step 64 determines that it is not a new client C, or after the completion of the ADDSON function, the process ends at 170.

If step 162 determines that it is not a 1 hop client (i.e. C is a multi-hop client) a step 162 determines whether the parent P of client C is known to client C. If not, a step 174 determines the parent P from the header of client C. If the client C does know its parent, or after the completion of step 174, a step 176 receives parent P from client C. Next, in a step 178, the function ADDSON(P,C) is evoked, and the process is completed at 170.

In FIG. 7, the ADDSON(P,C) function is explained in greater detail. More particularly, function steps 168–178 begin at 180 and, in a step 182, the variables C, P are received. In this notation, the string RSIB() refers to a string of right siblings, and the notation LSON() refers to a string of left sons. A step 184 sets RSIB(C)=LSON(P). A step 186 sets a string FATHER(C)=P and a step 188 sets the string LSON(P)=N2. The variable N2 is an in-memory pointer that points to the memory location of nodes. The string FATHER provides a pointer from a child C to its father, which in this case is P. The process is then completed as indicated at 190.

In FIGS. 7a and 7b, the ADDSON function is graphically illustrated. In FIG. 7a, a parent 192 has a left son 194 and a right sibling 196. The parent 192 and left son 194 have

16

mutual pointers to each other, while the right sibling 196 has only a pointer to the parent 192. The left son 194 also has a pointer to the right sibling 196. When the ADDSON function is evoked with the argument (P, C) C is added as the left son 198 and the pointer in the parent 192 is updated to point to the left son 198. The left son 198 has pointers to the parent and to the new right sibling 194. The new right sibling 194 still has a point to the older right sibling 196, and both siblings 194 and 196 have pointers to the parent 192. It should be noted, under all circumstances, that the parent is only directly aware of the left son, in that it only has a pointer to the left son.

In FIG. 8, the process 136 “Delete Client From Tree” is illustrated in flow- diagram form. The process 136 begins at 200 and in a step 202, it is determined whether the target is equal to the left son. The “target” is, of course, the client to be deleted. If the target is the left son, a step 204 determines if there are other siblings. If not, the left son is deleted in a step 206. If there are other siblings, a step 208 makes the next sibling the left son, and then the left son is deleted by step 206. The process is then completed at 210. If step 202 determines that the left target is not equal to the left son, the target is found in a step 212, and is then deleted in a step 214. A step 216 then changes the sibling pointers, and the process is completed at 210.

FIGS. 8a–8c are several scenarios used to illustrate the process of FIG. 8. Assume that there is a tree structure as illustrated in FIG. 8a. If the node “A” (i.e. a client A) of FIG. 8a “disappears” all nodes (clients) 218 that used client A as a path to the server P are dropped from the network as illustrated in FIG. 8b. With reference again to FIG. 8a, if the node C disappears, the sibling B will simply reset its pointer to point to sibling D without any loss of service to any of the nodes. The lost nodes 218 of FIG. 8b will need to reestablish themselves into the network as previously described.

FIG. 9a is a tree structure that will be used to illustrate the step 134 “Place Network Tree In Client Transmit Buffer” of FIG. 6. Since the tree structure 220 is a logical construct, it must be represented in a form suitable for digital transmission. This form is illustrated in FIG. 9b as a string 222. With reference to both FIGS. 9a and 9b, the string 222 represents the tree on a top-to-bottom, left-to-right basis. Therefore, the string 222 indicates for the parent X that its left son is 3 with a right sibling B. For the parent 3, there is a left son 9 with a right sibling Z. For the parent Z, there is a left son 8, a right sibling 5, and another right sibling Q. For the parent Q, there is a left son P. Therefore, the tree structure 220 has been completely and compactly represented by the notation of the string 222.

The converting of trees to strings and the reverse is well known to those skilled in the art. In short, a left parenthesis in the string indicates that a left son follows, and a comma in the string indicates that a right sibling follows. For example, the aforementioned book *SNOBOL 4: Techniques and Applications* describe the process for converting trees to “prefix form” as described above, and vice versa. The aforementioned book *ALGORITHMS IN SNOBOL 4* likewise describes the process. While the tree structure 9a is useful for representing and traversing a tree data structure, it is not well-adapted for rapid searching for particular nodes. For this purpose, the table of FIG. 9c is created to implement fast searching and other housekeeping functions. In this illustration, the table of FIG. 9c includes four columns. The first column is the sequential element or “node” number, a second column 226 is the node name, the third column 228 includes the time stamp of the creation of the node, and the fourth column includes the actual physical

6,044,062

17

memory location of the node. In this way, a particular node can be searched by element number, node name, time stamp, or memory location without resorting to the time consuming recursive search algorithms otherwise typically used to search tree structures.

FIG. 10 is a pictorial representation of a portion of the server of FIG. 3 that has been simplified to explain the steps 78 of FIG. 4 "Communicate With Network." The wireless network system 10 includes a number of clients and, perhaps, other servers, each of which has its own IP address. The radio modems of those clients and servers communicate with radio modem 62 of the server which provides digital data to the serial port of a server computer or host 38. A router, bridge or other device is used to connect the server to a network, such as a TCP/IP network 12. Of course, the radio packet modem 62 and the server computer 38 can be considered part of the wireless network system 10 as described previously. The combination of the server and the router or the like performs a "gateway" function, in that it provides translation services between the two networks 10 and 12.

Referring back to FIG. 4, the step 76 "Send Packets" simply involves sending the data packets stored in the client transmit buffer to the network 10 through the radio modem 62. Likewise, and in a straightforward matter, the step 78 "Communicate With Network" simply forwards the data stored in the network transmit buffer to the network through the router 14 or through another route, such as the Ethernet interface 58. The "Send Packets" and "Communicate With Network" processes will be easily understood by those skilled in the art. Again, the server process control 72 allocates system resources among the processes 74-80 on a round-robin basis.

In FIG. 11, the housekeeping process 80 of FIG. 4 is illustrated in greater detail. Since the housekeeping function 80 is of generally less importance than the other functions of process 70, it is possible that housekeeping function will be interrupted with a branch to one of functions 74, 76, and 78 of FIG. 4.

More particularly, in FIG. 11, the housekeeping function 80 of FIG. 4 is illustrated in greater detail. The process 80 begins at 232 and, in a decision step 234, it is determined whether a flag is set. If not, the next element is equal to 1, i.e. it is picking the first element on the list. If step 234 determines that a flag is set, the process 80 knows that the housekeeping has been interrupted in the middle of the list and therefore the next element is set equal to the stored mark point as indicated in step 238. Next, a step 240 determines whether if the end of the table has been reached. If so, the process is completed at 242. If the end of the table has not been reached, the next element retrieved in a step 244, and then in a step 246, it is determined whether the current time minus the time stamp is greater than a predetermined interval. If it is, a step 248 deletes the client from the tree and from the table. This step 248 is performed to ensure that a client node that has dropped out the network 10 without informing the server is deleted from the server tree at some point in time. A suitable interval may be 15 minutes, or any desired interval set by a network manager. Process control then returns to step 240.

If step 246 determines that a node (i.e. a client) corresponding to the next element has checked-in within the time INTERVAL, a step 250 determines whether there is heavy traffic on the server. If not, process control is returned to step 240. If there is heavy traffic, a step 252 marks the place in the table corresponding to the current element (i.e. the

18

marked point in the list is stored in memory) and then a step 254 determines the traffic type. Process control then branches to process 256 if it is heavy network traffic, 258 if it is heavy outgoing packet traffic, and process 260 if it is heavy incoming packet traffic.

In FIG. 12, a radio modem 62 (which can be similar to all of the radio modems described herein) is illustrated in block diagram form. Again, the radio modem 62 is commercially available from GRE America, Inc. as the Gína spread spectrum radio modem, models 6000N-5 or 8000N-5. Spread spectrum technology gives good reliability and some transmission security in that a 127 bit cyclical code must be known by both the transmitting and receiving node. However, for true data security, encryption techniques, well known to those skilled in the art, should be used. Gína modems do include the option of 64 bit built-in encryption as an option.

It should be further noted that the Gína radio modem hardware can be modified to incorporate the server process (or the client process for the client radio modems) of the present invention by storing program steps implementing those processes into a ROM or programmable ROM (PROM) 262 of the radio modem 62.

The radio modem 262 includes a microprocessor 264 coupled to a bus 268. The microprocessor is an Intel 80C188 microprocessor in the present example. The PROM 262 (which currently stores 512 Kbytes of code) is coupled to the bus, as is RAM 268, a serial interface 272, and an HDLC converter 272. Coupled to the HDLC 272 interface is a transceiver interface 274, and coupled to the transceiver interface 274 is a CSMA/CD unit 276. A transceiver unit 278 with an antenna 280 is coupled to the CSMA/CD unit 276.

The devices 272 and 276 are used for error correction and noise cancellation, as will be appreciated by those skilled in the art. The CSMA/CD detects if two packets have "collided" producing indecipherable noise. If so, no acknowledgment of the packet is sent by radio modem 62, and the senders of the two packets will wait a short random period before resending their packets. Since the waiting period is random, there is little likelihood that the packets will collide a second time. The HDLC performs a checksum on the received packets and, if the checksum fails, prevents the sending of the acknowledgment. This will cause the sending node to resend the packet after a random waiting period.

The currently used radio modems operate in the 902-928 MHz frequency range at about 725 mW, and have an outdoor range of up to 12 miles, line-of-sight. These characteristics are a good compromise for a light to moderately dense network. If the network becomes very dense, it may be preferable to reduce the power, since this will reduce the number of clients that hear a given packet. Also, other frequency ranges are also suitable, such as the 2.404 to 2.478 GHz range.

The currently sold Gína spread spectrum radio models have their transmission ("baud") rate artificially limited to 38.4 kHz. However, this artificial limit can be easily removed by a simple change to the program in PROM 262 to allow the modems to operate at 115.2 kHz, or nearly at full ISDN baud rates. At these baud rates, a single server can reasonably support three simultaneous WWW browser sessions and a dozen e-mail sessions. This compares very favorably to cellular networks which, as noted previously, can only support one user at a time. This also compares very favorably to the Ricochet system which, since it is limited to 28.8K baud, is not very useful for WWW browsing.

In FIG. 13, a client 18 including a computer 20 and a radio modem 22 of FIG. 1 is illustrated in greater detail. Again, the

6,044,062

19

client computer **20** can be any suitable form of digital processor including personal computer, workstation, PDA, etc. A computer **20** includes a microprocessor **282**, RAM **284**, and ROM **286**. The microprocessor is coupled to the RAM **284** and the ROM **286** by a memory bus **288**. The microprocessor **282** is also coupled to an input/output (I/O) bus **290** to which a number of peripherals **292** may be attached, including the radio modem **22**. As before, the radio modem **22** includes a control C portion and a radio R portion, where the control portion of the radio modem **22** is coupled to the I/O bus **290**. With brief reference to FIG. **12**, the control portion C is everything but the transceiver unit **278** and the antenna **280**, and the radio portion R corresponds to the transceiver unit **278**. Also, as before, the client process running on the client **18** can run on the computer **20**, in the control C portion of the modem **22**, or partially on both processors. The client **18** typically includes other peripherals **292** such as a removable media drive **94** receptive to removable media **296**, (such as a floppy disk or a CD ROM) and to a hard disk drive **298**. Those skilled in the design of computer system will readily understand how the hardware of client **18** is assembled and used.

In alternate embodiments of the present invention, uninterruptible power supplies and Global Positioning Systems (GPS) are added to the clients **18**. The uninterruptible power supplies ensure that the clients stay on the network, and the GPS can be used in conjunction with directional antennas (such as phased array antennas) attached to the radio modems **22** to direct the transmission to the desired next node in the link. This increases the efficiency of the system, and reduces packet "pollution" of the network. The GPS unit can be coupled to I/O bus **290**, or can be incorporated into the radio modem **22**.

In FIG. **14**, a client process **300** is implemented in the hardware of client **18**. Again, this process can run on the microprocessor **282**, or it can be partially or wholly run on the microprocessor of the controller C of the radio modem **22**. In this current preferred embodiment, the process **300** runs on the computer portion **20** of the client **18**. The client process **300** includes a client process control **302**, a process **304** for radio transmitting and receiving data packet, and a process **306** for maintaining a first-in-first-out (FIFO) buffer for send and receive data packets in the RAM **284** of the computer **20**.

In FIG. **15**, the process **304** of FIG. **14** is described in greater detail. The process **304** begins at **308** and, in a step **310**, it is determined whether the client is on the network. If not, the client needs to get on the network before it can send data to the server. This connection process begins at **312** to determine whether it is out of tries in trying to reach the server. If not, it sends a **01** packet in a step **314** and waits to receive a **02** packet from the server or another client in a step **316**. If it does not receive a **02** packet in response to **01** packet, process control is returned to step **312** until it runs out of server tries. When it does run out of server tries, process control is turned over to a step **318** which determines whether it is out of client tries. If yes, this particular client cannot reach either a server or another client and the process terminates at **320** with a failure. If it is not out of client tries in step **318**, a **03** packet is sent in a step **320** and the client waits to receive a **04** from another client in a step **322**. If a **04** is not received, the process control is returned to step **318** until they are out of client tries.

If a **02** is received in a step **316** or a **04** is received in a step **322**, then the client is in communication with the server or a client, respectively. In either instance, a step **324** stores the "link," i.e. the path to a server, whether it is direct to the

20

server or through one or more intermediate clients. Next, in a step **326**, a **05** is sent to the link, and a step **328** determines whether a **06** is returned. If not, the process is terminated as indicated at **320**. If a **06** has been received then a **07** is sent to the link in a step **330**, and a step **332** determines whether a **08** is returned. If not, a step **334** determines if they are out of tries, and if not, process control is returned to step **330** to send another **07** to the link. If after a certain number of tries, e.g. 3 tries, a **08** is not received in response to **07** transmitted by the client, the process terminates with a failure at a step **320**. If a **08** is received as determined by step **332**, a random check-in time is set in a step **336**. A random check-in time is set so that not all clients will try to check in with the server at the same time. Preferably, the random times will equally distribute the check-in times for the various clients equally within the aforementioned period INTERVAL. Finally, at this point, the client is connected into the network and the transmit/receive process is accomplished in a step **338**. Of course, if the client was on the network as determined by step **310**, the step **338** can be performed directly. The step **338** will be performed until there is a time-out of the transmit/receive process due to the round-robin scheduling by the client process control **302** (see FIG. **14**).

In FIG. **16**, the process **338** "Perform Transmit/Receive" is illustrated in greater detail. The process **338** has a transmit/receive process control **340** and three subprocesses **342**, **344**, and **346**. Again, time are allocated to the various subprocesses on a round-robin basis.

The subprocess **342** is the check-in routine where the client is required to check in on a periodic basis with the server to avoid being dropped from the server's routing list. As noted previously, the check-in start time is essentially random, and is within a given period INTERVAL. More particularly, the subprocess **342** begins with a decision **348** as to whether it is the proper time to check-in. If not, process control is immediately returned to process control **340**. If it is check-in time, a **07** is sent to the server. If a **08** is received from the server, all is well and process control is returned to process control **340**. If the expected **08** is not received, decision step **354** determines if there are any more tries. Typically, at least three tries will be allowed. If there are more tries, process control is returned to step **350**. If there aren't any more tries, a step **356** will authenticate and send an **11** to the left son of the client that the client is removing itself from the network. Authentication prevents the situation where a "promiscuous" spooler could masquerade as a client and transmit an "11" packet with downstream client addresses, thereby disconnecting those downstream clients from the network. The client then marks itself as being disconnected or "off" of the network in a step **358**, and process control is returned to process control **340**.

In FIG. **17**, the process **344** "Process Computer Received Packets" is shown in flow diagram form. The process **344** begins at **360** and, in a step **362**, the data is obtained from a buffer. Next, in a step **364**, the header is added to the data, including the link and the packet type "14" to indicate that this is a data-type data packet. Next, the data packet, complete with header, is transmitted. The process is completed at **368**.

FIG. **18** illustrates the process **346** "Process Radio Received Packets" of FIG. **16** in greater detail. The process **346** begins at **370** and, in a step **373**, determines if the received packet is for it. If yes, a step **374** will process the packet per the code type, as will be discussed in greater detail subsequently. Then, a step **376** determines if the father node of the client has been marked. Not, a new, shorter link is created since the packet was received without being

6,044,062

21

relayed by the father node. If the father node has been marked, or after a new link has been created, the process terminates at **380**.

If step **372** determines that it is not that client's packet, a step **382** determines if that client is on the route for the packet. If yes, a step **384** tests to see if the client is marked. If it is marked, it has already sent that packet and the process is completed at **380**. If the client hasn't been marked, it marks itself in the header of the data packet and transmits the packet in a step **386**. Process control is then given to step **376** to see if the client's link can be upgraded as discussed previously.

If step **382** determines that the packet is not for that client, and that the client is not part of the link, steps **388–392** still analyze the packet in a process known as "pooning." Since this client can hear this packet, there is an opportunity to upgrade its link. Step **388** determines whether the link to the last marked node plus one (i.e. the distance to the first unmarked node) is shorter than its own link. This is because this client is listening to the last marked node, and the number of hops through that last marked node is the number of hops of that last marked node plus one. If it is, the client's link is updated in a step **392** to this shorter link. If not, the alternative route is cached in case the client's current link becomes inoperative. Therefore, in the pooning process, the client listens to all packets to continuously and dynamically update its link to the best possible path.

In FIG. **18A**, a data packet **394** of the present invention includes a header portion **396** including a link section **398** and a data type section **400**, and a data portion **402**. The link **398** indicates that the destination of this data packet is the node P. The two digit data type **400** indicates what type of data is being sent, and the data field **402** includes the actual data and is terminated within EOD (end of data) marker. This packet corresponds to the tree of FIG. **9a**. Since all upstream nodes (i.e. nodes Q, Z, **3**, and X) are marked with asterisks ("*"), it is known that the data packet has passed through and has been marked by each of these nodes before reaching the node P. If, however, the data packet **394'** of FIG. **18B** is received where in only nodes X and **3** are marked, this means that the node **3** can hear the transmission of node (client) **3** directly. In this instance, there is no need to go through nodes Q and Z to reach the server X. As a result, the new, upgraded link is from node P to node **3** to the server X. This is represented by the notation: X(**3**(P)).

The table of FIG. **19** is used to illustrate the "Process Per Type Code" step **384** of FIG. **18**. The table of FIG. **19** includes three columns **404**, **406**, and **408**. The first column **404**, lists the codes that can be received. These codes corresponds to the 2 byte code **400** of the data packet **394** of FIG. **18A**. The second column **406** corresponds to the server responses to receiving such codes, and the third column **408** are the client responses to receiving the codes. We will now discuss each of the codes, in sequence.

When the server receives a **01** code, its response is a **02** code plus a one-way seed as discussed previously. Since a **01** code is never intended for a client, it will ignore or "drop" the **01** coded data packets.

For the **02**, **03**, and **04** codes, the server will ignore or drop those data packets because these data packets are only intended for clients. If a client receives a **02**, it responds with a **05** and a one-way response. In response to a **03**, a client will send a **04** and a seed or a null. In response to a **04**, the client will send a **05** and a one-way seed. Again, one-way seeds and responses to one-way seeds were discussed previously.

22

When a server receives a **05**, if it has previously sent a **02** and if the **05** is authentic, then it will send a **06**. Otherwise, it will drop the packet. When a client receives a **05**, if it had previously sent a **04**, and if the **05** is authentic, then it sends a **06**. Otherwise, the client will drop the data packet. If the server receives a **06**, it will drop the data packet. If a client receives a **06** after it sent a **05**, then it will send a **07**. Otherwise, it will drop the packet as well.

When a **07** is received from the server, it will immediately respond with a **08**. Since **07** coded packets are never intended for clients, it will be dropped.

Data packets coded with an **08**, **09**, **10**, or **11** are all dropped if received by a server. If a client receives a **08**, it will update the tree or repeat the data. In response to a **09**, a client will send a **10**. In response to a **10**, a client will update the tree or repeat the data. In response to a type **11**, it send an **11** to the left son with the address the departing node plus a **01** to reconnect to the network.

Data packets of type **12** and **86** are currently reserved. In response to a data packet type **13**, a server will delete the sender. Since this is a server destination data packet only, if a client receives a data packet of type **13**, it will drop the data packet.

Finally, if a server receives a data packet of type **14**, it will send it to the network transmit buffer. If a client receives a data packet of type **14**, it will send it to the computer transmit buffer.

FIG. **20** illustrates an initialization routine which connects a client CB to a server S through another client CA. The sequence is as follows. As indicated by arrow a, client CB sends a **03** to client CA. In return, the client CA sends a **04** and a seed back to client CB as indicated by arrow b. Client CB then sends a **05** and a one-way response as indicated by arrow c to client CA, and client CA sends a **06** and an acknowledgment with a **05** to client CD as indicated by arrow d. Then, client CB sends a **09** to client CA as indicated by arrow e, and client CA sends a **10** and the link to the client CB as indicated by arrow f. Client CB then sends a **07** and the neighbor's addresses to the client CA as indicated by arrow g, and a client CA relays the **07** and the neighbor's address to the server S as indicated by arrow g'. The server S then sends a **08** and the tree to the client CA as indicated by arrow h, and the client CA relays the **08** and the tree to the client CB as indicated by the arrow h'. At this point, the client CB has the link to the server S and the complete tree of the network in its memory.

FIGS. **21a–21d** illustrate a portion of the server process which deals with determining a return path from a received data packet at a server. Assume, for example, the tree is known to the server is as illustrated in FIG. **21a**. This is the same tree as was illustrated in an example of FIGS. **9a** and **9b**. Then, assume that the server X receives the packet from a client P as illustrated in FIG. **21b**. The simplest way of determining the reverse address is simply reverse the link section of the header portion of the data packet of FIG. **21b** to provide a return address of **21c**. However, if the part of the address of the header of the data packet of FIG. **21b** has been lost or corrupted during the transition process, the tree of FIG. **21a** can be used to reconstruct the return path. This is accomplished by jumping from parent to parent in reverse order as indicated to determine the return path. In this example, the reverse order parent jumping indicates that the original path the server X was P>Q>Z>**3**>X, which, when reversed, gives us the proper reverse path, namely X(**3**(Z(Q(P))))). As will be appreciated by those skilled in the art, this type of reverse tree traversal is easily accomplished with a recursive function.

6,044,062

23

While this invention has been described in terms of several preferred embodiments, there are alterations, permutations, and equivalents which fall within the scope of this invention. It should also be noted that there are many alternative ways of implementing both the process and apparatus of the present invention. It is therefore intended that the following appended claims be interpreted as including all such alterations, permutations, and equivalents as fall within the true spirit and scope of the present invention.

We claim:

1. A wireless network system comprising:

a server including a server controller and a server radio modem, said server controller implementing a server process that includes the control of said server radio modem, said server process including the receipt and transmission of data packets via said server radio modem;

a plurality of clients each including a client controller and a client radio modem, said client controller implementing a client process that includes the control of said client radio modem, said client process including the receipt and transmission of data packets via said client radio modem, wherein said client process of each of said clients initiates and selects a radio transmission path to said server that is one of a direct link to said server and an indirect link to said server through at least one the remainder of said plurality of clients; and

wherein said selected link to said server is a link utilizing the least number of other clients,

whereby said transmission path from said client to said server is optimized.

2. A wireless network system comprising:

a server including a server controller and a server radio modem, said server controller implementing a server process that includes the control of said server radio modem, said server process including the receipt and transmission of data packets via said server radio modem; and

a plurality of clients each including a client controller and a client radio modem, said client controller implementing a client process that includes the control of said client radio modem, said client process including the receipt and transmission of data packets via said client radio modem, wherein said client process of each of said clients initiates and selects a radio transmission path to said server that is one of a direct link to said server and an indirect link to said server through at least one the remainder of said plurality of clients,

wherein said server process further includes logic that maintains a client link tree having client link entries.

3. A wireless network system as recited in claim 2, wherein said server process further comprises:

logic that compares a selected link from said client to said server to a current client link entry in said client link tree; and

logic that updates said client link tree when said comparison meets predetermined conditions.

4. A wireless network system as recited in claim 3, wherein said server process further comprises:

logic that determines if said client is authentic;

logic that determines if said client is already in said client link tree if client is determined to be authentic;

logic that deletes said client from said client link tree if said client is already in said client link tree; and

logic that inserts said client in said client link tree if said client is authentic.

24

5. A wireless network system comprising:

server means providing a server process including receiving data packets via a server wireless communication means, sending data packets via said wireless communication means, communicating with a network, and performing housekeeping functions;

a plurality of client means, each client means providing a client process including sending and receiving data packet via a client wireless communication means, maintaining a send/receive data buffer in digital memory, and selecting a link to said server means that is one of a direct link to said server and an indirect link to said server through at least one of the remainder of said plurality of client means; and

wherein said selected link to said server is a link utilizing the least number of other clients,

whereby said transmission path from said client to said server is optimized.

6. A wireless network system comprising:

server means providing a server process including receiving data packets via a server wireless communication means, sending data packets via said wireless communication means, communicating with a network, and performing housekeeping functions; and

a plurality of client means, each client means providing a client process including sending and receiving data packet via a client wireless communication means, maintaining a send/receive data buffer in digital memory, and selecting a link to said server means that is one of a direct link to said server and an indirect link to said server through at least one of the remainder of said plurality of client means,

wherein said server process further comprises a means for maintaining a client link tree having client link entries.

7. A wireless network system as recited in claim 6, wherein said server process further comprises:

a means for comparing a selected link from said client to said server to a current client link entry in said client link tree; and

a means for updating said client link tree when said comparison meets predetermined conditions.

8. A wireless network system as recited in claim 7, wherein said server process further includes:

a means for determining if said client is authentic;

a means for determining if said client is already in said client link tree if client is determined to be authentic;

a means for deleting said client from said client link tree if said client is already in said client link tree; and

a means for inserting said client in said client link tree if said client is authentic.

9. A method for providing wireless network communication comprising:

providing a server implementing a server process including receiving data packets via r.f. transmission, sending data packets via r.f. transmission, communicating with a network, and performing housekeeping functions; and

providing a plurality of clients, each client providing a client process including sending and receiving data packet via r.f. transmission, maintaining a send/receive data buffer in digital memory, and selecting a transmission path to said server via that is one of a direct link to said server and an indirect link to said server through at least one of the remainder of said plurality of clients; and

25

wherein said selected link to said server is a link utilizing the least number of other clients,

whereby said transmission path from said client to said server is optimized.

10. A method for providing wireless network communication comprising:

providing a server implementing a server process including receiving data packets via r.f. transmission, sending data packets via r.f. transmission, communicating with a network, and performing housekeeping functions; and

providing a plurality of clients, each client providing a client process including sending and receiving data packet via r.f. transmission, maintaining a send/receive data buffer in digital memory, and selecting a transmission path to said server via that is one of a direct link to said server and an indirect link to said server through at least one of the remainder of said plurality of clients, wherein said server process further includes maintaining a client link tree having client link entries.

11. A method as recited in claim 10, wherein said server process further includes:

comparing a selected link from said client to said server to a current client link entry in said client link tree; and updating said client link tree when said comparison meets predetermined conditions.

12. A method as recited in claim 11, wherein said server process further includes:

determining if said client is authentic; determining if said client is already in said client link tree if client is determined to be authentic; deleting said client from said client link tree if said client is already in said client link tree; and inserting said client in said client link tree if said client is authentic.

13. A method for providing wireless network communication comprising the steps of:

a server process including a data packet reception step, a data packet transmission step, a network communication step, and a housekeeping step; and

26

a plurality of clients each providing a client process including a data sending and receiving step, a send and receive data buffer maintenance step, and a link selection step that is one of a direct link to a server and an indirect link to said server through at least one of the remainder of said plurality of clients; and

wherein said selected link to said server is a link utilizing the least number of other clients,

whereby said transmission path from said client to said server is optimized.

14. A method for providing wireless network communication comprising the steps of:

a server process including a data packet reception step, a data packet transmission step, a network communication step, and a housekeeping step; and

a plurality of clients each providing a client process including a data sending and receiving step, a send and receive data buffer maintenance step, and a link selection step that is one of a direct link to a server and an indirect link to said server through at least one of the remainder of said plurality of clients,

wherein said server process further comprising the step of maintaining a client link tree having client link entries.

15. A method as recited in claim 14, wherein said server process further comprises the steps of:

comparing a selected link from said client to said server to a current client link entry in said client link tree; and updating said client link tree when said comparison meets predetermined conditions.

16. A method as recited in claim 15, wherein said server process further comprises steps of:

determining if said client is authentic; determining if said client is already in said client link tree if client is determined to be authentic; deleting said client from said client link tree if said client is already in said client link tree; and inserting said client into said client link tree if said client is authentic.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 6,044,062
DATED : March 28, 2000
INVENTOR(S) : Brownrigg et al.

Page 1 of 2

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Title page.

Item [57], **ABSTRACT**,

Line 10, reads "maintains a radio transmission path to the server that is either"; it should read -- maintains a radio transmission path to the server that is either --;

Column 2.

Line 39, reads "modem of a particular personal computer in radio transmis-"; it should read -- modem of a particular personal computer is in radio transmis- --;

Column 3.

Line 30, reads " "Control Using Pacing in a Packet Radio Network", N."; it should read -- "Congestion Control Using Pacing in a Packet Radio Network", N. --;

Column 5.

Line 11, reads "each including a client controller and a client radio modern"; it should read -- each including a client controller and a client radio modem --;

Column 13.

Line 51, reads "those skilled in the art However, it should be noted that at"; it should read -- those skilled in the art. However, it should be noted that at --;

Column 17.

Line 60, reads "desired internal set by a network manager. Process control"; it should read -- desired interval set by a network manager. Process control --;

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 6,044,062
DATED : March 28, 2000
INVENTOR(S) : Brownrigg et al.

Page 2 of 2

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 26.

Line 28, reads "to a correct client link entry in said client link tree; and"; it should read -- to a current client link entry in said client link tree; and --;

Signed and Sealed this

Twenty-first Day of May, 2002

Attest:

A handwritten signature in black ink, appearing to read "James E. Rogan", with a horizontal line drawn underneath it.

Attesting Officer

JAMES E. ROGAN
Director of the United States Patent and Trademark Office

(12) **EX PARTE REEXAMINATION CERTIFICATE** (8193rd)
United States Patent
Brownrigg et al.

(10) **Number:** US 6,044,062 C1
 (45) **Certificate Issued:** May 3, 2011

(54) **WIRELESS NETWORK SYSTEM AND METHOD FOR PROVIDING SAME**

FOREIGN PATENT DOCUMENTS

(75) Inventors: **Edwin B. Brownrigg**, Roseville, CA (US); **Thomas W. Wilson**, Alameda, CA (US)

EP	0 483 547 A1	5/1992
EP	0 578 041 B1	1/1994
EP	0 663 746 B1	7/1995
EP	0718954 A1	6/1996
EP	0 740 873 B1	11/1996

(73) Assignee: **IPCO, LLC**, Atlanta, GA (US)

(Continued)

Reexamination Request:

No. 90/008,011, Apr. 17, 2006

OTHER PUBLICATIONS

Office Action dated Jun. 4, 2008, in co-pending U.S. Appl. No. 11/300,902, filed Dec. 15, 2005.

Reexamination Certificate for:

Patent No.: **6,044,062**
 Issued: **Mar. 28, 2000**
 Appl. No.: **08/760,895**
 Filed: **Dec. 6, 1996**

(Continued)

Primary Examiner—Scott L. Weaver

(57) **ABSTRACT**

A wireless network system includes a server having a server controller and a server radio modem, and a number of clients each including a client controller and a client radio modem. The server controller implements a server process that includes the receipt and the transmission of data packets via the radio modem. The client controllers of each of the clients implements a client process that includes the receipt and transmission of data packets via the client radio modem. The client process of each of the clients initiates, selects, and maintains a radio transmission path to the server that is either a direct path to the server, or is an indirect path or “link” to the server through at least one of the remainder of the clients. A method for providing wireless network communication includes providing a server implementing a server process including receiving data packets via a radio modem, sending data packets via the server radio modem, communicating with the network, and performing housekeeping functions, and further includes providing a number of clients, each implementing a client process sending and receiving data packets via a client radio modem, maintaining a send/receive data buffer, and selecting a radio transmission path to the server. The radio transmission path or “link” is either a direct path to the server, or an indirect path to the server through at least one of the remainder of the clients. The process preferably optimizes the link to minimize the number of “hops” to the server.

Certificate of Correction issued May 21, 2002.

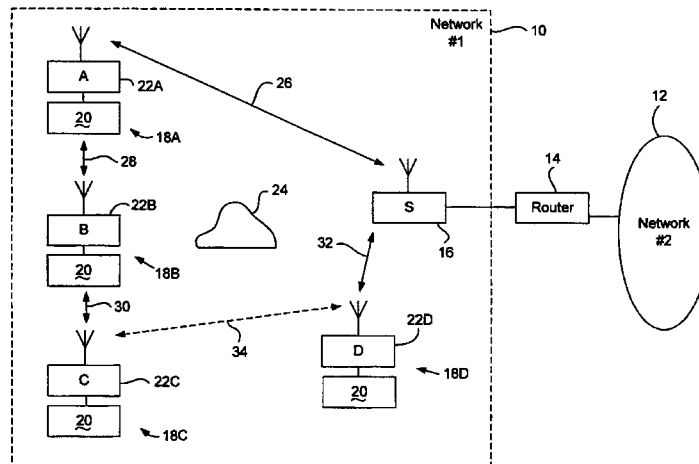
- (51) **Int. Cl.**
H04L 29/06 (2006.01)
H04Q 7/22 (2006.01)
- (52) **U.S. Cl.** **370/238**; 370/315; 455/11.1; 455/445
- (58) **Field of Classification Search** None
 See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

3,575,558 A	4/1971	Leyburn et al.
3,665,475 A	5/1972	Gram
3,705,385 A	12/1972	Batz
3,723,876 A	3/1973	Seaborn, Jr.
3,742,142 A	6/1973	Martin
3,848,231 A	11/1974	Wootton
3,892,948 A	7/1975	Constable
3,906,460 A	9/1975	Halpern
3,914,692 A	10/1975	Seaborn, Jr.
3,922,492 A	11/1975	Lumsden

(Continued)



US 6,044,062 C1

Page 2

U.S. PATENT DOCUMENTS						
			4,967,366	A	10/1990	Kaehler
			4,968,970	A	11/1990	LaPorte
			4,968,978	A	11/1990	Stolarczyk
			4,972,504	A	11/1990	Daniel, Jr. et al.
			4,973,957	A	11/1990	Shimizu et al.
			4,973,970	A	11/1990	Reeser
			4,977,612	A	12/1990	Wilson
			4,980,907	A	12/1990	Raith et al.
			4,987,536	A	1/1991	Humblet
			4,989,230	A	1/1991	Gillig et al.
			4,993,059	A	2/1991	Smith et al.
			5,007,052	A	4/1991	Flammer
			5,079,768	A	1/1992	Flammer
			5,088,032	A	2/1992	Bosack
			5,115,433	A	5/1992	Baran et al.
			5,117,422	A	5/1992	Hauptschein et al.
			5,130,987	A	7/1992	Flammer
			5,138,615	A	8/1992	Lamport et al.
			5,142,694	A	8/1992	Jackson et al.
			5,159,592	A	10/1992	Perkins
			5,170,393	A	12/1992	Peterson et al.
			5,216,502	A	6/1993	Katz
			5,221,838	A	6/1993	Gutman et al.
			5,223,844	A	6/1993	Mansell et al.
			5,231,658	A	7/1993	Eftechiou
			5,235,630	A	8/1993	Moody et al.
			5,239,294	A	8/1993	Flanders et al.
			5,239,575	A	8/1993	White et al.
			5,241,410	A	8/1993	Streck et al.
			5,243,338	A	9/1993	Brennan, Jr. et al.
			5,245,633	A	9/1993	Schwartz et al.
			5,252,967	A	10/1993	Brennan et al.
			5,253,167	A	10/1993	Yoshida et al.
			5,265,150	A	11/1993	Helmkamp et al.
			5,265,162	A	11/1993	Bush et al.
			5,266,782	A	11/1993	Alanara et al.
			5,272,747	A	12/1993	Meads
			5,276,680	A	1/1994	Messenger
			5,282,250	A	1/1994	Dent et al.
			5,289,165	A	2/1994	Belin
			5,289,362	A	2/1994	Liebl et al.
			5,291,516	A	3/1994	Dixon et al.
			5,295,154	A	3/1994	Meier et al.
			5,305,370	A	4/1994	Kearns et al.
			5,309,501	A	5/1994	Kozik et al.
			5,315,645	A	5/1994	Matheny
			5,317,309	A	5/1994	Vercellotti et al.
			5,319,364	A	6/1994	Waraksa et al.
			5,319,698	A	6/1994	Glidewell et al.
			5,319,711	A	6/1994	Servi
			5,323,384	A	6/1994	Norwood et al.
			5,325,429	A	6/1994	Kurgan
			5,329,394	A	7/1994	Calvani et al.
			5,331,318	A	7/1994	Montgomery
			5,334,974	A	8/1994	Simms et al.
			5,335,265	A	8/1994	Cooper et al.
			5,343,493	A	8/1994	Karimullah
			5,345,231	A	9/1994	Koo et al.
			5,345,595	A	9/1994	Johnson et al.
			5,347,263	A	9/1994	Carroll et al.
			5,352,278	A	10/1994	Korver et al.
			5,354,974	A	10/1994	Eisenberg
			5,355,513	A	10/1994	Clarke et al.
			5,365,217	A	11/1994	Toner
			5,371,736	A	12/1994	Evan
			5,382,778	A	1/1995	Takahira et al.
			5,383,134	A	1/1995	Wrzesinski
			5,383,187	A	1/1995	Vardakas et al.
			5,390,206	A	2/1995	Rein et al.
			5,406,619	A	4/1995	Akhteruzzaman et al.
			5,412,192	A	5/1995	Hoss

US 6,044,062 C1

Page 3

5,412,654 A	5/1995	Perkins	5,596,722 A	1/1997	Rahnema
5,412,760 A	5/1995	Peitz	5,602,843 A	2/1997	Gray
5,416,475 A	5/1995	Tolbert et al.	5,604,414 A	2/1997	Milligan et al.
5,418,812 A	5/1995	Reyes et al.	5,604,869 A	2/1997	Mincher et al.
5,420,910 A	5/1995	Rudokas et al.	5,606,361 A	2/1997	Davidsohn et al.
5,424,708 A	6/1995	Ballesty et al.	5,608,721 A	3/1997	Natarajan et al.
5,430,729 A	7/1995	Rahnema	5,608,786 A	3/1997	Gordon
5,432,507 A	7/1995	Mussino et al.	5,613,620 A	3/1997	Center et al.
5,438,329 A	8/1995	Gastouniotis et al.	5,615,227 A	3/1997	Schumacher, Jr. et al.
5,439,414 A	8/1995	Jacob	5,617,084 A	4/1997	Sears
5,440,545 A	8/1995	Buchholz et al.	5,623,495 A	4/1997	Eng et al.
5,442,553 A	8/1995	Parrillo	5,625,410 A	4/1997	Washino et al.
5,442,633 A	8/1995	Perkins et al.	5,628,050 A	5/1997	McGraw et al.
5,445,287 A	8/1995	Center et al.	5,629,687 A	5/1997	Sutton et al.
5,445,347 A	8/1995	Ng	5,629,875 A	5/1997	Adair, Jr.
5,451,929 A	9/1995	Adelman et al.	5,630,209 A	5/1997	Wizgall et al.
5,451,938 A	9/1995	Brennan, Jr.	5,631,554 A	5/1997	Briese et al.
5,452,344 A	9/1995	Larson	5,636,216 A	6/1997	Fox et al.
5,454,024 A	9/1995	Lebowitz	5,640,002 A	6/1997	Ruppert et al.
5,455,569 A	10/1995	Sherman et al.	5,644,294 A	7/1997	Ness
5,465,401 A	11/1995	Thompson	5,657,389 A	8/1997	Houvener
5,467,074 A	11/1995	Pedtko	5,659,300 A	8/1997	Dresselhuys et al.
5,467,082 A	11/1995	Sanderson	5,659,303 A	8/1997	Adair, Jr.
5,467,345 A	11/1995	Cutler, Jr. et al.	5,668,876 A	9/1997	Falk et al.
5,468,948 A	11/1995	Koenck et al.	5,673,252 A	9/1997	Johnson et al.
5,471,201 A	11/1995	Cerami et al.	5,673,304 A	9/1997	Connor et al.
5,473,322 A	12/1995	Carney	5,673,305 A	9/1997	Ross
5,475,689 A	12/1995	Kay et al.	5,682,139 A	10/1997	Pradeep et al.
5,479,400 A	12/1995	Dilworth et al.	5,682,476 A	10/1997	Tapperson et al.
5,481,259 A	1/1996	Bane	5,689,229 A	11/1997	Chaco et al.
5,481,532 A	1/1996	Hassan et al.	5,691,980 A	11/1997	Welles, II et al.
5,484,997 A	1/1996	Haynes	5,696,695 A	12/1997	Ehlers et al.
5,488,608 A	1/1996	Flammer	5,699,328 A	12/1997	Ishizaki et al.
5,493,273 A	2/1996	Smurlo et al.	5,701,002 A	12/1997	Oishi et al.
5,493,287 A	2/1996	Bane	5,702,059 A	12/1997	Chu et al.
5,502,726 A	3/1996	Fischer	5,704,046 A	12/1997	Hogan
5,504,746 A	4/1996	Meier	5,704,517 A	1/1998	Lancaster, Jr.
5,506,837 A	4/1996	Sollner et al.	5,706,976 A	1/1998	Purkey
5,508,412 A	4/1996	Kast et al.	5,708,223 A	1/1998	Wyss
5,509,073 A	4/1996	Monnin	5,708,655 A	1/1998	Toth et al.
5,513,244 A	4/1996	Joao et al.	5,712,619 A	1/1998	Simkin
5,515,419 A	5/1996	Sheffer	5,712,980 A	1/1998	Beeler et al.
5,517,188 A	5/1996	Carroll et al.	5,714,931 A	2/1998	Petite et al.
5,522,089 A	5/1996	Kikinis et al.	5,717,718 A	2/1998	Rowell et al.
5,528,215 A	6/1996	Siu et al.	5,726,534 A	3/1998	Seo
5,528,507 A	6/1996	McNamara et al.	5,726,544 A	3/1998	Lee
5,539,825 A	7/1996	Akiyama et al.	5,726,644 A	3/1998	Jednacz et al.
5,541,938 A	7/1996	Di Zenzo et al.	5,726,984 A	3/1998	Kubler et al.
5,542,100 A	7/1996	Hatakeyama	5,732,074 A	3/1998	Spaur et al.
5,544,784 A	8/1996	Malaspina	5,732,078 A	3/1998	Arango
5,548,632 A	8/1996	Walsh et al.	5,736,965 A	4/1998	Mosebrook et al.
5,550,358 A	8/1996	Tait et al.	5,737,318 A	4/1998	Melnik
5,550,359 A	8/1996	Bennett	5,740,232 A	4/1998	Pailles et al.
5,550,535 A	8/1996	Park	5,740,366 A	4/1998	Mahany et al.
5,553,094 A	9/1996	Johnson et al.	5,742,509 A	4/1998	Goldberg et al.
5,555,258 A	9/1996	Snelling et al.	5,745,849 A	4/1998	Britton
5,555,286 A	9/1996	Tendler	5,748,619 A	5/1998	Meier
5,557,320 A	9/1996	Krebs	5,754,111 A	5/1998	Garcia
5,557,748 A	9/1996	Norris	5,754,227 A	5/1998	Fukuoka
5,562,537 A	10/1996	Zver et al.	5,757,783 A	5/1998	Eng et al.
5,565,857 A	10/1996	Lee	5,757,788 A	5/1998	Tatsumi et al.
5,568,535 A	10/1996	Sheffer et al.	5,760,742 A	6/1998	Branch et al.
5,570,084 A	10/1996	Ritter et al.	5,761,083 A	6/1998	Brown, Jr. et al.
5,572,528 A	11/1996	Shuen	5,764,742 A	6/1998	Howard et al.
5,573,181 A	11/1996	Ahmed	5,767,791 A	6/1998	Stoop et al.
5,574,111 A	11/1996	Brichta et al.	5,771,274 A	6/1998	Harris
5,583,914 A	12/1996	Chang et al.	5,774,052 A	6/1998	Hamm et al.
5,588,005 A	12/1996	Ali et al.	5,781,143 A	7/1998	Rossin
5,589,878 A	12/1996	Cortjens et al.	5,790,644 A	8/1998	Kikinis
5,590,038 A	12/1996	Pitroda	5,790,662 A	8/1998	Valerij et al.
5,596,719 A	1/1997	Ramakrishnan et al.	5,796,727 A	8/1998	Harrison et al.

US 6,044,062 C1

Page 4

5,798,964 A	8/1998	Shimizu et al.	5,969,608 A	10/1999	Sojdehei et al.
5,801,643 A	9/1998	Williams et al.	5,973,756 A	10/1999	Erlin
5,812,531 A	9/1998	Cheung et al.	5,974,236 A	10/1999	Sherman
5,815,505 A	9/1998	Mills	5,978,364 A	11/1999	Melnik
5,818,822 A	10/1998	Thomas et al.	5,978,578 A	11/1999	Azarya et al.
5,822,273 A	10/1998	Bary et al.	5,987,011 A	11/1999	Toh
5,822,309 A	10/1998	Ayanoglu et al.	5,987,421 A	11/1999	Chuang
5,822,544 A	10/1998	Chaco et al.	5,991,625 A	11/1999	Vanderpool
5,825,772 A	10/1998	Dobbins et al.	5,991,639 A	11/1999	Rautiola et al.
5,826,195 A	10/1998	Westerlage et al.	5,995,022 A	11/1999	Plis et al.
5,828,044 A	10/1998	Jun et al.	5,995,592 A	11/1999	Shirai et al.
5,832,057 A	11/1998	Furman	5,995,593 A	11/1999	Cho
5,838,223 A	11/1998	Gallant et al.	5,997,170 A	12/1999	Brodbeck
5,838,237 A	11/1998	Revell et al.	5,999,094 A	12/1999	Nilssen
5,838,812 A	11/1998	Pare, Jr. et al.	6,005,759 A	12/1999	Hart et al.
5,841,118 A	11/1998	East et al.	6,005,963 A	12/1999	Bolle et al.
5,841,764 A	11/1998	Roderique et al.	6,018,659 A	1/2000	Ayyagari et al.
5,842,976 A	12/1998	Williamson	6,021,664 A	2/2000	Granato et al.
5,844,808 A	12/1998	Konsmo et al.	6,023,223 A	2/2000	Baxter, Jr.
5,845,230 A	12/1998	Lamberson	6,028,522 A	2/2000	Petite
5,848,054 A	12/1998	Mosebrook et al.	6,028,857 A	2/2000	Poor
5,852,658 A	12/1998	Knight et al.	6,031,455 A	2/2000	Grube et al.
5,854,994 A	12/1998	Canada et al.	6,032,197 A	2/2000	Birdwell et al.
5,856,974 A	1/1999	Gervais et al.	6,035,213 A	3/2000	Tokuda et al.
5,862,201 A	1/1999	Sands	6,035,266 A	3/2000	Williams et al.
5,864,772 A	1/1999	Alvarado et al.	6,036,086 A	3/2000	Sizer, II et al.
5,870,686 A	2/1999	Monson	6,038,491 A	3/2000	McGarry et al.
5,872,773 A	2/1999	Katzela et al.	6,044,062 A	3/2000	Brownrigg et al.
5,873,043 A	2/1999	Comer	6,046,978 A	4/2000	Melnik
5,874,903 A	2/1999	Shuey et al.	6,054,920 A	4/2000	Smith et al.
5,875,185 A	2/1999	Wang et al.	6,055,561 A	4/2000	Feldman et al.
5,880,677 A	3/1999	Lestician	6,060,994 A	5/2000	Chen
5,884,184 A	3/1999	Sheffer	6,064,318 A	5/2000	Kirchner et al.
5,884,271 A	3/1999	Pitroda	6,067,017 A	5/2000	Stewart et al.
5,886,333 A	3/1999	Miyake	6,067,030 A	5/2000	Burnett et al.
5,889,468 A	3/1999	Banga	6,069,886 A	5/2000	Ayerst et al.
5,892,690 A	4/1999	Boatman et al.	6,073,169 A	6/2000	Shuey et al.
5,892,924 A	4/1999	Lyon et al.	6,073,840 A	6/2000	Marion
5,896,097 A	4/1999	Cardozo	6,075,451 A	6/2000	Lebowitz et al.
5,898,369 A	4/1999	Godwin	6,078,251 A	6/2000	Landt et al.
5,898,733 A	4/1999	Satyanarayana	6,084,867 A	7/2000	Meier
5,905,438 A	5/1999	Weiss et al.	6,094,622 A	7/2000	Hubbard et al.
5,905,442 A	5/1999	Mosebrook et al.	6,097,703 A	8/2000	Larsen et al.
5,907,291 A	5/1999	Chen et al.	6,100,817 A	8/2000	Mason, Jr. et al.
5,907,491 A	5/1999	Canada et al.	6,101,427 A	8/2000	Yang
5,907,540 A	5/1999	Hayashi	6,101,445 A	8/2000	Alvarado et al.
5,907,807 A	5/1999	Chavez, Jr. et al.	6,112,983 A	9/2000	D'Anniballe et al.
5,909,429 A	6/1999	Satyanarayana et al.	6,115,393 A	9/2000	Engel et al.
5,914,673 A	6/1999	Jennings et al.	6,115,580 A	9/2000	Chuprun et al.
5,917,405 A	6/1999	Joao	6,119,076 A	9/2000	Williams et al.
5,917,629 A	6/1999	Hortensius et al.	6,121,885 A	9/2000	Masone et al.
5,923,269 A	7/1999	Shuey et al.	6,122,759 A	9/2000	Ayanoglu et al.
5,926,101 A	7/1999	Dasgupta	6,124,806 A	9/2000	Cunningham et al.
5,926,103 A	7/1999	Petite	6,127,917 A	10/2000	Tuttle
5,926,529 A	7/1999	Hache et al.	6,128,551 A	10/2000	Davis et al.
5,926,531 A	7/1999	Petite	6,130,622 A	10/2000	Hussey et al.
5,933,073 A	8/1999	Shuey	6,137,423 A	10/2000	Glorioso et al.
5,940,771 A	8/1999	Gollnick et al.	6,140,975 A	10/2000	Cohen
5,941,363 A	8/1999	Partyka et al.	6,141,347 A	10/2000	Shaughnessy et al.
5,941,955 A	8/1999	Wilby et al.	6,150,936 A	11/2000	Addy
5,946,631 A	8/1999	Melnik	6,150,955 A	11/2000	Tracy et al.
5,948,040 A	9/1999	DeLorme et al.	6,157,464 A	12/2000	Bloomfield et al.
5,949,799 A	9/1999	Grivna et al.	6,157,824 A	12/2000	Bailey
5,953,319 A	9/1999	Dutta et al.	6,167,239 A	12/2000	Wright et al.
5,953,371 A	9/1999	Rowell et al.	6,172,616 B1	1/2001	Johnson et al.
5,953,507 A	9/1999	Cheung et al.	6,173,159 B1	1/2001	Wright et al.
5,957,718 A	9/1999	Cheng et al.	6,174,205 B1	1/2001	Madsen et al.
5,960,074 A	9/1999	Clark	6,175,922 B1	1/2001	Wang
5,963,146 A	10/1999	Johnson et al.	6,177,883 B1	1/2001	Jennetti et al.
5,963,452 A	10/1999	Etoh et al.	6,181,255 B1	1/2001	Crimmins et al.
5,966,658 A	10/1999	Kennedy, III et al.	6,181,284 B1	1/2001	Madsen et al.

US 6,044,062 C1

Page 5

6,181,981 B1	1/2001	Varga et al.	6,438,575 B1	8/2002	Khan et al.
6,188,354 B1	2/2001	Soliman et al.	6,445,291 B2	9/2002	Addy et al.
6,188,675 B1	2/2001	Casper et al.	6,456,960 B1	9/2002	Williams et al.
6,192,282 B1	2/2001	Smith et al.	6,457,038 B1	9/2002	Defosse
6,192,390 B1	2/2001	Berger et al.	6,462,644 B1	10/2002	Howell et al.
6,195,018 B1	2/2001	Ragle et al.	6,462,672 B1	10/2002	Besson
6,198,390 B1	3/2001	Schlager et al.	6,477,558 B1	11/2002	Irving et al.
6,199,068 B1	3/2001	Carpenter	6,483,290 B1	11/2002	Hemminger et al.
6,201,962 B1	3/2001	Sturniolo et al.	6,484,939 B1	11/2002	Blaeuer
6,205,143 B1	3/2001	Lemieux	6,489,884 B1	12/2002	Lamberson et al.
6,208,247 B1	3/2001	Agre et al.	6,491,828 B1	12/2002	Sivavec et al.
6,208,266 B1	3/2001	Lyons et al.	6,492,910 B1	12/2002	Ragle et al.
6,212,175 B1	4/2001	Harsch	6,496,696 B1	12/2002	Melnik
6,218,953 B1	4/2001	Petite	6,504,357 B1	1/2003	Hemminger et al.
6,218,958 B1	4/2001	Eichstaedt et al.	6,504,834 B1	1/2003	Fifield
6,218,983 B1	4/2001	Kerry et al.	6,507,794 B1	1/2003	Hubbard et al.
6,233,327 B1	5/2001	Petite	6,509,722 B2	1/2003	Lopata
6,234,111 B1	5/2001	Ulman et al.	6,513,060 B1	1/2003	Nixon et al.
6,236,332 B1	5/2001	Conkright et al.	6,515,586 B1	2/2003	Wymore
6,243,010 B1	6/2001	Addy et al.	6,519,568 B1	2/2003	Harvey et al.
6,246,886 B1	6/2001	Oliva	6,542,076 B1	4/2003	Joao
6,249,516 B1	6/2001	Brownrigg et al.	6,542,077 B2	4/2003	Joao
6,259,369 B1	7/2001	Monico	6,543,690 B2	4/2003	Leydier et al.
6,271,752 B1	8/2001	Vaios	6,560,223 B1	5/2003	Egan et al.
6,275,166 B1	8/2001	del Castillo et al.	6,574,234 B1	6/2003	Myer et al.
6,275,707 B1	8/2001	Reed et al.	6,574,603 B1	6/2003	Dickson et al.
6,286,050 B1	9/2001	Pullen et al.	6,584,080 B1	6/2003	Ganz et al.
6,286,756 B1	9/2001	Stinson et al.	6,600,726 B1	7/2003	Nevo et al.
6,288,634 B1	9/2001	Weiss et al.	6,608,551 B1	8/2003	Anderson et al.
6,288,641 B1	9/2001	Casais	6,618,578 B1	9/2003	Petite
6,295,291 B1	9/2001	Larkins	6,618,709 B1	9/2003	Sneeringer
6,301,514 B1	10/2001	Canada et al.	6,628,764 B1	9/2003	Petite
6,304,556 B1	10/2001	Haas	6,628,965 B1	9/2003	LaRosa et al.
6,305,602 B1	10/2001	Grabowski et al.	6,653,945 B2	11/2003	Johnson et al.
6,307,843 B1	10/2001	Okanoue	6,665,278 B2	12/2003	Grayson
6,308,111 B1	10/2001	Koga	6,671,586 B2	12/2003	Davis et al.
6,311,167 B1	10/2001	Davis et al.	6,674,403 B2	1/2004	Gray et al.
6,314,169 B1	11/2001	Schelberg, Jr. et al.	6,678,255 B1	1/2004	Kuriyan
6,317,029 B1	11/2001	Fleeter	6,678,285 B1	1/2004	Garg
6,327,245 B1	12/2001	Satyanarayana et al.	6,691,173 B2	2/2004	Morris et al.
6,329,902 B1	12/2001	Lee et al.	6,731,201 B1	5/2004	Bailey et al.
6,334,117 B1	12/2001	Covert et al.	6,735,630 B1	5/2004	Gelvin et al.
6,351,223 B1	2/2002	DeWeerd et al.	6,747,557 B1	6/2004	Petite et al.
6,356,205 B1	3/2002	Salvo et al.	6,771,981 B1	8/2004	Zalewski et al.
6,357,034 B1	3/2002	Muller et al.	6,775,258 B1	8/2004	van Valkenburg et al.
6,362,745 B1	3/2002	Davis	6,804,532 B1	10/2004	Moon et al.
6,363,057 B1	3/2002	Ardalan et al.	6,826,607 B1	11/2004	Gelvin et al.
6,366,217 B1	4/2002	Cunningham et al.	6,832,251 B1	12/2004	Gelvin et al.
6,366,622 B1	4/2002	Brown et al.	6,842,430 B1	1/2005	Melnik
6,369,769 B1	4/2002	Nap et al.	6,859,831 B1	2/2005	Gelvin et al.
6,370,489 B1	4/2002	Williams et al.	6,891,838 B1	5/2005	Petite et al.
6,373,399 B1	4/2002	Johnson et al.	6,900,737 B1	5/2005	Ardalan et al.
6,380,851 B1	4/2002	Gilbert et al.	6,914,533 B2	7/2005	Petite
6,384,722 B1	5/2002	Williams	6,914,893 B2	7/2005	Petite
6,392,692 B1	5/2002	Monroe	6,922,558 B2	7/2005	Delp et al.
6,393,341 B1	5/2002	Lawrence et al.	6,959,550 B2	11/2005	Freeman et al.
6,393,381 B1	5/2002	Williams et al.	6,970,434 B1	11/2005	Mahany et al.
6,393,382 B1	5/2002	Williams et al.	7,020,701 B1	3/2006	Gelvin et al.
6,400,819 B1	6/2002	Nakano et al.	7,027,416 B1	4/2006	Kriz
6,401,081 B1	6/2002	Montgomery et al.	7,027,773 B1	4/2006	McMillin
6,405,018 B1	6/2002	Reudink et al.	7,053,767 B2	5/2006	Petite et al.
6,411,889 B1	6/2002	Mizunuma et al.	7,054,271 B2	5/2006	Brownrigg et al.
6,415,245 B2	7/2002	Williams et al.	7,064,679 B2	6/2006	Ehrke et al.
6,421,354 B1	7/2002	Godlewski	7,103,511 B2	9/2006	Petite
6,421,731 B1	7/2002	Ciotti, Jr. et al.	7,137,550 B1	11/2006	Petite
6,422,464 B1	7/2002	Terranova	7,349,682 B1	3/2008	Bennett, III et al.
6,424,270 B1	7/2002	Ali	7,468,661 B2	12/2008	Petite et al.
6,424,931 B1	7/2002	Sigmar et al.	7,480,501 B2	1/2009	Petite
6,430,268 B1	8/2002	Petite	7,484,008 B1	1/2009	Gelvin et al.
6,431,439 B1	8/2002	Suer et al.	7,573,813 B2	8/2009	Melnik
6,437,692 B1	8/2002	Petite et al.	7,653,394 B2	1/2010	McMillin

US 6,044,062 C1

Page 6

7,739,378	B2	6/2010	Petite	GB	2352004	A	1/2001
2001/0002210	A1	5/2001	Petite	GB	2352590	A	1/2001
2001/0003479	A1	6/2001	Fujiwara	JP	60261288	A	12/1985
2001/0021646	A1	9/2001	Antonucci et al.	JP	01255100	A	10/1989
2001/0024163	A1	9/2001	Petite	JP	11353573	A	12/1999
2001/0034223	A1	10/2001	Rieser et al.	JP	2000113590	A	4/2000
2001/0038343	A1	11/2001	Meyer et al.	JP	2001063425	A	3/2001
2002/0002444	A1	1/2002	Williams et al.	JP	2001088401	A	4/2001
2002/0012323	A1	1/2002	Petite et al.	JP	2001309069	A	11/2001
2002/0013679	A1	1/2002	Petite	JP	2001319284	A	11/2001
2002/0019725	A1	2/2002	Petite	JP	2001357483	A	12/2001
2002/0027504	A1	3/2002	Davis et al.	JP	2002007672	A	1/2002
2002/0031101	A1	3/2002	Petite et al.	JP	2002007826	A	1/2002
2002/0032746	A1	3/2002	Lazaridis	JP	2002085354	A	3/2002
2002/0061031	A1	5/2002	Sugar et al.	JP	2002171354	A	6/2002
2002/0072348	A1	6/2002	Wheeler et al.	KR	2001025431	A	4/2001
2002/0089428	A1	7/2002	Walden et al.	WO	WO 90/13197		11/1990
2002/0095399	A1	7/2002	Devine et al.	WO	WO 95/12942		5/1995
2002/0098858	A1	7/2002	Struhsaker	WO	WO 95/24177		9/1995
2002/0109607	A1	8/2002	Cumeralto et al.	WO	WO 95/34177		12/1995
2002/0136233	A1	9/2002	Chen et al.	WO	WO 96/10307		4/1996
2002/0158774	A1	10/2002	Johnson et al.	WO	WO 98/00056		1/1998
2002/0163442	A1	11/2002	Fischer	WO	WO 98/37528		8/1998
2002/0169643	A1	11/2002	Petite et al.	WO	WO 98/45717		10/1998
2002/0193144	A1	12/2002	Belski et al.	WO	WO 99/13426		3/1999
2003/0001754	A1	1/2003	Johnson et al.	WO	WO 00/23956		4/2000
2003/0028632	A1	2/2003	Davis	WO	WO 01/15114		3/2001
2003/0030926	A1	2/2003	Aguren et al.	WO	WO 01/24109		4/2001
2003/0034900	A1	2/2003	Han	WO	WO 02/08725		1/2002
2003/0036822	A1	2/2003	Davis et al.	WO	WO 02/08866		1/2002
2003/0046377	A1	3/2003	Daum et al.	WO	WO 02/052521		7/2002
2003/0058818	A1	3/2003	Wilkes et al.	WO	WO 03/007264		1/2003
2003/0069002	A1	4/2003	Hunter et al.	WO	WO 03/021877		3/2003
2003/0073406	A1	4/2003	Benjamin et al.	WO	WO 04/002014		12/2003
2003/0078029	A1	4/2003	Petite				
2003/0093484	A1	5/2003	Petite				
2003/0133473	A1	7/2003	Manis et al.				
2003/0169710	A1	9/2003	Fan et al.				
2003/0185204	A1	10/2003	Murdock				
2003/0210638	A1	11/2003	Yoo et al.				
2004/0047324	A1	3/2004	Diener				
2004/0053639	A1	3/2004	Petite et al.				
2004/0131125	A1	7/2004	Sanderford et al.				
2004/0183687	A1	9/2004	Petite et al.				
2004/0228330	A1	11/2004	Kubler et al.				
2005/0190055	A1	9/2005	Petite				
2005/0195768	A1	9/2005	Petite				
2005/0195775	A1	9/2005	Petite et al.				
2005/0201397	A1	9/2005	Petite				
2005/0243867	A1	11/2005	Petite				
2006/0095876	A1	5/2006	Chandra et al.				
2009/0006617	A1	1/2009	Petite				
2009/0243840	A1	10/2009	Petite et al.				

FOREIGN PATENT DOCUMENTS

EP	0 749 259	A2	12/1996
EP	0 749 260	A2	12/1996
EP	0 766 489	A2	4/1997
EP	0 768 777	A2	4/1997
EP	0 812 502	B1	12/1997
EP	0825577	A1	2/1998
EP	0 999 717	A2	5/2000
EP	1096454	A2	5/2001
FR	2817110	A1	5/2002
GB	2229302	A	9/1990
GB	2247761	A	3/1992
GB	2262683	A	6/1993
GB	2297663	A	8/1996
GB	2310779	A	9/1997
GB	2326002	A	12/1998
GB	2336272	A	10/1999

OTHER PUBLICATIONS

Baba et al., "Wireless Medium Access Control Protocol For CAN," 4th Int'l CAN Conf., Berlin, Germany, available at <http://www.can-cia.org/fileadmin/cia/files/icc/4/baba1.pdf> (1997).

Corbell et al., "Technical Implementation in Support Of The IAEA's Remote Monitoring Field Trial At The Oak Ridge Y-12 Plant," Dept. of Energy, Office of Scientific and Technical Information, Report No. Sand-096-1934C, available at http://www.osti.gov/bridge/product.biblio.jsp?query_id=1&page=0&osti_id=270678 (1996).

Corbell et al., "Technical Results Of Y-12/IAEA Field Trial Of Remote Monitoring System," Dept. of Energy, Office of Scientific and Technical Information, Report No. Sand-97-1781C, available at http://www.osti.gov/bridge/product.biblio.jsp?query_id=0&page=0&osti_id=505711 (1997).

Echelon Corp., "LonTalk® Protocol Specification," available at <http://www.wenerlon.com/JobAids/Lontalk%20Protocol%20Spec.pdf> (1994).

Echelon Corp., "Series 90™-30 PLC LonWorks® Bus Interface Module User's Manual," available at <http://www.pdfsupply.com/pdfs/gfk1322a.pdf> (1997).

Frank, Randy, "Understanding Smart Sensors," Artech House (1996).

Nilsen et al., "Storage Monitoring Systems For The Year 2000," Dept. of Energy, Office of Scientific and Technical Information, Report No. Sand-97-8532C, available at http://www.osti.gov/bridge/product.biblio.jsp?query_id=3&page=0&osti_id=303988 (1997).

Poor, Robert D., "Hyphos: A Self-Organizing, Wireless Network," Massachusetts Institute of Technology (Jun. 1997).

US 6,044,062 C1

Page 7

- Raji, Reza S., "Control Networks and the Internet," Echelon Corp., Rev. 2.0, available at http://www.echelon.com/solutions/opensystems/papers/Control_Internet.pdf (1998).
- Ross et al., "PNC/DOE Remote Monitoring Project at Japan's Joyo Facility," Office of Scientific and Technical Information, Report No. Sand-96-1937C, available at http://www.osti.gov/bridge/product.biblio.jsp?query_id=&page=0&osti_id=270680 (1996).
- Saffo, Paul, "Sensors: The Next Wave of Infotech Innovation," Institute for the Future (1997).
- Schneider et al., "International Remote Monitoring Project Argentina Nuclear Power Station Spent Fuel Transfer Remote Monitoring System," Dept. of Energy, Office of Scientific and Technical Information, Report No. Sand-97-1784C, available at http://www.osti.gov/bridge/product.biblio.jsp?query_id=1&page=&osti_id=505674 (1997).
- Tanenbaum, Andrew S., "Computer Networks," chapters 1, 5 (3d ed. Prentice Hall 1996).
- Thomas, "Extending CAN Networks By Incorporating Remote Bridging," 4th Int-CAN Conf., Berlin, Germany, available at http://www.can-cia.org/fileadm_in/cia/files/icc/4/thomas.pdf (1997).
- "Engineering Report—Johnson Controls Interface," ESTeem Radios (Nov. 1994).
- ESTeem Application Paper—AgriNorthwest Employee's Provide Wireless Control System (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Allen-Bradley Goes Wireless on Alaska's North Slope (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Wireless Control of Polluted Water (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Wireless Networking for Natural Gas Extraction (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Build Your Own Wireless Power Distribution System (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Lost Cabin Gas Plant Uses Wireless Control to Enhance Production & Safety (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Northwest Farm Applies Wireless Solution (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Wireless Networking for Kodiak's Coast Guard Station (describing a system that was in use prior to Mar. 1999).
- ESTeem Application Paper—Wireless Mobile Mapping System (describing a system that was in use prior to Mar. 1999).
- "ESTeem Engineering Report, Johnson Controls Interface No: 91-102" (Nov. 1994).
- ESTeem Models 85, 95, 96, & 98 User's Manual (describing the ESTeem 96C and 96F radios used prior to 1999).
- "ESTeem Model 96C," ESTeem Radios (describing a system that was for sale at least as early as 1994).
- "ESTeem Model 96F," ESTeem Radios (describing a system that was for sale at least as early as 1994).
- "Site Survey Report," ESTeem Radios (Sep. 24, 1993).
- "Technical Bulletin—Johnson Controls," ESTeem Radios (Jan. 29, 1998).
- 1997 Project Summary, Held Untethered Nodes, University of California at Los Angeles, available at <http://web.archive.org/web/199812052324758/http://www.darpa.mil/leaving.asp?url=http://www.janet.ucla.edu/glomo>, Jul. 25, 2008, pp. 1-5.
- 1997 Project Summary, Mobile Versatile Radios (MoVeR), University of California at Los Angeles, available at <http://web.archive.org/web/19990222140122/http://www.darpa.mil/leaving.asp?url=http://www.janet.ucla.edu/>, Jul. 25, 2008, pp. 1-4.
- 1997 Project Summary, Towards a Wireless Overlay Internetworking Architecture, University of California at Berkeley, available at <http://web.archive.org/web/19990202065939/http://www.darpa.mil/leaving.asp?url=http://daedalus.cs.berkeley.edu>, Jul. 25, 2008, pp. 1-8.
- 3Com Invests in Coactive Networks, *Coactive* (press release), Author: unknown, Dec. 14, 1999, pp. 1-4.
- 5808 Photoelectric Smoke/Heat Detector with Built-in Wireless Transmitter Installation Instructions, *ADEMCO*; Author: unknown; 1998.
- ABB Kent-Taylor Interfacing*, Author: unknown, *Engineering Report*, No. 93-011, Jun. 18, 1996, pp. 1-9.
- Abbott et al., *Wireless Product Applications for Utilities, Electric Power Research Institute*, Feb. 1996, pp. 1-137.
- About AES Corporation, *AES IntelliNet*, Author: unknown, available at <http://web.archive.org/web/19990127093116/www.aes-intellinet.com/ae>, on Mar. 5, 2009, pp. 1-2.
- ADEMCO Group, 4110DL Security System, Installation Instructions, Oct. 1996, *ADEMCO Group*, Author: unknown, pp. 1-15.
- ADEMCO Group, 4110XM Security System, Installation Instructions, Jul. 1996, *ADEMCO Group*, Author: unknown, pp. 1-20.
- ADEMCO Group, 4120EC Security System, Installation Instructions, Nov. 1990, *ADEMCO Group*, Author: unknown, pp. 1-17.
- ADEMCO Group, 4120XM Security System, Installation Instructions, Oct. 1993, *ADEMCO Group*, Author: unknown, pp. 1-80.
- ADEMCO Group, 4140XMPT2 Partitioned Security System with Scheduling User's Manual, May 1993, *ADEMCO Group*, Author: unknown; pp. 1-54.
- ADEMCO Group, 4281, 5881 and 5882 Series RF Receivers Installation Instructions, Oct. 1996, *ADEMCO Group*, Author: unknown; pp. 1-6.
- ADEMCO Group, 5330 Alpha Console, Installation Instructions, May 90, *ADEMCO Group*, Author: unknown, pp. 1-24.
- ADEMCO Group, 5706 Smoke Detector with Built-in Wireless Transmitter, Installation Instructions, Dec. 1991, *ADEMCO Group*, Author: unknown, pp. 1-8.
- ADEMCO Group, 5707 Smoke Detector with Built-in Wireless Transmitter, Installation Instructions, Aug. 1992, *ADEMCO Group*, Author: unknown, pp. 1-12.
- ADEMCO Group, 5715 Universal Transmitter, Installation Instructions, Mar. 1989, *ADEMCO Group*, Author: unknown, pp. 1-4.
- ADEMCO Group, 5775 Passive Infrared Motion Detector/Transmitter, Installation Instructions, Jul. 1991, *ADEMCO Group*; Author: unknown; pp. 1-4.
- ADEMCO Group, 58008C Photoelectronic Smoke/Detector with Built-In Wireless Transmitter Installation Instructions, 1998, *ADEMCO Group*, Author: unknown; pp. 1-4.

US 6,044,062 C1

Page 8

- ADEMCO Group, 5800TM Transmitter Module Installation Instructions, Apr. 1994, *ADEMCO Group*, Author: unknown; pp. 1.
- ADEMCO Group, 5801 Remote Wireless Panic Transmitter Installation Instructions, Apr. 1994, *ADEMCO Group*, Author: unknown; pp. 2.
- ADEMCO Group, 5802CP Belt Clip Transmitter Installation Instructions, Nov. 1994, *ADEMCO Group*, Author: unknown; pp. 1.
- ADEMCO Group, 5802MN Supervised Miniature Transmitter Installation Instructions, Jan. 1995, *ADEMCO Group*, Author: unknown; pp. 1.
- ADEMCO Group, 5802MN2 Supervised Miniature Transmitter Installation Instructions, Jun. 1997, *ADEMCO Group*, Author: unknown; pp. 1.
- ADEMCO Group, 5803 Wireless Key Transmitter Installation Instructions, Nov. 1994, *ADEMCO Group*, Author: unknown; pp. 2.
- ADEMCO Group, 5804 Wireless Key Transmitter Installation Instructions, Jul. 1995, *ADEMCO Group*, Author: unknown; pp. 3.
- ADEMCO Group, 5804BD Bi-Directional Wireless Key Installation Instructions, Apr. 1997, *ADEMCO Group*, Author: unknown; pp. 4.
- ADEMCO Group, 5806 Smoke Detector with Built-In Wireless Transmitter Installation Instructions, May 1998, *ADEMCO Group*, Author: unknown; pp. 1-4.
- ADEMCO Group, 5807 Smoke Detector with Built-In Wireless Transmitter Installation Instructions, May 1998, *ADEMCO Group*, Author: unknown; pp. 1-6.
- ADEMCO Group, 5808 Photoelectronic Smoke/Heat Detector with Built-In Wireless Transmitter Installation Instructions, 1998, *ADEMCO Group*, Author: unknown; pp. 1-8.
- ADEMCO Group, 5808 Wireless Smoke Detector, 1999, available at <http://web.archive.org/web/20000118015507/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-4.
- ADEMCO Group, 5809 Rate-of-Rise Heat Detector/Transmitter Installation Instructions, May 1995, *ADEMCO Group*, Author: unknown; pp. 1-2.
- ADEMCO Group, 5816 Door/Window Transmitter Installation Instructions, Nov. 1994, *ADEMCO Group*, Author: unknown; pp. 1-2.
- ADEMCO Group, 5816TEMP Low Temperature Transmitter Installation Instructions May 1998, *ADEMCO Group*, Author: unknown; pp. 1-2.
- ADEMCO Group, 5818 Recessed Transmitter, Installation Instruction, Jan. 1994, *ADEMCO Group*, Author: unknown; pp. 1-2.
- ADEMCO Group, 5819 Shock Processor Transmitter Installation Instructions, May 1998, *ADEMCO Group*, Author: unknown; pp. 1-2.
- ADEMCO Group, 5819WHS/5819BRS Shock Processor Transmitter Installation Instructions, May 1998, *ADEMCO Group*, Author: unknown; pp. 1-2.
- ADEMCO Group, 5827 Remote Wireless Keypad/Transmitter Installation Instructions, Apr. 1994, *ADEMCO Group*, Author: unknown; pp. 1.
- ADEMCO Group, 5827BD and 5827BDE Wireless Bi-Directional Keypads Installation Instructions and Operating Guide, Mar. 1996, *ADEMCO Group*, Author: unknown; pp. 1-6.
- ADEMCO Group, 5849 Glass Break Detector/Transmitter Installation Instructions, Oct. 1997, *ADEMCO Group*, Author: unknown; pp. 1-4.
- ADEMCO Group, 5850 Glass Break Detector/Transmitter Installation Instructions, May 1998, *ADEMCO Group*, Author: unknown; pp. 1-4.
- ADEMCO Group, 5890 Passive Infrared Motion Detector/Transmitter Installation Instructions, May 1998, *ADEMCO Group*, Author: unknown; pp. 1-8.
- ADEMCO Group, 5890 Wireless PIR Motion Detector, 1997, available at <http://web.archive.org/web/19990429054256/www.ademco.com/asc/> on Mar. 5, 2009, pp. 1-3.
- ADEMCO Group, 5890PI Passive Infrared Motion Detector/Transmitter Installation Instructions, Mar. 1998, *ADEMCO Group*, Author: unknown; pp. 1-4.
- ADEMCO Group, 6128RF Keypad/Receiver—full wireless capability, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19981206111450/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-2.
- ADEMCO Group, 6128RF Keypad/Transceiver, Installation Instructions, Jul. 1998, *ADEMCO Group*, Author: unknown, pp. 1-8.
- ADEMCO Group, 6128RF Keypad/Transceiver, User Guide, May 1998, *ADEMCO Group*, Author: unknown, pp. 1.
- ADEMCO Group, 6128WL Keypad/Receiver, Installation Instructions, Oct. 1998, *ADEMCO Group*, Author: unknown, pp. 1-8.
- ADEMCO Group, 6128WL Keypad/Receiver, User Guide, Oct. 1998, *ADEMCO Group*, Author: unknown, pp. 1.
- ADEMCO Group, 7715DF MicroFAST Installation Tool, User Manual, Feb. 1998, *ADEMCO Group*, Author: unknown, pp. 1-32.
- ADEMCO Group, 7720 Subscriber Radio, Installation Instructions, Jan. 1992, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1-18.
- ADEMCO Group, 7720NX Network Extender, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990220035932/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-3.
- ADEMCO Group, 7720NX Network Extender, *ADEMCO Group*, Author: unknown, 1998, pp. 1-2.
- ADEMCO Group, 7720P Programming Tool, User Guide, Mar. 1992, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1-8.
- ADEMCO Group, 7720Plus Subscriber Radio, Installation Instructions, Oct. 1996, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1-100.
- ADEMCO Group, 7720ULF Combination Fire Control and Long Range Radio Transmitter, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990501210612/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-1.
- ADEMCO Group, 7720ULF Subscriber Radio, Installation Instructions, Mar. 1995, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1-20.

US 6,044,062 C1

Page 9

- ADEMCO Group, 7720V2 Self-Contained Long Range Radio Transmitter, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990501212349/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–4.
- ADEMCO Group, 7720V2 Subscriber Radio, Installation Instructions, Jun. 1996, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–24.
- ADEMCO Group, 7810iR Internet Receiver, Installation and Setup Guide, May 2002, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–58.
- ADEMCO Group, 7820 Appendices, *ADEMCO Group*, Author: unknown, Date: unknown, available at <http://www.guardianalarms.net>, pp. 1–2.
- ADEMCO Group, 7820 Integrated Radio Transmitter, Installation Instructions, Aug. 1995, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–52.
- ADEMCO Group, 7825 Outdoor Antenna with Bracket, Installation Instructions, Feb. 1995, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–2.
- ADEMCO Group, 7830R SafetyNet Subscriber Radio, Installation Instructions, Jun. 1996, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–32.
- ADEMCO Group, 7830R Subscriber Transmitter, 1997, available at <http://web.archive.org/web/1999050125427/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–3.
- ADEMCO Group, 7835C Cellular Control Channel Transceiver, Installation and Setup Guide, Sep. 1998, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–32.
- ADEMCO Group, 7835C Cellular SafetyNet Subscriber Radio Transceiver, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990801221202/www.ademco.com/> on Mar. 5, 2009, pp. 1–3.
- ADEMCO Group, 7845C Cellular Control Channel Transceiver, Installation and Setup Guide, Sep. 1990, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–104.
- ADEMCO Group, 7845CZ Seven Zone Cellular Control Channel Transceiver, Installation and Setup Guide, Sep. 2001, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–64.
- ADEMCO Group, 7845i Internet Communications Module, Installation and Setup Guide, May 2002, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–42.
- ADEMCO Group, 7920SE 900MHz Fully Synthesized Transceiver, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990501222639/www.ademco.com/> on Mar. 5, 2009, pp. 1–3.
- ADEMCO Group, 7920SE Transceiver, Installation Instructions, Apr. 1995, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1–80.
- ADEMCO Group, ADEMCO World Leader in Home Security Products, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990428164624/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–2.
- ADEMCO Group, AlarmNet Introduces Control Channel Cellular for Commercial Fire/Burglary Applications, *ADEMCO Group*, (press release), Aug. 31, 1999, available at <http://web.archive.org/web/20000119053724/www.ademco.com/pr0831> on Mar. 31, 2009, pp. 1.
- ADEMCO Group, AlarmNet, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990420234130/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–3.
- ADEMCO Group, Alpha Vista No. 5130XT Security System, Installation Instructions, Mar. 1989, *ADEMCO Group*, Author: unknown, pp. 1–96.
- ADEMCO Group, Compass Network Downloader, *ADEMCO Group*, Author: unknown, Date: unknown, available at <http://www.guardianalarms.net> pp. 1–109.
- ADEMCO Group, Compass, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990209094401/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–3.
- ADEMCO Group, Control/Communicator 5110XM, Installation Instructions, Apr. 1996, *ADEMCO Group*, Author: unknown, pp. 1–76.
- ADEMCO Group, Fire Alarm Control/Communicator Model 5110XM User's Manual, Apr. 1996, *ADEMCO Group*, Author: unknown; pp. 1–30.
- ADEMCO Group, Fire and Burglary System Model 5120XM User's Manual, Jun. 1996, *ADEMCO Group*, Author: unknown; pp. 1–40.
- Ademco Group, Home Page, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19961023204954/http://ademco.com/> on Mar. 5, 2009, pp. 1.
- ADEMCO Group, Lynx—Quick Install Security System, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990116225005/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–3.
- ADEMCO Group, Lynx Quick Star Guide, Oct. 1998, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Lynx Security System Programming Form & Summary of Connections, Oct. 1998, *ADEMCO Group*, Author: unknown, pp. 1–16.
- ADEMCO Group, Lynx Security System User Guide, Oct. 1998, *ADEMCO Group*, Author: unknown; pp. 1–40.
- ADEMCO Group, Lynx Security System, Installation and Setup Guide, Oct. 1998, *ADEMCO Group*, Author: unknown, pp. 1–48.
- ADEMCO Group, Powerline Carrier Device Modules, 1997 *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990218035115/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–2.
- ADEMCO Group, Remote Keypads 6128, 6137, 6137R, 6138, 6139 & 6139R, Installation Guide, Aug. 1998, *ADEMCO Group*, Author: unknown, pp. 1–2.
- ADEMCO Group, Security System Model 4110DL Programming Form, Oct. 1996, *ADEMCO Group*, Author: unknown, pp. 1–8.
- ADEMCO Group, Security System Model 4110XM Programming Form, Jul. 1996, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model 4120EC Programming Form, Sep. 1993, *ADEMCO Group*, Author: unknown, pp. 1–2.

US 6,044,062 C1

Page 10

- ADEMCO Group, Security System Model 4120XM Programming Form, Sep. 1992, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model 4130XM, 4140XM, 5130XM Programming Form, *ADEMCO Group*, Author: unknown, Date: unknown, pp. 1–4.
- ADEMCO Group, Security System Model 4130XT/4140/5130XT Programming Form, Jul. 1989, *ADEMCO Group*, Author: unknown, pp. 1–2.
- ADEMCO Group, Security System Model 4140XMP Programming Form, Jan. 1992, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model 4140XMPT Programming Form, *ADEMCO Group*, Author: unknown, Date: unknown, pp. 1.
- ADEMCO Group, Security System Model 4140XMPT2 Programming Form, Apr. 1996, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model 5110XM Programming Form, Apr. 1996, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model 5120XM Programming Form, Jun. 1996, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model 5140XM Programming Form, Jun. 1993, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model Vista–10 Programming Form, Sep. 1994, *ADEMCO Group*, Author: unknown, pp. 1–4.
- ADEMCO Group, Security System Model Vista–10_{SE} Programming Guide, Apr. 1997, *ADEMCO Group*, Author: unknown, pp. 1–24.
- ADEMCO Group, Security System Model Vista–128B Commercial Burglary Partitioned Security System with Scheduling, Quick Start Guide, Jul. 1998, *ADEMCO Group*, Author: unknown, pp. 1–39.
- ADEMCO Group, Security System User’s Manual, Sep. 1996, *ADEMCO Group*, Author: unknown; pp. 1–88.
- ADEMCO Group, The Vista–100 Series, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19970620010543/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–7.
- ADEMCO Group, The Vista–10SE, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990502214402/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1–4.
- ADEMCO Group, via16 Programming Form, Jul. 1993, *ADEMCO Group*, Author: unknown, pp. 1–2.
- ADEMCO Group, Via–16 Security System, Installation Instructions, Jan. 1992, *ADEMCO Group*, Author: unknown, pp. 1–24.
- ADEMCO Group, Via–30+, Vista 10, 4111XM Security System User’s Manual, Jul. 1994, *ADEMCO Group*, Author: unknown, pp. 1–44.
- ADEMCO Group, Via–30PSE and Vista–10_{SE} Security System User’s Manual, Jan. 1997, *ADEMCO Group*, Author: unknown; pp. 1–88.
- ADEMCO Group, Via–30P_{SE} Security System, Programming Guide, Apr. 1997, *ADEMCO Group*, Author: unknown, pp. 1–24.
- ADEMCO Group, Vista 4120XM and 4140XMP Security System User’s Manual, Jan. 1994, *ADEMCO Group*, Author: unknown; pp. 1–60.
- ADEMCO Group, Vista 4130XT Security System, Installation Instructions, Oct. 1988, *ADEMCO Group*, Author: unknown, pp. 1–84.
- ADEMCO Group, Vista 4140XMPT2 Partitioned Security System with Scheduling, Installation Instructions, May 1993, *ADEMCO Group*, Author: unknown, pp. 1–68.
- ADEMCO Group, Vista 5140XM Commercial Fire and Burglary Alarm System, Installation Instructions, Jun. 1993, *ADEMCO Group*, Author: unknown, pp. 1–74.
- ADEMCO Group, Vista AT 4140 Security System, Installation Instructions, Sep. 1988, *ADEMCO Group*, Author: unknown, pp. 1–68.
- ADEMCO Group, Vista Series 4120EC Security System User’s Manual, Sep. 1992, *ADEMCO Group*, Author: unknown; pp. 1–28.
- ADEMCO Group, Vista Series 4130XM, 5130XM, 4140XMP Security System User’s Manual, Feb. 1992, *ADEMCO Group*, Author: unknown; pp. 1–32.
- ADEMCO Group, Vista Series 4140XMP, Installation Instructions, Jan. 1992, *ADEMCO Group*, Author: unknown, pp. 1–52.
- ADEMCO Group, Vista Series 4140XMPT/4140XMPT–UL Partitioned Security System User’s Manual, Jun. 1993, *ADEMCO Group*, Author: unknown; pp. 1–32.
- ADEMCO Group, Vista Series 5140XM User’s Manual, Aug. 1992, *ADEMCO Group*, Author: unknown; pp. 1–28.
- ADEMCO Group, Vista Series Partitioned Security Systems Model 4140XMPT, Installation Instructions, Feb. 1992, *ADEMCO Group*, Author: unknown, pp. 1–60.
- ADEMCO Group, Vista XM Series 4140XM, 5130XM, 4130XM, Installation Instructions, Jul. 1990, *ADEMCO Group*, Author: unknown, pp. 1–26.
- ADEMCO Group, Vista XM Series, Installation Instructions, *ADEMCO Group*, Author: unknown, Oct. 1991, pp. 1–16.
- ADEMCO Group, Vista–10 Security System, Installation Instructions, Sep. 1994, *ADEMCO Group*, Author: unknown, pp. 1–56.
- ADEMCO Group, Vista–100 Commercial Fire & Burglary Alarm System User’s Manual, Nov. 1995, *ADEMCO Group*, Author: unknown; pp. 1–66.
- ADEMCO Group, Vista–100 Commercial Fire & Burglary Alarm System with Scheduling Quick Start, Apr. 1996, *ADEMCO Group*, Author: unknown; pp. 1–24.
- ADEMCO Group, Vista–100 Commercial Fire and Burglary Partitioned Security System with Scheduling, Installation Instructions and Programming Guide, Jan. 1998, *ADEMCO Group*, Author: unknown, pp. 1–233.
- ADEMCO Group, Vista–10_{SE}, Installation Instructions, May 1997, *ADEMCO Group*, Author: unknown, pp. 1–88.
- ADEMCO Group, Vista–128B Commercial Burglary Partitioned Security System with Scheduling, Installation and Setup Guide, Jul. 1998, *ADEMCO Group*, Author: unknown, pp. 1–252.
- ADEMCO Group, Vista–128FB Commercial Fire and Burglary Partitioned Security System User Guide, Oct. 1998, *ADEMCO Group*, Author: unknown; pp. 1–80.
- ADEMCO Group, Vista–128FB Commercial Fire and Burglary Partitioned Security System with Scheduling, Installation and Setup Guide, Oct. 1998, *ADEMCO Group*, Author: unknown, pp. 1–220.

US 6,044,062 C1

Page 11

- ADEMCO Group, Vista-20 2-Partitioned Security System, Installation Instructions, Nov. 1995, *ADEMCO Group*, Author: unknown, pp. 1-120.
- ADEMCO Group, Vista-20 2-Partitioned Security System, Programming Form, Apr. 1996, *ADEMCO Group*, Author: unknown, pp. 1-8.
- ADEMCO Group, Vista-20 Security System User's Manual, Apr. 1995, *ADEMCO Group*, Author: unknown; pp. 1-52.
- ADEMCO Group, Vista-20HW 2-Partitioned Security System, Installation Instructions, Apr. 1996, *ADEMCO Group*, Author: unknown, pp. 1-100.
- ADEMCO Group, Vista-20HW 2-Partitioned Security System, Programming Form, Apr. 1996, *ADEMCO Group*, Author: unknown, pp. 1-8.
- ADEMCO Group, Vista-20HW_{SE} 2-Partitioned Security System, Installation Instructions, Aug. 1997, *ADEMCO Group*, Author: unknown, pp. 1-84.
- ADEMCO Group, Vista-20HW_{SE} 2-Partitioned Security System, Programming Form, Aug. 1997, *ADEMCO Group*, Author: unknown, pp. 1-8.
- ADEMCO Group, Vista-20_{SE} 2-Partitioned Security System, Installation Instructions, Aug. 1997, *ADEMCO Group*, Author: unknown, pp. 1-100.
- ADEMCO Group, Vista-20_{SE} 2-Partitioned Security System, Programming Form, Aug. 1997, *ADEMCO Group*, Author: unknown, pp. 1-8.
- ADEMCO Group, Vista-20_{SE}/Vista-20HW_{SE} Security System User's Manual, Jun. 1997, *ADEMCO Group*, Author: unknown; pp. 1-52.
- ADEMCO Group, Vista-30P_{SE} Security System, Installation Instructions, Apr. 1997, *ADEMCO Group*, Author: unknown, pp. 1-104.
- ADEMCO Group, Vista-40 2-Partition Security System, Programming Guide, Jul. 1998, *ADEMCO Group*, Author: unknown, pp. 1-24.
- ADEMCO Group, Vista-40 2-Partitioned Security System, Installation and Setup Guide, Jul. 1998, *ADEMCO Group*, Author: unknown, pp. 1-380.
- ADEMCO Group, Vista-40 Programming Guide, Jun. 1997, *ADEMCO Group*, Author: unknown, available at <http://www.guardianalarms.net>, pp. 1-20.
- ADEMCO Group, Vista-40 Security System User's Guide, Jul. 1998, *ADEMCO Group*, Author: unknown, pp. 1-60.
- ADEMCO Group, Vista-50, Vista-50UL Security System, Nov. 1994, *ADEMCO Group*, Author: unknown, pp. 1-66.
- ADEMCO Group, Vista-50P, Vista-50PUL Partitioned Security System with Scheduling, Installation Instructions and Programming Guide, Oct. 1997, *ADEMCO Group*, Author: unknown, pp. 1-199.
- ADEMCO Group, Vista-50P, Vista-50PUL Security System User's Manual, Jul. 1995, *ADEMCO Group*, Author: unknown, pp. 1-66.
- ADEMCO Group, Vista-50P, Vista-50PUL, Partitioned Security System with Scheduling, Quick Start, Aug. 1995, *ADEMCO Group*, Author: unknown, pp. 1-28.
- ADEMCO Group, Vista-AT Security System User's Manual, Sep. 1988, *ADEMCO Group*, Author: unknown, pp. 1-56.
- ADEMCO Group, V-Link Downloading Software User's Guide, Jun. 1994, *ADEMCO Group*, Author: unknown, available at <http://guardianalarms.net>, pp. 1-126.
- ADEMCO Group, V-Plex Security Technology, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990421110527/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-6.
- ADEMCO Group, Wireless Transmitters/Receivers: 5700 Wireless Transmitters, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990127120423/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-2.
- ADEMCO Group, Wireless Transmitters/Receivers: 5800 Wireless Transmitters, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990218181254/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-2.
- ADEMCO Group, Wireless User Interface Devices, 1997, *ADEMCO Group*, Author: unknown, available at <http://web.archive.org/web/19990421190353/www.ademco.com/ademco> on Mar. 5, 2009, pp. 1-4.
- AES • 7700 Central Station, Installation & Operation Manual, Document 40-0551u, *AES Corporation*, Author: unknown, Dec. 1996, pp. 1-40.
- AES • IntelliGuard 7470, *AES IntelliNet*, Author: unknown, Nov. 2003, pp. 1-15.
- AES • IntelliNet Theory of Operation, *AES IntelliNet*, Author: unknown, Dec. 1996, downloaded from <http://www.guardianalarms.net>, pp. 1-18.
- AES • IntelliNet Wireless Network Glossary of Terms, document 40-0551u, *AES IntelliNet*, Author: unknown, Dec. 1996, pp. 1-15.
- AES 700 Smart Central Station InstaCentral Station Installation & Operation Manual, Document No. 40-0551e, *AES Intellinet*, Author: unknown; Nov. 20, 1996, pp. 1-57.
- AES 7067 IntelliTap-II Digital Dialer Interface: A Supplemental Alarm Supporting Device, *AES IntelliNet*, Author: unknown; Aug. 5, 2004, pp. 1-4.
- AES 7099 Central Station Installation & Operation Manual, Document No. 40-0050, *AES Intellinet*, Author: unknown; 1998, pp. 1-20.
- AES 7450 RF Subscriber Unit Installation Manual, *AES IntelliNet*, Author: unknown; Jun. 21, 2000, pp. 1-8.
- AES 7750-F RF Smart Subscriber Unit Version 2, Including 7750-F-4x4 and 7750-F-8, Installation & Operation Manual, *AES IntelliNet*, Author: unknown, Apr. 2001 (Updated Nov. 2003), pp. 1-60.
- AES 7750-F RF Smart Subscriber Unit Version 2, Installation & Operation Manual, *AES IntelliNet*, Author: unknown, Aug. 2000, pp. 1-30.
- AES Central Alarm Monitoring, Author: unknown, available at <http://web.archive.org/web/19990225163745/www.aes-intellinet.com/ae>, on Mar. 5, 2009, pp. 1-3.
- AES IntelliNet Model 7440 & 7440-XL RF Subscriber Unit, *Addendum, AES Intellinet*, Author: unknown, Aug. 29, 2002, pp. 1.
- AES IntelliNet 7450 *Addendum, AES Corporation*, Author: unknown, Jul. 9, 2002, pp. 1-2.
- AES IntelliNet Dealer's List By State, Author: unknown, available at <http://web.archive.org/web/20010216234026/www.aes-intellinet.com/list> on Mar. 5, 2009, pp. 1-13.
- AES IntelliNet Model 7003 Central Station, Installation & Operation Manual, *AES IntelliNet*, Author: unknown, Jan. 9, 2001, available at www.guardianalarms.net, pp. 1-25.

US 6,044,062 C1

Page 12

- AES *IntelliNet* Model 7050, 7750, Subscriber Unit, Version 1.62, Installation & Operation Manual, *AES IntelliNet*, Author: unknown, Dec. 1996, available at www.guardianalarms.net, pp. 1–110.
- AES *IntelliNet* Model 7050–E & 7750–E, RF Subscriber Unit, Version 1.71, Installation & Operation Manual, *AES IntelliNet*, Author: unknown, Feb. 24, 1997, available at www.guardianalarms.net, pp. 1–54.
- AES *IntelliNet* Model 7050–E Radio Subscriber Unit Installation Manual, *AES IntelliNet*, Author: unknown, Jul. 17, 2000, available at www.guardianalarms.net, pp. 1–4.
- AES *IntelliNet* Net 77 Version 1.48.30, Installation & Operation Manual, Document 40–0551u, *AES Corporation*, Author: unknown, Jun. 1999, pp. 1–30.
- AES *IntelliNet* Net 77 Version 1.48.4, Installation & Operation Manual, Document 40–0551u, *AES Corporation*, Author: unknown, Nov. 2000, pp. 1–36.
- AES *IntelliNet* Net7K Version 1.48.4, Installation & Operation Manual, Document 40–0551, *AES Corporation*, Nov. 2000, pp. 1–36.
- AES *IntelliNet* Radio Communication Subscriber Unit 7050, Sep. 16, 1997, available at <http://web.archive.org/web/1999020361203/www.aes-intellinet.com/sp> on Mar. 5, 2009, pp. 1–2.
- AES *IntelliNotes* Universal Serial Data Interface/ USDI, Bulletin No. 55, *AES Corporation*, Author: unknown, Apr. 5, 2001, pp. 1–12.
- AES *IntelliTAP* Model 7068, Version 1.08, Installation Guide, *AES IntelliNet*, Author: unknown, Jun. 15, 2000, pp. 1–11.
- AES *IntelliTRAK* 7555–RT GPS Based Vehicle Tracking Unit, Version 2.12, *AES IntelliNet*, Author: unknown, Nov. 6, 2002, pp. 1–16.
- AES *IntelliTRAK* 7555–RT GPS Based Vehicle Tracking Unit, Version 2.0a, *AES IntelliNet*, Author: unknown, Feb. 20, 2001, pp. 1–16.
- AES Net7000, Installation & Operation Manual, *AES IntelliNet*, Author: unknown; Nov. 24, 1996, pp. 1–76.
- AES Net77 Wireless Network Management Software Installation & Operation Manual Central Station Manual, Section 3, *AES IntelliNet*; Author: unknown; Dec. 1996, pp. 1–87.
- AES UL/ULC System Configuration, *AES Corporation*, Author: unknown, May 1, 2003, pp. 1.
- Agre et al., Autoconfigurable Distributed Control Systems, *ISADS*, Apr. 25–27, 1995, pp. 162–168.
- Agre et al., Development Platform for Self–Organizing Wireless Sensor Networks, *UCLA, Rockwell Science Center*; Date: unknown, pp. 1–25.
- Agre et al., *Development Platform for Self–Organizing Wireless Sensor Networks*, Publisher: unknown, Date: unknown; pp. 1–10.
- Agre et al., *Development Platform for Self–Organizing Wireless Sensor Networks*, *SPIE*, vol. 3713, Apr. 1999, pp. 257–268.
- Agre et al., Technical and Management Proposal for Adaptive Wireless Arrays for Interactive Reconnaissance, Surveillance and Target Acquisition in Small Unit Operations (AWAIRS), Defense Advanced Research Projects Agency Broad Agency Announcement 96–26, *UCLA*, Date: unknown, pp. 1–50.
- AlarmLink, Inc., *A Brief History*, available at <http://www.alarmlink.com/Default.aspx?tabid=28>, on Mar. 23, 2009, pp. 1.
- AlarmLink, Inc., *Alarm Over IP Products*, available at <http://www.alarmlink.com/Default.aspx?tabid=38>, on Mar. 24, 2009, pp. 1.
- AlarmLink, Inc., *Central Stations*, available at <http://www.alarmlink.com/Default.aspx?tabid=35>, on Mar. 24, 2009, pp. 1–3.
- AlarmLink, Inc., *Home Page*, available at <http://www.alarmlink.com/>, on Mar. 24, 2009, pp. 1–2.
- AlarmLink, Inc., *MeshWorks of Los Angeles*, available at <http://www.alarmlink.com/Default.aspx?tabid=39>, on Mar. 24, 2009, pp. 1.
- AlarmNet–C Service Shutdown, *Honeywell, Inc.*, Author: unknown, Date: unknown, pp. 1.
- Allen–Bradley Interfacing*, Author: unknown, *Engineering Report*, No. 90–023, Jul. 21, 1999, pp. 1–11.
- Alwan et al., Adaptive Mobile Multimedia Networks, *IEEE Personal Communications*, Apr. 1996, pp. 34–51.
- American National Standards Institute, Inc., *ANSI C12.18–1996: Protocol Specification for ANSI Type 2 Optical Port*, *National Electrical Manufacturers Association*, 1996.
- American National Standards Institute, Inc., *ANSI C12.19–1997: Utility Industry End Device Data Tables*, *National Electrical Manufacturers Association*, 1997.
- Amir, *The Ricochet System Architecture*, available at http://www.lariat.org/g/B_erkeley/node2.html, on May 1996, pp. 1–5.
- Amir et al., An Evaluation of the Metricom Ricochet Wireless Network, CS 294–7 Class Project, Department of Electrical Engineering and Computer Science of the University of California at Berkeley, Publisher: unknown, May 7, 1996, pp. 1–20.
- Amir, *The Ricochet System Architecture* (May 7, 1996), available at <http://www.lariat.org/Berkeley/node2.html>, Sep. 17, 2009, pp. 1–4.
- AN/TSQ–129 Position Location Reporting System (PLRS)*, Author: unknown, available at <http://www.fas.org/man/dod-101/sys/land/plrs.htm> on Feb. 22, 2010, pp. 1–3.
- Asada et al., Low Power Wireless Communication and Signal Processing Circuits for Distributed Microsensors; Proceedings of the International Circuits and Systems Symposium, *ISCAS '97; UCLA, Rockwell Science Center*; Jun. 1997, pp. 1–5.
- Asada, Wireless Integrated Network Sensors (WINS), *UCLA, SPIE* vol. 3673, Mar. 1999, pp. 11–18.
- Asada et al., Wireless Integrated Network Sensors: Low Power Systems on a Chip, *UCLA, Rockwell Science Center*; Date: unknown, pp. 1–24.
- Asada et al., *Wireless Integrated Network Sensors: Low Power Systems on a Chip*, *UCLA*, 1998, pp. 1–16.
- Asada et al., *Wireless Integrated Sensors Network: Low Power Systems on a Chip*, Publisher: unknown, Date: unknown, pp. 1–8.
- Atlanta Building News, *The Voice of the Greater Atlanta Home Builders Association*, vol. 7, No. 5, May 2006, pp. 1–60.
- Bagby, Calypso Ventures Inc.—WLAN background, 2 pages.
- Baker et al., The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm, *IEEE*; Nov. 1981, pp. 1694–1701.
- Ball et al., Reliability of Packet Switching Broadcast Radio Networks, *IEEE Transactions on Circuits and Systems*, vol. CAS–23, No. 12, Dec. 1976, pp. 806–813.

US 6,044,062 C1

Page 13

- Bapna et al., Antenna Pointing for High Bandwidth Communications from Mobile Robots, *Paper, Field Robotics Center, The Robotics Institute, Carnegie Mellon University*, Date: unknown, pp. 1–6.
- Barrington Interface, Author: unknown, *Engineering Report*, No. 90–013, Revised: Oct. 1994, pp. 1.
- Beech et al., *AX.25 Link Access Protocol for Amateur Packet Radio*, Version 2.2, *American Relay & Tucson Amateur Packet Radio Corporation*, Jul. 1993, Revised Jul. 1998, pp. 1–143.
- BGE, 5743 Wireless Dual Switch™ Glass Break Detector, Installation and Operating Instructions, *BGE*, Author: unknown; Date: unknown, pp. 1–2.
- BGE, 5746 Wireless Audio Switch™ Glass Break Detector, Installation and Operating Instructions, *BGE*, Author: unknown; Date: unknown, pp. 1–10.
- Bhatnagar et al., Layer netNet: A New Self-Organizing Network Protocol, *Dept. of Electrical Engineering, SUNY; IEEE*; 1990, pp. 1–5.
- Black, Lutron RF Technology, Reliable, First, Forward Thinking, *Lutron Electronics Co. Inc.*, Aug. 2006, pp. 1–16.
- Blaney, HomeRF™ Working Group, 4th Liaison Report, *IEEE 802.11-98/360*, Nov. 1998, Slides 1–12.
- Brain, *How Motes Work: Ad hoc Networks*, available at <http://computer.howstuffworks.com/mote3.htm> on Feb. 25, 2010, pp. 1–3.
- Brain, *How Motes Work*, available at <http://computer.howstuffworks.com/mote.htm>, on Feb. 25, 2010, pp. 1–2.
- Brain, *How Motes Work: The Basic Idea*, available at <http://compute.rhowstuffworks.com/motel.htm>, on Feb. 25, 2010, pp. 1–2.
- Brain, *How Motes Work: Typical Applications*, available at <http://compute.rhowstuffworks.com/mote2.htm>, on Feb. 25, 2010, pp. 1–2.
- Brain, *How Motes Work: A Typical Mote*, available at <http://compute.rhowstuffworks.com/mote4.htm>, on Feb. 25, 2010, pp. 1–2.
- Brayer, Implementation and Performance of Survivable Computer Communication with Autonomous Decentralized Control, *IEEE Communications Magazine*, Jul. 1983, pp. 34–41.
- Bristol Babcock Interfacing*, Author: unknown, *Engineering Report*, No. 95–001, Revised: Apr. 17, 1996, pp. 1–4.
- Brownrigg et al., Development of a Packet-Switching Network for Library Automation, Proceedings of The National Online Meeting Apr. 12–14, 1983), pp. 67–74.
- Brownrigg et al., *Electrons, Electronic Publishing, and Electronic Display, Information Technology and Libraries* (Sep. 1985), pp. 201–207.
- Brownrigg et al., Implementing Library Automation Plans in a University Computing Environment, Planning for Computing in Higher Education 5, *EDUCOM Series in Computing and Telecommunications in Higher Education*, 1980, pp. 215–225.
- Brownrigg et al., Online Catalogues: Through a Glass Darkly, *Information Technology and Libraries*, Mar. 1983, pp. 104–115.
- Brownrigg et al., Packet Radio for Library Automation, *Information Technology and Libraries* 3 (Sep. 1984), pp. 229–244.
- Brownrigg et al., *Packet Radio Networks; Architectures, Protocols, Technologies and Applications* (1987), (introduction pp. ix–xviii); pp. 3–274.
- Brownrigg et al., Packet Switching and Library Automation: A Management Perspective, Proceedings of the 45th ASIS Annual Meeting Oct. 17–21, 1982, vol. 19, pp. 54–57.
- Brownrigg et al., Technical Services in the Age of Electronic Publishing, *Library Resource & Technical Services*, Jan./Mar. 1984, pp. 59–67.
- Brownrigg, The Organization of Computer Resources into a Packet Radio Network, *IEEE*, Jan. 1977, vol. Com–25 No. 1, pp. 169–178.
- Brownrigg et al., User Provided Access to the Internet, available at <http://web.simmons.edu/~chen/nit/NIT'92/033-bro.htm>, Jun. 9, 2005, pp. 1–6.
- Brownrigg, Continuing Development of California State Radio Packet Project, Proceedings of the ASIS 1992 Mid-Year Meeting (Silver-Spring, MD: American Society for Information Science, 1992), pp. 97–100.
- Bruninga, A Worldwide Packet Radio Network, *Signal*, vol. 42, No. 10, Jun. 1988, pp. 221–230.
- Bryan, *Man-Portable Networked Sensor System*, Publisher: unknown, Date: unknown, pp. 1–10.
- Bult et al., A Distributed, Wireless MEMS Technology for Condition Based mMaintenance, *EED, Defense Technical Information Center, UCLA, Electrical Engineering Department, Rockwell Science Center*; Apr. 22–26, 1996.
- Bult et al., *A Distributed, Wireless MEMS Technology for Condition Based Maintenance*, Publisher: unknown; Nov. 1997, pp. 1–8.
- Bult et al. Low Power Systems for Wireless Microsensors, *UCLA Electrical Engineering Department*, 1996 ISLPED, pp. 1–5.
- Bult et al., Low Power Systems for Wireless Microsensors, *EED, UCLA; ILSPED*; 1996, pp. 1–15.
- Bult et al., Wireless Integrated Microsensors, *EED, UCLA Electrical Engineering Department, Rockwell Science Center, TRF*; Jun. 6, 1996, pp. 205–210.
- Caddx Controls, Inc., *NetworX Caddx NetworX NX-8 Control/Communicator Installation Manual*, *Caddx Controls*; Author: unknown; 1996 Jul. 15, 1999, pp. 1–116.
- Caddx Installation Instructions Package, document No. 466–1786, *Caddx Installation Controls, Inc., Caddx Controls*; Author: unknown; Aug. 1998, pp. 1–58.
- Caddx–Caddi Controls, Inc., *Ranger 9000E, User's Manual*, downloaded from <http://www.guardianalarms.net>, May 17, 1996, pp. 1–9.
- Carlisle, Edison's Netcomm Project (Sep. 1988), 1989 *IEEE*, pp. B5–1 to B5–4.
- Case Study: Genentech Uses Coactive's Technology to Centralize Monitor and Control Functions in a Mixed Legacy and New Equipment Environment, *Coactive*, Author: unknown, 1998, pp. 1–4.
- Case Study: Ingham Regional Medical Center Uses Coactive Technology to Monitor and Control Critical Power Generations in a Multi-Campus Environment, *Coactive*, 1998, pp. 1–4.
- Chen, Emerging Home Digital Networking Needs, *Paper, DSP Solutions R&D Center, Texas Instruments, Inc.*, pp. 1–6.
- Chen et al., Route Optimization and Location Updates for Mobile Hosts, 1996 *IEEE*, Proceedings of the 16th ICDCS, pp. 319–326.
- Circon Systems Partners with Coactive Networks to Deliver Circon WebControl™, *Coactive* (press release), Author: unknown; Feb. 7, 2000, pp. 1–4.

US 6,044,062 C1

Page 14

- Circon Technology Connects Building Management Systems to the Internet Using Coactive Routers, *Coactive* (press release); Author: unknown; May 20, 1997.
- Circon Technology Connects Building Management Systems to the Internet Using Coactive Routers, *Coactive* (press release); Author: unknown; May 20, 1997, pp. 3.
- Cisco Systems, Inc., Enhanced Interior Gateway Routing Protocol, Cisco Systems, Inc., Updated Sep. 9, 2005, pp. 1–44.
- Cisco's John Chambers Discusses the Internet Consumer Revolution at CES Using Demo Based on Echelon's LonWorks Technology, *Home Toys* (press release); Author: unknown; Jan. 8, 1999, available at <http://hometoy.s.com/hinews/dec98/releases/echelon04.htm> on Apr. 22, 2009, pp. 1–6.
- Cisco Systems, RFC1812—Requirements for IP Version 4 Routers, Fred Baker ed. (Jun. 1995), available at <http://www.faqs.org/rfcs/rfc1812.html>, Sep. 14, 2009, pp. 1–129.
- Clare et al., Self-Organizing Distributed Sensor Networks, *EED, UCLA, Rockwell Science Center*; Date: unknown, pp. 1–9.
- Clare, Awairs Progress Review: Planned Milestones, *UCLA: Rockwell Science Center*, Nov. 20, 1998, pp. 1–12.
- Clement, Scada System Using Packet Radios Helps to Lower Cincinnati's Telemetry Costs, *Water/Engineering & Management*, Aug. 1996, pp. 18–20.
- Cleveland, Performance and Design Considerations for Mobile Mesh Networks, *Milcom '96 Conference Proceedings*, vol. 1 of 3, Oct. 22–24, 1996, pp. 245–249.
- Clever Solutions—Metricom offers wireless data networks—includes related articles on Metricom's technology and the SOnTech company—Company Profile*, available at http://findarticles.com/p/articles/mi_m0REL/is_n_11_v93/ai_14770465/?tag=content;coll, on Nov. 22, 1993 (3 pages).
- Coactive Bridges Gap between Control Systems and Corporate Data Networks with New Off-the-Shelf Router Family, *Coactive* (press release); Author: unknown; Jun. 8, 1998.
- Coactive Enhances Residential Getaway to Enable Multiple Home Networks, *Coactive* (press release), Author: unknown; Jan. 6, 2000, pp. 1–4.
- Coactive Joins 3Com to Demonstrate Convergence of Control and Enterprise Networks at Retail Systems '98, *Coactive* (press release), Author: unknown; Jun. 16, 1998, pp. 1–4.
- Coactive Launches First Architecture to Support the Convergence Between Control and IP Networks, *Coactive* (press release), Author: unknown; May 20, 1998, pp. 1–4.
- Coactive Leads Standardization Effort for LonTalk Routers, *Coactive* (press release); Author: unknown; May 20, 1997.
- Coactive Leads Standardization Effort for LonTalk/IP Routers, *Coactive* (press release), Author: unknown; May 20, 1997, pp. 3.
- Coactive Networks and Diverse Networks Team to Deliver End-to-End Infrastructure for Enabling the Digital Home, *Coactive* (press release), Author: unknown; Aug. 28, 2000, pp. 1–4.
- Coactive Networks and Innovex Technologies Deliver Internet Access to Home Security, Lighting and Climate Control, *Coactive* (press release), Author: unknown; Feb. 29, 2000, pp. 1–4.
- Coactive Networks and Silicon Energy Partner to Deliver an End-to-End Solution for Internet-Based Energy Monitoring and Analysis, *Coactive* (press release), Author: unknown; Sep. 19, 2000, pp. 1–4.
- Coactive Networks and Vicinium Systems Team to Deliver a Complete Television-Based Interface to Digital Homes and Neighborhoods, *Coactive* (press release), Author: unknown; Jun. 19, 2000, pp. 1–4.
- Coactive Networks Announces First Shipments of Internet Gateway to Home Control Systems, *Coactive* (press release), Author: unknown; May 3, 1999, pp. 1–4.
- Coactive Networks Announces Formation of Technical Advisory Board, *Coactive* (press release), Author: unknown; Oct. 5, 1998, pp. 1–4.
- Coactive Networks Announces System Provider Partner Program, *Coactive* (press release), Author: unknown; Jan. 25, 1999, pp. 1–4.
- Coactive Networks Bridges Gap between Control Systems and Corporate Data Networks with New Off-the-Shelf Router Family, *Coactive* (press release), Author: unknown; Jun. 8, 1998, pp. 1–6.
- Coactive Networks Expands Support for Management and HMI Applications, *Coactive* (press release), Author: unknown; Nov. 2, 1998, pp. 1–4.
- Coactive Networks Introduces Multi-Service Home Control Network Access to U.S. Market, *Coactive* (press release), Author: unknown; Feb. 16, 1999, pp. 1–4.
- Coactive Networks Names Gus Ezcurra Vice President of Sales, *Coactive* (press release), Author: unknown; Jul. 20, 1998, pp. 2.
- Coactive Networks Names Janice Roberts, 3Com Senior VP, to Board of Directors, *Coactive* (press release), Author: unknown; Jun. 2, 1998, pp. 2.
- Coactive Networks Powers Innovative Energy Management Solution, *Coactive* (press release), Author: unknown; Jan. 5, 2001, pp. 1–4.
- Coactive Networks President Named to LonMark Board of Directors; *Coactive* (press release), *Coactive* (press release); Author: unknown; Jun. 14, 1998, pp. 1–3.
- Coactive Networks Shatters Price Barriers with New IP Gateway to Home Control Systems, *Coactive* (press release), Author: unknown; Oct. 26, 1998, pp. 1–4.
- Coactive Networks to Supply Internet-Based Home Gateways for up to 400,000 Customers; First Phase of Deliveries Valued at US\$22 Million, *Coactive* (press release), Author: unknown; Oct. 25, 1999, pp. 1–8.
- Coactive Networks Unveils the First Full-Service Residential Getaway, *Coactive* (press release), Author: unknown; May 3, 2000, pp. 1–4.
- Coactive Networks, Inc., A New Solution for Offering Multiple Telemetry Services to the Home, *Coactive*, 1999, pp. 1–8.
- Coactive Networks, Inc., Coactive Connector® 1000 Series, *Coactive*, 2000, pp. 1–4.
- Coactive Networks, Inc., Coactive Connector® 2000 Series, *Coactive*, Date: unknown, pp. 1–8.
- Coactive Networks, Inc., Connecting Networks to the Real World™, *Coactive*, Date: unknown, pp. 1–4.
- Coactive Networks, Inc., Corporate Background, *Coactive*, 2001, pp. 1–6.
- Coactive Networks, Inc., Corporate Fact Sheet, *Coactive*, 2001, pp. 2.
- Coactive Networks, Inc., Router-LE: Remote Access to LonWorks Over Ethernet, *Coactive*, 1998, pp. 1–4.
- Coactive Networks, Inc., Router-LL: Connect LonWorks Networks Across Internet Protocol, *Coactive*, 1998, pp. 1–4.
- Coactive Networks, Inc., The Coactive Connector® Residential Gateway, Date: unknown, pp. 1–10.

US 6,044,062 C1

Page 15

- Coactive Receives \$2 Million in Funding, *Coactive* (press release), *Coactive* (press release); Author: unknown; Oct. 15, 1997, pp. 3.
- Coactive Receives First Round of Venture Funding Investors Embrace Control Network Connectivity Technology, *Coactive* (press release), Author: unknown, Dec. 1, 1997, pp. 2.
- Cohen et al., IP Addressing and Routing in a Local Wireless Network, 1992 IEEE, 1992, pp. 626–632.
- Cook et al., *Water Distribution and Control by Wireless Networking, Electronic Systems Technology*; Date: unknown, pp. 1–3.
- Corcoran et al., Browser-Style Interfaces to a Home Automation Network, *IEEE Transactions on Consumer Electronics*, vol. 43, No. 4, Nov. 1997, pp. 1063–1069.
- Corcoran et al., CEBus Network Access via the World-Wide-Web, available at http://ieeexploreieee.org/xpl/freeabs_all.jsp?arnumber=517285, on Mar. 29, 2009, Paper Published on *Consumer Electronics*, 1996, Digest of Technical Papers, pp. 236.
- Corcoran et al., *CEBus Network Sccess via the World-Wide-Web, IEEE*, 1996.
- Corcoran et al., *CEBus Network Access via the World-Wide-Web, IEEE*, 1996, pp. 236–237.
- Corson et al., Architectural Considerations for Mobile Mesh Networking, *Milcom '96 Conference Proceedings* vol. 1 of 3, Oct. 22–24, 1996, pp. 225–229.
- Corson et al., Internet-Based Mobile *Ad Hoc* Networking, *IEEE Internet Computing*, Jul.–Aug. 1999, pp. 63–70.
- Custom Solutions, Inc., Accessories, available at http://web.archive.org/web/19981206221844/www.csi3.com/hv_p4.htm on Feb. 27, 2009, pp. 1–3.
- Custom Solutions, Inc., HomeVision 2000 for HomeVision, *Press Release*, available at http://web.archive.org/web/19981207075734/www.csi3.com/HV_PR_0 on Feb. 27, 2009, pp. 1–2.
- Custom Solutions, Inc., HomeVision 2.7 “How To” Information, Date: unknown; pp. 1–146.
- Custom Solutions, Inc., HomeVision 2.7 Auto Report Feature, Date: unknown; pp. 1–10.
- Custom Solutions, Inc., HomeVision 2.7 Interface Command Protocol, Date: unknown; pp. 1–40.
- Custom Solutions, Inc., HomeVision 2.7, Date: unknown; pp. 1–42.
- Custom Solutions, Inc., HomeVision 2.7, *Document Purpose*, Date: unknown; pp. 1–28.
- Custom Solutions, Inc., HomeVision 2.7, *Summary of Changes—2.7*, Date: unknown; pp. 1–26.
- Custom Solutions, Inc., HomeVision 2.7, *Welcome to Home Vision*, Date: unknown; pp. 1–18.
- Custom Solutions, Inc., HomeVision 2.7e, *Owner's Manual* (1999) Date: unknown; pp. 1–596.
- Custom Solutions, Inc., HomeVision 2.7e, *Version History Overview*, Date: unknown; pp. 1–38.
- Custom Solutions, Inc., HomeVision Description, available at <http://web.archive.org/web/19981206004955/http://www.csi3.com/HV.htm> on Mar. 2, 2009, pp. 1–14.
- Custom Solutions, Inc., HomeVision-PC 2.62 Interface Command Protocol, Date: unknown; pp. 1–36.
- Custom Solutions, Inc., HomeVision-PC 2.62, *Document Purpose*, Date: unknown; pp. 1–24.
- Custom Solutions, Inc., HomeVision-PC 2.62, *Summary of Changes—2.62*, Date: unknown; pp. 1–8.
- Custom Solutions, Inc., HomeVision-PC 2.62, *Version History Overview*, Date: unknown; pp. 1–6.
- Custom Solutions, Inc., HomeVision-PC 2.62, *Welcome to HomeVision PC*, Date: unknown; pp. 1–12.
- Custom Solutions, Inc., HomeVision-PC Description, available at http://web.archive.org/web/19981205094024/http://www.csi3.com/hv_pc.htm on Mar. 2, 2009, pp. 1–6.
- Custom Solutions, Inc., HomeVision-PC Software, available at http://web.archive.org/web/19990224053817/http://www.csi3.com/hv_p3pc.htm on Feb. 27, 2009, pp. 1–2.
- Custom Solutions, Inc., HomeVision-PC Version 2.62, *Owner's Manual* (1997) pp. 1–234.
- Custom Solutions, Inc., Media Information, Feb. 16, 1999, available at http://web.archive.com/web/19990502073249/www.csi3.com/hv_media.htm on Feb. 27, 2009, pp. 1.
- Custom Solutions, Inc., Using Enerzone StatNet Thermostats with HomeVision (1998) pp. 1–16.
- Davies et al., The Application of Packet Switching Techniques to Combat Net Radio, *Proceedings of the IEEE*, vol. 75, No. 1, Jan. 1987, pp. 43–55.
- Davies, et al., Internetworking in the Military Environment, *Proceedings of IEEE Infocom '82* (1982) pp. 19–29.
- Davis et al., Knowledge-Based Management of Cellular Clone Fraud, *IEEE* (1992), pp. 230–234.
- Deering et al., *Internet Protocol, Version 6 (IPv6) Specification*, RFC 2460, *The Internet Security*, Dec. 1998 (40 pages).
- Distributions, Networks, and Networking: Options for Dissemination; Workshop on Electronic Texts, Session III, available at <http://palimpsest.stanford.edu/byorg/lc/etextw/sess3.html>, Jul. 17, 2007, pp. 1–10, Author: unknown.
- Dixon et al., Addressing, Bridging and Source Routing, *IEEE Network*, Jan. 1988, vol. 2, No. 1, pp. 25–32.
- Dong et al., Low Power Signal Processing Architectures for Network Microsensors; *UCLA*; Date: unknown; pp. 1–6.
- Dong et al., *Low Power Signal Processing Architectures for Network Microsensors, ACM*, 1997, pp. 173–177.
- DSC-3500 Meeting the Control and Conservation Challenge, Johnson Controls*, 1984, pp. 1–6.
- DTE Energy Technologies Selects Coactive Networks Internet Getaways to Roll Out New Class of E-Services to Businesses, *Coactive* (press release), Author: unknown, May 3, 2000, pp. 1–4.
- DTE Energy Technologies Selects Coactive Networks to Power Distributed Generation Solutions Worldwide, *Coactive* (press release), Author: unknown, Aug. 1, 2001, pp. 1–4.
- Echelon Corporation Demonstrates Internet Connectivity in Digital Home Applications at 1999 International Consumer Electronics Show, *Home Toys* (press release); Author: unknown; Dec. 15, 1998, available at <http://hometoys.com/hтинews/dec98/releases/echelon03.htm> on Apr. 22, 2009, pp. 1–4.
- Eight Leading Controls Companies Join Coactive Partner Program, *Coactive* (press release), Author: unknown, Aug. 21, 2000, pp. 1–4.
- Elson et al., Fine-Grained Nnetwork Time Synchronization Using Reference Broadcasts; *UCLA Computer Science Department*; May 17, 2002, pp. 1–14.
- Eng et al., Bahama: A Broadband Ad-Hoc Wireless ATM Local-Area Network, 1995 IEEE International Conference on Communications, Jun. 18–22, 1995, pp. 1216–1223.
- Enhanced Position Location Reporting System (EPLRS)*, Author: unknown, available at <http://www.globalsecurity.org/military/systems/ground/eplrs.htm> on Feb. 22, 2010, pp. 1–3.

US 6,044,062 C1

Page 16

- Ephremides et al., A Design Concept for Reliable Mobile Radio Networks with a Frequency Hopping Signaling, *IEEE*; 1987, pp. 56–73.
- ESTeem Engineering Report, Johnson Controls Interface No. 91–102*, Author: unknown, Publisher: unknown, Nov. 1994.
- ESTeem Model 96F*, Author: unknown, *ESTeem Radios*; Sep. 6, 1996 (2 pages).
- Estrin et al., Next Century Challenges: Scalable Coordination in Sensor Networks, *ACM*, 1999, pp. 263–270.
- Estrin et al., RFC1940–Source Demand Routing: Packet Format and Forwarding Specification (Version 1), Network Working Group, May 1996, available at <http://www.faqs.org/rfcs/rfc1940.html>, Sep. 14, 2009, pp. 1–20.
- Estrin et al., Source Demand Routing: Packet Format and Forwarding Specification (Version 1), Network Working Group, *Internet Draft*, Jan. 19, 1995, pp. 1–28.
- Expert Report of Randy H. Katz, Ph. D, of *SIPCO, LLC et al. v. The Toro Company et al.*, Case No. 2:08–cv–00505.
- Fort Riley Water Distribution Monitoring Control System Drawings 1–30*, Author: unknown; Publisher: unknown; Date: unknown.
- Foxboro Interfacing*, Author: unknown, *Engineering Report*, No. 91–023, Revised: Jun. 19, 1996, pp. 1–5.
- Frank, Transmission of IP Datagrams Over NET/ROM Networks, ARRL Amateur Radio 7th Computer Networking Conference, Oct. 1988, pp. 65–70.
- Frankel, Packet Radios Provide Link for Distributed Survivable Command Control Communications in Post–Attack Scenarios, *Microwave System News*, Jun. 1983, Circle Reader Service No. 77, pp. 80–108.
- Franz, HiperLAN—Der ETSI–Standard für locale Funknetze, *NTZ*, Sep. 1995, 10 pages.
- Fullmer, Collision Avoidance Techniques for Packet–Radio Networks, *Dissertation*, University of California at Santa Cruz, Jun. 1998, pp. 1–162.
- Gale et al., The Impact of Optical Media on Information Publishing, *Bulletin of the American Society For Information Science*, vol. 12, No. 6, Aug./Sep. 1986, pp. 12–14.
- Garbee, Thoughts on the Issues of Address Resolution and Routing in Amateur Packet Radio TCP/IP Networks, ARRL Amateur Radio 6th Computer Networking Conference, Aug. 1987, pp. 56–58.
- García–Luna–Aceves et al., Wireless Internet Gateways (Wings), 1997 *IEEE*, pp. 1271–1276.
- García–Luna–Aceves, A Fail–Safe Routing Algorithm for Multishop Packet–Radio Networks, *IEEE Infocom '86*, Technical Sessions: Apr. 8–10, 1986, pp. 434–442.
- García–Luna–Aceves, A Minimum–hop Routing Algorithm Based on Distributed Information, Elsevier Science Publishers, B.V. (North Holland), 1989, pp. 367–382.
- García–Luna–Aceves, Routing Management in Very Large Scale Networks, Elsevier Science Publishers, B.V. (North Holland), 1988, pp. 81–93.
- GE Fanuc Interfacing*, Author: unknown, *Engineering Report*, No. 91–010, Revised: Apr. 11, 1996, pp. 1–8.
- General PLC/RTU Interfacing*, Author: unknown, *Engineering Report*, No. 92–010, Revised: Jun. 18, 1996, pp. 1–5.
- GE Security, NetworX NX–4, 2004, pp. 1–2.
- GE Security, NetworX NX–548E, 2006, pp. 1–2.
- Geier et al., Networking Routing Techniques and their Relevance to Packet Radio Networks, ARRL/CRRL Amateur Radio 6th Computer Networking Conference, London, Ontario, Canada, Sep. 1990, pp. 105–117.
- Gerla et al., Multicluster, mobile, multimedia radio network, *CSD, UCLA; Blatzer Journals*; Jul. 12, 1995, pp. 1–26.
- Gerla et al., Multicluster, Mobile, Multimedia Radio Network, UCLA Computer Science Department; Baltzer Journals; *Wireless Networks*; Jul. 12, 1995, pp. 255–265.
- Golden Power Manufacturing, 6030 PCT Programmable Communicating Thermostat, Author: unknown, 2007, pp. 1–3.
- Golden Power Manufacturing, Ritetemp Universal Wireless Thermostat, Author: unknown, 2007, pp. 1–2.
- Goldman et al., *Impact of Information and Communications Technologies on Residential Customer Energy Services*, Paper, *Berkeley: UCLA*, Oct. 1996, pp. 1–89.
- Grady et al., *Telemetry Options for Small Water Systems*, Special Report SR14–1999, Publisher: unknown, Sep. 1999, pp. 1–23.
- Guardian Alarms, Inc., Home Security System—Model 7068 Digital Dialer Interface, Author: unknown, available at www.guardianalarms.net, 2007, pp. 1.
- Guardian Alarms, Inc., Security Company—Home Alarm System Monitoring—AES 7067 IntelliTap–II Digital Dialer Interface, Author: unknown, available at www.guardianalarms.net, 2007, pp. 1.
- Guardian Alarms, Inc., Security System—Alarm System Monitoring—7160 EZ Router, Author: unknown, available at www.guardianalarms.net, 2007, pp. 1.
- Guardian Alarms, Inc., Security System—Alarm System Monitoring—Net 7000, Author: unknown, available at <http://www.guardianalarms.net>, 2007, pp. 1.
- Guardian Alarms, Inc., Security System—Alarm System Monitoring—Radionics FDX, Author: unknown, available at www.guardianalarms.net, 2007, pp. 1.
- Haartsen, Bluetooth—*The Universal Radio Interface for Ad Hoc, Wireless Connectivity*, *Ericsson Review*, No. 3, 1998, pp. 110–117.
- Haartsen et al., *Bluetooth: Vision, Goals and Architecture*, *Mobile Computing and Communications Review*, vol. 1, No. 2, Date: unknown, pp. 1–8.
- Hahn et al., Packet Radio Network Routing Algorithms: A Survey, *IEEE Communications Magazine*, vol. 22, No. 11, Nov. 1984, pp. 41–47.
- Hai Omni, Features & Specifications, *Home Automation, Inc.*, available at <http://web.archive.org/web/19970216055832/www.homeauto.com/omni> on Feb. 17, 2009, pp. 1–6.
- Hall, Tactical Internet System Architecture for Task Force XXI, 1996 *IEEE*, pp. 219–230.
- Hamilton et al., Optimal Routing in Multihop Packet Radio Networks, 1990 *IEEE*, pp. 389–396.
- Harrison, Microwave Radio In The British Telecom Access Network, Second IEE National Conference on Telecommunications, Conference Publication No. 300, Date: unknown, pp. 208–213.
- Hedrick, An Introduction To IGRP, Rutgers, The State University of New Jersey, Center for Computers and Information Services, Laboratory for Computer Science Research, Aug. 22, 1991 (Updated Aug. 10, 2005), pp. 1–21.
- Hedrick, Routing Information Protocol (Jun. 1988), RFC 1058, available at [Http://Tools.Ietf.Org/Html/Rfc1058](http://Tools.Ietf.Org/Html/Rfc1058), Jun. 24, 2009, pp. 1–34.
- Hinden et al., The DARPA Internet Gateway, RFC 823, Publisher: unknown, Sep. 1982, pp. 1–43.

- Holtville et al., *Symbol Technologies, Telxon and Aironet Commit to Future Interoperability of Their Wireless Local Area Networks Based on the IEEE 802.11 Specification*, *Business Wire*, Jun. 24, 1996, available at http://www.thefreelibrary.com/_/print/PrintArticle.aspx?id=18414624, pp. 1–3.
- Hsu et al., *Wireless Communications for Smart Dust*, Berkeley: UCLA, Jan. 30, 1998, pp. 1–20.
- Home Automation, Inc., *Home Page*, HAI Omni: Features & Specifications, *Home Automation, Inc.* available at [http://web.archive.org/web/19961219004403/http://www.homeauto.com\(archived web page\)](http://web.archive.org/web/19961219004403/http://www.homeauto.com(archived%20web%20page)) on Feb. 17, 2009; Author: unknown; pp. 1.
- Home Automation, Inc., HAI Company Background; Publisher: Unknown; Date: unknown; pp. 1–2.
- Home Toys, Inc., HTINews Review, available at <http://www.hometoys.com/htinews/aug97/reviews/homevis/homevis1.htm> on Mar. 2, 2009, pp. 1–26.
- Honeywell, Inc., Honeywell Home Control Version 2.0 Demonstration, available at <http://web.archive.org/web/19980630195929/www.hbc.honeywell.com/> on Mar. 5, 2009, (7 pages).
- Hong et al., U.S. Lighting Market Characterization, vol. II: Energy Efficient Lighting Technology Options, Sep. 30, 2005, Report prepared for Building Technologies Program, Office of Energy Efficiency and Renewable Energy, pp. 1–36.
- Hotel Technology Next Generation, *A Guide for Understanding Wireless in Hospitality, An HTNG White Paper*, Jun. 2006, (Jayne O'Neill, ed.), pp. 1–77.
- How Does the New Power Company Deliver on the Promise of Energy Reconstructing?, *NewPower* (press release), Author: unknown, May 31, 2001, pp. 1–6.
- Hruschka et al., Packet Radio, Drahtlose Datenübertragung im Amateurfunk, *Elektronik*, Jun. 1991, pp. 54–57 and 84.
- Hubner et al., A Distributed Multihop Protocol for Mobile Stations to Contact a Stationary Infrastructure, The Third IEE Conference on Telecommunications, Conference Publication No. 331, Date: unknown, pp. 204–207.
- Humpal, *Extended Timers for Fort Riley*, Publisher: unknown; Mar. 1993.
- Humpal, *Modified Download Files for Fort Riley*, Publisher: unknown; Apr. 1994.
- IIS—Contract Detail, *Project Name: Ft. Riley Radio Expansion*, Author: unknown, *Johnson Controls*, Sep. 1998.
- Important Dealer Notification—Honeywell AlarmNet—M Network Alert, Source: unknown; Author: unknown; Apr. 2007, pp. 1.
- Industrial Communications*, Author: unknown, available at <http://web.archive.org/web/19990222162354/www.metrocom.com/industrial/> on May 10, 2010, pp. 1–3.
- Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, Author: unknown, *IEEE*, Std. 802.11–1997, 1997, pp. 1–445.
- Information Sciences Institute (UCLA), *Internet Protocol (IPv4) Defense Advanced Research Projects Agency*, Sep. 1981.
- IOConnect Architecture™, *Coactive*, 2001, pp. 1–4.
- Iwata et al., *Scalable Routing Strategies for Ad Hoc Wireless Networks*, *IEEE Journal on Selected Areas in Communications*, vol. 17, No. 8, Aug. 1999, pp. 1369–1379.
- Jacobsen, The Building Blocks of a Smart Sensor for Distributed Control Networks, *IEEE Technical Applications Conference Northcon*, Nov. 4–6, 1998, pp. 285–290.
- JC/83RF System: Cost-effective Multiple Facility Management by Radio Network*, *Johnson Controls*, Date: unknown, pp. 2214–2219.
- JC/83RF System: Multiple Facility Management by Radio Network*, *Johnson Controls*, Publication No. 2161, 1983, pp. 1–4.
- JDS Technologies, Infrared Xpander, IR-XP², *User Manual*, Date: unknown; pp. 1–15.
- JDS Technologies, Model: 8R5PR, 8 Channel RS485 Relay Xpander, *Installation Manual*, pp. 1–5.
- JDS Technologies, Stargate 8 Channel RS-485 HUB, Publisher: unknown; Date: unknown, pp. 1.
- JDS Technologies, Stargate Interactive Automation System, 1998, pp. 1–2.
- JDS Technologies, Stargate, *Operation Manual*, Mar. 2000, pp. 1–114.
- JDS Technologies, Stargate-IP System Layout, Publisher: unknown; Date: unknown, pp. 1.
- JDS Technologies, *Support: Protocol Specifications*, available at <http://jdstechologies.com/protocol.htm>, on Feb. 16, 2009, pp. 1–32.
- JDS Technologies, TimeCommander, TimeCommander Plus, *User Guide*, Jun. 1998, pp. 1–95.
- JDS Technologies, Web Xpander, Installation and Operation Manual, Feb. 2004, pp. 1–34.
- Jimenez-Cedeno et al., Centralized Packet Radio Network: A Communication Approach Suited for Data Collection in a Real-Time Flash Flood Prediction System, *ACM-SAC* 1993, pp. 709–713.
- Johnson Controls Interface*, Author: unknown, *Engineering Report*, No. 91–012, Revised: Nov. 1994, pp. 1–14.
- Johnson, Mobile Host Internetworking Using IP Loose Source Routing, Carnegie Mellon University CMU-CS-93-128, DARPA Order No. 7330, Feb. 1993, pp. 1–18.
- Johnson, Routing in *Ad Hoc* Networks of Mobile Hosts, 1995 IEEE, pp. 158–163.
- Johnson, Scalable and Robust Internetwork Routing for Mobile Hosts, 1994 IEEE, pp. 1–11.
- Johnson Controls, Inc., LonWorks® Digital Controller, 1998, pp. 1–12.
- Johnson et al., Dynamic Source Routing in *Ad Hoc* Wireless Networks, *Paper*, Publisher: unknown, pp. 1–18.
- Johnson et al., Dynamic Source Routing in *Ad Hoc* Wireless Networks, reprinted in *Mobile Computing*; Tomasz Imielinski and Hank Korth eds., 1996; Kluwer Academic Publishers, pp. 153–181.
- Johnson et al., Protocols for Adaptive Wireless and Mobile Networking, *IEEE Personal Communications*, 3(1), Feb. 1996, pp. 1–18.
- Johnson et al., Route Optimization in Mobile IP, *Internet Draft* (Nov. 28, 1994), available at <http://www.monarch.cs.rice.edu/internet-drafts/draft-ietf-mobileip-optim-00.txt>, Sep. 26, 2009, pp. 1–29.
- Jubin et al., The DARPA Packet Radio Network Protocols, *Proceedings of the IEEE*, vol. 75, No. 1, Jan. 1987, pp. 21–32.
- Kaashoek et al., FLIP: An Internetwork Protocol for Supporting Distributed Systems, *ACM Transactions on Computer Systems*, vol. 11, No. 1, Feb. 1993, pp. 73–106.

US 6,044,062 C1

Page 18

- Kaiser, *Circuits and Systems for Embedded Wireless Devices: Low Power Sensor, Interface, Signal Processing, Communication, and Network Systems*, École Polytechnique Fédérale de Lausanne, pp. 1–40.
- Kaiser, *Embedded Wireless Devices: Sensors*, Outline, École Polytechnique Fédérale de Lausanne, pp. 1–53.
- Kaiser, *Embedded Wireless Devices: Signal Processing*, Outline, École Polytechnique Fédérale de Lausanne, pp. 1–19.
- Kaiser, *Embedded Wireless Devices: Wireless Networking*, Outline, École Polytechnique Fédérale de Lausanne, pp. 1–16.
- Kaiser, *Embedded Wireless Devices: Wireless Physical Layer*, Outline, École Polytechnique Fédérale de Lausanne, pp. 1–29.
- Kaiser et al., Low Power Wireless Integrated Microsensor (LWIM), *Progam Mission, UCLA*; Jan. 1997.
- Kaiser et al., Low Power Wireless Integrated Microsensors (LWIM), BAA 94–15 Proposal Abstract, UCLA Electrical Engineering Department, Rockwell Science Center, Date: unknown, 15 pages.
- Kaiser et al., Low Power Wireless Integrated Microsensors (LWIM), Request for Support to Project, UCLA Electrical Engineering Department, Rockwell Science Center, Sep. 13, 1994, 71 pages.
- Kaiser et al., Low Power Wireless Integrated Microsensors (LWIM); UCLA; Rockwell Science Center; LWIM Kickoff Meeting, Aug. 8, 1995, Presented to Dr. Ken Gabriel (ARPA), Dr. Elissa Sobolewski (ARPA), and Dr. Joseph Kielman (FBI), 62 pages.
- Karn et al., Packet Radio in the Amateur Service, *IEEE Journal on Selected Areas in Communications*, vol. SAC–3, No. 3, May 1985, pp. 431–439.
- Katz et al., The Bay Area Research Wireless Access Network (BARWAN) (Jun. 1996) (presentation paper), <http://daedalus.cs.berkeley.edu/talks/retreat.6.97/BARWAN.597.ppt>, pp. 1–66.
- Katz et al., The Bay Area Research Wireless Access Network (BARWAN), University of California at Berkeley, available at http://www.cs.berkeley.edu/~randy/Daedalus/BARWAN/BARWAN_index.html, 6 pages.
- Keltron's Home Page With Frames, *Index*, available at <http://web.archive.org/web/19990831161957/http://www.keltroncorp.com>, on Mar. 24, 2009, pp. 1.
- Kemp, Home Automation Application Guide, Applications for Home Automation in Any Home, vol. 1, 2000, pp. 1–106.
- Khan, Robert E., Issues in Distributed Routing for Mobile Packet Radio Network, *IEEE* 1982, pp. 233–238.
- Kleinrock et al., Hierarchical Routing for Large Networks, Performance Evaluation, and Optimization, *Computer Networks 1* (1977), pp. 155–174.
- Kocom, Digital Home Network, Kitchen TV Phone KTD–505, *User's Manual*, pp. 1–7.
- Kohno et al., An Adaptive Sensor Network System for Complex Environments, in *Intelligent Autonomous Systems* (Kakazu et al., eds.), *IOS Press*, 1998, pp. 21–28.
- Krishnamachari, *Networking Wireless Sensors*, Cambridge University Press, Date: unknown, pp. 1–10.
- Lacoss, *Distributed Sensor Networks*, Final Report, *Lincoln Laboratory at Massachusetts Institute of Technology*, Sep. 30, 1986, pp. 1–225.
- Lauer et al., Survivable Protocols for Large Scale Packet Radio Networks, *IEEE Global Telecommunications Conference*, Nov. 26–29, 1984, vol. 1 of 3, pp. 468–471.
- Lauer, Packet–Radio Routing, *Routing in Communications Networks*, Ch. 11 (1995) pp. 351–396.
- Lee et al., Distributed Measurement and Control Based on the IEEE 1451 Smart Transducer Interface Standards, *Proceedings of the 16th IEEE Instrumentation and Measurement Technology Conference*, vol. 1, May 24–26, 1999, *IEEE*, pp. 608–613.
- Leiner et al., *Goals and Challenges of the DARPA GloMo Program*, *IEEE Personal Communications*, Dec. 1996, pp. 34–43.
- Leviton Manufacturing Co., Inc., *The DECORA® Collection of Designer Devices*, 2006, pp. 1–85.
- Lewis et al., Packet–Switching Applique for Tactical VHF Radios, 1987 *IEEE Military Communications Conference*, Oct. 19–22, 1987, *Conference Record* vol. 2 of 3, pp. 449–455.
- Lin et al., Adaptive Clustering for Mobile Wireless Networks; Publisher: unknown; Date: unknown; pp. 1–21.
- Lin et al., CMOS Front End Components for Micropower RF Wireless Systems; EED, *UCLA Electrical Engineering Department*; 1998, pp. 1–5.
- Lin et al., Wireless Integrated Network Sensors (WINS) for Tactical Information Systems, *UCLA, Rockwell Science Center*; Date: unknown; pp. 1–5.
- Linear Corporation, Supervised Digital Security Transmitters TX–91, TX–92, TX–94, *Operation Instructions*, 1993, pp. 1.
- Linear Corporation, Supervised Digital Security Transmitter T–90, *Installation Instructions*, 2006, pp. 1–2.
- Linear Corporation, Supervised Wireless Receiver and Zone Expander SRX–64A, *Installation Instructions*, 2003, pp. 1–2.
- Local and Metropolitan Area Networks: Wireless Medium Access Control (MAC) and Physical (PHY) Specifications, Annex A: Protocol Implementation Conformance Statement (PICS) Proforma, Author: unknown; *IEEE*, Nov. 1997, pp. 1–75.
- Local and Metropolitan Area Networks: Wireless Medium Access Control (MAC) and Physical (PHY) Specifications, Author: unknown; *IEEE*, Nov. 1997, pp. 1–98.
- Long Range Radio, *AlarmNet®*, Author: unknown; Date: unknown, pp. 1–10.
- LonTalk Protocol, LonWorks™ Engineering Bulletin, *Echelon Corp.*; Author: unknown; Apr. 1993, pp. 1–27.
- LonWorks® Products, 1998, *Version A, Echelon Corp.*, pp. 1–21.
- LonWorks® Router User's Guide, *Echelon Corp.*; Author: unknown; 1995, pp. 1–136.
- LonWorks® SMX™ Transceiver, datasheet, *Echelon Corp.*; Author: unknown; 1997, pp. 1–18.
- Lougheed et al., A Border Gateway Protocol 3 (BGP–3), RFC 1267, (Oct. 1991), available at <http://tools.ietf.org/html/rfc1267>, Jun. 24, 2009, pp. 1–36.
- Lowe et al., Publishing Bibliographic Data on Optical Disks: A Prototypical Application and Its Implications, Third International Conference on Optical Mass Data Storage, *Proceedings of SPIE*, vol. 529, pp. 227–236.
- Lutron Electronics Co. Inc., *Connecting to a RadioRA System via a Local Area Network*, Application Note #127, unknown, pp. 1–16.
- Lutron Electronics Co. Inc., *Homeowner's Guide for the RadioRA® Quick Start Package*, 2004, pp. 1–8.

US 6,044,062 C1

Page 19

- Lutron Electronics Co. Inc., *How to retrofit RadioRA® Wall-Mounted Master Control into an existing home*, Application #41, 2004, pp. 1–2.
- Lutron Electronics Co. Inc., *Interfacing RadioRA® to Security and Fire Alarm Systems*, Application Note #59, pp. 1–4.
- Lutron Electronics Co. Inc., *IR/RS232 Interface for Bang & Olufsen® Beo4® Remote Control and RadioRA®*, Application Note #199, 2004, pp. 1–3.
- Lutron Electronics Co. Inc., *Level Capture with a RadioRA® Master Control*, Application Note #73, 2003, pp. 1–3.
- Lutron Electronics Co. Inc., *Modem Installation for HomeWorks®*, Application Note #9, 1998, pp. 1–4.
- Lutron Electronics Co. Inc., *RadioRA® RA-IR-KIT Installation Instructions*, Application Note #61, 2000, pp. 1–4.
- Lutron Electronics Co. Inc., *RadioRA® RF Signal Repeater*, 1998, pp. 1–2.
- Lutron Electronics Co. Inc., *RadioRA® Single-Location Switch, Controls for Permanently Installed Lighting Loads*, 1998, pp. 1–2.
- Lutron Electronics Co. Inc., *RadioRA® Table Lamp Controls, Dimming and Switching Controls for Table and Floor Lamps*, 1999, pp. 1–2.
- Lutron Electronics Co. Inc., *Using a Photocell with the RadioRA® System*, Application Note #45, 1998, pp. 1–4.
- Lutron Electronics Co. Inc., *Using an Astronomic Timeclock with the RadioRA® System*, Application Note #42, 1998, pp. 1–2.
- Lutron Electronics Co. Inc., *Using the RadioRA® System to Activate Scenes 5–16 on a GRAFIK Eye® Control Unit*, Application Note #48, 1998, pp. 1–4.
- Lutron Electronics Co. Inc., *Using the RadioRA® Telephone Interface*, Application Note #46, 1998, pp. 1–2.
- Lynch et al., *Application of Data Compression Techniques to a Large Bibliographic Database*, Proceeding of the Seventh International Conference on Very Large Database, Cannes, France, Sep. 9–11, 1981 (Washington, DC: IEEE Computer Society Press, 1981), pp. 435–447.
- Lynch et al., *Beyond the Integrated Library System Concept: Bibliographic Networking at the University of California*, Proceedings of the Second National Conference on Integrated Online Library Systems Proceedings, Sep. 1984, pp. 243–252.
- Lynch et al., *Conservation, Preservation and Digitization, Energies for Transition: Proceedings of the Fourth National Conference of the Association of College and Research Libraries*, Baltimore, MD, Apr. 9–12, 1986 (Chicago, IL: Association of College and Research Libraries, 1986), pp. 225–228.
- Lynch et al., *Document Delivery and Packet Facsimile*, Proceedings of the 48th ASIS Annual Meeting, vol. 22, Oct. 20–24, 1985, pp. 11–14.
- Lynch et al., *Electronic Publishing, Electronic Imaging, and Document Delivery, Electronic Imaging '86* (Boston, MA: Institute for Graphic Communication, Inc., 1986), pp. 662–667.
- Lynch et al., *Library Applications of Electronic Imaging Technology, Information Technology and Libraries*, Jun. 1986, pp. 100–105.
- Lynch et al., *Packet Radio Networks: Architectures, Protocols, Technologies and Applications*, Pergamon Press, 1ed., 1987, pp. 1–275.
- Lynch et al., *Public Access Bibliographic Databases in a Multicampus University Environment, Databases in the Humanities and Social Sciences—4*, Proceedings of the International Conference on Databases in the Humanities and Social Sciences, Jul. 1987, Learned Information, Inc., 1989, pp. 411–419.
- Lynch et al., *The Telecommunications Landscape: 1986 Library Journal*, Oct. 1, 1986, pp. 40–46.
- M100 Series Motor Actuator*, Author: unknown, *Johnson Controls, Inc.*; Apr. 1993, pp. 1–20.
- M100C Series Actuator with Digital Control Signal Input and R81CAA-2 Interface Board*, Installation Bulletin, *Johnson Controls*, 2000, pp. 1–12.
- Mak et al., *Design Consideration for Implementation of Large Scale Automatic Meter Reading Systems*, *IEEE Transactions of Power Delivery*, vol. 10, No. 1, Jan. 1995, pp. 97–103.
- Mak et al., *Design Considerations for Implementation of Large Scale Automatic Meter Reading Systems IEEE Transactions on Power Delivery*, vol. 10, No. 1, Jan. 1995, pp. 97–103.
- Malkin, RFC 2453, RIP Version 2 (Nov. 1998), available at <http://tools.ietf.org/html/rfc2453>, Jun. 24, 2009, pp. 1–40.
- Maltz, *Experiences Designing and Building a Multi-Hop Wireless Ad Hoc Network Testbed, Paper*, Mar. 5, 1999, School of Computer Science Carnegie Mellon University, pp. 1–20.
- Maltz et al., *Experiences Designing and Building a Multi-Hop Wireless Ad Hoc Network Testbed, Paper*, Mar. 5, 1999, available at <http://reportsarchive.adm.cs.cmu.edu/anon/1999/CMU-C S-99-116.pdf>, pp. 1–20.
- Maltz, *On-Demand Routing in Multi-Hop Wireless Mobile Ad Hoc Networks, Thesis*, May 2001, pp. 1–192.
- Man-Portable Networked Sensor System* (1997–), Author: unknown, available at <http://www.spawar.navy.mil/depts/d30/d37/d371/mpnss/mpnss.html> on May 20, 2010, pp. 1–4.
- March of the Motes*, Author: unknown, *New Scientist*, vol. 179, issue 2409, Aug. 23, 2003, pp. 26.
- Marcy et al., *Wireless Sensor Networks for Area Monitoring and Integrated Vehicle Health Management Applications; Rockwell Science Center*, Thousand Oaks, CA, AIAA-99-4557; Date: unknown, pp. 1–11.
- Markie et al., *LonWorks and PC/104: A winning Combination, PC/104 Embedded Solutions*; Summer 1998, pp. 1–8.
- Martel et al., *Home Automation Report: A Modular Minimum Complexity, High-Resolution and Low Cost Field Device Implementation for Home Automation and Healthcare, MIT*; Publisher: unknown; Mar. 31, 1998; pp. 1–29.
- McQuillan et al., *The ARPA Network Design Decisions; Computer Networks*, vol. 1, No. 5, Aug. 1977 pp. 243–289.
- McQuillan et al., *The New Routing Algorithm for the ARPANET*, *IEEE Transactions on Communications*, vol. COM-28, No. 5, May 1980, pp. 711–719.
- Metasys Compatible Products*, Author: unknown; *Johnson Controls, Inc.*, 1997 (9 pages).
- Metasys Extended System Architecture*, vol. II, Author: unknown, Publisher: unknown, Sep. 1999.
- Metasys N2 System Protocol Specification for Vendors*, Author: unknown, Publisher: unknown, Jun. 1996.
- Mills, *Exterior Gateway Protocol Formal Specification* (Apr. 1984), RFC 904, available at <http://tools.ietf.org/html/rfc904>, Jun. 24, 2009, pp. 1–32.
- Modicon Interfacing*, Author: unknown, *Engineering Report*, No. 90-022, Revised: Apr. 12, 1996, pp. 1–9.

US 6,044,062 C1

Page 20

- Moore Products—Hart Protocol Interfacing, Author: unknown, *Engineering Report*, No. 94-007, Revised: Mar. 1, 1996, pp. 1-3.
- Moorman, *Packet Radio Used in a Cost-Effective Automated Weather Meso-Net*, available at <http://www.wrh.noaa.gov/wrh/96TAs/TA9631/ta96-31.html>, Dec. 3, 1996 (5 pages).
- Moy, RFC 2328, OSPF Version 2 (Apr. 1998), available at <http://tools.ietf.org/html/rfc2328>, Jun. 24, 2009, pp. 1-245.
- Mozer et al., The Neural Network House: An Overview, in L. Niklasson & Boden (Eds.), *Current trends in connectionism* (pp. 371-380); Hillsdale: Erlbaum, 1995; pp. 1-9.
- MTC Teams with Coactive Networks to Deliver an Advanced Energy Communications and Management Solution, *Coactive* (press release), Author: unknown, Feb. 5, 2001, pp. 1-4.
- Murthy et al., An Efficient Routing Protocol for Wireless Networks, *Mobile Networks and Applications 1* (1996), pp. 183-197.
- Natkunanathan et al., A Signal Search Engine for Wireless Integrated Network Sensors, EED, *UCLA Electrical Engineering Department*; Date: unknown; pp. 1-4.
- Natunathan et al., Wins: Signal Search Engine for Signal Classification, EED, *UCLA*; Date: unknown; pp. 1-6.
- Negus et al, HomeRF™ and SWAP: Wireless Networking for the Connected Home, *ACM Sigmoblie Mobile Computing and Communications Review*, vol. 2, Issue 4, Oct. 1998, available at <http://portal.acm.org/citation.cfm?id=1321400.1321401>, on Mar. 29, 2009, pp. 1-2.
- Negus et al., HomeRF™ and SWAP: Wireless Networking for the Connected Home, *Mobile Computing and Communications Review*, vol. 2, No. 4, Date: unknown, pp. 28-37.
- Network Working Group, *Internet Protocol (IPv6)*, *Internet Engineering Task Force*, 1998.
- NewPower and Coactive Networks Announce Strategic Alliance to Deliver the Connected Home, *Coactive* (press release), Author: unknown, Mar. 14, 2001, pp. 1-4.
- Nextgen Searches, *IPCO v. The Wireless Sensor Network Industry?* Special Report on *IPCOv. Oncor et al.*, Corporate Manager's Edition, 2009, pp. 1-16.
- NX-480 Wireless Motion Sensor, document No. 466-1479 Rev. D; *Caddx Controls*; AuAuthor: unknown; Caddx Controls, Inc.; May 1, 1998, pp. 1.
- Omni Automation System, Author: unknown; *Home Automation, Inc.*, Date: unknown, pp. 1-266.
- Omni Instalation Manual, Author: unknown; *Home Automation, Inc.*, Oct. 1997, pp. 1-88.
- Omni Owner's Manual, Author: unknown; *Home Automation, Inc.*, Date: unknown, pp. 1-136.
- Omni user Manual, *Home Automation, Inc.*; Author: unknown; 1997.
- Omron Interfacing, Author: unknown, *Engineering Report*, No. 95-003, Revised: Apr. 17, 1996, pp. 1-4.
- Ondo, *PLRS/JTIDS Hybrid, Filled Artillery Journal*, Jan.-Feb. 1981, pp. 20-25.
- Opto-22 Protocol, Author: unknown, *Engineering Report*, No. 93-010, Revised: May 31, 1996, pp. 1-8.
- Oran (ed.), OSI IS-IS Intra-Domain Routing Protocol, RFC 1142 (Feb. 1990), available at <http://tools.ietf.org/html/rfc1142>, Jun. 24, 2009, pp. 1-665.
- Park et al., SensorSim: A Simulation Framework for Sensor Networks, *ACM*, 2000, pp. 104-111.
- Perkins et al., A Mobile Networking System Based on Internet Protocol, Publisher: unknown, Date: unknown, pp. 1-17.
- Perkins et al., *Ad-Hoc On-Demand Distance Vector Routing "AODV"*, <http://moment.cs.ucsb.edu/AODV/aodv.html>, Aug. 25, 2009, pp. 1-5.
- Perkins et al., Continuous, transparent network access for portable users, A Mobile Networking System Based on Internet Protocol, IEEE Personal Communications, First Quarter 1994, pp. 32-41.
- Perkins et al., Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers; SIGCOM Conference on Communications Architectures, Protocols and Applications, London England UK (Aug. 1994); pp. 234-244.
- Perkins et al., Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, ACM SIGCOMM Computer Communications Review archive vol. 24, Issue 4 (Oct. 1994), pp. 234-244.
- Perkins et al., Mobility Support in IPv6, *Internet Draft* (Sep. 22, 1994), available at <http://www.monarch.cs.rice.edu/internet-draft/draft-perkins-ipv6-mobility-sup-oo.txt>, Sep. 26, 2009, pp. 1-13.
- Perkins et al., RFC3561—*Ad Hoc On-Demand Distance Vector (AODV) Routing* (Jul. 2003), available at <http://tools.ietf.org/html/rfc3561>, Aug. 25, 2009, pp. 1-38.
- Phoenix Contact Interfacing*, Author: unknown, *Engineering Report*, No. 94-001, Revised: Jun. 20, 1996, pp. 1-7.
- Pittway Corporation, Company History, available at <http://www.fundinguniverse.com/company-histories/Pittway-Corporation> Mar. 6, 2009, pp. 1-5.
- PLC Direct (Koyo) Interfacing*, Author: unknown, *Engineering Report*, No. 96-001, Revised: Apr. 10, 1996, pp. 1-8.
- Postel (Editor), Internet Protocol, DARPA Internet Program Protocol Specification, RFC 791 (Sep. 1981), Information Sciences Institute, University of So. Cal., pp. 1-45.
- Postel (Editor), Internet Protocol, DARPA Internet Program Protocol Specification, RFC 791 (Sep. 1981), Information Sciences Institute, University of So. Cal., available at <http://www.rfc-editor.org/rfc/rfc791.txt> on Sep. 14, 2009, 51 pages.
- Postel (ed.), *Transmission Control Protocol, Version 4*, RFC793, available at <http://www.faqs.org/rfcs/rfc793.html>, Sep. 1981, pp. 1-85.
- Pottie et al., Adaptive Wireless Arrays for Interactive RSTA in SUO (Awairs), *UCLA, Electrical Engineering Department*; Date: unknown, pp. 1-20.
- Pottie et al., Adaptive Wireless Arrays Interactive Reconnaissance, Surveillance, and Target Acquisition in Small Unit Operations (Awairs); Lower Power Wireless Integrated Microsensors (LWIM), Presented to Dr. E. Carapezza, Dr. D. Lao and Lt. Col. J. Hernandez, *UCLA, Rockwell Science Center*; Mar. 21, 1997, pp. 1-110.
- Pottie, Awairs Mini-Site Review [Presentation], *Rockwell Science Center*; Feb. 23, 1998, pp. 1-58.
- Pottie, Awairs: Mini-Site Review, *Project Status*, UCLA: Rockwell Science Center, Feb. 23, 1998, pp. 1-58.
- Pottie, *Hierarchical Information Processing in Distributed Sensor Networks*, *ISIT*, Aug. 16-21, 1998, IEEE, 1998, pp. 163.
- Pottie et al., Wins: Principles and Practice, *EDD, UCLA*; Date: unknown, pp. 1-10.
- Pottie et al., Wireless Integrated Network Sensors: Towards Low Cost and Robust Self-Organizing Security Networks; *EED, UCLA; Rockwell Science Center; SPIE* vol. 3577, Nov. 1, 1998, pp. 86-95 (20 pages).

- Pottie et al., *Wireless Integrated Network Sensors: Towards Low Cost and Robust Self-Organizing Security Networks*, Publisher: unknown, Date: unknown, pp. 1–10.
- Pottie et al., *Wireless Integrated Network Sensors*, *UCLA; Communications of the ACM*, vol. 43, No. 5, May 2000, pp. 51–58.
- Pottie, *Wireless Sensor Networks*, ITW 1998, Jun. 22–26, 1998, available at <http://dantzig.ee.ucla.edu/oclab/Pottie.html>, 2 pages.
- Power/Perfect Energy Management Systems, Author: unknown, *Johnson Controls*, 1983, pp. 1–4.
- Rabaey et al., *PicoRadio Supports Ad Hoc Ultra-Low Power Wireless Networking*, *Computer*; IEEE, Jul. 2000, pp. 42–48.
- Radlherr, *Datentransfer Ohne Draht und Telefon*, *Funkschau*, Nov. 1991, pp. 49–52.
- Raji, *Control Networks and the Internet*, *Echelon Corp.*; 1998, pp. 1–39.
- Raji, *End-to-End Solutions with LonWorks® Control Technology: Any Point, Any Time, Any Where*, *Echelon Corp.*; 1998, pp. 1–30.
- Rehker et al., *A Border Gateway Protocol 4 (BGP-4)*, RFC 1771, (Mar. 1995), available at <http://tools.ietf.org/html/rfc1771>, Jun. 24, 2009, pp. 1–58.
- Ritter et al. *The Architecture of Metricom's Microcellular Data Network™ (MCDN) and Detail of its Implementation as the Second and Third Generation Ricochet™ Wide-Area Mobile Data Service*, *IEEE*, 2001, pp. 143–152.
- Rosen, *Exterior Gateway Protocol (EGP)*, RFC 827 (Oct. 1982), available at <http://tools.ietf.org/html/rfc827>, Jun. 24, 2009, pp. 1–48.
- Salkintzisa et al., *Design and implementation of a low-cost wireless network for remote control and monitoring applications*, *Elsevier, Microprocessors and Microsystems*, 1997, pp. 79–88.
- Saltzer et al., *Source Routing of Campus-wide Internet Transport* (Sep. 15, 1980), available at <http://groups.csail.mit.edu/ana/publications/pubPDFs/Sourcerouting.html>, Sep. 21, 2009, pp. 1–14.
- Schulman et al., *SINCGARS Internet Controller-Heart of the Digitized Battlefield*, *Proceedings of the 1996 Tactical Communications Conference*, Apr. 30–May 2, 1996, pp. 417–421.
- Selected Vendor Telecommunications Products*, Author: unknown, available at <http://eetd.lbl.gov/ea/ems/reports/39015a.pdf>, Date: unknown; pp. 1–83.
- Shacham et al., *A Packet Radio Network for Library Automation; 1987 IEEE Military Communications Conference*, vol. 2, at 21.3.1 (Oct. 1987); pp. 456–462.
- Shacham et al., *Dynamic Routing for Real-Time Data Transport in Packet Radio Networks*, *IEEE Proceedings of INFOCOM '82*, pp. 152–159.
- Shacham et al., *Future Directions In Packet Radio Architectures And Protocols*, *Proceedings of The IEEE*, vol. 75, No. 1, Jan. 1987, pp. 83–99.
- Shacham et al., *Packet Radio Networking*, *Telecommunications* vol. 20, No. 9, Sep. 1986, pp. 42,43,46,48,64 and 82.
- Shoch, *Inter-Network Naming, Addressing and Routing*, *Internet Experiment Note # 19*, Notebook section 2.3.3.5, Xerox Palo Alto Research Center, Jan. 29, 1978, Publisher: unknown, pp. 1–9.
- Smart Home Technology Leader Intelli Selects Coactive Networks Internet Gateways, *Coactive* (press release), Author: unknown, Sep. 11, 2000, pp. 1–4.
- Sohrabi et al., *Protocols for Self-Organization of a Wireless Sensor Network*, *IEEE Personal Communications*, Oct. 2000, pp. 16–27.
- Special Poll Feature*, Author: unknown, *Engineering Report*, No. 93–008, Sep. 1993, pp. 1–5.
- Square D Interfacing* Author: unknown, *Engineering Report*, No. 88–010, Revised: Apr. 18, 1996, pp. 1–9.
- Subramanian et al., *An Architectural for Building Self-Configurable Systems*, *IEEE*, 2000, pp. 63–73.
- Sunshine, *Addressing Problems in Multi-Network Systems* (Apr. 1981), available at <ftp://ftp.isi.edu/in-notes/ien/ien178.txt>, Sep. 14, 2009, pp. 1–26.
- Sunshine, *Addressing Problems in Multi-Network Systems*, *Proceedings INFOCOM '82*, 1982 IEEE, pp. 12–18.
- Sunshine, *Network Interconnection and Gateways*, *IEEE Journal on Selected Areas in Communications*, vol. 8, No. 1, Jan. 1990, pp. 4–11.
- Sunshine, *Source Routing in Computer Networks*, *Information Sciences Department of The Rand Corporation* (1977), Publisher: unknown, pp. 29–33.
- Technology Review*, *Metricom's Ricochet Packet Radio Network*, *Ham Radio Online*, 1996, Author: unknown, pp. 1–3.
- Technology Review: Metricom's Ricochet Packet Radio Network*, Author: unknown, *Virtual Publishing*, 1996, available at http://www.hamradio-online.com/1996/jan/met_ricom.html on May 4, 2010, pp. 1–3.
- Texas Instruments Interface*, Author: unknown, *Engineering Report*, No. 91–021, Revised: Nov. 1994, pp. 1–3.
- The Institute of Electrical and Electronics Engineers, Inc., *IEEE Std 802.11-1997*, available at www.ieee.org on Jun. 26, 1997.
- The New Power Company Announces Revolutionary Energy-Saving Program that Gives Consumers Remote Control of their Thermostats via the Internet, *NewPower* (press release), Author: unknown, Apr. 24, 2001, pp. 1–6.
- Theodorides, *Wireless Integrated Network Sensors*, *Power Point Presentation*, Publisher: unknown, Apr. 15, 2003, pp. 1–19.
- The SNVT Master Master List and Programmers's Guide, *Echelon Corp.*; Author: unknown; Mar. 1996, pp. 1–23.
- Tobagi-et-al, *Packet Radio and Satellite Networks*, *IEEE Communications Magazine*, vol. 22, No. 11, Nov. 1984, pp. 24–40.
- Toh, *A Novel Distributed Routing Protocol to Support Ad-Hoc Mobile Computing*, *Conference Proceedings of the 1996 IEEE Fifteenth Annual International Phoenix Conference on Computers and Communications*, Mar. 27–29, 1996, pp. 480–486.
- Toshiba Interfacing*; Author: unknown, *Engineering Report*, No. 91–011, Revised: Jun. 19, 1996, pp. 1–4.
- Totolo, *HomeRF, A New Protocol on the Horizon*, Feb. 1999, available at <http://www.hometoys.com/htinews/feb99/articles/totolo/totolo.htm>, Mar. 2, 2009, pp. 1–3.
- TranstexT® Advanced Energy Management System*, Article, Author: unknown, Publisher: unknown, Date: unknown, pp. 1–2.
- TranstexT® Advanced Energy Management System*, Brochure, Author: unknown, *Integrated Communication Systems, Inc.*, 1990, pp.1–8.
- Varadhan et al., *SDRP Route Construction*, *Internet Draft*, available at [draft-ietf-sdr-route-construction-01.{ps,txt}](http://draft-ietf-sdr-route-construction-01.ps.txt), Feb. 27, 2005, pp. 1–12.

US 6,044,062 C1

Page 22

- Vardhan et al., Wireless Integrated Network Sensors (WI(NS): Distributed In Situ Sensing for Mission and Flight Systems, *2000 IEEE Aerospace Conference Proceedings*; 2000, pp. 459–463 (12 pages).
- Vista–40 2–Partition Security System Installation and Setup Guide; *Ademco*, Author: unknown; Jul. 1, 1998.
- Wang et al., Energy–Scalable Protocols for Battery–Operated MicroSensor Networks, *EED, MIT*; Date: unknown, pp. 1–11.
- Warrock, *Scool Give Report on Radio–Based FMS, Energy User News*, Nov. 7, 1983, pp. 1.
- Weiser, Some Computer Science Issues in Ubiquitous Computing, Mar. 23, 1993, *Communications of the ACM*; Jul. 1993, pp. 1–13.
- Weiser, The Computer for the 21st Century, *Scientific American*; Sep. 1991, available at <http://www.ubiq.com/hypertext/weiser/SciAmdraft3.html> on Apr. 20, 2009, pp. 1–9.
- Welcome to UtiliNet: A Wireless Data Communications Solution from Metricom, Inc.*, Author: unknown, available at <http://web.archive.org/web/199806028045812/www.metricom.com/industrial/utilinet.html> on May 10, 2010, pp. 1–10.
- Westcott et al., Hierarchical Routing for Very Large Networks, IEEE Military Communications Conference, Oct. 21–24, 1984, Conference Record vol. 2, pp. 214–218.
- Westinghouse Numa Logic Interface*, Author: unknown, *Engineering Report*, No. 91–013, Date: unknown, pp. 1–7.
- Wey et al., Clone Terminator: An Authentication Service for Advanced Mobile Phone System; IEEE (1995); pp. 175–179.
- What's Behind Ricochet: A Network Overview*; Author: unknown, available at http://web.archive.org/web/20000815090824/www.ricochet.com/ricochet_t_advantage/tech_overview.html, Aug. 15, 2000, pp. 1–4.
- Wikipedia, *Ad Hoc On–Demand Distance Vector Routing*, available at http://en.wikipedia.org/wiki/Ad_Hoc_On-Demand_Distance_Vector_Routing on Aug. 25, 2009, pp. 1–3.
- Wikipedia, Bellman–Ford Algorithm, available at <http://en.wikipedia.org/wiki/Bellman-Ford>, Jun. 24, 2009, pp. 1–4.
- Wikipedia, Border Gateway Protocol, available at http://en.wikipedia.org/wiki/Border_Gateway_Protocol, Jun. 24, 2009, pp. 1–13.
- Wikipedia, Distance–Vector Routing Protocol, available at http://en.wikipedia.org/wiki/Distance-Vector_Routing_Protocol, Jun. 24, 2009, pp. 1–4.
- Wikipedia, Enhanced Interior Gateway Routing Protocol, available at <http://en.wikipedia.org/wiki/EIGRP>, Jun. 24, 2009, pp. 1–7.
- Wikipedia, Exterior Gateway Protocol, available at http://en.wikipedia.org/wiki/Exterior_Gateway_Protocol, Jun. 24, 2009, pp. 1.
- Wikipedia, Interior Gateway Routing Protocol, available at http://en.wikipedia.org/wiki/Interior_Gateway_Routing_Protocol, Jun. 24, 2009, pp. 1–2.
- Wikipedia, IS–IS, available at <http://en.wikipedia.org/wiki/IS-IS>, Jun. 24, 2009, pp. 1–3.
- Wikipedia, L. R. Ford, Jr., available at http://en.wikipedia.org/wiki/L._R._Ford,_Jr, Jun. 24, 2009, pp. 1.
- Wikipedia, Open Shortest Path First, available at http://en.wikipedia.org/wiki/Open_Shortest_Path_First, Jun. 24, 2009, pp. 1–9.
- Wikipedia, Richard E. Bellman, available at http://en.wikipedia.org/wiki/Richard_Bellman, Jun. 24, 2009, pp. 1–3.
- Wikipedia, Routing Information Protocol, available at http://en.wikipedia.org/wiki/Routing_Information_Protocol, Jun. 24, 2009, pp. 1–4.
- Will et al., *Wireless Networking for Control and Automation of Off–road Equipment, ASAE*, Jul. 18–21, 1999, pp. 1–10.
- Wilson, Lexicon 700t Touchscreen Remote, Jan. 1, 1999, available at <http://avrev.com/home-theater-remotes-system-control/remotes-system> on Mar. 2, 2009, pp. 1–3.
- Wireless Accessories, catalog pages, *Home Automation, Inc.*; available at <http://web.archive.org/web/19970216060056/www.homeauto.com/> on Feb. 17, 2009 (archived web page);, Author: unknown; pp. 1–2.
- Wright (ed.), Home–automation networks mature while the PC industry chases a new home LAN, *EDN Design Feature*, Date: unknown, pp. 1–9.
- Wu, Distributed System Design; CRC Press (1999); pp. 177–180 and 204.
- Wunnava et al., Web Based Remote Security System (WRSS) Model Development, *IEEE*, Apr. 7–9, 2000, pp. 379–382.
- X10, CK11A ActiveHome, Home Automation System, *Owner's Manual*, Oct. 23, 1997, pp. 1–56.
- X10.com: The Supersite for Home Automation, *What's in the Kit*, available at <http://web.archive.org/web/19991111133653/www.com/products/x>, on Mar. 2, 2009, pp. 1–2.
- X10.com: The Supersite for Home Automation, *Wireless Remote Control System (RC5000)*, available at <http://web.archive.org/web/1999111453227/www.x10.com/products/x1>, on Mar. 2, 2009, pp. 1.
- X10: The Supersite for Home Automation, *Transceiver Module*, available at <http://web.archive.org/web/20000229141517/www.x10.com/products/x>, on Mar. 2, 2009, pp. 1.
- Yadav, *Border Security Using Wireless Integrated Network Sensors (WINS)* Power Point Presentation, Publisher: unknown, Date: unknown, pp. 1–22.
- Young, A Unifying Dynamic Distributed Multichannel TDMA Slot Assignment Protocol, Working paper, *Rockwell International*; Oct. 25, 1995, pp. 1–29.
- Young, USAP: A Unifying Dynamic Distributed Multichannel TDMA Slot Assignment Protocol, *Rockwell International Communication Systems Division; IEEE*; 1996, pp. 235.
- Yu, *Target Identification Processor for Wireless Sensor Network*, Dissertation, Los Angeles: University of California, 1999, pp. 1–110.
- Zander et al., The Softnet Project: A Retrospect, 1988 IEEE, pp. 343–345.
- Zimmermann et al., *Daten Funken*, Publisher: unknown; Date: unknown, pp. 1–6.
- U.S. Appl. No. 09/271,517, in the name of Thomas D. Petite, for a System for Monitoring Conditions in a Residential Living Community, abandoned as per Notice of Abandoned mailed Oct. 14, 2004.
- Brownrigg, E.B. et al.; A Packet Radio Network for Library Automation; IEEE (1987); pp. 456–462.
- Brownrigg, E.B. et al.; Packet Radio Networks; Architectures, Protocols, Technologies and Applications (1987), (introduction pp. ix–xviii); pp. 3–274.
- Brownrigg, E.B. et al.; Distribution, Networks, and Networking: Options for Dissemination; Workshop on Electronic Texts Session III (<http://palimpsest.stanford.edu/by-org/lc/etextw/sess3.html> 1992); pp. 1–10.

US 6,044,062 C1

Page 23

Brownrigg, E.B.; User Provided Access to the Internet; (<http://web.simmons.edu/~chen/nit/NIT'92/033-bro.htm> 2005); pp. 1–6.

Perkins, C.E. et al.; Highly Dynamic Destination–Sequenced Distance–Vector Routing (DSDV) for Mobile Computers; SIGCOMM 94–9/94 London England UK (1994); pp. 234–244.

Wu, J.; Distributed System Design; CRC Press (1999); pp. 177–180 and 204.

Kahn, Robert E., Gronemeyer, Steven A., Burchfiel, Jerry, and Kunzelman, Ronald C., “*Advances in Packet Radio*

Technology,” IEEE Nov. 1978, vol. 66, No. 11, pp. 1468–1496.

Wey, Jyhi–Kong, Chang, Han–Tsung, Sun, Lir–Fan and Yang, Wei–Pang, “*Clone Terminator: An Authentication Service for Advanced Mobile Phone System*,” IEEE 1995, pp. 175–179.

Jubin, John and Tornow, Janet D., “*The DARPA Packet Radio Network Protocols*,” Proceedings of the IEEE, vol. 75, No. 1, Jan. 1987, pp. 21–32.

Kleinrock, Leonard and Kamoun, Farouk, “*Hierarchical Routing for Large Networks*,” North–Holland Publishing Company, Computer Networks 1, 1997, pp. 155–174.

US 6,044,062 C1

1
EX PARTE
REEXAMINATION CERTIFICATE
ISSUED UNDER 35 U.S.C. 307

THE PATENT IS HEREBY AMENDED AS
INDICATED BELOW.

2
AS A RESULT OF REEXAMINATION, IT HAS BEEN
DETERMINED THAT:

The patentability of claims **2-4, 6-8, 10-12** and **14-16** is
5 confirmed.

Claims **1, 5, 9** and **13** are cancelled.

* * * * *

Superseded by a more recent version



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.233

(11/93)

DATA COMMUNICATION NETWORKS

**INFORMATION TECHNOLOGY –
PROTOCOL FOR PROVIDING THE
CONNECTIONLESS-MODE NETWORK
SERVICE: PROTOCOL SPECIFICATION**

ITU-T Recommendation X.233

Superseded by a more recent version

(Previously "CCITT Recommendation")

Superseded by a more recent version

Foreword

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. Some 179 member countries, 84 telecom operating entities, 145 scientific and industrial organizations and 38 international organizations participate in ITU-T which is the body which sets world telecommunications standards (Recommendations).

The approval of Recommendations by the members of ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, 1993). In addition, the World Telecommunication Standardization Conference (WTSC), which meets every four years, approves Recommendations submitted to it and establishes the study programme for the following period.

In some areas of information technology, which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC. The text of ITU-T Recommendation X.233 was approved on 11th November 1993. The identical text is also published as ISO/IEC International Standard 8473-1.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

© ITU 1994

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the ITU.

Superseded by a more recent version**CONTENTS**

	<i>Page</i>
1 Scope	1
2 Normative references.....	1
2.1 Identical Recommendations International Standards.....	2
2.2 Paired Recommendations International Standards identical in technical content	2
2.3 Additional references	2
3 Definitions	2
3.1 Reference model definitions.....	2
3.2 Service conventions definitions.....	3
3.3 Network layer architecture definitions	3
3.4 Network layer addressing definitions	3
3.5 Local area network definitions	3
3.6 PICS definitions	3
3.7 Additional definitions.....	3
4 Abbreviations	4
4.1 Data units.....	4
4.2 Protocol data units	4
4.3 Protocol data unit fields.....	4
4.4 Parameters	4
4.5 Miscellaneous.....	4
5 Overview of the protocol.....	5
5.1 Internal organization of the Network layer	5
5.2 Subsets of the protocol	5
5.3 Addresses and titles	5
5.4 Service provided by the protocol.....	6
5.5 Underlying service assumed by the protocol.....	6
6 Protocol functions.....	6
6.1 PDU composition function.....	7
6.2 PDU decomposition function	7
6.3 Header format analysis function.....	7
6.4 PDU lifetime control function	7
6.5 Route PDU function	8
6.6 Forward PDU function	8
6.7 Segmentation function.....	8
6.8 Reassembly function	9
6.9 Discard PDU function	10
6.10 Error reporting function	10
6.11 PDU header error detection function.....	12
6.12 Padding function	12
6.13 Security function	12
6.14 Source routing function	13
6.15 Record route function.....	13
6.16 Quality of service maintenance function	14
6.17 Priority function	14
6.18 Congestion notification function	14
6.19 Echo request function.....	14
6.20 Echo response function	15
6.21 Classification of functions.....	16

Superseded by a more recent version*Page*

7	Structure and encoding of PDUs	17
	7.1 Structure	17
	7.2 Fixed part.....	18
	7.3 Address part.....	20
	7.4 Segmentation part.....	21
	7.5 Options part	21
	7.6 Data part	25
	7.7 Data PDU	26
	7.8 Inactive Network layer protocol.....	27
	7.9 Error Report PDU.....	28
	7.10 Echo Request PDU.....	30
	7.11 Echo Response PDU	30
8	Provision of the underlying service	30
	8.1 Subnetwork points of attachment	30
	8.2 Subnetwork quality of service	30
	8.3 Subnetwork user data	32
	8.4 Subnetwork dependent convergence functions	32
9	Conformance	32
	9.1 Static conformance	32
	9.2 Dynamic conformance.....	34
	9.3 PICS proforma.....	34
Annex A	– PICS proforma.....	35
	A.1 Introduction	35
	A.2 Abbreviations and special symbols	35
	A.3 Instructions for completing the PICS proforma	35
	A.4 Identification	37
	A.5 Major capabilities	38
	A.6 End systems.....	38
	A.7 Intermediate systems	44
Annex B	– Supporting technical material	49
	B.1 Data unit lifetime.....	49
	B.2 Reassembly lifetime control	50
	B.3 The power of the header error detection function	51
Annex C	– Algorithms for PDU header error detection function	53
	C.1 Symbols used in algorithms.....	53
	C.2 Arithmetic conventions	53
	C.3 Algorithm for generating checksum parameters	53
	C.4 Algorithm for checking checksum parameters	53
	C.5 Algorithm to adjust the checksum parameter when an octet is altered	54

Superseded by a more recent version

Summary

This Recommendation specifies a protocol which is used to provide a connectionless-mode service as defined in the network service definition and associated management functions. The specification consists of the protocol mechanisms and PDU structure for the exchange of data between end systems and intermediate systems using connectionless data transfers. The protocol defines: data transfer, error reporting and echo function.

The Recommendation includes a description of the mapping from the protocol operation onto a generalized OSI data link service which can be used in conjunction with real subnetworks.

Introduction

This is one of a set of Recommendations and International Standards produced to facilitate the interconnection of open systems. The set covers the services and protocols required to achieve such interconnection.

This Recommendation | International Standard is positioned with respect to other related Recommendations and International Standards by the layers defined in ITU-T Rec. X.200 | ISO/IEC 7498-1. In particular, it is a protocol of the Network layer. The protocol specified by this Recommendation | International Standard may be used between Network entities in end systems, between Network entities in intermediate systems, or between a Network entity in an end system and a Network entity in an intermediate system. In an end system, it provides the connectionless-mode Network service defined in CCITT Rec. X.213 | ISO/IEC 8348.

The interrelationship of the protocol specification and the related service definitions is illustrated in Figure Intro.1.

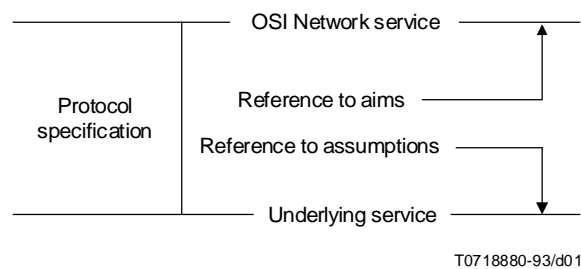


Figure Intro. 1 – Interrelationship of protocol and services

In order to evaluate the conformance of a particular implementation of this protocol, it is necessary to have a statement of which of the protocol's capabilities and options have been implemented. Such a statement is called a Protocol Implementation Conformance Statement (PICS), as defined in CCITT Rec. X.290 | ISO/IEC 9646-1. A PICS proforma, from which a PICS may be prepared for a specific implementation, is included in this Recommendation | International Standard as normative Annex A.

INTERNATIONAL STANDARD**CCITT RECOMMENDATION****INFORMATION TECHNOLOGY – PROTOCOL FOR PROVIDING
THE CONNECTIONLESS-MODE NETWORK SERVICE:
PROTOCOL SPECIFICATION****1 Scope**

This Recommendation | International Standard specifies a protocol that is used to provide the connectionless-mode Network service described in CCITT Rec. X.213 | ISO/IEC 8348 and to perform certain Network layer management functions. The protocol relies upon the provision of an underlying connectionless-mode service by real subnetworks and/or data links. The underlying connectionless-mode service assumed by the protocol may be obtained either directly, from a connectionless-mode real subnetwork, or indirectly, through the operation of an appropriate Subnetwork Dependent Convergence Function (SNDCF) or Protocol (SNDCP) over a connection-mode real subnetwork, as described in ISO/IEC 8648. This Recommendation | International Standard specifies the operation of the protocol with respect to a uniform, abstract “underlying subnetwork service”. Other Recommendations | International Standards specify the way in which this “underlying subnetwork service” is obtained from real subnetworks, such as those which conform to ISO/IEC 8802 or ISO/IEC 8208. The “underlying subnetwork service” may be obtained from real subnetworks other than those that are specifically covered by the other Recommendations | International Standards.

This Recommendation | International Standard specifies

- a) procedures for the connectionless transmission of data and control information from one Network entity to a peer Network entity;
- b) the encoding of the protocol data units (PDUs) used for the transmission of data and control information, comprising a variable-length protocol header format;
- c) procedures for the correct interpretation of protocol control information; and
- d) the functional requirements for implementations claiming conformance to this Recommendation | International Standard.

The procedures are defined in terms of

- a) the interactions among peer Network entities through the exchange of protocol data units;
- b) the interactions between a Network entity and a Network service user through the exchange of Network service primitives; and
- c) the interactions between a Network entity and an abstract underlying service provider through the exchange of service primitives.

This Recommendation | International Standard also provides the PICS proforma for this protocol, in compliance with the relevant requirements, and in accordance with the relevant guidance, given in CCITT Rec. X.290 | ISO/IEC 9646-1.

2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent editions of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of the currently valid ITU-T Recommendations.

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Reference Model: Basic Reference Model.*
- ITU-T Recommendation X.210 (1993) | ISO/IEC 10731:1993, *Information technology – Open Systems Interconnection – Conventions for the definition of OSI services.*
- CCITT Recommendation X.213 (1992) | ISO/IEC 8348:1992, *Information technology – Network service definition for Open Systems Interconnection.*

2.2 Paired Recommendations | International Standards identical in technical content

- ITU-T Recommendation X.224 (1993), *Protocol for providing the OSI connection-mode Transport service.*

ISO/IEC 8073:1992, *Information technology – Telecommunications and information exchange between systems – Open Systems Interconnection – Protocol for providing the connection-mode transport service.*
- CCITT Recommendation X.290 (1992), *OSI conformance testing methodology and framework for protocol Recommendations for CCITT applications – General concepts.*

ISO/IEC 9646-1:1991, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework: General concepts.*

2.3 Additional references

- ITU-T Recommendation X.25 (1993), *Interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) for terminals operating in the packet mode and connected to public data networks by dedicated circuit.*
- ISO/IEC 8208:1990, *Information technology – Data communications – X.25 Packet Layer Protocol for Data Terminal Equipment.*
- ISO/IEC 8648:1988, *Information processing systems – Open Systems Interconnection – Internal organization of the network layer.*
- ISO/IEC 8802:1990, *Information processing systems – Data communications – Local area networks.*

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply.

3.1 Reference model definitions

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.200 | ISO/IEC 7498-1:

- a) end system;
- b) Network entity;
- c) Network layer;
- d) Network protocol;
- e) Network protocol data unit;
- f) Network relay;
- g) Network service;
- h) Network service access point;
- i) Network service access point address;
- j) routing;
- k) service;
- l) service data unit;
- m) service primitive.

3.2 Service conventions definitions

This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.210 | ISO/IEC 10731:

- a) service provider;
- b) service user.

3.3 Network layer architecture definitions

This Recommendation | International Standard makes use of the following terms defined in ISO/IEC 8648:

- a) intermediate system;
- b) relay system;
- c) subnetwork;
- d) subnetwork dependent convergence protocol;
- e) subnetwork dependent convergence function;
- f) subnetwork independent convergence protocol;
- g) subnetwork independent convergence function;
- h) subnetwork access protocol.

3.4 Network layer addressing definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.213 | ISO/IEC 8348:

- a) Network addressing domain;
- b) Network protocol address information;
- c) subnetwork point of attachment.

3.5 Local area network definitions

This Recommendation | International Standard makes use of the following term defined in ISO/IEC 8802:

local area network

3.6 PICS definitions

This Recommendation | International Standard makes use of the following terms defined in CCITT Rec. X.290 | ISO/IEC 9646-1:

- a) PICS proforma;
- b) protocol implementation conformance statement.

3.7 Additional definitions

3.7.1 derived PDU: A protocol data unit the fields of which are identical to those of an initial PDU, except that it carries only a segment of the user data from an N-UNITDATA request.

3.7.2 initial PDU: A protocol data unit carrying the whole of the user data from an N-UNITDATA request.

3.7.3 local matter: A decision made by a system concerning its behaviour in the Network layer that is not prescribed or constrained by this Recommendation | International Standard.

3.7.4 Network entity title: An identifier for a Network entity which has the same abstract syntax as an NSAP address, and which can be used to unambiguously identify a Network entity in an end or intermediate system.

3.7.5 reassembly: The act of regenerating an initial PDU from two or more derived PDUs.

3.7.6 segment: A distinct unit of data consisting of part of the user data provided in the N-UNITDATA request and delivered in the N-UNITDATA indication.

3.7.7 segmentation: The act of generating two or more derived PDUs from an initial or derived PDU. The derived PDUs together carry the entire user data of the initial or derived PDU from which they were generated.

4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply:

4.1 Data units

NSDU	Network service data unit
PDU	protocol data unit
SDU	service data unit
SNSDU	subnetwork service data unit

4.2 Protocol data units

DT PDU	data protocol data unit
ER PDU	error report protocol data unit
ERP PDU	echo reply protocol data unit
ERQ PDU	echo request protocol data unit

4.3 Protocol data unit fields

DA	destination address
DAL	destination address length
DUID	data unit identifier
E/R	error report flag
LI	length indicator
LT	lifetime
MS	more segments flag
NLPID	Network layer protocol identifier
SA	source address
SAL	source address length
SL	segment length
SO	segment offset
SP	segmentation permitted flag

4.4 Parameters

DA	destination address
QOS	quality of service
SA	source address

4.5 Miscellaneous

CLNP	connectionless-mode network protocol (i.e. the protocol defined in this Recommendation International Standard)
NPAI	Network protocol address information
NS	Network service
NSAP	Network service access point
PICS	protocol implementation conformance statement
SN	subnetwork
SNAcP	subnetwork access protocol
SNDCF	subnetwork dependent convergence function
SNDCP	subnetwork dependent convergence protocol
SNICP	subnetwork independent convergence protocol
SNPA	subnetwork point of attachment

5 Overview of the protocol

5.1 Internal organization of the Network layer

The architectural organization of the Network layer is described in ISO/IEC 8648. ISO/IEC 8648 identifies and categorizes the way in which functions can be performed within the Network layer by Network layer protocols, thus providing a uniform framework for describing how protocols operating either individually or cooperatively in the Network layer can be used to provide the OSI Network service. This protocol is designed to be used in the context of the internetworking protocol approach to the provision of the connectionless-mode Network service defined in ISO/IEC 8648.

This protocol is intended for use in the Subnetwork Independent Convergence Protocol (SNICP) role. A protocol which fulfills the SNICP role operates to construct the OSI Network service over a defined set of underlying services, performing functions which are necessary to support the uniform appearance of the OSI connectionless-mode Network service over a homogeneous or heterogeneous set of interconnected subnetworks. This protocol is defined to accommodate variability where subnetwork dependent convergence protocols and/or subnetwork access protocols do not provide all of the functions necessary to support the connectionless-mode Network service over all or part of the path from one Network Service Access Point (NSAP) to another.

As described in ISO/IEC 8648, a protocol at the Network layer may fulfill different roles in different configurations. Although this protocol is designed particularly to be suitable for a SNICP role in the context of the internetworking protocol approach to the provision of the connectionless-mode Network service, it may also be used to fulfill other roles, and may therefore be used in the context of other approaches to subnetwork interconnection.

The operation of this protocol is specified with respect to an “underlying subnetwork service” which is made available through the operation of other Network layer protocols or through provision of the Data Link service. The “underlying subnetwork service” assumed by this protocol is described in 5.5.

5.2 Subsets of the protocol

Two subsets of the full protocol are defined, which exploit the known subnetwork characteristics of particular configurations and are therefore not subnetwork independent.

The Inactive Network Layer Protocol Subset is a null-function subset which can be used when it is known that the source and destination end systems are connected by a single subnetwork, and when none of the functions performed by the full protocol is required to provide the connectionless-mode Network service between any pair of end systems.

The Non-segmenting Protocol Subset permits simplification of the header when it is known that the source and destination end systems are connected by subnetworks whose individual service data unit sizes are greater than or equal to a known bound which is large enough so that segmentation is not required. This subset is selected by setting the segmentation permitted flag to zero (see 6.7).

5.3 Addresses and titles

The following clauses describe the addresses and titles used by this protocol.

5.3.1 Addresses

The source address and destination address parameters referred to in 7.3 are NSAP addresses. The syntax and semantics of an NSAP address are described in CCITT Rec. X.213 | ISO/IEC 8348.

The encoding used by this protocol to convey NSAP addresses is the “preferred encoding” specified in CCITT Rec. X.213 | ISO/IEC 8348. The NSAP address, encoded as a string of binary octets according to CCITT Rec. X.213 | ISO/IEC 8348, is conveyed in its entirety in the address fields described in 7.3.

5.3.2 Network entity titles

A Network Entity Title (NET) is an identifier for a Network entity in an end system or intermediate system. Network entity titles are allocated from the same name space as NSAP addresses, and the determination of whether a name is an NSAP address or a Network entity title depends on the context in which the name is interpreted. The values of the source route and record route parameters defined in 7.5.4 and 7.5.5 respectively are Network entity titles. The values of the source address and destination address parameters in the Error Report PDU defined in 7.9, in the Echo Request PDU defined in 7.10, and in the Echo Response PDU defined in 7.11 are also Network entity titles.

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

The encoding used by this protocol to convey Network entity titles is the “preferred encoding” specified in CCITT Rec. X.213 | ISO/IEC 8348. The Network entity title, encoded as a string of binary octets according to CCITT Rec. X.213 | ISO/IEC 8348, is conveyed in its entirety in the appropriate fields.

5.4 Service provided by the protocol

This protocol provides the connectionless-mode Network service described in CCITT Rec. X.213 | ISO/IEC 8348. The relevant Network service primitive and its parameters are shown in Table 1.

NOTE – CCITT Rec. X.213 | ISO/IEC 8348 states that the maximum size of a connectionless-mode Network service data unit (NSDU) is 64 512 octets.

Table 1 – Connectionless-mode Network service primitive

Primitive		Parameters
N-UNITDATA	Request Indication	NS-Source-Address, NS-Destination-Address, NS-Quality-of-Service, NS-Userdata

5.5 Underlying service assumed by the protocol

It is intended that this protocol be capable of operating over connectionless-mode services derived from a wide variety of real subnetworks and data links. Therefore, in order to simplify the specification of the protocol, its operation is defined (in clause 6) with respect to an abstract “underlying subnetwork service” rather than any particular real subnetwork service. This underlying service consists of a single SN-UNITDATA primitive which conveys the source and destination subnetwork point of attachment addresses, a subnetwork quality of service parameter, and a certain number of octets of user data.

The SN-UNITDATA primitive is used to describe the abstract interface that exists between the CLNP protocol machine and an underlying real subnetwork or a subnetwork dependent convergence function that operates over a real subnetwork or real data link to provide the required underlying service.

The primitive provided and its parameters are shown in Table 2.

Provision of the “underlying subnetwork service” by real subnetworks and data links is described in clause 8 and in other Recommendations | International Standards.

Table 2 – Underlying service primitive

Primitive		Parameters
SN-UNITDATA	Request Indication	SN-Source-Address, SN-Destination-Address, SN-Quality-of-Service, SN-Userdata

6 Protocol functions

This clause describes the functions performed as part of the protocol.

Not all of the functions must be performed by every implementation. Subclause 6.21 specifies which functions may be omitted, and the correct behavior when requested functions are not implemented.

6.1 PDU composition function

This function is responsible for the construction of a protocol data unit according to the rules governing the encoding of PDUs given in clause 7. The Protocol Control Information (PCI) required is determined from current state and local information and from the parameters associated with the N-UNITDATA request.

Network Protocol Address Information (NPAI) for the source address and destination address fields of the PDU header is derived from the NS-Source-Address and NS-Destination-Address parameters. The NS-Destination-Address and NS-Quality-of-Service parameters, together with current state and local information, are used to determine which optional functions are to be selected. User data passed from the Network service user (NS-Userdata) form the data part of the protocol data unit.

During the composition of the protocol data unit, a Data Unit Identifier (DUID) is assigned to distinguish this request to transmit NS-Userdata to a particular destination Network service user from other such requests. The originator of the PDU shall choose the DUID so that it remains unique (for this source and destination address pair) for the maximum lifetime of the Initial PDU in the network; this rule applies for any PDUs derived from the Initial PDU as a result of the application of the segmentation function (see 6.7). Derived PDUs are considered to correspond to the same Initial PDU, and hence to the same N-UNITDATA request, if they have the same source address, destination address, and data unit identifier.

The DUID is also available for ancillary functions such as error reporting (see 6.10).

The total length of the PDU in octets is determined by the originator and placed in the total length field of the PDU header. This field is not changed for the lifetime of the protocol data unit, and has the same value in the Initial PDU and in each of any Derived PDUs that may be created from the Initial PDU.

When the non-segmenting protocol subset is employed, neither the total length field nor the data unit identifier field is present. The rules governing the PDU composition function are modified in this case as follows. During the composition of the protocol data unit, the total length of the PDU in octets is determined by the originator and placed in the segment length field of the PDU header. This field is not changed for the lifetime of the PDU. No data unit identification is provided.

6.2 PDU decomposition function

This function is responsible for removing the protocol control information from the protocol data unit. During this process, information pertinent to the generation of the N-UNITDATA indication is determined as follows. The NS-Source-Address and NS-Destination-Address parameters of the N-UNITDATA indication are recovered from the NPAI in the source address and destination address fields of the PDU header. The data part of the received PDU is retained until all segments of the original service data unit have been received; collectively, these form the NS-Userdata parameter of the N-UNITDATA indication. Information relating to the Quality of Service (QOS) provided during the transmission of the PDU is determined from the quality of service and other information contained in the options part of the PDU header. This information constitutes the NS-Quality-of-Service parameter of the N-UNITDATA indication.

6.3 Header format analysis function

This function determines whether the full protocol or the inactive Network layer protocol is in use, and whether or not a received PDU has reached its final destination. If the Network layer protocol identifier (NLPID) field in a received PDU contains a value that identifies the protocol defined by this Recommendation | International Standard, then either the full protocol or the non-segmenting subset is in use; the header format analysis function determines whether or not the received PDU has reached its destination, using the destination address in the PDU header. If the destination address provided in the PDU identifies either a Network entity title of this Network entity or an NSAP served by this Network entity, then the PDU has reached its destination; if not, it shall be forwarded.

If the NLPID field contains a value that identifies the inactive Network layer protocol, then no further analysis of the PDU header is required. The Network entity in this case determines that either the Subnetwork Point of Attachment (SNPA) address encoded as NPAI in the supporting subnetwork protocol (see 8.1) corresponds directly to an NSAP address serviced by this Network entity, or that an error has occurred.

6.4 PDU lifetime control function

This function is used to enforce the maximum PDU lifetime. It determines whether a received PDU may be forwarded or whether its assigned lifetime has expired, in which case it shall be discarded.

The operation of the PDU lifetime control function depends upon the lifetime field in the PDU header. This field contains, at any time, the remaining lifetime of the PDU (represented in units of 500 ms). The lifetime of the Initial PDU is determined by the originating Network entity and placed in the lifetime field of the PDU. If and when the segmentation function is applied to a PDU, the value of the lifetime field of the Initial PDU is copied into all of the corresponding Derived PDUs.

The value of the lifetime field of a PDU is decremented by every Network entity that processes the PDU. When a Network entity processes a PDU, it decrements the PDU lifetime by at least one. The value of the PDU lifetime field shall be decremented by more than one if the sum of

- a) the transit delay in the underlying service from which the PDU was received, and
- b) the delay within the system processing the PDU

exceeds or is estimated to exceed 500 ms. In this case, the lifetime field shall be decremented by one for each additional 500 ms of actual or estimated delay. The determination of delay need not be precise, but where a precise value cannot be ascertained, the value used shall be an overestimate, not an underestimate.

If the lifetime field reaches a value of zero before the PDU is delivered to its destination, the PDU shall be discarded. The error reporting function shall be invoked as described in 6.10. This may result in the generation of an Error Report PDU.

It is a local matter whether or not the destination Network entity performs the lifetime control function.

6.5 Route PDU function

This function determines the Network entity to which a PDU should be forwarded and the underlying service that must be used to reach that Network entity, using the destination address field and either the segment length field (if present) or the total length field (if the segment length field is not present). Where segmentation is required, the route PDU function further determines over which underlying service Derived PDUs shall be sent in order to reach that Network entity. The results of the route PDU function are passed to the forward PDU function (along with the PDU itself) for further processing.

Selection of the underlying service that shall be used to reach the “next” system in the route to the destination is initially influenced by the NS-Quality-of-Service parameter of the N-UNITDATA request, which specifies the QOS requested by the sending NS user. Whether this QOS is to be provided directly by the protocol, through the selection of the quality of service maintenance parameter and other optional parameters, or through the QOS facilities offered by each of the underlying services, or both, is determined prior to invocation of the forward PDU function. Route selection by intermediate systems may subsequently be influenced by the values of the quality of service maintenance parameter (if present), and other optional parameters (if present).

6.6 Forward PDU function

This function issues an SN-UNITDATA request primitive (see 5.5), supplying the subnetwork or SNDCEF identified by the route PDU function with the protocol data unit as user data to be transmitted, the address information required by that subnetwork or SNDCEF to identify the “next” system within the subnetwork-specific addressing domain (this may be an intermediate system or the destination end system), and quality of service constraints (if any) to be considered in the processing of the user data.

When the PDU to be forwarded is longer than the maximum service data unit size provided by the underlying service, the segmentation function is applied (see 6.7).

6.7 Segmentation function

Segmentation is performed when the length of a protocol data unit is greater than the maximum service data unit size supported by the underlying service to be used to transmit the PDU.

Segmentation consists of composing two or more new PDUs (Derived PDUs) from the too-long Initial or Derived PDU that is to be segmented. All of the header information from the PDU to be segmented, with the exception of the segment length and checksum fields of the fixed part, and the segment offset field of the segmentation part, is duplicated in each Derived PDU, including all of the address part, the data unit identifier and total length of the segmentation part, and the options part (if present).

NOTE – The rules for forwarding and segmentation guarantee that the header length is the same for all segments (Derived PDUs) of an Initial PDU, and is the same as the header length of the Initial PDU. The size of a PDU header therefore will not change due to the operation of any protocol function.

The user data field of the PDU to be segmented is divided and apportioned among the user data fields of the Derived PDUs in such a way that the Derived PDUs satisfy the maximum-length requirements of the SN-Userdata parameter of the SN-UNITDATA request primitive used to access the selected underlying service. The user data field of each derived PDU, except for the last, shall contain a number of octets that is a non-zero multiple of 8. Thus, the value of the segment offset field in any PDU is either zero or a non-zero multiple of 8. Segmentation shall not result in the generation of a Derived PDU containing fewer than eight octets of user data.

Derived PDUs are identified as being from the same Initial PDU by means of

- a) the source address field;
- b) the destination address field; and
- c) the data unit identifier field.

The following fields of the PDU header are used in conjunction with the segmentation function:

- a) *Segment offset* – Identifies the octet at which the segment begins with respect to the start of the data part of the Initial PDU;
- b) *Segment length* – Specifies the number of octets in the Derived PDU, including both header and data;
- c) *More segments flag* – Set to one if this Derived PDU does not contain the final octet of the user data from the Initial PDU as its final octet of user data; and
- d) *Total length* – Specifies the number of octets in the Initial PDU, including both header and data.

Derived PDUs may be further segmented without constraining the routing of the individual Derived PDUs.

The segmentation permitted flag is set to one to indicate that segmentation is permitted. If the Initial PDU is not to be segmented at any point during its lifetime, the flag is set to zero by the source Network entity. The setting of the segmentation permitted flag may not be changed by any other Network entity for the lifetime of the Initial PDU and any Derived PDUs.

6.8 Reassembly function

The reassembly function reconstructs the Initial PDU from the Derived PDUs generated by the operation of the segmentation function on the Initial PDU (and, recursively, on subsequent Derived PDUs).

A bound on the time during which segments (Derived PDUs) of an Initial PDU may be held at a reassembly point before being discarded is provided, so that reassembly resources may be released when it is no longer expected that missing segments of the Initial PDU will arrive at the reassembly point. Upon reception of a Derived PDU, a reassembly timer shall be initiated with a value that indicates the amount of time that shall elapse before any unreceived (missing) segments of the Initial PDU are assumed to be lost. When this timer expires, all segments (Derived PDUs) of the Initial PDU held at the reassembly point shall be discarded, the resources allocated for those segments may be freed, and, if selected, an error report shall be generated (see 6.10).

While the exact relationship between reassembly lifetime and PDU lifetime is a local matter, the reassembly function shall preserve the intent of the PDU lifetime. Consequently, the reassembly function shall discard PDUs whose lifetime would otherwise have expired had they not been under the control of the reassembly function; that is, the reassembly lifetime for a given PDU shall be less than the PDU lifetime in all derived PDUs being held at the reassembly point.

NOTES

- 1 Methods of bounding reassembly lifetime are discussed in Annex B.
- 2 The segmentation and reassembly functions are intended to be used in such a way that the fewest possible segments are generated at each segmentation point and reassembly takes place at the final destination of a PDU. However, other schemes which
 - a) interact with the routing algorithm to favor paths on which fewer segments are generated, or
 - b) generate more segments than absolutely required in order to avoid additional segmentation at some subsequent point

are not precluded. The information necessary to enable the use of one of these alternative strategies may be made available through the operation of a Network layer management function or by other means.

- 3 The originator of the Initial PDU determines the value of the segmentation permitted flag in the Initial PDU and all Derived PDUs (if any). An intermediate system may not change this value in the Initial PDU or any PDU derived from it, and may not therefore add or remove the segmentation part of the header.

6.9 Discard PDU function

This function performs all of the actions necessary to free the resources reserved by the Network entity when any of the following situations are encountered.

NOTE 1 – The following list is not exhaustive.

- a) A violation of protocol procedure has occurred.
- b) A PDU is received whose checksum is inconsistent with its contents.
- c) A PDU is received, but due to local congestion, it cannot be processed.
- d) A PDU is received whose header cannot be analyzed.
- e) A PDU is received which cannot be segmented and cannot be forwarded because its length exceeds the maximum service data unit size supported by any underlying service available for transmission of the PDU to the next Network entity on the chosen route.
- f) A PDU is received whose destination address is unreachable or unknown.
- g) Incorrect or invalid source routing was specified. This may include a syntax error in the source routing field, an unknown or unreachable Network entity title in the source routing field, or a path which is not acceptable for other reasons.
- h) A PDU is received whose PDU lifetime has expired or whose lifetime expires during reassembly.
- i) A PDU is received which contains an unsupported option corresponding to a Type 2 function (see 6.21).

NOTE 2 – In general, it is not always possible to determine whether a destination NSAP address is invalid (does not follow CCITT Rec. X.213 | ISO/IEC 8348), unprocessable (in that there is no routing table entry for the address), or incorrectly coded (as NPAI). Therefore, with respect to generating an Error Report PDU, the situation described in f) may or may not be distinguished from the situation described in d), and the “reason for discard” (see 6.10 and Table 8) may be “header syntax error” or “destination address unknown”.

6.10 Error reporting function

6.10.1 Overview

This function attempts to return an Error Report PDU to the source Network entity when a protocol data unit originated by that Network entity is discarded in accordance with 6.9.

The Error Report PDU identifies the discarded PDU, specifies the type of error detected, and identifies the location in the header of the discarded PDU at which the error was detected. At least the entire header of the discarded PDU and, at the discretion of the originator of the Error Report PDU, none, all, or part of the data part of the discarded PDU are placed in the data part of the Error Report PDU.

The originator of a PDU controls the subsequent generation of Error Report PDUs that refer to it. The error report (E/R) flag in the original PDU is set by the source Network entity to indicate that an Error Report PDU is to be generated if the Initial PDU or any PDUs derived from it are discarded; if the flag is not set, error reports are not generated.

NOTES

1 The suppression of Error Report PDUs is controlled by the originating Network entity and not by the NS user. Care should be exercised by the originator with regard to suppressing ER PDUs so that error reporting is not suppressed for every PDU generated.

2 Non-receipt of an Error Report PDU does not imply correct delivery of a PDU issued by a source Network entity.

6.10.2 Requirements

An Error Report PDU shall not be generated to report the discard of an Error Report PDU.

An Error Report PDU shall not be generated to report the discard of a PDU unless that PDU has the error report flag set to allow error reports.

If a PDU is discarded, and the error report flag in the discarded PDU is set to allow error reports, an Error Report PDU shall be generated if the reason for discard is one of the reasons for discard enumerated in 6.9, subject to the conditions described in 6.10.4. If a PDU with the E/R flag set to allow error reports is discarded for any other reason, an ER PDU may be generated (as an implementation option).

Error reports may be suppressed in circumstances in which the validity of the information in the PDU that caused the error condition is uncertain. These circumstances include, but are not limited to, those described in items b), c) and d) of 6.9.

6.10.3 Processing of error reports

An Error Report PDU is composed from information contained in the header of the discarded PDU to which the error report refers. The contents of the source address field of the discarded PDU are used as the destination address of the Error Report PDU. This value, which in the context of the discarded PDU was used as an NSAP address, is used in the context of the Error Report PDU as the Network entity title of the Network entity that originated the discarded PDU. The Network entity title of the originator of the Error Report PDU is conveyed in the source address field of the header of the Error Report PDU. The value of the lifetime field is determined in accordance with 6.4. Optional parameters are selected in accordance with 6.10.4.

The segmentation of Error Report PDUs is not permitted; hence, no segmentation part is present. The total length of the ER PDU in octets is placed in the segment length field of the ER PDU header. This field is not changed during the lifetime of the ER PDU. If the originator of the ER PDU determines that the size of the ER PDU exceeds the maximum service data unit size of the underlying service, the ER PDU shall be truncated to the maximum service data unit size (see 8.3) and forwarded with no other change. Error Report PDUs are routed and forwarded by intermediate system Network entities in the same way as Data PDUs.

NOTE – The requirement stated in 8.3 that the underlying service assumed by the protocol shall be capable of supporting a service data unit size of 512 octets guarantees that at least the entire header of the discarded PDU can be conveyed in the data part of an ER PDU.

When an ER PDU is decomposed upon reaching its destination, information that may be used to interpret and act upon the error report is obtained as follows. The Network entity title recovered from the NPAI in the source address field of the ER PDU header is used to identify the Network entity that generated the error report. The reason for generating the error report is extracted from the options part of the PDU header. The entire header of the discarded PDU, and part or all of the original user data (if present), are extracted from the data part of the ER PDU to assist in ascertaining the nature of the error.

6.10.4 Relationship of discarded PDU options to error reports

The generation of an error report is affected by options that are present in the corresponding discarded PDU. The presence of options in the discarded PDU that are not supported by the system that has discarded that PDU, or the presence of an unrecognized PDU type code parameter, may cause the suppression of an error report even if the discarded PDU indicated that an error report should be generated in the event of a discard.

The processing of an error report is also affected by options that are present in the corresponding discarded PDU. In particular, options selected in the discarded PDU affect which options are included in the corresponding Error Report PDU. The selection of options for an Error Report PDU is governed by the following requirements:

- a) If the priority, QOS maintenance, or security option is selected in the discarded PDU, and the system generating the Error Report PDU supports the option, then the Error Report PDU shall specify the same option, using the value that was specified in the discarded PDU.
- b) If the system generating the Error Report PDU does not support the security option, an error report shall not be generated for a discarded PDU that selected the security option.
- c) If the complete source route option is selected in the discarded PDU, and the system generating the Error Report PDU supports the option, then the Error Report PDU shall specify the complete source route option. The source route parameter value is obtained by extracting from the discarded PDU that portion of the complete source route list that has already been processed, and reversing the order of Network entity titles which comprise that portion of the list.
- d) If the system generating the Error Report PDU does not support the complete source route option, an Error Report PDU shall not be generated for a discarded PDU that selects the complete source route option.
- e) The padding, partial source route, and record route options, if supported, may be specified in the Error Report PDU.

NOTE – The values of the optional parameters in e) above may be derived as a local matter, or they may be based upon the corresponding values in the discarded PDU.

6.11 PDU header error detection function

The PDU header error detection function protects against failure of intermediate or end system Network entities due to the processing of erroneous information in the PDU header. The function is realized by a checksum computed on the entire PDU header. The checksum is verified at each point at which the PDU header is processed. If the checksum calculation fails, the PDU shall be discarded. If PDU header fields are modified (for example, due to the operation of the lifetime function), then the checksum shall be modified so that the checksum remains valid.

The use of the header error detection function is optional and is selected by the originating Network entity. If the function is not used, the checksum field of the PDU header shall be set to zero.

If the function is selected by the originating Network entity, the value of the checksum field is calculated so as to cause the following formulae to be satisfied:

$$\sum_{i=1}^L a_i \pmod{255} = 0$$

$$\sum_{i=1}^L (L - i + 1) a_i \pmod{255} = 0$$

in which L is the number of octets in the PDU header, and a_i is the value of the octet at position i . The first octet in the PDU header is considered to occupy position $i = 1$.

When the function is in use, neither octet of the checksum field may be set to zero.

To ensure that inadvertent modification of a header while a PDU is being processed by an intermediate system (for example, due to a memory fault) may still be detected by the PDU header error detection function, an intermediate system Network entity shall not recompute the checksum for the entire header, even if fields are modified.

NOTE – Annex C contains descriptions of algorithms which may be used to calculate the correct value of the checksum field when the PDU is created, and to update the value of the checksum field when the header is modified.

6.12 Padding function

The padding function is provided to allow space to be reserved in the PDU header which is not used to support any other function. Octet alignment shall be maintained.

NOTE – An example of the use of this function is to cause the data part of a PDU to begin on a convenient boundary, such as a computer word boundary.

6.13 Security function

The provision of protection services (e.g. data origin authentication, data confidentiality, and data integrity of a single connectionless-mode NSDU) is performed by the security function.

The security function is related to the protection from unauthorized access quality of service parameter described in Rec. X.213 | ISO/IEC 8348. The function is realized through the selection of the security parameter in the options part of the PDU header.

This Recommendation | International Standard does not specify the way in which protection services are to be provided; it provides only for the encoding of security information in the PDU header. To facilitate interoperation among end systems and intermediate systems by avoiding different interpretations of the same encoding, a means to distinguish user-defined security encodings from standardized security encodings is described in 7.5.3.

NOTE – As an implementation consideration, data origin authentication may be provided through the use of a cryptographically generated or enciphered checksum (distinct from the PDU header error detection mechanism); data confidentiality and data integrity may be provided via route control mechanisms.

6.14 Source routing function

The source routing function allows a Network entity to specify the path that a generated PDU shall take. Source routing may be selected only by the originator of a PDU. Source routing is accomplished using a list of Network entity titles held in a parameter within the options part of the PDU header. The length of this parameter is determined by the originating Network entity, and does not change during the lifetime of a PDU. Only the titles of intermediate system Network entities shall be included in the list; the Network entity titles of the source and destination of the PDU shall not be included in the list.

Associated with the list of Network entity titles is an indicator that identifies the next entry in the list to be used; this indicator is advanced by the receiver of a PDU when the next title in the list matches its own. The indicator is updated as the PDU is forwarded so as to identify the appropriate next entry at each point along the route.

Two forms of the source routing function are provided. The first form, referred to as complete source routing, requires that the specified path shall be taken; that is, only those systems identified in the list may be visited by the PDU while *en route* to the destination, and each system shall be visited in the order specified. If the specified path cannot be taken, the PDU shall be discarded. Subclause 6.10 describes the circumstances in which an attempt shall be made to inform the PDU's originator of the discard using the error reporting function.

The second form is referred to as partial source routing. As with complete source routing, each system identified in the list shall be visited in the order specified while *en route* to the destination. However, with this form of source routing the PDU may take any path necessary to arrive at the next intermediate system in the list, which may include visiting intermediate systems that are not identified in the list. The PDU shall not be discarded (for source routing related reasons) unless one of the systems specified cannot be reached by any available route.

6.15 Record route function

The record route function records the path taken by a PDU as it traverses a series of intermediate systems. A recorded route consists of a list of Network entity titles held in a parameter within the options part of the PDU header. The length of this parameter is determined by the originating Network entity, and does not change during the lifetime of the PDU.

The list is constructed as the PDU is forwarded along a path towards its destination. Only the titles of intermediate system Network entities shall be included in the recorded route. The Network entity title of the originator of the PDU shall not be recorded in the list.

When an intermediate system Network entity processes a PDU containing the record route parameter, the Network entity adds its own Network entity title at the end of the list of recorded Network entity titles. An indicator is maintained to identify the next available octet to be used for recording of route. This indicator is updated as entries are added to the list as follows. The length of the entry to be added to the list is added to the value of the next available octet indicator, and this sum is compared with the length of the record route parameter. If the addition of the entry to the list would exceed the size of the parameter, the next available octet indicator is set to indicate that route recording has been terminated. The Network entity title is not added to the list. The PDU may still be forwarded to its final destination, without further addition of Network entity titles.

If the addition of the entry would not exceed the size of the record route parameter, the next available octet indicator is updated with the new value, and the Network entity title is added to the end of the list.

Two forms of the record route function are provided. The first form is referred to as complete route recording. It requires that the list of Network entity titles be a complete and accurate record of all intermediate systems visited by a PDU (including Derived PDUs), except when a shortage of space in the record route option field causes termination of recording of route, as described above. When complete route recording is selected, PDU reassembly at intermediate systems may be performed only when the Derived PDUs that are reassembled all took the same route.

The second form is referred to as partial route recording. It also requires a record of intermediate systems visited by a PDU. When partial route recording is selected, PDU reassembly at intermediate systems may be performed whether or not the Derived PDUs that are reassembled all took the same route; the route recorded in any of the Derived PDUs may be placed in the PDU resulting from the reassembly.

NOTE – The record route function is intended to be used in the diagnosis of subnetwork problems and/or to provide a return path that could be used as a source route in a subsequent PDU.

6.16 Quality of service maintenance function

The quality of service maintenance function provides information to Network entities in intermediate systems which may be used to make routing decisions where such decisions affect the overall QOS provided to NS users. This information is conveyed to intermediate system Network entities in a parameter in the options part of the PDU header.

In those instances in which the QOS requested cannot be maintained, intermediate system Network entities shall attempt to deliver the PDU at a QOS different from the QOS requested. Intermediate system Network entities may, but need not, provide a notification of failure to meet the requested quality of service.

6.17 Priority function

The priority function allows a PDU to be processed preferentially with respect to other PDUs. The function is realized through the selection of a parameter in the options part of the PDU header.

The lowest priority value is zero; numerically greater values signify successively higher priority. The priority function provides a means whereby the resources of end and intermediate system Network entities, such as outgoing transmission queues and buffers, can be used preferentially to process higher-priority PDUs ahead of lower-priority PDUs. The specific action taken by an individual Network entity to support the priority function is a local matter.

6.18 Congestion notification function

To allow NS users to take appropriate action when congestion is experienced within the NS provider, intermediate systems may inform the destination Network entity of congestion through the use of a flag in the QOS maintenance parameter in the options part of the PDU header. The value of this flag is initially set to zero (0) by the originator of the PDU and may be set to one (1) by any intermediate system which processes the PDU to indicate that it is experiencing congestion. The criteria for determining when this action is to be taken are a local matter.

NOTE – Congestion typically corresponds to the unavailability of buffer space to maintain output queues. An appropriate policy for indicating congestion may be based upon the depth of the output queue selected for a PDU (according to its destination address or other routing information). When the depth of a particular output queue exceeds a certain proportion of the maximum depth of that queue, an intermediate system may start to discard PDUs. The intermediate system may then set the congestion experienced flag in the next PDU to be forwarded and may continue to do so until the congestion is alleviated.

6.19 Echo request function

This function is invoked by Network layer management to obtain information about the dynamic state of the Network layer with respect to (a) the reachability of specific Network entities, and (b) the characteristics of the path or paths that can be created between Network entities through the operation of Network layer routing functions.

When invoked, the echo request function causes an Echo Request (ERQ) PDU to be created. The ERQ PDU shall be constructed and processed by Network entities in end systems and in intermediate systems that support the echo request function in exactly the same way as the DT PDU, with the following exceptions:

- a) Since the echo request function is invoked by Network layer management, rather than by an N-UNITDATA request, the information available to the PDU composition function (see 6.1) consists of current state, local information, and information supplied by Network layer management; the references in 6.1 to information obtained from parameters of the N-UNITDATA request do not apply to the composition of an ERQ PDU.
- b) The source address field of the ERQ PDU shall contain either the Network entity title of the originating Network entity or the NSAP address of an NSAP within the originating end or intermediate system. The destination address field of the ERQ PDU shall contain either the Network entity title of the destination Network entity or the NSAP address of an NSAP within the destination end or intermediate system.

NOTE 1 – A Network entity title is syntactically indistinguishable from an NSAP address. The additional information in an NSAP address, if any, beyond that which is present in a Network entity title, is relevant only to the operation of the PDU decomposition function in a destination end system, and therefore is not needed for the processing of an ERQ PDU (from which no N-UNITDATA indication is ever produced). The fact that the source and destination address fields of the ERQ PDU may contain either NETs or NSAP addresses therefore does not affect the processing of an ERQ PDU by any Network entity, and the information that is obtained through the operation of the echo request function is the same in either case.

- c) When an ERQ PDU has reached its destination, as determined by the header format analysis function, the echo response function (see 6.20), rather than the PDU decomposition function, shall be invoked. It is a local matter whether or not this involves an interaction with Network layer management.

NOTE 2 – Since the echo response function is a Type 2 function (see 6.21), the destination Network entity may or may not perform the echo response function upon receiving an ERQ PDU. Network layer management must therefore consider, when the echo request function is invoked, that non-receipt of a corresponding echo response PDU may be due to non-support of the echo response function by the destination Network entity.

- d) The maximum length of the ERQ PDU is equal to the maximum length of the Echo Response PDU minus the maximum length of the Echo Response PDU header. This ensures that the entire ERQ PDU can be contained within the data field of the Echo Response PDU (see 6.20).
- e) The data part of the ERQ PDU may, as a local matter, contain zero or more octets with any values (subject to the overall maximum length of the ERQ PDU specified in (d) above). If the first octet of the data part contains the binary value 1000 0001 (the NLPID for this protocol), then the first n octets of the data part (where n is the value of the second octet of the data part) shall contain an entire Echo Response PDU header, in which every field in the fixed part and address part, except the segment length and checksum fields, shall contain a valid value. The more segments flag shall have the value zero. If and only if the segmentation permitted flag is set to 1, the segmentation part shall be present. The options part, if present, may contain any of the options described in 7.5.

NOTE 3 – This Echo Response PDU header, if present in the data part of an ERQ PDU, may be, but is not required to be, used in whole or in part by the destination Network entity to compose an Echo Response PDU [see 6.20 (d)]. If this information is *not* present in the data part of the ERQ PDU, it may not be possible for the echo response function of the destination Network entity to select an appropriate value for the lifetime field of the Echo Response PDU.

6.20 Echo response function

This function is performed by a Network entity when it has received an ERQ PDU that has reached its destination, as determined by the header format analysis function – that is, an ERQ PDU that contains, in its destination address field, a Network entity title or NSAP address that identifies the Network entity or any of its NSAPs.

When invoked, the echo response function causes an Echo Response (ERP) PDU to be created. The ERP PDU shall be constructed and processed by Network entities in end systems and in intermediate systems that support the echo response function in exactly the same way as the DT PDU, with the following exceptions:

- a) Since the echo response function is not invoked by an N-UNITDATA request, the information available to the PDU composition function consists of current state, local information, and information contained in the corresponding ERQ PDU; the references in 6.1 to information obtained from parameters of the N-UNITDATA request do not apply to the composition of an ERP PDU.
- b) The source address field of the ERP PDU shall contain the value of the destination address field of the corresponding ERQ PDU. The destination address field of the ERP PDU shall contain the value of the source address field of the corresponding ERQ PDU.

NOTE – The observation contained in NOTE 1 of 6.19 applies also to the ERP PDU.

- c) The ERQ PDU, in its entirety, shall be placed into the data part of the ERP PDU. The data part of the ERP PDU shall contain *only* the corresponding ERQ PDU.
- d) If the data part of the ERQ PDU contains an ERP PDU header [see 6.19 e)], the PDU composition function may, but is not required to, use some or all of the information contained therein to select values for the fields of the ERP PDU header. In this case, however, the value of the lifetime field contained in the ERP PDU header in the ERQ PDU data part shall be used as the value of the lifetime field in the ERP PDU. The values of the segment length and checksum fields shall be computed by the Network entity regardless of the contents of those fields in the ERP PDU header in the data part of the ERQ PDU.
- e) The options part of the ERP PDU may contain any (or none) of the options described in 7.5. The values for these options, if present, are determined by the Network entity as a local matter. They may be, but are not required to be, either identical to or derived from the corresponding options in the ERQ PDU and/or the ERP PDU header contained in the data part of the ERQ PDU (if present). The source routing option in the ERP PDU shall not be identical to (copied from) the source routing option in the ERQ PDU header. If the recording of route option in the ERP PDU is identical to (copied from) the recording of route option in the ERQ PDU header, the second octet of the parameter value field shall be set to the value 3.
- f) It is a local matter whether or not the destination Network entity performs the lifetime control function on an ERQ PDU before performing the echo response function. The destination Network entity shall make the same decision in this regard that it would make, as a local matter, for a DT PDU in accordance with 6.4.

6.21 Classification of functions

Implementations of this Recommendation | International Standard are not required to support all of the functions described in 6.1 through 6.20. Functions are divided into three categories:

Type 1 – These functions shall be supported.

Type 2 – These functions may or may not be supported. If an implementation does not support a Type 2 function and the function is selected in a PDU, then that PDU shall be discarded, and an Error Report PDU shall be generated and forwarded to the originating Network entity, providing that the error report flag is set and the conditions of 6.10.4 are satisfied.

Type 3 – These functions may or may not be supported. If an implementation does not support a Type 3 function and the function is selected in a PDU, then the function is not performed, and the PDU is processed exactly as though the function had not been selected. The PDU shall not be discarded for this reason.

Table 3 shows how the functions are divided into these three categories.

Table 3 – Categorization of protocol functions

Function	Full Protocol	Non-Segmenting Subset	Inactive Subset
PDU Composition	1	1	1
PDU Decomposition	1	1	1
Header Format Analysis	1	1	1
PDU Lifetime Control	1	1	N/A
Route PDU	1	1	N/A
Forward PDU	1	1	N/A
Segment PDU	1	N/A	N/A
Reassemble PDU	1	N/A	N/A
Discard PDU	1	1	N/A
Error Reporting	1	1	N/A
Header Error Detection	1	1	N/A
Security	2	2	N/A
Complete Source Routeing	2	2	N/A
Complete Route Recording	2	2	N/A
Echo request	2	2	N/A
Echo response	2	2	N/A
Partial Source Routeing	3	3	N/A
Partial Route Recording	3	3	N/A
Priority	3	3	N/A
QOS Maintenance	3	3	N/A
Congestion Notification	3	3	N/A
Padding	3	3	N/A

NOTES

1 While the error reporting and header error detection functions shall be provided, they are invoked only when selected by the originating Network entity.

2 The rationale for the definition of Type 3 functions is that in the case of some functions it is more important to forward the PDUs between intermediate systems or deliver them to an end system than it is to support the functions. Type 3 functions should be used in those cases in which they are of an advisory nature; they cannot cause a PDU to be discarded when they are not supported.

7 Structure and encoding of PDUs

7.1 Structure

All protocol data units shall contain an integral number of octets. The octets in a PDU are numbered starting from one (1) and increasing in the order in which they are submitted to the underlying service. The bits in an octet are numbered from one (1) to eight (8), where bit one (1) is the low-order (least significant) bit.

When consecutive octets are used to represent a binary number, the lower-numbered octet has the most significant value.

Any implementation supporting this protocol is required to state in its specification the way in which octets are transferred, using the terms “most significant bit” and “least significant bit”. The PDUs of this protocol are defined using the terms “most significant bit” and “least significant bit”.

NOTE – When the encoding of a PDU is represented using a diagram in this clause, the following representation is used:

- a) octets are shown with the lowest-numbered octet to the left, higher-numbered octets being further to the right; and
- b) within an octet, bits are shown with bit eight (8) to the left and bit one (1) to the right.

With the exception of the inactive Network layer subset, PDUs shall contain, in the following order:

- a) the fixed part;
- b) the address part;
- c) the segmentation part, if present;
- d) the options part, if present;
- e) the reason for discard parameter (ER PDU only); and
- f) the data part, if present.

Items a) through e) comprise the PDU header.

In the case of the inactive Network layer subset, only the elements identified in 7.8 are present. Subclauses 7.2 through 7.5 do not apply to the inactive Network layer subset.

The structure is illustrated in Figure 1. For the purposes of Figure 1 and subclause 7.5, the reason for discard parameter contained in the ER PDU is considered to be the final element of the options part.

Part	Described in
Fixed Part	Subclause 7.2
Address Part	Subclause 7.3
Segmentation Part	Subclause 7.4
Options Part	Subclause 7.5
Data	Subclause 7.6

T0718890-93/d02

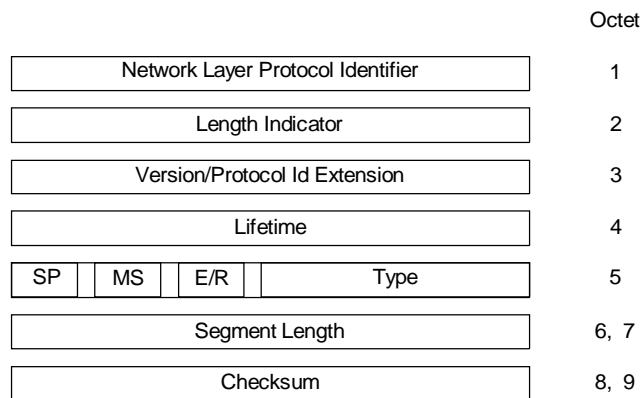
Figure 1 – PDU structure

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

7.2 Fixed part

7.2.1 General

The fixed part has the format illustrated in Figure 2.



T0718900-93/d03

Figure 2 – PDU header – Fixed part

7.2.2 Network layer protocol identifier

The value of this field is set to binary 1000 0001 to identify this Network layer protocol. The value of this field is set to binary 0000 0000 to identify the inactive Network layer protocol subset.

7.2.3 Length indicator

The length is indicated by a binary number, with a maximum value of 254 (1111 1110). The length indicated is the length in octets of the header, as described in 7.1. The value 255 (1111 1111) is reserved for possible future extensions.

NOTE – The rules for forwarding and segmentation guarantee that the header length is the same for all segments (Derived PDUs) of an Initial PDU, and is the same as the header length of the Initial PDU. The size of a PDU header therefore will not change due to the operation of any protocol function.

7.2.4 Version/protocol identifier extension

The value of this field is binary 0000 0001, which identifies the standard version 1 of this protocol.

7.2.5 PDU lifetime

The PDU lifetime field is encoded as a binary number representing the remaining lifetime of the PDU, in units of 500 ms.

7.2.6 Flags

7.2.6.1 Segmentation permitted

The segmentation permitted flag indicates whether segmentation is permitted. Its value is determined by the originator of the PDU and cannot be changed by any other Network entity for the lifetime of the Initial PDU and any Derived PDUs.

A value of one (1) indicates that segmentation is permitted. A value of zero (0) indicates that segmentation is not permitted. When the value of zero is selected, the segmentation part of the PDU header is not present, and the value of the segment length field gives the total length of the PDU (see 7.2.8 and 7.4.3).

7.2.6.2 More segments

The more segments flag indicates whether or not the data part of this PDU contains (as its last octet) the last octet of the user data in the NSDU. When the more segments flag is set to one (1), segmentation has occurred and the last octet of the NSDU is not contained in this PDU. The more segments flag shall not be set to one (1) if the segmentation permitted flag is not set to one (1).

When the more segments flag is set to zero (0), the last octet of the data part of the PDU is the last octet of the NSDU.

7.2.6.3 Error report

When the error report flag is set to one (1), the rules in 6.10 are used to determine whether or not to generate an Error Report PDU if it is necessary to discard this PDU.

When the error report flag is set to zero (0), discard of the PDU will not cause the generation of an Error Report PDU.

7.2.7 Type code

The type code field identifies the type of the protocol data unit. Allowed values are given in Table 4.

Table 4 – PDU type codes

PDU Type	Type Code					
	Bits	5	4	3	2	1
DT PDU		1	1	1	0	0
ER PDU		0	0	0	0	1
ERQ PDU		1	1	1	1	0
ERP PDU		1	1	1	1	1

7.2.8 PDU segment length

The segment length field specifies the entire length of the PDU in octets, including both header and data (if present). When the full protocol is employed and a PDU is not segmented, the value of this field is identical to the value of the total length field located in the segmentation part of the header.

When the non-segmenting protocol subset is employed, no segmentation part is present in the header. In this case, the segment length field specifies the entire length of the Initial PDU, including both header and data (if present).

The value of the segment length field shall not be changed for the lifetime of the PDU.

7.2.9 PDU checksum

The checksum is computed on the entire PDU header. For the Data, Echo Request, and Echo Reply PDUs, this includes the segmentation and options parts (if present). For the Error Report PDU, this includes the reason for discard field as well.

A checksum value of zero (0) is reserved to indicate that the checksum is to be ignored. The operation of the PDU header error detection function (see 6.11) ensures that the value zero does not represent a valid checksum. A non-zero value indicates that the checksum shall be processed; if the checksum calculation fails, the PDU shall be discarded.

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

7.3 Address part

7.3.1 General

The address part immediately follows the fixed part of the PDU header. The address part is illustrated in Figure 3.

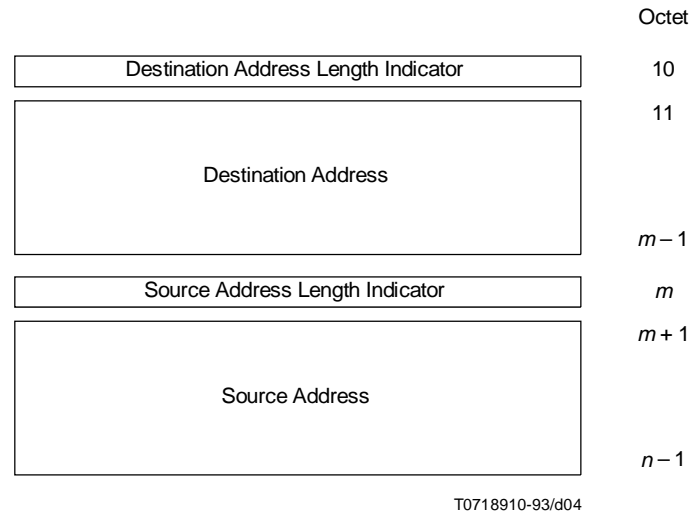


Figure 3 – PDU header – Address part

7.3.2 Destination and source addresses

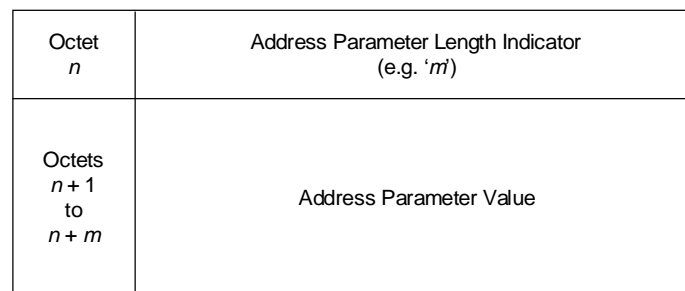
The destination and source addresses used by this protocol are Network service access point addresses or Network entity titles as defined in CCITT Rec. X.213 | ISO/IEC 8348.

The destination and source addresses are of variable length. The destination and source addresses are encoded as Network protocol address information in the destination address and source address fields using the “preferred encoding” defined in CCITT Rec. X.213 | ISO/IEC 8348.

The destination address length indicator field specifies the length of the destination address in octets. The destination address field follows the destination address length indicator field.

The source address length indicator field specifies the length of the source address in octets. The source address length indicator field follows the destination address field. The source address field follows the source address length indicator field.

Each address parameter is encoded as illustrated in Figure 4.



T0718920-93/d05

Figure 4 – Address parameters

7.4 Segmentation part

7.4.1 General

If the segmentation permitted flag in the fixed part of the PDU header (see 7.2.6.1) is set to one (1), the segmentation part of the header, illustrated in Figure 5, shall be present.

If the segmentation permitted flag is set to zero (0), the segmentation part shall not be present (the non-segmenting protocol subset is in use).

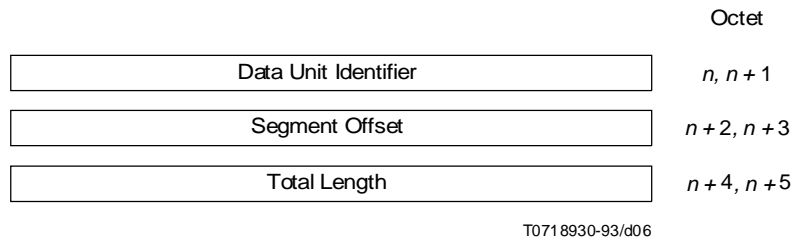


Figure 5 – PDU header – Segmentation part

7.4.2 Data unit identifier

The data unit identifier identifies an Initial PDU (and hence, its Derived PDUs) so that a segmented data unit may be correctly reassembled. The data unit identifier size is two (2) octets.

7.4.3 Segment offset

For each Derived PDU, the segment offset field specifies the relative position of the segment contained in the data part of the Derived PDU with respect to the start of the data part of the Initial PDU. The offset is measured in units of octets. The offset of the first segment (and hence, the Initial PDU) is zero (0); an unsegmented (Initial) PDU has a segment offset value of zero (0). The value of this field shall be a multiple of eight (8).

7.4.3 PDU total length

The total length field specifies the entire length of the Initial PDU in octets, including both the header and data. The value of this field shall not be changed for the lifetime of the Initial PDU (and hence, its Derived PDUs).

7.5 Options part

7.5.1 General

The options part of the PDU header is illustrated in Figure 6.

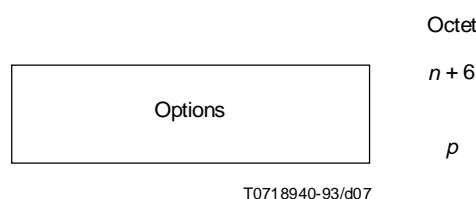


Figure 6 – PDU header – Options part

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

If the options part is present, it may contain one or more parameters. The number of parameters that may be contained in the options part is constrained by the length of the options part, which is determined by the following formula:

$$\text{Options part length} = \text{PDU header length} - (\text{length of fixed part} + \text{length of address part} + \text{length of segmentation part})$$

and by the length of the individual optional parameters.

Parameters defined in the options part may appear in any order. Duplication of options is not permitted. Receipt of a PDU with a duplicated option shall be treated as a protocol error. The rules governing the treatment of protocol errors are described in 6.10.

The encoding of parameters contained within the options part of the PDU header is illustrated in Figure 7.

Octets

n	Parameter Code
$n + 1$	Parameter Length (e.g. ' m ')
$n + 2$ to $n + m + 1$	Parameter Value

T071 8950-93/d08

Figure 7 – Encoding of option parameters

The parameter code field is encoded in binary and provides for a maximum of 255 different parameters. No parameter code uses bits 8 and 7 with the value 00, so the actual maximum number of parameters is lower. A parameter code of 255 (binary 1111 1111) is reserved for possible future extensions.

The parameter length field indicates the length, in octets, of the parameter value field. The length is indicated by a positive binary number, m , with a minimum value of 1 and a theoretical maximum value of 254. The practical maximum value of m is lower. For example, in the case of a single parameter contained within the options part, two octets are required for the parameter code and the parameter length indicators. Thus, the value of m is limited to:

$$m = 252 - (\text{length of fixed part} + \text{length of address part} + \text{length of segmentation part})$$

Accordingly, for each successive parameter the maximum value of m decreases.

The parameter value field contains the value of the parameter identified in the parameter code field.

The following parameters are permitted in the options part.

7.5.2 Padding

The padding parameter is used to lengthen the PDU header to a convenient size (see 6.12).

Parameter Code: 1100 1100
 Parameter Length: variable
 Parameter Value: any value is allowed.

Notwithstanding the requirement stated in 7.5.1 that the value of the parameter length field be no less than 1, the receiver of a PDU containing a value of 0 for the parameter length field of the padding option (and containing, therefore, no parameter value field for the padding option) may, but is not required to, treat this as a protocol error.

7.5.3 Security

This parameter allows a unique and unambiguous security level to be assigned to a protocol data unit (see 6.13).

Parameter Code: 1100 0101
 Parameter Length: variable
 Parameter Value: The high order two bits of the first octet specify the security format code, as shown in Table 5.

The rest of the first octet is reserved and shall be set to zero (0) when transmitted and ignored when received. The remainder of the parameter value field specifies the security level as described in the following clauses.

Table 5 – Security format codes

Security Format Code	Type of Security Field
00	Reserved
01	Source Address Specific
10	Destination Address Specific
11	Globally Unique

7.5.3.1 Source address specific

The security format code value of binary 01 indicates that the remaining octets of the parameter value field specify a security level which is unique and unambiguous in the context of the security classification system employed by the authority responsible for assigning the source NSAP address.

7.5.3.2 Destination address specific

The security format code value of binary 10 indicates that the remaining octets of the parameter value field specify a security level which is unique and unambiguous in the context of the security classification system employed by the authority responsible for assigning the destination NSAP address.

7.5.3.3 Globally unique security

The security format code value of binary 11 indicates that the remaining octets of the parameter value field specify a globally unique and unambiguous security level. This security classification system is not specified by this Recommendation | International Standard.

7.5.4 Source routing

The source routing parameter specifies, either completely or partially, the route to be taken from the originating Network entity to the destination Network entity (see 6.14).

Parameter Code: 1100 1000
 Parameter Length: variable
 Parameter Value: 2 octets of control information followed by a concatenation of Network entity title entries ordered from source to destination.

The first octet of the parameter value is the type code, which has the following significance:

0000 0000 partial source routing
 0000 0001 complete source routing
 <all other values reserved>

The second octet indicates the octet offset of the next Network entity title entry to be processed in the list. It is relative to the start of the parameter, such that a value of three (3) indicates that the next Network entity title entry begins immediately after this control octet. Successive octets are indicated by correspondingly larger values of this indicator.

The third octet begins the Network entity title list. The list consists of variable length Network entity title entries. The first octet of each entry gives the length of the Network entity title that comprises the remainder of the entry.

7.5.5 Recording of route

The recording of route parameter identifies the intermediate systems traversed by the PDU (see 6.15).

Parameter Code: 1100 1011
 Parameter Length: variable
 Parameter Value: 2 octets of control information followed by a concatenation of Network entity title entries ordered from source to destination.

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

The first octet of the parameter value is the type code, which has the following significance:

0000 0000	partial recording of route in progress
0000 0001	complete recording of route in progress
	<all other values reserved>

The second octet identifies the first octet not currently used for a recorded Network entity title, and therefore also the current end of the list. It is encoded relative to the start of the parameter value, such that a value of three (3) indicates that no Network entity titles have yet been recorded. The value of 255 is used to indicate that route recording has been terminated.

The third octet begins the Network entity title list. The list consists of variable length Network entity title entries. The first octet of each entry gives the length of the Network entity title comprising the remainder of the entry. Network entity title entries are always added to the end of the list.

NOTE – The length of the record route parameter is determined by the originator of the PDU and is not changed during the lifetime of the PDU; hence, the operation of the record route function does not affect the length of the header.

7.5.6 Quality of service maintenance

The quality of service maintenance parameter conveys information about the quality of service requested by the originating NS user.

Network entities in intermediate systems may, but are not required to, make use of this information as an aid in selecting a route when more than one route satisfying other routing criteria is available and the available routes are known to differ with respect to quality of service (see 6.16).

Parameter Code:	1100 0011
Parameter Length:	variable
Parameter Value:	The high order two bits of the first octet specify the QOS format code, as shown in Table 6.

Table 6 – QOS format codes

QOS Format Code	Type of QOS Field
00	Reserved
01	Source Address Specific
10	Destination Address Specific
11	Globally Unique

The rest of the first octet is reserved for use by the globally unique QOS format, as described in 7.5.6.3. If any other QOS format code is selected, bits 6-1 of the first octet shall be zero (0). The remainder of the parameter value field specifies the QOS as described in the following clauses.

7.5.6.1 Source address specific

The QOS format code value of binary 01 indicates that the remaining octets of the parameter value field specify a QOS which is unique and unambiguous in the context of the QOS maintenance system employed by the authority responsible for assigning the source NSAP address.

7.5.6.2 Destination address specific

The QOS format code value of binary 10 indicates that the remaining octets of the parameter value field specify a QOS which is unique and unambiguous in the context of the QOS maintenance system employed by the authority responsible for assigning the destination NSAP address.

7.5.6.3 Globally unique QOS

The QOS format code value of binary 11 indicates that the remainder of the parameter value field specifies a globally unique QOS maintenance field. When the globally unique QOS maintenance function is employed, the parameter value field shall have a total length of one octet, which is assigned the values shown in Table 7.

Table 7 – Globally unique QOS parameter values

Bits	Usage
8 and 7	QOS format code of binary 11
6	Reserved
5	Sequencing vs. transit delay
4	Congestion experienced
3	Transit delay vs. cost
2	Residual error probability vs. transit delay
1	Residual error probability vs. cost

Bit 6 is reserved, and shall be set to zero (0) when transmitted and ignored when received.

Bit 5 is set to one to indicate that, where possible, routing decisions should favor sending all PDUs to the specified destination NSAP address over a single path (in order to maintain sequence) over minimizing transit delay. A value of zero (0) indicates that, where possible, routing decisions should favor low transit delay over sequence preservation.

Bit 4 is set to zero by the Network entity which originates the protocol data unit. It is set to one by an intermediate system to indicate that this PDU has visited a congested intermediate system, and appropriate action should be taken by the destination Network entity. Once the congestion experienced bit is set by an intermediate system, it may not be reset by any intermediate system traversed by the PDU further along the path towards the destination.

Bit 3 is set to one to indicate that, where possible, routing decisions should favor low transit delay over low cost. A value of 0 indicates that routing decisions should favor low cost over low transit delay.

Bit 2 is set to one to indicate that, where possible, routing decisions should favor low residual error probability over low transit delay. A value of zero indicates that routing decisions should favor low transit delay over low residual error probability.

Bit 1 is set to one to indicate that, where possible, routing decisions should favor low residual error probability over low cost. A value of 0 indicates that routing decisions should favor low cost over low residual error probability.

7.5.7 Priority

The value of the priority parameter indicates the relative priority of the protocol data unit. Intermediate systems that support this option shall make use of this information in routing and in ordering PDUs for transmission (see 6.17).

Parameter Code: 1100 1101
 Parameter Length: 1 octet
 Parameter Value: 0000 0000 – Normal (Default)
 through
 0000 1110 – Highest
 <all other values reserved>

The values 0000 0001 through 0000 1110 are to be used for higher priority protocol data units. If an intermediate system does not support this option, all PDUs shall be treated as if the field had the value 0000 0000.

7.6 Data part

The data part of the PDU header is illustrated in Figure 8.



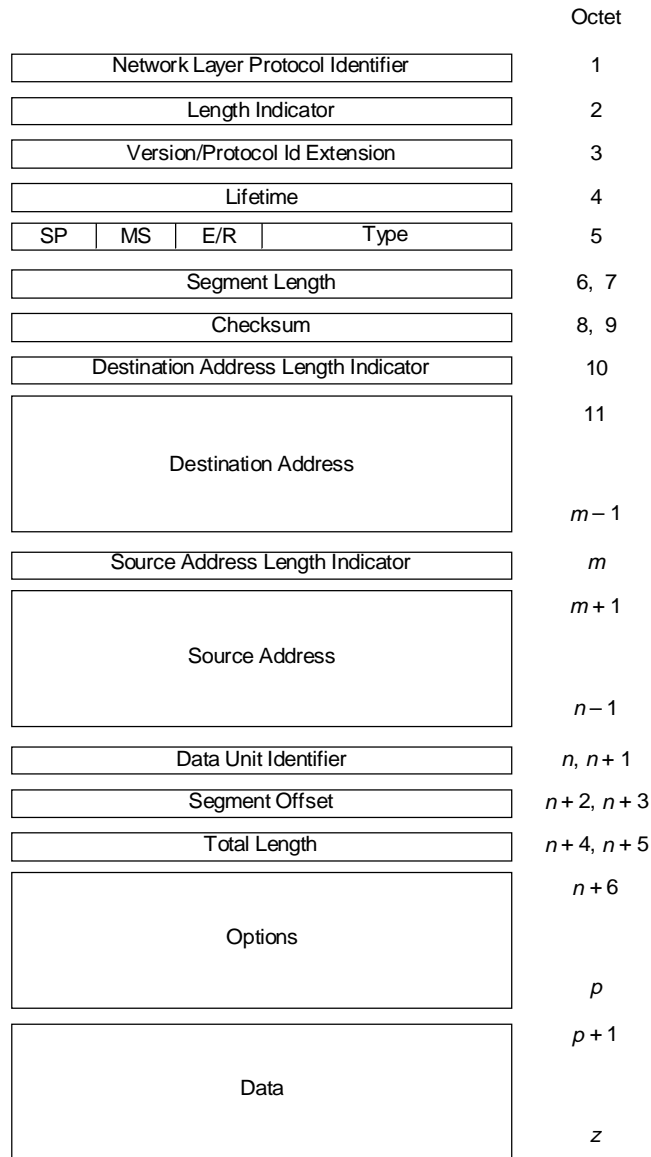
T0718960-93/d09

Figure 8 – PDU header – Data part

7.7 Data PDU

7.7.1 Structure

The Data PDU has the format illustrated in Figure 9.



T0718970-93/d10

Figure 9 – Data PDU

7.7.2 Fixed part

- | | |
|--------------------------------------|-------------|
| 1) Network Layer Protocol Identifier | (See 7.2.2) |
| 2) Length Indicator | (See 7.2.3) |
| 3) Version/Protocol Id Extension | (See 7.2.4) |
| 4) Lifetime | (See 7.2.5) |
| 5) SP, MS, E/R | (See 7.2.6) |
| 6) Type Code | (See 7.2.7) |
| 7) Segment Length | (See 7.2.8) |
| 8) Checksum | (See 7.2.9) |

7.7.3 Addresses

See 7.3.

7.7.4 Segmentation

See 7.4.

7.7.5 Options

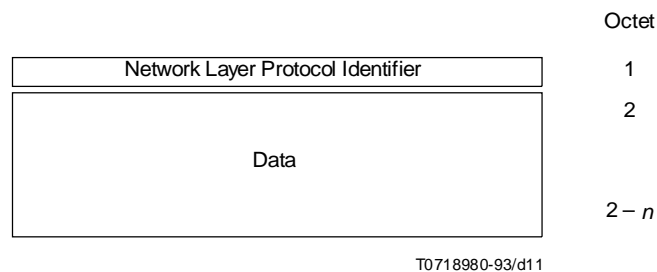
See 7.5.

7.7.6 Data

See 7.6.

7.8 Inactive Network layer protocol**7.8.1 Structure**

The Inactive Network Layer Protocol PDU has the format illustrated in Figure 10.

**Figure 10 – Inactive Network Layer Protocol PDU****7.8.2 Network layer protocol identifier**

The value of the Network layer protocol identifier field is binary zero (0000 0000).

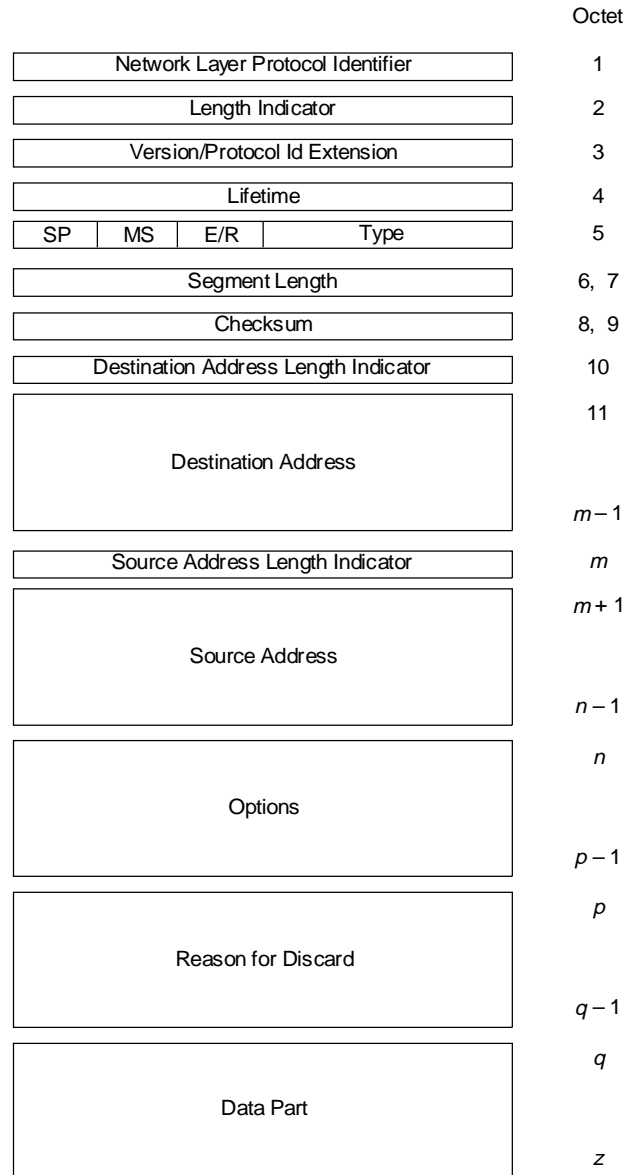
7.8.3 Data part

The data part may contain any number of octets up to one less than the maximum number that can be placed in the SN-Userdata parameter of the underlying SN-UNITDATA primitive. Therefore, the inactive Network layer protocol can be used only when the length of the NS-Userdata parameter in the N-UNITDATA primitive is constrained to be less than or equal to the value of the length of the SN-Userdata parameter minus one (see 7.6).

7.9 Error Report PDU

7.9.1 Structure

The format of the Error Report PDU is illustrated in Figure 11.



T0718990-93/d12

Figure 11 – Error Report PDU

7.9.2 Fixed part

The fixed part of the Error Report PDU is composed in the same way as a new (Initial) Data PDU.

- | | |
|--|-------------|
| 1) Network Layer Protocol Identifier | (See 7.2.2) |
| 2) Length Indicator | (See 7.2.3) |
| 3) Version/Protocol Id Extension | (See 7.2.4) |
| 4) Lifetime | (See 7.2.5) |
| 5) SP, MS, E/R (<i>Always set to zero</i>) | (See 6.10) |
| 6) Type Code | (See 7.2.7) |
| 7) Segment Length | (See 7.2.8) |
| 8) Checksum | (See 7.2.9) |

7.9.3 Addresses

The destination address specifies the Network entity title of the originator of the discarded PDU. The source address specifies the title of the intermediate system or end system Network entity initiating the Error Report PDU (see 7.3).

7.9.4 Options

See 7.5.

7.9.5 Reason for discard

This parameter is valid only for the Error Report PDU.

- Parameter Code: 1100 0001
 Parameter Length: two octets
 Parameter Value: type of error encoded in binary.

The parameter values are listed in Table 8.

The first octet of the parameter value contains an error type code. If the error in the discarded PDU can be localized to a particular field, the number of the first octet of that field is stored in the second octet of the reason for discard parameter field. If the error cannot be localized to a particular field, or if the error is a checksum error, then the value zero (0) is stored in the second octet of the reason for discard parameter field.

Table 8 – Reason for discard parameter values

Parameter value	Class of error	Meaning
0000 0000 0001 0010 0011 0100 0101 0110 0111	General	Reason not specified Protocol procedure error Incorrect checksum PDU discarded due to congestion Header syntax error (cannot be parsed) Segmentation needed but not permitted Incomplete PDU received Duplicate option
1000 0000 0001	Address	Destination address unreachable Destination address unknown
1001 0000 0001 0010 0011	Source Routeing	Unspecified source routeing error Syntax error in source routeing field Unknown address in source routeing field Path not acceptable
1010 0000 0001	Lifetime	Lifetime expired while data unit in transit Lifetime expired during reassembly
1011 0000 0001 0010 0011 0100	PDU Discarded	Unsupported option not specified Unsupported protocol version Unsupported security option Unsupported source routeing option Unsupported recording of route option
1100 0000	Reassembly	Reassembly interference

7.9.6 Data part

This field contains the entire header of the discarded PDU, and may contain none, some, or all of the data part of the discarded PDU.

7.10 Echo Request PDU

The ERQ PDU has the same format as the DT PDU (see 7.7).

7.11 Echo Response PDU

The ERP PDU has the same format as the DT PDU (see 7.7).

8 Provision of the underlying service

Subnetwork dependent convergence functions may be performed to provide an underlying connectionless-mode service when a real subnetwork does not inherently provide the underlying connectionless-mode service assumed by the protocol. If a subnetwork inherently provides a connection-mode service, a subnetwork dependent convergence function provides a mapping into the required underlying connectionless-mode service. Subnetwork dependent convergence functions may also be required in those cases in which functions assumed from the underlying service are not performed. In some cases, this may require the operation of an explicit protocol (i.e. a protocol involving explicit exchanges of protocol control information between peer Network entities) in the subnetwork dependent convergence protocol (SNDCP) role. However, there may also be cases in which the functionality required to fulfill the SNDCP role consists simply of a set of rules for manipulating the underlying service (without the exchange of PCI between peer Network entities).

8.1 Subnetwork points of attachment

The source address and destination address parameters in the SN-UNITDATA primitive specify the points of attachment to a public or private subnetwork(s). Subnetwork point of attachment addresses (SNPAs) are defined by each individual subnetwork authority. The syntax and semantics of SNPAs are not defined by this Recommendation | International Standard.

8.2 Subnetwork quality of service

Associated with each connectionless-mode transmission, certain measures of quality of service are requested when the SN-UNITDATA primitive action is initiated. These requested measures (or parameter values and options) are based on *a priori* knowledge of the service available from the subnetwork. Knowledge of the nature and type of service available is typically obtained prior to an invocation of the underlying connectionless-mode service.

The quality of service parameters identified for the underlying connectionless-mode service may in some circumstances be directly derivable from or mappable onto those identified in the connectionless-mode Network service. The following parameters as defined in CCITT Rec. X.213 | ISO/IEC 8348 may be employed:

- a) transit delay;
- b) protection against unauthorized access;
- c) cost determinants;
- d) priority; and
- e) residual error probability.

NOTE – For those real subnetworks which do not inherently provide quality of service as a parameter, it is a local matter as to how the semantics of the service requested might be preserved. In particular, there may be instances in which the quality of service requested cannot be maintained. In such circumstances, an attempt shall be made to deliver the protocol data unit at whatever quality of service is available.

In general, either the SNDCF or the subnetwork itself may perform functions associated with specific QOS requests. These functions may be optionally selected by the CLNP. The relevant subnetwork QOS parameters are classified as follows:

- a) those QOS parameters for which the SNDCF or the subnetwork itself performs functions expressly designed to provide information for the route PDU function of the CLNP;

- b) those QOS parameters for which the SNDCF or the subnetwork itself performs functions expressly designed to provide the desired QOS; and
- c) those QOS parameters for which the SNDCF or the subnetwork itself may be called upon to perform either of the functions (a) or (b) above.

The determination of values for these QOS parameters is provided in the following clauses.

8.2.1 Transit delay

Transit delay is the elapsed time between an SN-UNITDATA request and the corresponding SN-UNITDATA indication. Elapsed time values are calculated on SNSDUs that are successfully transmitted. Successful transmission of an SNSDU is defined to occur when an SNSDU transmitted by the sending SNDCF is delivered to the intended destination SNDCF. Transit delay is based on an SNSDU size of 512 octets, and is specified in units of 500 ms.

Transit delay is determined by the SNDCF prior to the processing of any user data by the subnetwork. The mechanism whereby transit delay information is passed to the route PDU function of the CLNP is a local matter. Transit delay may be either measured or estimated. The SNDCFs described herein do not provide any means for measuring or estimating transit delay beyond any such means provided by the underlying subnetwork.

NOTES

1 If transit delay is to be measured, an SNDCP designed to bound the transit time of SNSDUs that cross the subnetwork should be used prior to the processing of any data requests to determine the actual delay.

2 Transit delay within a given subnetwork may vary. Where transit delay is measured, it may be necessary to periodically repeat the measurement process in order to maintain accurate measures in any routing information maintained by the Network entity.

3 If no better measures are available, transit delay may be estimated by sending an SNSDU (via some uniquely identified protocol data unit which prompts a response) and by measuring the elapsed time between the SN-UNITDATA requests and the corresponding SN-UNITDATA indications. This results in an overestimate of delay such that the CLNP may be expected to operate correctly. If transit delay is estimated, it is preferred that estimates be high rather than low in order that uncertainties in transit delay do not prevent the CLNP from discarding protocol data units whose intended lifetime has expired.

8.2.2 Protection from unauthorized access

No recommendation is made concerning how to provide protection against passive monitoring, modification, replay, addition, or deletion of SN-Userdata.

8.2.3 Residual error probability

Residual error probability is estimated as the ratio of lost, duplicated, or incorrectly delivered SNSDUs to total SNSDUs transmitted by the SNDCF during a measurement period. The mechanism whereby residual error probability is passed to the route PDU function of the CLNP is a local matter.

Residual error probability is known by the SNDCF prior to the processing of any user data by the subnetwork, either as a result of the SNDCF having maintained a history of measures of residual error probability, or as a result of information obtained from the provider of the underlying service.

NOTE – For subnetworks which provide a connection-mode service, residual error probability is determined on an individual connection basis.

8.2.4 Cost determinants

This subclause is applicable only to ISO/IEC 8473.

The attempt to satisfy the constraints imposed by the NS user via the cost determinants quality of service parameter is performed by the route PDU function invoked by the CLNP. Where pertinent, information relating to tariff(s) assessed on a per packet or per connection basis is passed to the route PDU function of the CLNP. The mechanism by which this is accomplished is a local matter.

NOTE – The route PDU function invoked by the CLNP may be required to perform the following cost assessments. If:

- a) there is to be no incremental cost incurred in the processing of the SNSDU submitted, and there is a tariff assessed on a per packet basis;
- b) there is to be no additional cost incurred, and no connection is currently available to the specified destination, and a tariff is assessed on a per connection basis by the subnetwork (e.g. for virtual circuit setup, holding time of the virtual circuit, etc.); or
- c) a maximum acceptable cost has been specified for the processing of the NSDU, and that cost is likely to be exceeded,

then the route PDU function should return a result indicating that the CLNP should attempt to deliver the NSDU via some alternate route. If an alternate route cannot be found, a local function may be invoked to notify the NS user of the inability of the NS provider to deliver this NSDU (and possibly subsequent NSDUs) under the stated constraint.

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

8.3 Subnetwork user data

The SN-Userdata is an ordered multiple of octets, and is transferred transparently between the specified subnetwork points of attachment.

The underlying service assumed by the CLNP is required to support a service data unit size of at least 512 octets.

If the minimum service data unit sizes supported by all of the subnetworks involved in the transmission of a particular PDU are known to be large enough that segmentation is not required, then either the full protocol or the non-segmenting protocol subset may be used.

Data received from a subnetwork with protocol identification specifying this protocol (see 7.2.2) shall be processed according to this Recommendation | International Standard.

NOTE – Data with other protocol identification should be ignored, since it may have been sent by an implementation supporting additional protocols intended for use with this protocol.

8.4 Subnetwork dependent convergence functions

The general model for providing the underlying service assumed by the protocol in conjunction with a real subnetwork that uses a connectionless subnetwork access protocol is as follows. The generation of an SN-UNITDATA request by the CLNP results in the generation of a corresponding subnetwork-specific UNITDATA request by the subnetwork dependent convergence function. The receipt of a subnetwork-specific UNITDATA indication associated with delivery of a connectionless data unit to its destination causes the SND CF to generate an SN-UNITDATA indication to the CLNP.

The general model for providing the underlying service assumed by the CLNP in conjunction with a real subnetwork that uses a connection-mode subnetwork access protocol is as follows. The generation of an SN-UNITDATA request by the CLNP causes a connection (logical channel, logical link, or the equivalent) to be made available for the transmission of SN-Userdata. If a connection cannot be made available, the SN-UNITDATA request is discarded. The receipt of subnetwork-specific PDUs containing SN-Userdata causes the SND CF to generate an SN-UNITDATA indication to the CLNP.

Where a real subnetwork is designed to use either a connectionless-mode or a connection-mode subnetwork access protocol, the provision of the underlying service assumed by the CLNP is achieved by using the connectionless-mode alternative.

The way in which the underlying service is provided by specific subnetwork types is defined in other Recommendations | International Standards.

9 Conformance

9.1 Static conformance

9.1.1 End systems

An implementation claiming conformance to this Recommendation | International Standard as an end system shall

- a) support the transmission and reception of NPDU s using the full protocol;
- b) support the reception of NPDU s conveyed using the non-segmenting protocol subset;
- c) support the protocol functions identified in Table 9 as mandatory for end systems; and
- d) be capable of operating over one or more subnetworks, using the appropriate subnetwork dependent convergence function(s) specified in other Recommendations | International Standards.

Such an end system may (as implementation options), but is not required to

- e) support the transmission of NPDU's using the non-segmenting protocol subset;
- f) support the transmission and reception of NPDU's using the inactive Network layer protocol subset; and
- g) support any of the protocol functions identified in Table 9 as optional for end systems.

NOTE – Although item (a) above requires end systems to support both the transmission and the reception of NPDU's, the requirements for transmission and reception are specified separately in Table 9. In general, the procedures to be followed in order to support a given function are different for the sending and receiving senses. The separate specification (1) distinguishes between the requirements for two functions (PDU lifetime control and padding) for which support is mandatory for one sense of PDU transfer and optional for the other; and (2) clarifies the fact that support of several of the functions is applicable only for one sense of PDU transfer.

Table 9 – Static conformance requirements

Protocol function	Reference	End system (Note 1)		Intermediate system
		Sending	Receiving	
PDU Composition (Note 2)	6.1	M	N/A	N/A
PDU Decompositon (Note 2)	6.2	N/A	M	N/A
Header Format Analysis	6.3	N/A	M	M
PDU Lifetime Control	6.4	M	O	M
Route PDU	6.5	M	N/A	M
Forward PDU	6.6	M	N/A	M
Segmentation (Note 2)	6.7	M	N/A	(Note 3)
Reassembly (Note 2)	6.8	N/A	M	O (Note 4)
Discard PDU	6.9	N/A	M	M
Error Reporting	6.10	M	M	M
Header Error Detection	6.11	M	M	M
Security	6.13	O	O (Note 4)	O (Note 4)
Complete Source Routeing	6.14	O	N/A	O (Note 4)
Complete Route Recording	6.15	O	O (Note 4)	O (Note 4)
Echo request	6.19	O	O (Note 4)	O (Note 4)
Echo response	6.20	N/A	O (Note 4)	O (Note 4)
Partial Source Routeing	6.14	O	N/A	O (Note 4)
Partial Route Recording	6.15	O	O (Note 4)	O (Note 4)
Priority	6.17	O	O (Note 4)	O (Note 4)
QOS Maintenance	6.16	O	O (Note 4)	O (Note 4)
Congestion Notification	6.18	N/A	O (Note 4)	O (Note 4)
Padding	6.12	O	M	M

M Mandatory function; this function shall be implemented.
O Implementation option, as described in the text.
N/A Not applicable.

NOTES

1 The status in the “sending” column applies to the support of the given function for DT, ER, ERQ, and ERP PDUs sent by the end system; similarly, the status in the “receiving” column applies to the support of the given function for DT, ER, ERQ, and ERP PDUs received by the end system.

2 The PDU composition, PDU decomposition, segmentation, and reassembly functions are not relevant for ER PDUs.

3 The segment PDU function is in general mandatory for an intermediate system. However, a system which is to be connected only to subnetworks that all offer the same maximum SDU size (such as identical local area networks) will not need to perform this function, and therefore does not need to implement it.

4 See 9.2 for related dynamic conformance requirements that apply when this option is not supported.

9.1.2 Intermediate systems

An implementation claiming conformance to this Recommendation | International Standard as an intermediate system shall

- a) support the protocol functions identified in Table 9 as mandatory for intermediate systems; and
- b) be capable of operating over one or more subnetworks, using the appropriate subnetwork dependent convergence function(s) specified in other Recommendations | International Standards.

Such an intermediate system may (as an implementation option), but is not required to

- c) support any of the protocol functions identified in Table 9 as optional for intermediate systems.

9.2 Dynamic conformance

An implementation claiming conformance to this Recommendation | International Standard shall exhibit externally observable behaviour consistent with its having implemented:

- a) each protocol function that it supports in accordance with the function's specification, as contained in the subclause referenced from Table 9; and
- b) the relevant subnetwork dependent convergence function(s) in accordance with the specification contained in other Recommendations | International Standards.

All PDUs transmitted shall be structured as specified in clause 7.

An implementation that does not support a function identified in Table 9 as optional shall, upon receiving a PDU in which that function is selected, either discard the PDU and invoke the error reporting function, or process the PDU as though the function had not been selected, in accordance with the specification contained in 6.21.

9.3 PICS proforma

The supplier of a protocol implementation that claims to conform to this Recommendation | International Standard shall complete a copy of the PICS proforma provided in Annex A, including the information necessary to identify both the supplier and the implementation.

Annex A¹⁾ PICS proforma

(This annex forms an integral part of this Recommendation | International Standard)

A.1 Introduction

The supplier of a protocol implementation which is claimed to conform to this Recommendation | International Standard shall complete the following Protocol Implementation Conformance Statement (PICS) proforma.

A completed PICS proforma is the PICS for the implementation in question. The PICS is a statement of which capabilities and options of the protocol have been implemented. The PICS can have a number of uses, including use

- by the protocol implementor, as a check-list to reduce the risk of failure to conform to the standard through oversight;
- by the supplier and acquirer – or potential acquirer – of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma;
- by the user – or potential user – of the implementation, as a basis for initially checking the possibility of interworking with another implementation (note that, while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible PICSs);
- by a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A.2 Abbreviations and special symbols

A.2.1 Status symbols

M	mandatory
O	optional
O.<n>	optional, but support of at least one of the group of options labelled by the same numeral <n> is required
X	prohibited
<pred>	conditional-item symbol, including predicate identification (see A.3.4)
^	logical negation, applied to a conditional item's predicate

A.2.2 Other symbols

<t>	receive aspects of an item
<s>	send aspects of an item

A.3 Instructions for completing the PICS proforma

A.3.1 General structure of the PICS proforma

The first part of the PICS proforma – Implementation Identification and Protocol Summary – is to be completed as indicated with the information necessary to identify fully both the supplier and the implementation.

¹⁾ Copyright release for PICS proformas.

Users of this Recommendation | International Standard may freely reproduce the PICS proforma in this annex so that it can be used for its intended purpose and may further publish the completed PICS.

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

The main part of the PICS proforma is a fixed-format questionnaire divided into a number of major subclauses; these can be divided into further subclauses each containing a group of individual items. Answers to the questionnaire items are to be provided in the rightmost column, either by simply marking an answer to indicate a restricted choice (usually Yes or No), or by entering a value or a set or range of values.

NOTE – There are some items for which two or more choices from a set of possible answers can apply. All relevant choices are to be marked in these cases.

Each item is identified by an item reference in the first column; the second column contains the question to be answered; and the third column contains the reference or references to the material that specifies the item in the main body of this Recommendation | International Standard. The remaining columns record the status of the item – whether support is mandatory, optional, prohibited, or conditional – and provide space for the answers (see also A.3.4).

A supplier may also provide further information, categorized as either Additional Information or Exception Information. When present, each kind of further information is to be provided in a further subclause of items labelled A<i> or X<i>, respectively, for cross-referencing purposes, where <i> is any unambiguous identification for the item (e.g. a number); there are no other restrictions on its format or presentation.

A completed PICS proforma, including any Additional Information and Exception Information, is the Protocol Implementation Conformance Statement for the implementation in question.

NOTE – Where an implementation is capable of being configured in more than one way, a single PICS may be able to describe all such configurations. However, the supplier has the choice of providing more than one PICS, each covering some subset of the implementation's configuration capabilities, in cases where this makes for easier and clearer presentation of the information.

A.3.2 Additional information

Items of Additional Information allow a supplier to provide further information intended to assist in the interpretation of the PICS. It is not intended or expected that a large quantity will be supplied, and a PICS can be considered complete without any such information. Examples might be an outline of the ways in which a (single) implementation can be set up to operate in a variety of environments and configurations, or a brief rationale – based perhaps upon specific application needs – for the exclusion of features which, although optional, are nonetheless commonly present in implementations of this protocol.

References to items of Additional Information may be entered next to any answer in the questionnaire, and may be included in items of Exception Information.

A.3.3 Exception information

It may occasionally happen that a supplier will wish to answer an item with mandatory or prohibited status (after any conditions have been applied) in a way that conflicts with the indicated requirement. No pre-printed answer will be found in the support column for this; instead, the supplier shall write the missing answer into the Support column, together with an X<i> reference to an item of Exception Information, and shall provide the appropriate rationale in the Exception Information item itself.

An implementation for which an Exception Information item is required in this way does not conform to this Recommendation | International Standard.

NOTE – A possible reason for the situation described above is that a defect in the standard has been reported, a correction for which is expected to change the requirement not met by the implementation.

A.3.4 Conditional status

A.3.4.1 Conditional items

The PICS proforma contains a number of conditional items. These are items for which the status – mandatory, optional, or prohibited – that applies is dependent upon whether or not certain other items are supported, or upon the values supported for other items.

In many cases, whether or not the item applies at all is conditional in this way, as well as the status when the item does apply.

Where a group of items is subject to the same condition for applicability, a separate preliminary question about the condition appears at the head of the group, with an instruction to skip to a later point in the questionnaire if the “Not Applicable” answer is selected. Otherwise, individual conditional items are indicated by one or more conditional symbols (on separate lines) in the status column.

A conditional symbol is of the form “<pred>:<x>” where “<pred>” is a predicate as described in A.3.4.2, and “<x>” is one of the status symbols M, O, O.<n>, or X.

If the value of the predicate in any line of a conditional item is true (see A.3.4.2), then the conditional item is applicable, and its status is that indicated by the status symbol following the predicate; the answer column is to be marked in the usual way. If the value of a predicate is false, the Not Applicable (N/A) answer is to be marked in the relevant line. Each line in a multi-line conditional item should be marked: at most one line will require an answer other than N/A.

A.3.4.2 Predicates

A predicate is one of the following:

- a) an item-reference for an item in the PICS proforma: the value of the predicate is true if the item is marked as supported, and is false otherwise;
- b) a predicate name, for a predicate defined elsewhere in the PICS proforma (usually in the Major Capabilities section or at the end of the section containing the conditional item) (see below); or
- c) the logical negation symbol “^” prefixed to an item-reference or predicate name: the value of the predicate is true if the value of the predicate formed by omitting the “^” is false, and vice versa.

The definition for a predicate name is one of the following:

- a) an item-reference, evaluated as at (a) above;
- b) a relation containing a comparison operator (=, <, etc.) with at least one of its operands being an item-reference for an item taking numerical values as its answer; the predicate is true if the relation holds when each item-reference is replaced by the value entered in the Support column as an answer to the item referred to; or
- c) a boolean expression constructed by combining simple predicates, as in (a) and (b), using the boolean operators AND, OR, and NOT, and parentheses, in the usual way; the value of such a predicate is true if the boolean expression evaluates to true when the simple predicates are interpreted as described above.

Each item whose reference is used in a predicate or predicate definition is indicated by an asterisk in the Item column.

A.4 Identification

A.4.1 Implementation identification

Supplier	
Contact point for queries about the PICS	
Implementation name(s) and version(s)	
Other information necessary for full identification [e.g. name(s) and version(s) of machines and/or operating systems, system name(s)]	
NOTES	
1 Only the first three items are required for all implementations; other information may be completed as appropriate in meeting the requirement for full identification.	
2 The terms Name and Version should be interpreted appropriately to correspond with a supplier’s terminology (e.g. Type, Series, Model).	

A.4.2 Protocol summary

Identification of protocol specification	ITU-T Recommendation X.233 (1994) ISO/IEC 8473-1:1994
Identification of corrigenda and amendments to the PICS proforma	
Protocol version(s) supported	
Have any Exception Information items been required (see A.3.3)?	YES <input type="checkbox"/> NO <input type="checkbox"/>
(The answer YES means that the implementation does not conform to this Recommendation International Standard)	

Date of statement	
-------------------	--

A.5 Major capabilities

Item	Capability	Reference	Status	Support
* ES	End system		O.1	YES <input type="checkbox"/> NO <input type="checkbox"/>
* IS	Intermediate system		O.1	YES <input type="checkbox"/> NO <input type="checkbox"/>
FL-r	<r> Full protocol	6	M	YES <input type="checkbox"/>
FL-s	<s> Full protocol	6	M	YES <input type="checkbox"/>
NSS-r	<r> Non-segmenting subset	5.2	M	YES <input type="checkbox"/>
* NSS-s	<s> Non-segmenting subset	5.2	IS:M ^IS:O	N/A <input type="checkbox"/> YES <input type="checkbox"/> YES <input type="checkbox"/> NO <input type="checkbox"/>
* IAS-r	<r> Inactive subset	5.2	ES:O	N/A <input type="checkbox"/> YES <input type="checkbox"/> NO <input type="checkbox"/>
* IAS-s	<s> Inactive subset	5.2	IAS-r:M ^IAS-r:X	N/A <input type="checkbox"/> YES <input type="checkbox"/> N/A <input type="checkbox"/> NO <input type="checkbox"/>

A.6 End systems**A.6.1 Applicability**

The PICS proforma items in A.6 are applicable only to end system implementations; i.e. those in which item ES in A.5 is supported. The items in A.6.4.4 are applicable only to end system implementations that support the echo request function; i.e., those in which item eEreq in A.6.2 is supported. The items in A.6.4.5 are applicable only to end system implementations that support the echo response function; i.e. those in which item eErsp in A.6.2 is supported.

A.6.2 Supported functions

Item	Function	Reference	Status	Support
ePDUC	PDU composition	6.1	M	YES ☺
ePDUD	PDU decomposition	6.2	M	YES ☺
eHFA	Header format analysis	6.3	M	YES ☺
ePDUL-s	<s> PDU lifetime control	6.4	M	YES ☺
ePDUL-r	<r> PDU lifetime control	6.4	O	YES ☺ NO ☺
eRout	Route PDU	6.5	M	YES ☺
eForw	Forward PDU	6.6	M	YES ☺
eSegm	Segment PDU	6.7	M	YES ☺
eReas	Reassemble PDU	6.8	M	YES ☺
eDisc	Discard PDU	6.9	M	YES ☺
eErep	Error reporting	6.10	M	YES ☺
eEdec-s	<s> Header error detection	6.11	M	YES ☺
eEdec-r	<r> Header error detection	6.11	M	YES ☺
* eSecu-s	<s> Security	6.13	M	YES ☺ NO ☺
* eSecu-r	<r> Security	6.13	O	YES ☺ NO ☺
* eCRR-s	<s> Complete route recording	6.15	O	YES ☺ NO ☺
* eCRR-r	<r> Complete route recording	6.15	O	YES ☺ NO ☺
* ePRR-s	<s> Partial route recording	6.15	O	YES ☺ NO ☺
* ePRR-r	<r> Partial route recording	6.15	O	YES ☺ NO ☺
* eCSR	Complete source routing	6.14	O	YES ☺ NO ☺
* ePSR	Partial source routing	6.14	O	YES ☺ NO ☺
* ePri-s	<s> Priority	6.17	O	YES ☺ NO ☺
* ePri-r	<r> Priority	6.17	O	YES ☺ NO ☺
* eQOSM-s	<s> QOS maintenance	6.16	O	YES ☺ NO ☺
* eQOSM-r	<r> QOS maintenance	6.16	O	YES ☺ NO ☺
* eCong-s	<s> Congestion notification	6.18	eQOSM-s:M	N/A ☺ YES ☺
* eCong-r	<r> Congestion notification	6.18	O	YES ☺ NO ☺
* ePadd-s	<s> Padding	6.12	O	YES ☺ NO ☺
ePadd-r	<r> Padding	6.12	M	YES ☺
eEreq	Echo request	6.19	O	YES ☺ NO ☺
eErsp	Echo response	6.20	O	YES ☺ NO ☺
eSegS	Create segments smaller than necessary	6.8	O	YES ☺ NO ☺

A.6.3 Supported PDUs

Item	NPDU	Reference	Status	Support
eDT-t	DT (full protocol) transmit	7.7	M	YES ☺
eDT-r	DT (full protocol) receive	7.7	M	YES ☺
eDTNS-t	DT (non-segmenting) transmit	7.7	NSS-s:M	N/A ☺ YES ☺
eDTNS-r	DT (non-segmenting) receive	7.7	M	YES ☺
eER-t	ER transmit	7.9	M	YES ☺
eER-r	ER receive	7.9	M	YES ☺
eIN-t	Inactive PDU transmit	7.8	IAS-s:M	N/A ☺ YES ☺
eIN-r	Inactive PDU receive	7.8	IAS-r:M	N/A ☺ YES ☺
eERQ-t	ERQ transmit	7.10	eEreq:M	N/A ☺ YES ☺
eERQ-r	ERQ receive	7.10	M	YES ☺
eERP-t	ERP transmit	7.11	eErsp:M	N/A ☺ YES ☺
eERP-r	ERP receive	7.11	M	YES ☺

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

A.6.4 Supported parameters**A.6.4.1 DT parameters**

Item	Parameter	Reference	Status	Support
edFxFt-s	<s> Fixed part	7.2	M	YES ☺
edFxFt-r	<r> Fixed part	7.2	M	YES ☺
edAddr-s	<s> Addresses	7.3	M	YES ☺
edAddr-r	<r> Addresses	7.3	M	YES ☺
edSeg-s	<s> Segmentation part	7.4	M	YES ☺
edSeg-r	<r> Segmentation part	7.4	M	YES ☺
edPadd-s	<s> Padding	7.5.2	ePadd-s:M	N/A ☺ YES ☺
edPadd-r	<r> Padding	7.5.2	M	YES ☺
edSecu-s	<s> Security	7.5.3	eSecu-s:M	N/A ☺ YES ☺
edSecu-r	<r> Security	7.5.3	eSecu-r:M	N/A ☺ YES ☺
edCRR-s	<s> Complete route recording	7.5.5	eCRR-s:M	N/A ☺ YES ☺
edCRR-r	<r> Complete route recording	7.5.5	eCRR-r:M	N/A ☺ YES ☺
edPRR-s	<s> Partial route recording	7.5.5	ePRR-s:M	N/A ☺ YES ☺
edPRR-r	<r> Partial route recording	7.5.5	ePRR-r:M	N/A ☺ YES ☺
edCSR-s	<s> Complete source routeing	7.5.4	eCSR:M	N/A ☺ YES ☺
edPSR-s	<s> Partial source routeing	7.5.4	ePSR:M	N/A ☺ YES ☺
edQOSM-s	<s> QOS maintenance	7.5.6	c1:M	N/A ☺ YES ☺
edQOSM-r	<r> QOS maintenance	7.5.6	c2:M	N/A ☺ YES ☺
edPri-s	<s> Priority	7.5.7	ePri-s:M	N/A ☺ YES ☺
edPri-r	<r> Priority	7.5.7	ePri-r:M	N/A ☺ YES ☺
edData-s	<s> Data	7.6	M	YES ☺
edData-r	<r> Data	7.6	M	YES ☺
edUnSup2	Are received PDUs containing parameters selecting unsupported Type 2 functions discarded and where appropriate an Error Report PDU generated?	6.21	M	YES ☺
edUnSup3	Are parameters selecting unsupported Type 3 functions ignored?	6.21	M	YES ☺
Definition of conditional status entries:				
c1:	eQOSM-s OR eCong-s			
c2:	eQOSM-r OR eCong-r			






































A.6.4.2 ER parameters

Item	Parameter	Reference	Status	Support
eeFxFt-s	<s> Fixed part	7.2	M	YES ☺
eeFxFt-r	<r> Fixed part	7.2	M	YES ☺
eeAddr-s	<s> Addresses	7.3	M	YES ☺
eeAddr-r	<r> Addresses	7.3	M	YES ☺
eePadd-s	<s> Padding	7.5.2	ePadd-s:M	N/A ☺ YES ☺
eePadd-r	<r> Padding	7.5.2	M	YES ☺
eeSecu-s	<s> Security	7.5.3	eSecu-s:M	N/A ☺ YES ☺
eeSecu-r	<r> Security	7.5.3	eSecu-r:M	N/A ☺ YES ☺
eeCRR-s	<s> Complete route recording	7.5.5	eCRR-s:M	N/A ☺ YES ☺
eeCRR-r	<r> Complete route recording	7.5.5	eCRR-r:M	N/A ☺ YES ☺
eePRR-s	<s> Partial route recording	7.5.5	ePRR-s:M	N/A ☺ YES ☺
eePRR-r	<r> Partial route recording	7.5.5	ePRR-r:M	N/A ☺ YES ☺
eeCSR-s	<s> Complete source routeing	7.5.4	eCSR:M	N/A ☺ YES ☺
eePSR-r	<r> Partial source routeing	7.5.4	ePSR:M	N/A ☺ YES ☺
eeQOSM-s	<s> QOS maintenance	7.5.6	c1:M	N/A ☺ YES ☺
eeQOSM-r	<r> QOS maintenance	7.5.6	c2:M	N/A ☺ YES ☺
eePri-s	<s> Priority	7.5.7	ePri-s:M	N/A ☺ YES ☺
eePri-r	<r> Priority	7.5.7	ePri-r:M	N/A ☺ YES ☺
eeDisc-s	<s> Reason for discard	7.9.5	M	YES ☺
eeDisc-r	<r> Reason for discard	7.9.5	M	YES ☺
eeData-s	<s> Data	7.9.6	M	YES ☺
eeData-r	<r> Data	7.9.6	M	YES ☺
eeUnSup2	Are received PDUs containing parameters selecting unsupported Type 2 functions discarded?	6.21	M	YES ☺
edUnSup3	Are parameters selecting unsupported Type 3 functions ignored?	6.21		YES ☺
Definition of conditional status entries:				
c1:	eQOSM-s OR eCong-s			
c2:	eQOSM-r OR eCong-r			

A.6.4.3 Inactive network layer protocol PDU parameters

Item	Parameter	Reference	Status	Support
eiNLPI-s	<s> Inactive network layer protocol identifier	7.8.2	IAS-s:M	N/A ☺ YES ☺
eiNLPI-r	<r> Inactive network layer protocol identifier	7.8.2	IAS-r:M	N/A ☺ YES ☺
eiData-s	<s> Data	7.8.3	IAS-s:M	N/A ☺ YES ☺
eiData-r	<r> Data	7.8.3	IAS-r:M	N/A ☺ YES ☺

A.6.4.4 ERQ parameters

Item	Parameter	Reference	Status	Support
eqFxFt-s	<s> Fixed part	7.2	M	YES 
eqFxFt-r	<r> Fixed part	7.2	M	YES 
eqAddr-s	<s> Addresses	7.3	M	YES 
eqAddr-r	<r> Addresses	7.3	M	YES 
eqSeg-s	<s> Segmentation part	7.4	M	YES 
eqSeg-r	<r> Segmentation part	7.4	M	YES 
eqPadd-s	<s> Padding	7.5.2	ePadd-s:M	N/A  YES 
eqPadd-r	<r> Padding	7.5.2	M	YES 
eqSecu-s	<s> Security	7.5.3	eSecu-s:M	N/A  YES 
eqSecu-r	<r> Security	7.5.3	eSecu-r:M	N/A  YES 
eqCRR-s	<s> Complete route recording	7.5.5	eCRR-s:M	N/A  YES 
eqCRR-r	<r> Complete route recording	7.5.5	eCRR-r:M	N/A  YES 
eqPRR-s	<s> Partial route recording	7.5.5	ePRR-s:M	N/A  YES 
eqPRR-r	<r> Partial route recording	7.5.5	ePRR-r:M	N/A  YES 
eqCSR-s	<s> Complete source routeing	7.5.4	eCSR:M	N/A  YES 
eqPSR-s	<s> Partial source routeing	7.5.4	ePSR:M	N/A  YES 
eqQOSM-s	<s> QOS maintenance	7.5.6	c1:M	N/A  YES 
eqQOSM-r	<r> QOS maintenance	7.5.6	c2:M	N/A  YES 
eqPri-s	<s> Priority	7.5.7	ePri-s:M	N/A  YES 
eqPri-r	<r> Priority	7.5.7	ePri-r:M	N/A  YES 
eqData-s	<s> Data	7.6	M	YES 
eqData-r	<r> Data	7.6	M	YES 
eqUnSup2	Are received PDUs containing parameters selecting unsupported Type 2 functions discarded and where appropriate an Error Report PDU generated?	6.21	M	YES 
eqUnSup3	Are parameters selecting unsupported Type 3 functions ignored?	6.21	M	YES 
Definition of conditional status entries:				
c1: eQOSM-s OR eCong-s				
c2: eQOSM-r OR eCong-r				

A.6.4.5 ERP parameters

Item	Parameter	Reference	Status	Support
epFxFt-s	<s> Fixed part	7.2	M	YES ☺
epFxFt-r	<r> Fixed part	7.2	M	YES ☺
epAddr-s	<s> Addresses	7.3	M	YES ☺
epAddr-r	<r> Addresses	7.3	M	YES ☺
epSeg-s	<s> Segmentation part	7.4	M	YES ☺
epSeg-r	<r> Segmentation part	7.4	M	YES ☺
epPadd-s	<s> Padding	7.5.2	ePadd-s:M	N/A ☺ YES ☺
epPadd-r	<r> Padding	7.5.2	M	YES ☺
epSecu-s	<s> Security	7.5.3	eSecu-s:M	N/A ☺ YES ☺
epSecu-r	<r> Security	7.5.3	eSecu-r:M	N/A ☺ YES ☺
epCRR-s	<s> Complete route recording	7.5.5	eCRR-s:M	N/A ☺ YES ☺
epCRR-r	<r> Complete route recording	7.5.5	eCRR-r:M	N/A ☺ YES ☺
epPRR-s	<s> Partial route recording	7.5.5	ePRR-s:M	N/A ☺ YES ☺
epPRR-r	<r> Partial route recording	7.5.5	ePRR-r:M	N/A ☺ YES ☺
epCSR-s	<s> Complete source routeing	7.5.4	eCSR:M	N/A ☺ YES ☺
epPSR-s	<s> Partial source routeing	7.5.4	ePSR:M	N/A ☺ YES ☺
epQOSM-s	<s> QOS maintenance	7.5.6	c1:M	N/A ☺ YES ☺
epQOSM-r	<r> QOS maintenance	7.5.6	c2:M	N/A ☺ YES ☺
epPri-s	<s> Priority	7.5.7	ePri-s:M	N/A ☺ YES ☺
epPri-r	<r> Priority	7.5.7	ePri-r:M	N/A ☺ YES ☺
epData-s	<s> Data	7.6	M	YES ☺
epData-r	<r> Data	7.6	M	YES ☺
epUnSup2	Are received PDUs containing parameters selecting unsupported Type 2 functions discarded and where appropriate an Error Report PDU generated?	6.21	M	YES ☺
epUnSup3	Are parameters selecting unsupported Type 3 functions ignored?	6.21	M	YES ☺
Definition of conditional status entries:				
c1: eQOSM-s OR eCong-s				
c2: eQOSM-r OR eCong-r				

A.6.5 Timers

Item	Timer	Reference	Status	Values	Support	Values supported
eLifReas	Is reassembly timer <= received derived PDU lifetime?	6.8	M		YES ☺	
eReasLim	What values of the reassembly timer are supported?	6.8		500 ms to 127.5 s		

Superseded by a more recent version ISO/IEC 8473-1 : 1994 (E)

A.7 Intermediate systems**A.7.1 Applicability**

The PICS proforma items in A.7 are applicable only to intermediate system implementations; i.e. those in which item IS in A.5 is supported. The items in A.7.4.3 are applicable only to intermediate system implementations that support the echo request function; i.e. those in which item eReq in A.7.2 is supported. The items in A.7.4.4 are applicable only to intermediate system implementations that support the echo response function; i.e. those in which item eErsp in A.7.2 is supported.









A.7.2 Supported functions

Item	Function	Reference	Status	Support
iPDUC	PDU composition	6.1	M	YES ☺
iPDUD	PDU decomposition	6.2	M	YES ☺
iHFA	Header format analysis	6.3	M	YES ☺
iPDUL	<s> PDU lifetime control	6.4	M	YES ☺
iRout	Route PDU	6.5	M	YES ☺
iForw	Forward PDU	6.6	M	YES ☺
iSegm	Segment PDU	6.7	iDSNS:M	N/A ☺ YES ☺
iReas	Reassemble PDU	6.8	iErsp:M ^ iErsp:O	N/A ☺ YES ☺ YES ☺ NO ☺
iDisc	Discard PDU	6.9	M	YES ☺
iErep	Error reporting	6.10	M	YES ☺
iEdec	<s> Header error detection	6.11	M	YES ☺
* iSecu	<s> Security	6.13	O	YES ☺ NO ☺
* iCRR	<s> Complete route recording	6.15	O	YES ☺ NO ☺
* iPRR	<s> Partial route recording	6.15	O	YES ☺ NO ☺
* iCSR	Complete source routeing	6.14	O	YES ☺ NO ☺
* iPSR	Partial source routeing	6.14	O	YES ☺ NO ☺
* iPri	<s> Priority	6.17	O	YES ☺ NO ☺
* iQOSM	<s> QOS maintenance	6.16	O	YES ☺ NO ☺
* iCong	<s> Congestion notification	6.18	O	YES ☺ NO ☺
iPadd	<s> Padding	6.12	M	YES ☺
iEreq	Echo request	6.19	O	YES ☺ NO ☺
* iErsp	Echo response	6.20	O	YES ☺ NO ☺
iSegS	Create segments smaller than necessary	6.8	O	YES ☺ NO ☺
iDSNS	Simultaneous support of subnetworks with different SN-Userdata sizes	Table 9 Note 3	O	YES ☺ NO ☺





































A.7.3 Supported PDUs

Item	NPDU	Reference	Status	Support
iDT-t	DT (full protocol) transmit	7.7	M	YES ☺
iDT-r	DT (full protocol) receive	7.7	M	YES ☺
iDTNS-t	DT (non-segmenting) transmit	7.7	M	YES ☺
iDTNS-r	DT (non-segmenting) receive	7.7	M	YES ☺
iER-t	ER transmit	7.9	M	YES ☺
iER-r	ER receive	7.9	M	YES ☺
iERQ-t	ERQ transmit	7.10	iEreq:M	N/A ☺ YES ☺
iERQ-r	ERQ receive	7.10	M	YES ☺
iERP-t	ERP transmit	7.11	iErsp:M	N/A ☺ YES ☺
iERP-r	ERP receive	7.11	M	YES ☺





































A.7.4 Supported parameters**A.7.4.1 DT parameters**

Item	Parameter	Reference	Status	Support
idFxPt-s	<s> Fixed part	7.2	M	YES 
idFxPt-r	<r> Fixed part	7.2	M	YES 
idAddr-s	<s> Addresses	7.3	M	YES 
idAddr-r	<r> Addresses	7.3	M	YES 
idSeg-s	<s> Segmentation part	7.4	M	YES 
idSeg-r	<r> Segmentation part	7.4	M	YES 
idPadd-s	<s> Padding	ITU-T Rec. X.233 (1993 E) 7.5.2	Superseded by a more recent version M	YES 
idPadd-r	<r> Padding	7.5.2	M	YES 

A.7.4.2 ER parameters

Item	Parameter	Reference	Status	Support
ieFxFt-s	<s> Fixed part	7.2	M	YES 
ieFxFt-r	<r> Fixed part	7.2	M	YES 
ieAddr-s	<s> Addresses	7.3	M	YES 
ieAddr-r	<r> Addresses	7.3	M	YES 
iePadd-s	<s> Padding	7.5.2	M	YES 
iePadd-r	<r> Padding	7.5.2	M	YES 
ieSecu-s	<s> Security	7.5.3	iSecu:M	N/A  YES 
ieSecu-r	<r> Security	7.5.3	iSecu:M	N/A  YES 
ieCRR-s	<s> Complete route recording	7.5.5	iCRR:M	N/A  YES 
ieCRR-r	<r> Complete route recording	7.5.5	iCRR:M	N/A  YES 
iePRR-s	<s> Partial route recording	7.5.5	M	YES 
iePRR-r	<r> Partial route recording	7.5.5	iPRR:M	N/A  YES 
ieCSR-s	<s> Complete source routeing	7.5.4	iCSR:M	N/A  YES 
ieCSR-r	<r> Complete source routeing	7.5.4	iCSR:M	N/A  YES 
iePSR-s	<s> Partial source routeing	7.5.4	M	YES 
iePSR-r	<r> Partial source routeing	7.5.4	iPSR:M	N/A  YES 
ieQOSM-s	<s> QOS maintenance	7.5.6	M	YES 
ieQOSM-r	<r> QOS maintenance	7.5.6	c1:M	N/A  YES 
iePri-s	<s> Priority	7.5.7	M	YES 
iePri-r	<r> Priority	7.5.7	iPri:M	N/A  YES 
ieDisc-s	<s> Reason for discard	7.9.5	M	YES 
ieDisc-r	<r> Reason for discard	7.9.5	M	YES 
ieData-s	<s> Data	7.9.6	M	YES 
ieData-r	<r> Data	7.9.6	M	YES 
ieUnSup2	Are received PDUs containing parameters selecting unsupported Type 2 functions discarded?	6.21	M	YES 
ieUnSup3	Are parameters selecting unsupported Type 3 functions ignored?	6.21	M	YES 
Definition of conditional status entry: c1: iQOSM OR iCong				

A.7.4.3 ERQ parameters

Item	Parameter	Reference	Status	Support
iqFxFt-s	<s> Fixed part	7.2	M	YES 
iqFxFt-r	<r> Fixed part	7.2	M	YES 
iqAddr-s	<s> Addresses	7.3	M	YES 
iqAddr-r	<r> Addresses	7.3	M	YES 
iqSeg-s	<s> Segmentation part	7.4	M	YES 
iqSeg-r	<r> Segmentation part	7.4	M	YES 
iqPadd-s	<s> Padding	7.5.2	M	YES 
iqPadd-r	<r> Padding	7.5.2	M	YES 
iqSecu-s	<s> Security	7.5.3	iSecu:M	N/A  YES 
iqSecu-r	<r> Security	7.5.3	iSecu:M	N/A  YES 
iqCRR-s	<s> Complete route recording	7.5.5	iCRR:M	N/A  YES 
iqCRR-r	<r> Complete route recording	7.5.5	iCRR:M	N/A  YES 
iqPRR-s	<s> Partial route recording	7.5.5	M	YES 
iqPRR-r	<r> Partial route recording	7.5.5	iPRR:M	N/A  YES 
iqCSR-s	<s> Complete source routeing	7.5.4	iCSR:M	N/A  YES 
iqCSR-r	<r> Complete source routeing	7.5.4	iCSR:M	N/A  YES 
iqPSR-s	<s> Partial source routeing	7.5.4	M	YES 
iqPSR-r	<r> Partial source routeing	7.5.4	iPSR:M	N/A  YES 
iqQOSM-s	<s> QOS maintenance	7.5.6	M	YES 
iqQOSM-r	<r> QOS maintenance	7.5.6	c1:M	N/A  YES 
iqPri-s	<s> Priority	7.5.7	M	YES 
iqPri-r	<r> Priority	7.5.7	iPri:M	N/A  YES 
iqData-s	<s> Data	7.6	M	YES 
iqData-r	<r> Data	7.6	M	YES 
iqUnSup2	Are received PDUs containing parameters selecting unsupported Type 2 functions discarded and where appropriate an Error Report PDU generated?	6.21	M	YES 
iqUnSup3	Are parameters selecting unsupported Type 3 functions ignored?	6.21	M	YES 
Definition of conditional status entry: c1: iQOSM OR iCong				

A.7.4.4 ERP parameters

Item	Parameter	Reference	Status	Support
ipFxFt-s	<s> Fixed part	7.2	M	YES ☺
ipFxFt-r	<r> Fixed part	7.2	M	YES ☺
ipAddr-s	<s> Addresses	7.3	M	YES ☺
ipAddr-r	<r> Addresses	7.3	M	YES ☺
ipSeg-s	<s> Segmentation part	7.4	M	YES ☺
ipSeg-r	<r> Segmentation part	7.4	M	YES ☺
ipPadd-s	<s> Padding	7.5.2	M	YES ☺
ipPadd-r	<r> Padding	7.5.2	M	YES ☺
ipSecu-s	<s> Security	7.5.3	iSecu:M	N/A ☺ YES ☺
ipSecu-r	<r> Security	7.5.3	iSecu:M	N/A ☺ YES ☺
ipCRR-s	<s> Complete route recording	7.5.5	iCRR:M	N/A ☺ YES ☺
ipCRR-r	<r> Complete route recording	7.5.5	iCRR:M	N/A ☺ YES ☺
ipPRR-s	<s> Partial route recording	7.5.5	M	YES ☺
ipPRR-r	<r> Partial route recording	7.5.5	iPRR:M	N/A ☺ YES ☺
ipCSR-s	<s> Complete source routeing	7.5.4	iCSR:M	N/A ☺ YES ☺
ipCSR-r	<r> Complete source routeing	7.5.4	iCSR:M	N/A ☺ YES ☺
ipPSR-s	<s> Partial source routeing	7.5.4	M	YES ☺
ipPSR-r	<r> Partial source routeing	7.5.4	iPSR:M	N/A ☺ YES ☺
ipQOSM-s	<s> QOS maintenance	7.5.6	M	YES ☺
ipQOSM-r	<r> QOS maintenance	7.5.6	c1:M	N/A ☺ YES ☺
ipPri-s	<s> Priority	7.5.7	M	YES ☺
ipPri-r	<r> Priority	7.5.7	iPri:M	N/A ☺ YES ☺
ipData-s	<s> Data	7.6	M	YES ☺
ipData-r	<r> Data	7.6	M	YES ☺
ipUnSup2	Are received PDUs containing parameters selecting unsupported Type 2 functions discarded and where appropriate an Error Report PDU generated?	6.21	M	YES ☺
ipUnSup3	Are parameters selecting unsupported Type 3 functions ignored?	6.21	M	YES ☺
Definition of conditional status entry: c1: iQOSM OR iCong				

A.7.5 Timer and parameter values

Item	Timer	Reference	Status	Values	Support	Values supported
eLifReas	Is reassembly timer <= received derived PDU lifetime?	6.8	iReas:M		N/A ☺ YES ☺	
eReasLim	What values of the reassembly timer are supported?	6.8		500 ms to 127.5 s		

Annex B

Supporting technical material

(This annex does not form an integral part of this Recommendation | International Standard)

B.1 Data unit lifetime

There are two primary purposes for providing a PDU lifetime capability in the protocol defined by this Recommendation | International Standard. One purpose is to ensure against unlimited looping of protocol data units; while the routing algorithm should ensure that it will be very rare for data to loop, the PDU lifetime field provides additional assurance that loops will be limited in extent.

The other purpose of the lifetime capability is to provide a means by which the originating Network entity can limit the maximum NSDU lifetime. The OSI Transport protocol class 4 (ITU-T Rec. X.224 | ISO/IEC 8073) assumes that there is a particular maximum NSDU lifetime in order to protect against certain error states in the transport connection establishment and termination phases; *viz.* if a TPDU does not arrive within the maximum NSDU lifetime, then there is no chance that it will ever arrive. It is necessary to make this assumption, even if the Network layer does not guarantee any particular upper bound on NSDU lifetime; however, it is simpler for Transport protocol class 4 to deal with lost TPDU than to deal with late TPDU, and for this reason, it is preferable to discard late TPDU than to deliver them. It should be noted that NSDU lifetime is not directly associated with the retransmission of lost TPDU; rather, it is most useful for distinguishing old (duplicate) TPDU from new TPDU.

Maximum NSDU lifetime must be provided to a Transport protocol entity in units of time in order to be useful in determining Transport timer values (a Transport entity cannot count “hops”).

In the absence of any guaranteed upper bound, it is common to estimate a value for maximum NSDU lifetime. This value is often based upon observation of past performance, and may vary with source and destination. There are two possible ways to deal with the requirement for a limit on the maximum NSDU lifetime:

- 1) provide a mechanism in the Transport layer to recognize and discard old TPDU; or
- 2) specify lifetime in units of time.

The second method requires intermediate systems to decrement the lifetime field by a value which is an upper bound on the time elapsed since the PDU visited the previous intermediate system. The Transport layer relies on the Network layer to discard NSDU (and hence TPDU) whose lifetime has expired.

A major disadvantage to employing solution 1 is that Transport entities (instances) are created when required and released when their purpose has been fulfilled; hence, they are by nature temporary. In order to determine whether a particular TPDU is old, functions that recognize and discard old TPDU must be designed (and must always be present) in addition to those performed by each Transport entity instance. Such functions are extremely complex and impose a non-trivial overhead on Transport layer operation.

Conversely, the state machine associated with the provision of a connectionless-mode service does not require knowledge of previous connection state information to operate correctly. As no additional mechanisms beyond those necessary to correctly bound NPDU lifetime are required to ensure that old NSDU (and hence old TPDU) are not delivered to the Transport layer, it is preferable to have the Network layer discard NPDU whose lifetime has expired, and have the Transport layer deal with lost TPDU (solution 2).

B.1.1 Determining a value for NPDU lifetime

It is not necessary for each intermediate system to subtract a precise measure of the time that elapsed since an NPDU (containing the TPDU or a segment thereof) visited the previous intermediate system. Where a precise measure is not available, it is sufficient to subtract an overestimate of the actual time taken. In most cases, an intermediate system may simply subtract a constant value which depends upon the typical near-maximum delays that are encountered in a specific underlying service. A more accurate measure may be required for those subnetworks that have both a relatively large maximum delay and a relatively large variation in delay.

As an example, assume that a particular local area network has short average delays, with overall delays generally in the 1 ms to 5 ms range and with occasional delays up to 20 ms. In this case, although the relative range in delays might be large (a factor of twenty), it would still not be necessary to measure the actual delay for NPDU. A constant value of 20 ms (or more) could be subtracted for all NPDUs. Similarly, if a single hop satellite link had delays ranging from 0.5 s to 0.6 s then the higher value could always be used.

If a third subnetwork had normal delays ranging from 0.1 s to 1 s, but occasionally delivered an NPDU after a delay of 15 s, then an intermediate system attached to this subnetwork might find it necessary to determine how long it has actually taken the NPDU to be delivered. Even in this last example, it is more useful to have the intermediate systems determine when the delays are extreme and discard very old NPDUs, and allow the Transport protocol to detect the lost TPDU.

In addition to the time delay within each subnetwork, it is important to consider the time delay within intermediate systems. It should be relatively simple for those intermediate systems that expect to hold on to some data units for significant periods of time to decrement the lifetime appropriately.

B.2 Reassembly lifetime control

In order to ensure a bound on the lifetime of NSDUs, and to effectively manage reassembly buffers in the Network layer, the reassembly function described in clause 6 must control the lifetime of segments representing partially assembled PDUs. This clause discusses methods of bounding reassembly lifetime and suggests some implementation guidelines for the reassembly function.

When segments of a PDU arrive at a destination Network entity, they are buffered until an entire PDU is received, assembled, and passed to the PDU decomposition function. The protocol does not guarantee the delivery of PDUs; hence, it is possible for some segments of a PDU to be lost or delayed such that the entire PDU cannot be assembled in a reasonable length of time. In the case of loss of a PDU segment, for example, this could be forever. There are a number of possible schemes to prevent this:

- a) per-PDU reassembly timers;
- b) extension of the PDU lifetime control function; and
- c) coupling of the reassembly lifetime and Transport retransmission timers.

Each of these methods is discussed in the following clauses.

B.2.1 Method (a)

This method assigns a “reassembly lifetime” to each PDU received and identified by its data unit identifier. This is a local, real time which is assigned by the reassembly function and decremented while some, but not all, segments of the PDU are being buffered by the destination Network entity. If the timer expires, all segments of the PDU are discarded, thus freeing the reassembly buffers and preventing a “very old” PDU from being confused with a newer one bearing the same data unit identifier. For this scheme to function properly, the timers must be assigned in such a fashion as to prevent the phenomenon of reassembly interference (discussed below). In particular, the following guidelines should be followed:

- a) The reassembly lifetime must be much less than the maximum PDU lifetime of the network (to prevent the confusion of old and new data units).
- b) The lifetime should be less than the Transport protocol’s retransmission timers minus the average transit time of the network. If this is not the case, extra buffers are tied up holding data which has already been retransmitted by the Transport protocol. (Note that an assumption has been made that such timers are integral to the Transport protocol, which in some sense dictates that retransmission functions must exist in the Transport protocol employed).

B.2.2 Method (b)

This method is feasible if the PDU lifetime control function operates based on real or virtual time rather than hop count. In this scheme, the lifetime field of each PDU segment continues to be decremented by the reassembly function of the destination Network entity as if the PDU were still in transit (in a sense, it still is). When the lifetime of any segment of a partially reassembled PDU expires, all segments of that PDU are discarded. This scheme is attractive since the delivery behaviour of this protocol would be identical for segmented and unsegmented PDUs.

B.2.3 Method (c)

This method couples the reassembly lifetime directly to the Transport protocol's retransmission timers, and requires that Transport layer management make known to Network layer management (and hence, to the reassembly function) the values of its retransmission timers for each source from which it expects to be receiving traffic. When a PDU segment is received from a source, the retransmission time minus the anticipated transit time becomes the reassembly lifetime of that PDU. If this timer expires before the entire PDU has been reassembled, all segments of the PDU are discarded. This scheme is attractive since it has a low probability of holding PDU segments that have already been retransmitted by the source Transport entity; it has, however, the disadvantage of depending on reliable operation of the Transport protocol to work effectively. If the retransmission timers are not set correctly, it is possible that all PDUs would be discarded too soon, and the Transport protocol would make no progress.

B.3 The power of the header error detection function

B.3.1 General

The form of the checksum used for PDU header error detection is such that it is easily calculated in software or firmware using only two additions per octet of header, yet it has an error detection power approaching (but not quite equalling) that of techniques which involve calculations that are much more time- or space-consuming (such as cyclic polynomial checks). This clause discusses the power of this error detection function.

The checksum consists of two octets, either of which can assume any value except zero. That is, 255 distinct values for each octet are possible. The calculation of the two octets is such that the value of either is independent of the value of the other, so the checksum has a total of $255 \times 255 = 65\,025$ values. If one considers all ways in which the PDU header might be corrupted as equally likely, then there is only one chance in 65 025 that the checksum will have the correct value for any particular corruption. This corresponds to 0.0015 % of all possible errors.

The remainder of this clause considers particular classes of errors that are likely to be encountered. The hope is that the error detection function will be found to be more powerful, or at least no less powerful, against these classes as compared to errors in general.

B.3.2 Bit alteration errors

First considered are classes of errors in which bits are altered, but no bits are inserted or deleted.

A burst error of length b is a corruption of the header in which all of the altered bits (no more than b in number) are within a single span of consecutively transmitted bits that is b bits long. Checksums are usually expected to do well against burst errors of a length not exceeding the number of bits (16) in the header error detection parameter. The PDU header error detection parameter in fact fails to detect only 0.000 019 % of all such errors, each distinct burst error of length 16 or less being considered to be equally likely. In particular, it cannot detect an 8-bit burst in which an octet of zero is altered to an octet of 255 (all bits = 1) or vice versa. Similarly, it fails to detect the swapping of two adjacent octets only if one is zero and the other is 255.

The PDU header error detection, as should be expected, detects all errors involving only a single altered bit.

Undetected errors involving only two altered bits should occur only if the two bits are widely separated (and even then only rarely). The PDU header error detection detects all double bit errors for which the spacing between the two altered bits is less than 2040 bits = 255 octets. Since this separation exceeds the maximum header length, all double bit errors are detected.

The power to detect double bit errors is an advantage of the checksum algorithm used for the protocol, versus a simple modulo 65 536 summation of the header split into 16 bit fields. The simple summation would not catch all such double bit errors. In fact, double bit errors with a spacing as little as 16 bits apart could go undetected.

This annex does not consider the case in which the checksum itself is erroneously set to be all zero; this case is discussed in B.3.4.

B.3.3 Bit insertion/deletion errors

Although errors involving the insertion or deletion of bits are in general neither more nor less likely to go undetected than are all other kinds of general errors, at least one class of such errors is of special concern. If octets, all equal to either zero or 255, are inserted at a point such that the simple sum C_0 in the running calculation (described in B.3) happens to equal zero, then the error will go undetected. This is of concern primarily because there are two points in the calculation for which this value for the sum is not a rare occurrence, but is expected; namely, at the beginning and the end. That is, if the header is preceded or followed by inserted octets all equal to zero or 255, then no error will be detected. Both cases are examined separately.

Insertion of erroneous octets at the beginning of the header completely misaligns the header fields, causing them to be misinterpreted. In particular, the first inserted octet is interpreted as the Network layer protocol identifier, probably eliminating any knowledge that the data unit is related to this protocol, and thereby eliminating any attempt to perform the checksum calculation or invoking a different form of checksum calculation.

Undetected insertion of erroneous octets at the end of the header, in the absence of other errors, is impossible because the length field unequivocally defines where the header ends. Insertion or deletion of octets at the end of the header requires an alteration in the value of the octet defining the header length. Such an alteration implies that the value of the calculated sum at the end of the header would not be expected to have the dangerous value of zero, and consequently that the error is just as likely to be detected as is any error in general.

Insertion of an erroneous octet in the middle of the header is primarily of concern if the inserted octet has either the value zero or 255, and if the variable C_0 happens to have the value zero at this point. In most cases, this error will completely destroy the parsing of the header, which will cause the data unit to be discarded. In addition, in the absence of any other error, the last octet of the header will be thought to be data. This in turn will cause the header to end in the wrong place. In the case in which the header otherwise parses correctly, the last field will be found to be missing. Even in the case in which the last field is the padding option, and therefore not necessary, the length field for the padding function will be inconsistent with the header length field, and therefore the error can be detected.

B.3.4 Checksum non-calculation errors

Use of the header error detection function is optional. The choice of not using it is indicated by a checksum parameter value of zero. This creates the possibility that the two octets of the checksum parameter (neither of which is generated as being zero) could both be altered to zero. This would in effect be an error not detected by the checksum, since the check would not be made. One of three possibilities exists:

- a) A burst error of length sixteen (16) which sets the entire checksum to zero. Such an error could not be detected; however, it requires a particular positioning of the burst within the header. (A calculation of its effect on overall detectability of burst errors depends upon the length of the header.)
- b) All single bit errors are detected. Since both octets of the checksum field must be non-zero when the checksum is being used, no single bit error can set the checksum to zero.
- c) Where each of the two octets of the checksum parameter has a value that is a power of two, such that only one bit in each equals one (1), then a zeroing of the checksum parameter could result in an undetected double bit error. Furthermore, the two altered bits have a separation of less than sixteen (16), and could be consecutive. This is clearly a decline from the complete detectability previously described.

Where there is particular concern about the possibility of accidental zeroing of the checksum among data units within a network addressing domain, then a restriction may be imposed that all data units whose source or destination lie within the network addressing domain must make use of the header error detection function. Any data units which do not could be discarded, or could be prevented from leaving the local network addressing domain. This protects against errors that occur within the network addressing domain, and would protect all data units whose source or destination lies within the network addressing domain, even where the data path between all such pairs crosses other network addressing domains (errors outside the protected network addressing domain notwithstanding).

Annex C

Algorithms for PDU header error detection function

(This annex does not form an integral part of this Recommendation | International Standard)

C.1 Symbols used in algorithms

- C_0, C_1 are variables used in the algorithm;
- i is the number (i.e. the position) of an octet within the header (the position of the first octet is $i = 1$);
- O_i is the value of octet i of the PDU header;
- n is the number (i.e. the position) of the first octet of the checksum parameter ($n = 8$);
- L is the length of the PDU header in octets;
- X is the value of octet one of the checksum parameter;
- Y is the value of octet two of the checksum parameter.

C.2 Arithmetic conventions

Addition is performed in one of the two following modes:

- a) modulo 255 arithmetic;
- b) eight-bit one's complement arithmetic, in which, if any of the variables has the value minus zero (i.e. 255), it shall be regarded as though it had the value plus zero (i.e. 0).

C.3 Algorithm for generating checksum parameters

Construct the complete PDU header with the value of the checksum parameter field set to zero;

A: $C_0 \leftarrow C_1 \leftarrow 0$

B: Process each octet of the PDU header sequentially from $i = 1$ to L by

$$\begin{aligned} C_0 &\leftarrow C_0 + O_i \\ C_1 &\leftarrow C_1 + C_0 \end{aligned}$$

C: Calculate:

$$\begin{aligned} X &\leftarrow (L - 8)C_0 - C_1 \pmod{255} \\ Y &\leftarrow (L - 7)(-C_0) + C_1 \pmod{255} \end{aligned}$$

D: If $X = 0$, then $X \leftarrow 255$;

E: If $Y = 0$, then $Y \leftarrow 255$;

F: Place the values of X and Y in octets 8 and 9 respectively.

C.4 Algorithm for checking checksum parameters

A: If octets 8 and 9 of the PDU header both contain 0, then the checksum calculation has succeeded; else if either but not both of these octets contains the value zero, then the checksum is incorrect; otherwise, initialize

$$C_0 \leftarrow C_1 \leftarrow 0$$

B: Process each octet of the PDU header sequentially from $i =$ to L by

$$\begin{aligned} C_0 &\leftarrow C_0 + O_i \\ C_1 &\leftarrow C_1 + C_0 \end{aligned}$$

C: If, when all of the octets have been processed, $C_0 = C_1 = 0$, then the checksum calculation has succeeded; otherwise, the checksum calculation has failed.

C.5 Algorithm to adjust the checksum parameter when an octet is altered

This algorithm adjusts the checksum when an octet (such as the lifetime field) is altered. Suppose the value in octet k is changed by $Z = \text{newvalue} - \text{oldvalue}$.

If X and Y denote the checksum values held in octets n and $n + 1$, respectively, then adjust X and Y as follows:

A: If $X = 0$ and $Y = 0$ then do nothing; else if $X = 0$ or $Y = 0$ then the checksum is incorrect; else:

$$\begin{aligned} X &\leftarrow (k - n - 1)Z + X \pmod{255} \\ Y &\leftarrow (n - k)Z + Y \pmod{255} \end{aligned}$$

B: If $X = 0$, then $X \leftarrow 255$;

C: If $Y = 0$, then $Y \leftarrow 255$.

For this protocol, $n = 8$. If the octet being altered is the lifetime field, $k = 4$. For the case in which the lifetime is decreased by one unit ($z = -1$), the assignment statements for the new values of X and Y in the immediately preceding algorithm simplify to:

$$\begin{aligned} X &\leftarrow X + 5 \pmod{255} \\ Y &\leftarrow Y - 4 \pmod{255} \end{aligned}$$

NOTE – To derive this result, assume that when octet k has the value Z added to it, then X and Y have values Z_x and Z_y added to them. For the checksum parameters to satisfy the conditions of 6.11 both before and after the values are added, the following is required:

$$\begin{aligned} Z + Z_x + Z_y &= 0 \pmod{255} \\ \text{and} \\ (L - k + 1)Z + (L - n + 1)Z_x + (L - n)Z_y &= 0 \pmod{255} \end{aligned}$$

Solving these equations simultaneously yields:

$$\begin{aligned} Z_x &= (k - n - 1)Z \\ \text{and} \\ Z_y &= (n - k)Z \end{aligned}$$

Network Working Group
Request for Comments: 986

Ross Callon (BBN)
Hans-Werner Braun (UMich)
June 1986

WORKING DRAFT

Guidelines for the use of Internet-IP addresses in the
ISO Connectionless-Mode Network Protocol

Status of This Memo

This RFC suggests a method to allow the existing IP addressing, including the IP protocol field, to be used for the ISO Connectionless Network Protocol (CLNP). This is a draft solution to one of the problems inherent in the use of "ISO-grams" in the DOD Internet. Related issues will be discussed in subsequent RFCs. This RFC suggests a proposed protocol for the ARPA-Internet community, and requests discussion and suggestions for improvements. Distribution of this memo is unlimited.

1. Introduction

The CLNP is documented in [1], but for matters of completeness the following illustration of the CLNP header is included here as Figure 1.

The addressing part of the header is the subject of this RFC, i.e., the source and the destination address, respectively. These addresses are generally discussed in [2] and [3], with this document presenting a specific method for addressing in the DOD Internetwork environment, consistent with the international standardized NSAP addresses.

RFC 986
DOD IP Addresses in ISO CLNP

June 1986

	Octet	
Network Layer Protocol Identifier	1	:
Length Indicator	2	:
Version/Protocol Id Extension	3	: Fixed
Lifetime	4	: Part
SP MS E/R Type	5	:
Segment Length	6,7	:
Checksum	8,9	:
Destination Address Length Indicator	10	:
Destination Address	11 through m-1	: Address
Source Address Length Indicator	m	: Part
Source Address	m+1 through n-1	:
Data Unit Identifier	n,n+1	: Segment
Segment Offset	n+2,n+3	: ation
Total Length	n+4,n+5	: Part
Options	n+6 through p	: Options Part
Data	p+1 through z	: Data

Figure 1: PDU Header Format

2. Addresses for Use in the Internet

This section describes the primary addresses used to address NSAPs in the Internet. A later section will describe a separate address format for end systems and individual simple LANs that are attached to the Internet only through intervening Public Data Networks.

The appropriate Authority and Format Identifier (AFI) is one octet in length.

"The AFI consists of an integer with a value between 0 and 99 with an abstract syntax of two decimal digits" [3], that is, the AFI codes are binary coded decimal (BCD).

It specifies an ISO-6523-ICD assignment, and also that the Domain Specific Part (DSP) of the address is based on binary. The AFI octet uses the value "47". The ISO-6523-ICD format is used to emphasize that this is an administrative assignment. The usage of an ISO DCC (Data Country Code) would be possible, but could be misleading due to the fairly far spread geographical extent of the Internet-IP.

As required by the ISO addressing standard, the next two octets of the address, in this case, specify the Initial Domain Identifier. This two octet value is the International Code Designator (ICD) assigned to the DOD Internet, "0006".

The remainder of the NSAP address is the Domain Specific Part (DSP). This is assigned by the Internet administration, which is considered to be an addressing domain. The remainder of the address specifies a one byte version number, the four byte Internet Protocol address and a one byte IP user protocol field. The version number allows for future extensions. The IP address used is the same as the current four octet IP address. The user protocol field is the same as the user protocol field in the current IP header. This is necessary because the ISO protocol considers identification of the user protocol to be an addressing issue, and therefore does not allow for the user protocol to be specified in the protocol header independently from the address.

RFC 986
DOD IP Addresses in ISO CLNP

June 1986

Therefore a source or destination address within the ISO Connectionless Protocol, when used in the DOD Internet, looks as follows:

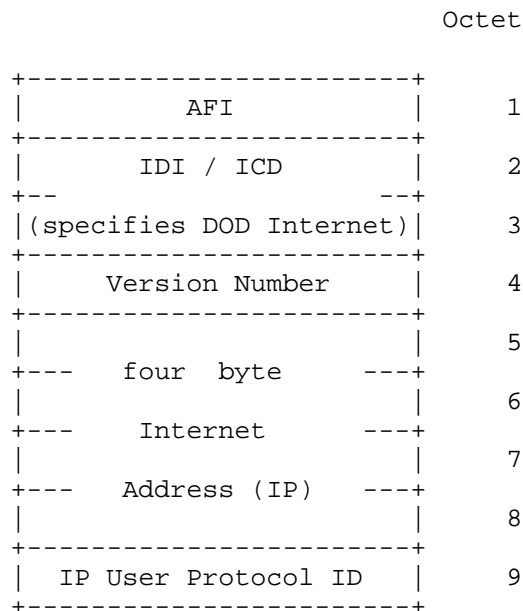


Figure 2: ISO IP address structure

The Authority and Format Identifier (AFI) is "47" (BCD). The Initial Domain Identifier (IDI) consists of the International Code Designator (ICD) assigned to the Internet, and must contain the value "0006". The Version Number must contain the value "01". The Current IP addresses and IP user protocol numbers can be found in [4].

3. Devices Attached to PDNs

Otherwise isolated end systems, which are attached to the Internet only indirectly via public data networks, and simple LANs which are similarly attached only via Public Data Networks, will make use of a separate address format based on their X.121 address.

Figure 3 specifies the address for use by end systems attached to PDNs. Here the AFI specifies an ISO-X.121 address format, with the DSP based on binary. The AFI occupies a single octet, and must take the value "37" (hexadecimal). The IDI contains the X.121 addresses

RFC 986
DOD IP Addresses in ISO CLNP

June 1986

encoded in binary (using BCD), padded at the end if necessary with all ones (binary "1111") to make up 7 full octets. Finally, the DSP contains a single octet, which specifies the user protocol.

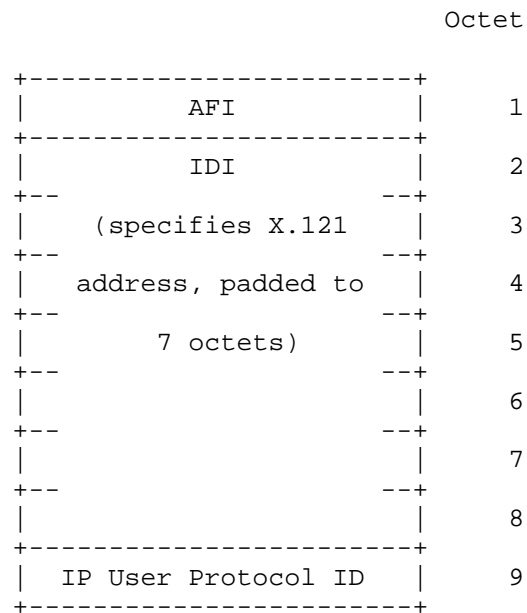


Figure 3: ISO IP address structure for isolated end systems

Figure 4 specifies the address for use by simple LANs attached to PDNs. This address is the same as the address used for end systems directly attached to PDNs, except for the addition of the (variable length) local address as used on the LAN. Whether the address is of the form shown in figure 3, or of the form shown in figure 4, is determined by looking at the length of the address.

RFC 986
 DOD IP Addresses in ISO CLNP

June 1986

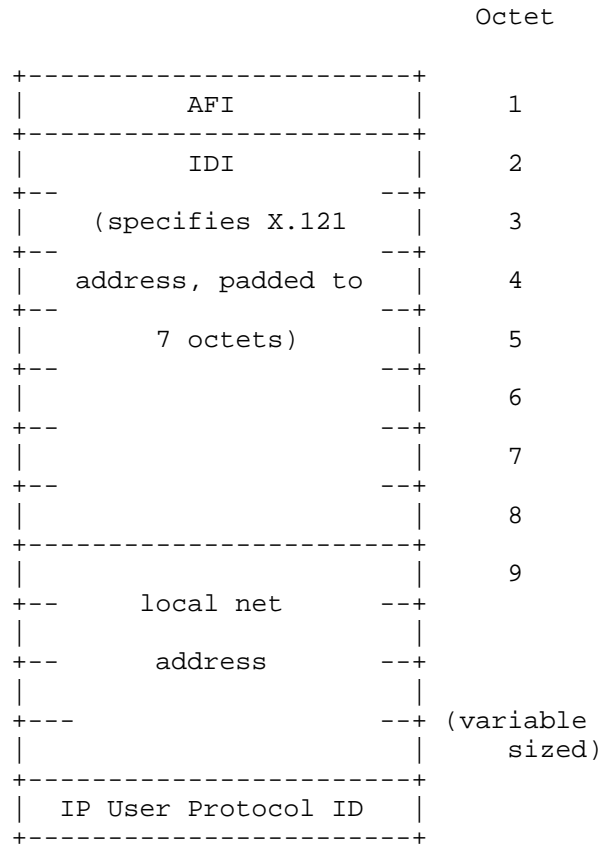


Figure 4: ISO IP address structure for isolated LANs

RFC 986
DOD IP Addresses in ISO CLNP

June 1986

References

- [1] ISO, "Protocol for Providing the Connectionless-Mode Network Services", RFC-926, ISO, December 1984.
- [2] ANSI, "Guidelines for the Specification of the Structure of the Domain Specific Part (DSP) of the ISO Standard NSAP Address", RFC-982, ANSI Working Document X3S3.3/85-258, April 1986.
- [3] ISO, Draft International Standard 8348/DAD2, "Information Processing Systems -- Data Communications -- Network Service Definition, Addendum 2 Covering Network Layer Addressing", RFC-941, April 1985.
- [4] Reynolds, J. and J. Postel, "Assigned Numbers", RFC-960, USC Information Sciences Institute, December 1985.

Network Working Group
Request for Comments: 1365

K. Siyan
Siyan Consulting Services
September 1992

An IP Address Extension Proposal

Status of This Memo

This memo provides information for the Internet community. It does not specify an Internet standard. Distribution of this memo is unlimited.

Abstract

This RFC suggests an extension to the IP protocol to solve the shortage of IP address problem, and requests discussion and suggestions for improvements.

1. Introduction and Background

The Internet community has a well-developed, mature set of protocols that have been quite successful in providing network and transport services to users. However, because of the spectacular success of the TCP/IP protocols and the number of networks that desire connection to the Internet, there is a shortage of network numbers that can be assigned.

The current network addressing scheme uses a 32-bit IP address that has a network part and a local address part. The division between the network part and the local address part has been defined in terms of 5 address classes: class A, B, C, D, E. Of these, only class A, B, C addresses are assigned to hosts. Class D is used for multicasting and class E is reserved.

Class A has the highest order bit set to 0, a 7 bit network number and a 24 bit host address.

Class B has the two higher order bits set to 10, a 14 bit network number and a 16 bit host address.

Class C has the three higher order bit set to 110, a 21 bit network number and a 8 bit host address.

Class D has the four higher order bits set to 1110.

Class E has four higher address bits set to 1111.

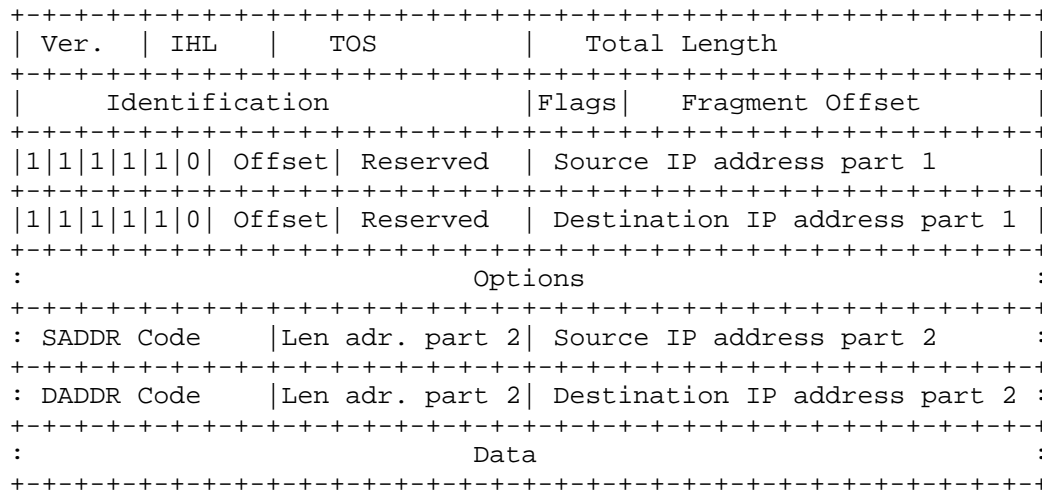
Increasing the size of the IP address field to more than 32 bits would solve the problem, but at the expense of making a new IP header definition that would be incompatible with the existing base of IP implementations. OSI based solutions such as using CLNP have been proposed but would take time to implement.

2. Proposal for IP extension

The IP header format should not be modified to minimize the changes necessary for supporting the address extensions that are proposed in this RFC. Instead an "escape" mechanism can be used to specify larger address. The IP header length field is 4 bits and this allows a maximum of fifteen 32-bit words where each word is 4 octets. The minimum size of the IP header without options is 5 words, which leaves 10 words for options. One can reserve 6 words (24 octets) for the normal options and leave the remaining (4 words or 16 octets) for a new option type that specifies an extended address. The details of this mechanism are discussed below.

Class E should be defined with the its five high order bits set to 11110. Its current definition is that four 1's in the most significant bits represent a class E address.

A new class F is proposed with its six high order bits set to 111110. The new class F address would be placed in the same locations that are used for source and destination IP address in the IP header, but would specify that part of the addressing information is in the options part of the IP header. This is illustrated in the figure below:



The "Offset" field specifies the offset in words from the beginning of the IP header where the second part of the IP address is located. Its purpose is to avoid searching the options part for addressing information. The address in the options part is in the Type-Length-Value form for consistency with other IP options that are found in this part. The "Len adr. part" indicates the length of the second IP address part in octets. The lengths should be defined so that the second part of the IP address ends on a word boundary. For example, the possible length values are 4, 8 octets. It is proposed that new IP option codes be used for the SADDR and DADDR codes respectively.

The IP address is the 2 bytes in the fixed IP header part plus the address field defined in the options part.

If the "Len adr. part" field has a value of 4, the new class is designated as the F-4 class (Class F with IP address length of 4 octets).

If the "Len adr. part" field has a value of 8, the new class is designated as the F-8 class (Class F with IP address length of 8 octets).

Each of the F-4 and F-8 IP address class can be further subdivided into a network number and a host number field in a manner that is similar to the current IP addressing scheme.

The sub-class definitions for F-4 class are shown below. Though the 4 octets are drawn contiguously, the first 2 octets and the last 2 octets are not contiguous in the IP header.

Class F-4A has the highest order bit set to 0, a 7 bit network number and a 24 bit host address.

```

+-----+
|0| net number | local part |
+-----+

```

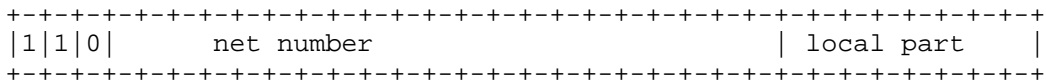
Class F-4B has the two higher order bits set to 10, a 14 bit network number and a 16 bit host address.

```

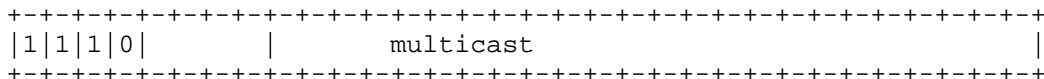
+-----+
|1|0| net number | local part |
+-----+

```

Class F-4C has the three higher order bit set to 110, a 21 bit network number and a 8 bit host address.

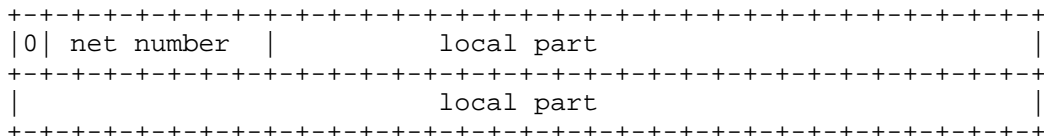


Class F-4D has the four higher order bits set to 1110. Class F-4D is reserved for multicasting.

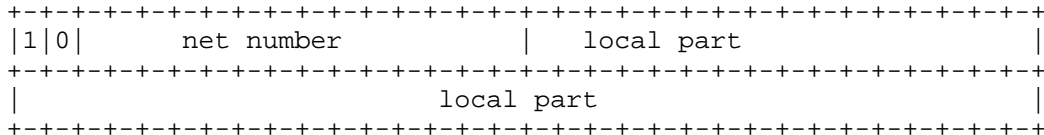


The sub-class definitions for F-8 class are shown below. Though the 8 octets are drawn contiguously, the first 2 octets and the last 6 octets are not contiguous in the IP header.

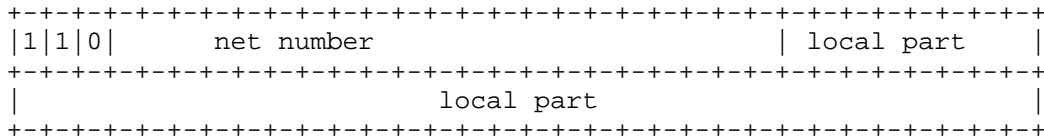
Class F-8A has the highest order bit set to 0, a 7 bit network number and a 56 bit host address.



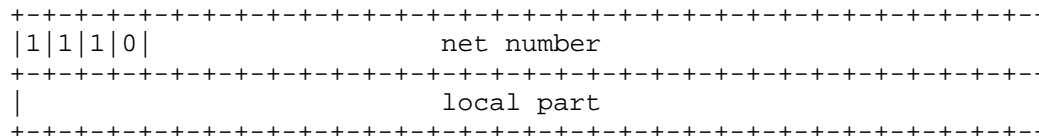
Class F-8B has the two higher order bits set to 10, a 14 bit network number and a 48 bit host address.



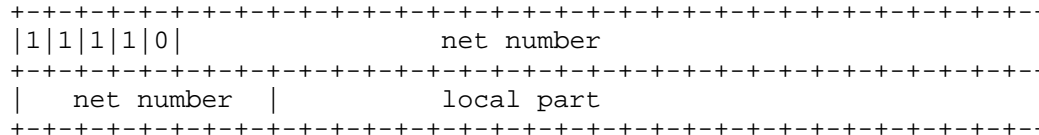
Class F-8C has the three higher order bit set to 110, a 21 bit network number and a 40 bit host address.



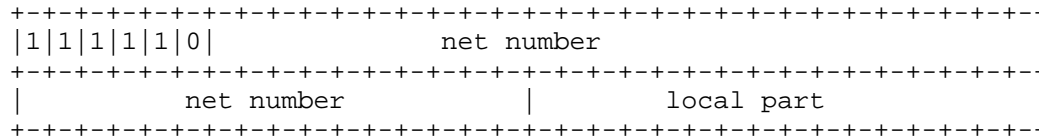
Class F-8D has the four higher order bits set to 1110, a 28 bit network number and a 32 bit host address.



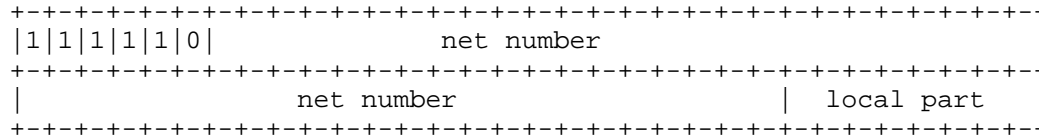
Class F-8E has the five higher order bits set to 11110, a 35 bit network number and a 24 bit host address.



Class F-8F has the six higher order bits set to 111110, a 44 bit network number and a 16 bit host address.



Class F-8G has the seven higher order bits set to 1111110, a 49 bit network number and a 8 bit host address.



3. Interoperability Issues

If the new class F address is seen by a host that does not support it the IP datagram will be ignored. So communication will not be possible with existing hosts, but the amount of modification for existing hosts is much less than implementing an entirely different IP header structure or a different protocol.

The receiving host must be modified to contain the following code sketched below:


```
if (Destination_IP_address & 0xFC000000 == 0xF8000000)
{
    /* New extended class F address */
    Class_F_Processing(Destination_IP_address);
}
```

The Class_F_Processing() procedure can be defined in a separate module. There will be other changes required to communicate the results of processing the class F address to the main IP processing module but they should not be so extensive.

Security Considerations

Security issues are not discussed in this memo.

Author's Address

Karanjit Siyan
Siyan Consulting Services
49 Taurus Road, Box 960
North Glastonbury
Emigrant, Montana 59027

Phone: 406-333-4491

E-Mail: 72550.1634@compuserve.com

Network Working Group
Request for Comments: 1375

P. Robinson
Tansin A. Darcos & Co.
October 1992

Suggestion for New Classes of IP Addresses

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard. Distribution of this memo is unlimited.

Abstract

This RFC suggests a change in the method of specifying the IP address to add new classes of networks to be called F, G, H, and K, to reduce the amount of wasted address space, and to increase the available IP address number space, especially for smaller organizations or classes of connectors that do not need or do not want a full Class C IP address.

Table of Contents

Introduction	1
Suggestion for new IP address classes	2
Current Class C Address	3
Proposed new Class C Address	4
Proposed "Class F" address	4
Proposed "Class G" address	4
Proposed "Class H" address	5
Proposed "Class K" address	5
Optional selection of routing codes by region	5
Summary	6
Notes	6
Security Considerations	7
Author's Address	7

Introduction

Currently, IP addresses on the Internet are 32-bit quantities which are generally represented as four decimal numbers from 0 to 255, separated by periods, sometimes called a "dotted" decimal number. The current numbering scheme provides in general for three classes of networks in general use (A,B, and C), and two other classes of networks (D, E).

The Class A networks assign a large address space for the particular

network to allow up to 254^3 local machines [1]. The Class B network assigns a somewhat smaller address space for the particular network to allow up to 254^2 local machines. The Class C network assigns a still smaller address space for the particular network to allow up to 254 local machines.

This memo proposes to assign part of the unused Class C address space for smaller networks than are currently available. The term "Class D" is used for the "multicast" capability and addresses in "Class E" are reserved for future use. Therefore, these new features for which capability is to be added is being referred to as classes F, G, H and K.

Suggestion for new IP address classes

The most worrisome problem which appears in the literature is the possibility of running out of address space for IP addresses. Various schemes are being suggested such as subrouting, introduction of additional bits, and other possibilities.

There is an even more serious matter. In all probability, I suspect that eventually the Internet backbone will either become available to anyone who wants to use it (like public highways) and the costs paid for out of taxes or some other method which gets someone else to pay for it, or eventually the Internet will be fully commercialized and made available to anyone who wants to buy a permanent connection. With the cost of hardware and connections dropping, some Computer Bulletin Board Systems (BBSs) which are currently accessible via telephone call may become accessible via TELNET or FTP. When a 9600 baud connection can be obtained for around the price of a phone line, the demand for internet access will skyrocket. This almost certain eventual availability to virtually anyone who wants a connection will cause an even greater demand for internet addresses, which will exacerbate this situation. One problem is in the granularity of IP addressing, in that the smallest possible IP address one may obtain allows for as high as 254 IP addresses. If someone wanted only to put four or five computers on the Internet, more than 240 addresses are wasted.

Many smaller installations would probably be interested either in placing their computers and/or servers on the Internet (and perhaps helping to pay the cost of running it) or in being able to access the Internet directly, and perhaps making facilities on their machines available to others; the problem being that IP addresses on Internet are not readily available to small classes of users. Also, the possibility exists of eventually placing non-computer and output-only devices such as printers, facsimile machines, and visual pagers directly on the Internet to allow people to send a message to a local

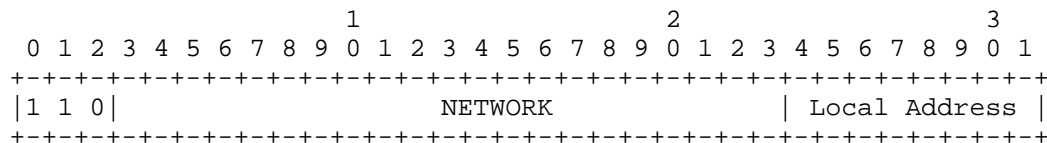
device simply by directing it to a specific internet site as an E-Mail message.

The scheme proposed by this paper proposes to make a slight change in one of the classes of network address in a manner which should not be a significant problem for implementing, and should not cause a significant hardship as the addresses to use for this purpose are not now allocated anyway, and may draw some of the drain which would have consumed Class C addresses in large quantity into quantities of Class F, H, or K addresses which waste less IP address space.

This scheme I am proposing is to allow for very small networks (1 or 2, 1-7, or 1-15, depending on the number of addresses the administrator of that site thinks he will need), by reconstructing the network address to include what is nominally part of the local address. If bridges and routers (and other hardware and software) do not assume that only the last 8 bits make up a local address and permit smaller spaces for local addresses, then this method should not cause problems. Sites needing less than a close order of 256 IP addresses could simply apply for 2 or more contiguous blocks of Class F numbers.

Currently, a Class C address consists of a 32-bit number in which the leftmost 3 bits consist of "110" [2]:

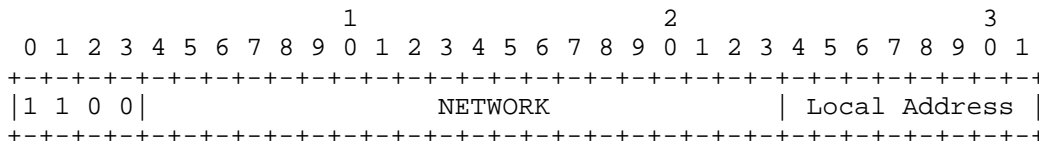
The third type of address, class C, has a 21-bit network number and a 8-bit local address. The three highest-order bits are set to 1-1-0. This allows 2,097,152 class C networks.



Current Class C Address

This memo proposes to change Class C addresses to be 4-bit numbers beginning with "1100":

The third type of address, class C, has a 20-bit network number and a 8-bit local address. The four highest-order bits are set to 1-1-0-0, This allows 1,048,576 class C networks.

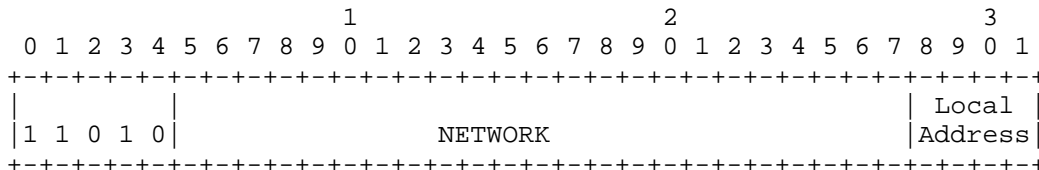


Proposed new Class C Address

This memo then proposes to add four new types of addresses, to be referred to as "Class F", "Class G", "Class H", and "Class K" [3]. These would all use part of the "old" class C address by all using IP addresses that begin with the 4-bit sequence "1101". The Class F addresses would begin with the binary code sequence "11010", Class G addresses begin with "110110", Class H addresses with "1101110", and Class K with "1101111".

Class F addresses will be used for networks having from 1-15 sites [4], where the number could be expected to exceed 7. Class F addresses are defined as follows:

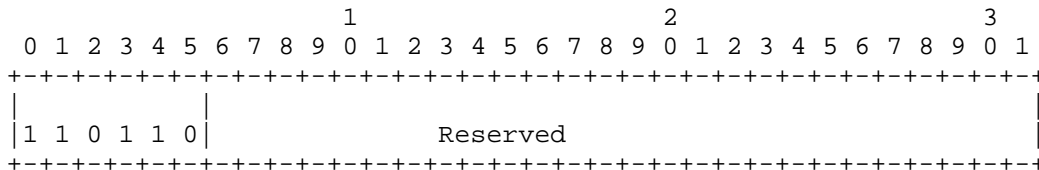
The sixth type of address, class F, has a 23-bit network number, and a 4-bit local address. The five highest-order bits are set to 1-1-0-1-0. This allows 16,777,256 class F networks.



Proposed "Class F" address

Class G is to be defined as follows:

The seventh type of address, class G, is reserved for future use. The six highest-order bits are set to 1-1-0-1-1-0.

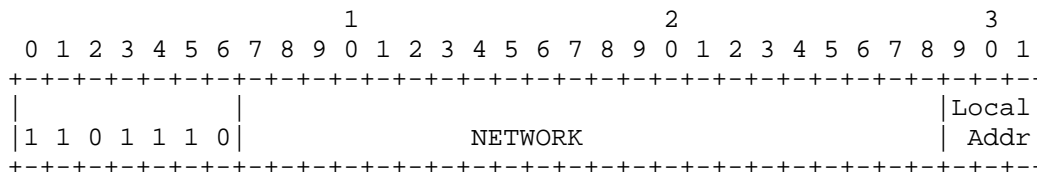


Proposed "Class G" address

Class H is for small networks which are not expected to exceed 7

connected IP addresses. Class H is to be defined as follows:

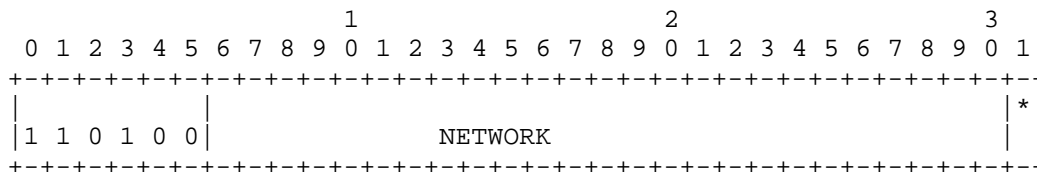
The eighth type of address, class H, has a 22-bit network number, and a 3-bit local address. The seven highest-order bits are set to 1-1-0-1-1-1-0. This allows 4,194,304 Class H addresses [5].



Proposed "Class H" address

Class K is for sites which either will only have one or two connected addresses [6]. Class K is to be defined as follows:

The eighth type of address, class K, has a 25-bit network number, and a 1-bit local address. The seven highest-order bits are set to 1-1-0-1-1-1-1. This allows 33,554,432 Class K addresses [7].



* = Local Address, 1 or 0
Proposed "Class K" address

Optional selection of routing codes by region

Because of the possibility of confusion, some method similar to the international dialing plan might be set up, in which bits 5-8 in Class F, bits 7-10 in Class H, and bits 6-9 in Class K could be used to define what part of the world the particular address is in, in a manner similar to the international telephone dialing system, which uses the first digit of the international telephone number to determine the region being used. The current method for assigning international dialing codes is:

1 North America	6 Oceania, Australia
2 Africa	7 Ex-Soviet Union Countries
3 Europe	8 Asia
4 Europe	9 Mideast
5 South America and Mexico	

If a similar method is used, I would recommend assigning 0,1,10 and 11 to North America, 8 and 12 to Asia, and leaving 13 through 15 for other areas as needed. Note that this would simply make some routing choices easier, it is not precisely necessary that this be done, since currently routing is generally done using the shortest path to a site and IP numbers don't really relate to any specific address anywhere in the world.

The number form of a class F, G, H or K address could still be listed in the standard form n.n.n.n, as long as it is not assumed that the 4th chunk number alone identifies a local address and that numbers with the same preceding 3 chunks do not necessarily belong to the same network.

Summary

In order to make the address space available, even if the method to implement this feature is not presently available, it is suggested that Class F, G, H, and K address space should be taken out of Class C space and reserved for the purpose of allowing smaller-sized networks so that this feature may be made available. Since Class C addresses currently are only using the equivalent of one Class A number anyway, this should not cause a problem.

Notes

- [1] Common practice dictates that neither an address 0 nor 255 should be used in any "dotted" address.
- [2] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1340, USC/Information Sciences Institute, July 1992.
- [3] To prevent confusion, no class "I" or "J" address was created by this memo.
- [4] It is expected that if the higher part of the network address occupying the 4-bits to the left of the Class F address are neither 0 nor 15, that a class F site could have 16 machines. If the 4-bits to the left are all 0, the Class F site must not use number 0. If the four bits are all 1, the site must not use number 15.

- [5] It may be that Class H numbers are more appropriate for classes of addresses that will not have as high a demand for access via Internet addresses such as facsimile machines and pagers. (The end digit could be used to determine class of service, i.e., 0 for tone only, 1 and 2 for numeric only, 3 4 and 5 for alphanumeric, and 6 and 7 for facsimile machines. Or some combination of these according to the demand. Remember, Internet won't always be just text messages and file transfers; we may eventually see things like voice telephone calls or voice data being placed to an Internet address just like calls made via the telephone system. This would require a whole change in the way things are done, but it's always best to look at the future.
- [6] It is suggested that addresses in this range not be assigned where the 7 bits to the left of the local number are all the same (all 0 or all 1), to allow all Class K addresses to have two local addresses.
- [7] Different things can be done with different capabilities. One thought was to set up some group of numbers and use them to indicate systems which are "gateway" systems, i.e., the top set of numbers in Class K could indicate that subnets are required after those numbers, similar to the use of an extension number on the switchboard of a large organization. Another possibility is to assign some of the numbers to specific classes of devices, such as number-only pagers and electronic display devices.

Security Considerations

Security issues are not discussed in this memo.

Author's Address

Paul Robinson
Tansin A. Darcos & Company
8604 Second Avenue #104
Silver Spring, MD 20910 USA

Phone: 202-310-1011
Telex: 6505066432MCI UW
E-mail: TDARCOS@MCIMAIL.COM

Network Working Group
Request for Comments: 1385

Z. Wang
University College London
November 1992

EIP: The Extended Internet Protocol
A Framework for Maintaining Backward Compatibility

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard. Distribution of this memo is unlimited.

Summary

The Extended Internet Protocol (EIP) provides a framework for solving the problem of address space exhaustion with a new addressing and routing scheme, yet maintaining maximum backward compatibility with current IP. EIP can substantially reduce the amount of modifications needed to the current Internet systems and greatly ease the difficulties of transition. This is an "idea" paper and discussion is strongly encouraged on Big-Internet@munnari.oz.au.

Introduction

The Internet faces two serious scaling problems: address exhaustion and routing explosion [1-2]. The Internet will run out of Class B numbers soon and the 32-bit IP address space will be exhausted altogether in a few years time. The total number of IP networks will also grow to a point where routing algorithms will not be able to perform routing based a flat network number.

A number of short-term solutions have been proposed recently which attempt to make more efficient use of the the remaining address space and to ease the immediate difficulties [3-5]. However, it is important that a long term solution be developed and deployed before the 32-bit address space runs out.

An obvious approach to this problem is to replace the current IP with a new internet protocol that has no backward compatibility with the current IP. A number of proposals have been put forward: Pip[7], Nimrod [8], TUBA [6] and SIP [14]. However, as IP is really the cornerstone of the current Internet, replacing it with a new "IP" requires fundamental changes to many aspects of the Internet system (e.g., routing, routers, hosts, ARP, RARP, ICMP, TCP, UDP, DNS, FTP).

Migrating to a new "IP" in effect creates a new "Internet". The

development and deployment of such a new "Internet" would take a very large amount of time and effort. In particular, in order to maintain interoperability between the old and new systems during the transition period, almost all upgraded systems have to run both the new and old versions in parallel and also have to determine which version to use depending on whether the other side is upgraded or not.

Let us now have a look at the detailed changes that will be required to replace the current IP with a completely new "IP" and to maintain the interoperability between the new and the old systems.

- 1) Border Routers: Border routers have to be upgraded and to provide address translation service for IP packets across the boundaries. Note that the translation has to be done on the outgoing IP packets as well as the in-coming packets to IP hosts.
- 2) Subnet Routers: Subnet Routers have to be upgraded and have to deal with both the new and the old packet formats.
- 3) Hosts: Hosts have to be upgraded and run both the new and the old protocols in parallel. Upgraded hosts also have to determine whether the other side is upgraded or not in order to choose the correct protocol to use.
- 4) DNS: The DNS has to be modified to provide mapping for domain names and new addresses.
- 5) ARP/RARP: ARP/RARP have to be modified, and upgraded hosts and routers have to deal with both the old and new ARP/RARP packets.
- 6) ICMP: ICMP has to be modified, and the upgraded routers have to be able to generate both both old and new ICMP packets. However, it may be impossible for a backbone router to determine whether the packet comes from an upgraded host or an IP host but translated by the border router.
- 7) TCP/UDP Checksum: The pseudo headers may have to be modified to use the new addresses.
- 8) FTP: The DATA PORT (PORT) command has to be changed to pass new addresses.

In this paper, we argue that an evolutionary approach can extend the addressing space yet maintain backward compatibility. The Extended Internet Protocol (EIP) we present here can be used as a framework by which a new routing and addressing scheme may solve the problem of address exhaustion yet maintain maximum backward compatibility to

current IP.

EIP has a number of very desirable features:

- 1) EIP allows the Internet to have virtually unlimited number of network numbers and over 10^{**9} hosts in each network.
- 2) EIP is flexible to accommodate most routing and addressing schemes, such as those proposed in Nimrod [8], Pip [7], NSAP [9] and CityCodes [10]. EIP also allows new fields such as Handling Directive [7] or CI [11] to be added.
- 3) EIP can substantially reduce the amount of modifications to current systems and greatly ease the difficulties in transition. In particular, it does not require the upgraded hosts and subnet routers to run two set of protocols in parallel.
- 4) EIP requires no changes to all assigned IP addresses and subnet structures in local are networks. and requires no modifications to ARP/RARP, ICMP, TCP/UDP checksum.
- 5) EIP can greatly ease the difficulties of transition. During the transition period, no upgrade is required to the subnet routers. EIP hosts maintain full compatibility with IP hosts within the same network, even after the transition period. During the transition period, IP hosts can communicate with any hosts in other networks via a simple translation service.

In the rest of the paper, IP refers to the current Internet Protocol and EIP refers to the Extended Internet Protocol (EIP is pronounced as "ape" - a step forward in the evolution :-).

Extended Internet Protocol (EIP)

The EIP header format is shown in Figure 1 and the contents of the header follows.

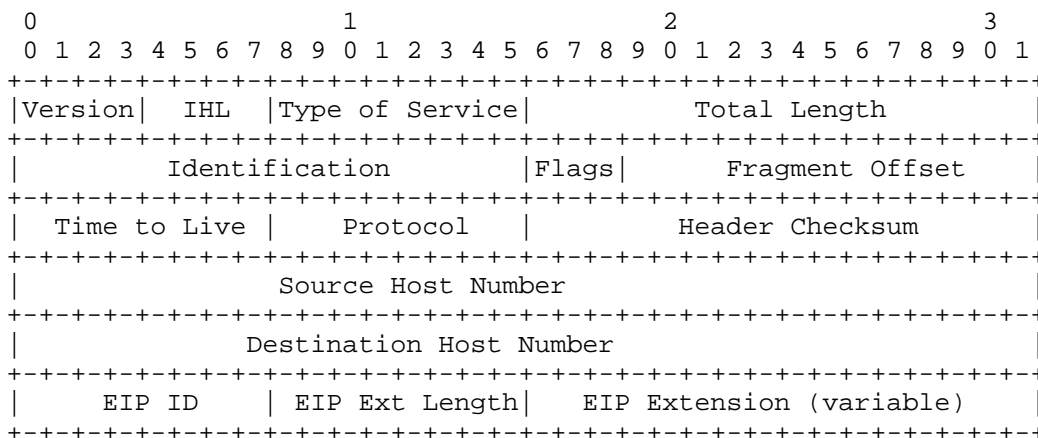


Figure 1: EIP Header Format

Version: 4 bits

The Version field is identical to that of IP. This field is set purely for compatibility with IP hosts. It is not checked by EIP hosts.

IHL: 4 bits

Internet Header Length is identical to that of IP. IHL is set to the length of EIP header purely for compatibility with IP. This field is not checked by EIP hosts. see below the EIP Extension Length field for more details)

Type of Service: 8 bits

The ToS field is identical to that of IP.

Total Length: 16 bits

The Total Length field is identical to that of IP.

Identification: 16 bits

The Identification field is identical to that of IP.

Flags: 3 bits

The Flags field is identical to that of IP.

Fragment Offset: 13 bits

The Fragment Offset field is identical to that of IP.

Time to Live: 8 bits

The Time to Live field is identical to that of IP.

Protocol: 8 bits

The Protocol field is identical to that of IP.

Header Checksum: 16 bits

The Header Checksum field is identical to that of IP.

Source Host Number: 32 bits

The Source Host Number field is used for identifying the source host but is unique only within the source network (the equivalent of the host portion of the source IP address).

Destination Host Number: 32 bits

The Destination Host Number field is used for identifying the destination host but is unique only within the destination network (the equivalent of the host portion of the destination IP address).

EIP ID: 8 bits

The EIP ID field equals to 0x8A. The EIP ID value is chosen in such a way that, to IP hosts and IP routers, an EIP appears to be an IP packet with a new IP option of following parameters:

COPY	CLASS	NUMBER	LENGTH	DESCRIPTION
1	0	TBD	var	

Option: Type=TBD

EIP Extension Length: 8 bits

The EIP Extension Length field indicates the total length of the EIP ID field, EIP Extension Length field and the EIP Extension field in octets. The maximum length that the IHL field above can specify is 60 bytes, which is considered too short in future. EIP hosts actually use the EIP Extension Length field to calculate the total header length:

The total header length = EIP Extension Length + 20.

The maximum header length of an EIP packet is then 276 bytes.

EIP Extension: variable

The EIP Extension field holds the Source Network Number, Destination Network Number and other fields. The format of the Extension field is not specified here. In its simplest form, it can be used to hold two fixed size fields as the Source Network Number and Destination Network Number as the extension to the addressing space. Since the Extension field is variable in length, it can accommodate almost any routing and addressing schemes. For example, the Extension field can be used to hold "Routing Directive" etc specified in Pip [7] or "Endpoint IDs" suggested in Nimrod [8], or the "CityCode" [10]. It can also hold other fields such as the "Handling Directive" [7] or "Connectionless ID" [11].

EIP achieves maximum backward compatibility with IP by making the extended space appear to be an IP option to the IP hosts and routers.

When an IP host receives an EIP packets, the EIP Extension field is safely ignored as it appears to the IP hosts as an new, therefore an unknown, IP option. As a result, there is no need for translation for in-coming EIP packets destined to IP hosts and there is also no need for subnet routers to be upgraded during the transition period (see later section for more details).

EIP hosts or routers can, however, determine whether a packet is an IP packet or an EIP packet by examining the EIP ID field, whose position is fixed in the header.

The EIP Extension field holds the Source and Destination Network Numbers, which are only used for communications between different networks. For communications within the same network, the Network Numbers may be omitted. When the Extension field is omitted, there is little difference between an IP packet and an EIP packet. Therefore, EIP hosts can maintain completely compatibility with IP hosts within one network.

In EIP, the Network Numbers and Host Numbers are separate and the IP address field is used for the Host Number in EIP. There are a number of advantages:

- 1) It maintains full compatibility between IP hosts and EIP hosts for communications within one network. Note that the Network Number is not needed for communications within one network. A

host can omit the Extension field if it does not need any other information in the Extension field, when it communicates with another host within the same network.

- 2) It allows the IP subnet routers to route EIP packet by treating the Host Number as the IP address during the transition period, therefore the subnet routers are not required to be updated along the border routers.
- 3) It allows ARP/RARP to work with both EIP and IP hosts without any modifications.
- 4) It allows the translation at the border routers much easier. During the transition period when the IP addresses are still unique, the network portion of the IP addresses can be directly extracted and mapped to EIP Network Numbers.

Modifications to IP Systems

In this section, we outline the modifications to the IP systems that are needed for transition to EIP. Because of the similarity between the EIP and IP, the amount of modifications needed to current systems are substantially reduced.

- 1) Network Numbers: Each network has to be assigned a new EIP Network Number based on the addressing scheme used. The mapping between the IP network numbers and the EIP Network Numbers can be used for translation service (see below).
- 2) Host Numbers: There is no need for assigning EIP Host Numbers. All existing hosts can use their current IP addresses as their EIP Host Numbers. New machines may be allocated any number from the 32-bit Host Number space since the structure posed on IP addressing is no longer necessary. However, during the transition, allocation of EIP Host Numbers should still follow the IP addressing rule, so that the EIP Host Numbers are still globally unique and can still be interpreted as IP addresses. This will allow a more gradual transition to EIP (see below).
- 3) Translation Service: During the transition period when the EIP Host Numbers are still unique, an address translation service can be provided to IP hosts that need communicate with hosts in other networks cross the upgraded backbone networks. The translation service can be provided by the border routers. When a border router receives an IP packet, it obtains the Destination Network Number by looking up in the mapping table between IP network numbers and EIP Network Numbers. The border router then adds the Extension field with the Source and Destination Network

Numbers into the packet, and forwards to the backbone networks. It is only necessary to translate the out-going IP packets to the EIP packets. There is no need to translate the EIP packets back to IP packets even when they are destined to IP hosts. This is because that the Extension field in the EIP packets appears to IP hosts just an unknown IP option and is ignored by the IP hosts during the processing.

- 4) Border Routers: The new EIP Extension has to be implemented and routing has to be done based on the Network Number in the EIP Extension field. The border routers may have to provide the translation service for out-going IP packets during the transition period.
- 5) Subnet Routers: No modifications are required during the transition period when EIP Host Numbers (which equals to the IP addresses) are still globally unique. The subnet routing is carried out based on the EIP Host Numbers and when the EIP Host IDs are still unique, subnet routers can determine, by treating the EIP Host Number as the IP addresses, whether a packet is destined to remote networks or not and forward correctly. The Extension field in the EIP packets also appear to the IP subnet routers an unknown IP option and is ignored in the processing. However, subnet routers eventually have to implement the EIP Extension and carry out routing based on Network Numbers when EIP Host Numbers are no longer globally unique.
- 6) Hosts: The EIP Extension has to be implemented. routing has to be done based on the Network Number in the EIP Extension field, and also based on the Host Number and subnet mask if subnetting is used. IP hosts may communication with any hosts within the same network at any time. During the transition period when the EIP Host Numbers are still unique, IP hosts can communicate with any hosts in other network via the translation service.
- 7) DNS: A new resource record (RR) type "N" has to be added for EIP Network Numbers. The RR type "A", currently used for IP addresses, can still be used for EIP Host Numbers. RR type "N" entries have to be added and RR type "PTR" to be updated. All other entries remain unchanged.
- 8) ARP/RARP: No modifications are required. The ARP and RARP are used for mapping between EIP Host Numbers and physical addresses.
- 9) ICMP: No modifications are required.
- 10) TCP/UDP Checksum: No modifications are required. The pseudo

header includes the EIP Source and Destination IDs instead of the source and destination IP addresses.

- 11) FTP: No modifications are required during the transition period when the IP hosts can still communicate with hosts in other networks via the translation service. After the transition period, however, the "DATA Port (Port)" command has to be modified to pass the port number only and ignore the IP address. A new FTP command may be created to pass both the port number and the EIP address to allow a third party to be involved in the file transfer.

Transition to EIP

In this section, we outline a plan for transition to EIP.

EIP can greatly reduce the complexity of transition. In particular, there is no need for the updated hosts and subnet routers to run two protocols in parallel in order to achieve interoperability between old and new systems. During the transition, IP hosts can still communicate with any machines in the same network without any changes. When the EIP Host Numbers (i.e., the 32-bit IP addresses) are still globally unique, IP hosts can contact hosts in other networks via translation service provided in the border routers.

The transition goes as follows:

Phase 0:

- a) Choose an addressing and routing scheme for the Internet.
- b) Implement the routing protocol.
- c) Assign new Network Numbers to existing networks.

Phase 1:

- a) Update all backbone routers and border routers.
- b) Update DNS servers.
- c) Start translation service.

Phase 2:

- a) Update first the key hosts such as mail servers, DNS servers, FTP servers and central machines.
- b) Update gradually the rest of the hosts.

Phase 3:

- a) Update subnet routers
- b) Withdraw the translation service.

The translation service can be provided as long as the Host IDs (i.e., the 32-bit IP address) are still globally unique. When the IP

address space is exhausted, the translation service will be withdrawn and the remaining IP hosts can only communicate with hosts within the the same network. At the same time, networks can use any numbers in the 32-bit space for addressing their hosts.

Related Work

A recent proposal called IPAE by Hinden and Crocker also attempts to minimize the modifications to the current IP system yet to extend the addressing space [12]. IPAE uses encapsulation so that the extended space is carried as IP data. However, it has been found that the 64 bits IP data returned by an ICMP packet is too small to hold the Global IP addresses. Thus, when a router receives an ICMP generated by an old IP host, it is not able to convert it into a proper ICMP packet. More details can be found in [13].

Discussions

EIP does not necessary increase the header length significantly as most of the fields in the current IP will be still needed in the new internet protocol. There are debates as to whether fragmentation and header checksum are necessary in the new internet protocol but no consensus has been reached.

EIP assumes that IP hosts and routers ignore unknown IP option silently as required by [15,16]. Some people have expressed some concerns as to whether current IP routers and hosts in the Internet can deal with unknown IP options properly.

In order to look into the issues further, we carried out a number of experiments over the use of IP option. We selected 35 hosts over 30 countries across the Internet. A TCP test program (based on `ttcp.c`) then transmitted data to the echo port (tcp port 7) of each of the hosts. Two tests were carried out for each host, one with an unknown option (type 0x8A, length 40 bytes) and the other without any options.

It is difficult to ensure that the conditions under which the two tests run are identical but we tried to make them as close as possible. The two tests (test-opt and test-noopt) run on the same machine a Sun4) in parallel, i.e., "test-opt & ; test-noopt" and then again in the reverse order, i.e., "test-noopt & ; test-opt", so each test pair actually run twice. Each host was ping'ed before the tests so that the domain name information was cached before the name lookup.

The experiments were carried out at three sites: UCL, Bellcore and Cambridge University. The tcp echo throughput (KB/Sec) results are

listed in Appendix.

The results show that the IP option was dealt with properly and there is no visible performance difference under the test setup.

References

- [1] Chiappa, N., "The IP Addressing Issue", Work in Progress, October 1990.
- [2] Clark, D., Chapin, L., Cerf, V., Braden, R., and R. Hobby, "Towards the Future Architecture", RFC 1287, MIT, BBN, CNRI, ISI, UC Davis, December 1991.
- [3] Solensky, F. and F. Kastenholz, "A Revision to IP Address Classifications", Work in Progress, March 1992.
- [4] Fuller, V., Li, T., Yu, J., and K. Varadhan, "Supernetting: an Address Assignment and Aggregation Strategy", RFC 1338, BARRNet, cisco, Merit, OARnet, June 1992.
- [5] Wang, Z., and J. Crowcroft, "A Two-Tier Address Structure for the Internet: a solution to the problem of address space exhaustion", RFC 1335, University College London, May 1992.
- [6] Callon, R., "TCP and UDP with Bigger Addresses (TUBA), a Simple Proposal for Internet Addressing and Routing", RFC 1347, DEC, June 1992.
- [7] Tsuchiya, P., "Pip: The 'P' Internet Protocol", Work in Progress, May 1992
- [8] Chiappa N., "A New IP Routing and Addressing Architecture", Work in Progress, 1992.
- [9] Colella, R., Gardner, E., and R. Callon, "Guidelines for OSI NSAP Allocation in the Internet" RFC 1237, NIST, Mitre, DEC, July 1991.
- [10] Deering, S., "City Codes: An Alternative Scheme for OSI NSAP Allocation in the Internet", Work in Progress, January 1992.
- [11] Clark, D., "Building routers for the routing of tomorrow", in his message to Big-Interent@munari.oz.au, 14 July 1992.
- [12] Hinden, R., and D. Crocker, "A Proposal for IP Address Encapsulation (IPAE): A Compatible Version of IP with Large Addresses", Work in Progress, July 1992.

- [13] Partridge, C., "Re: Note on implementing IPAE", in his message to Big-Interent@munnari.oz.au, 17 July 1992.
- [14] Deering, S., "SIP: Simple Internet Protocol", Work in Progress, September 1992.
- [15] Braden, R., Editor, "Requirements for Internet Hosts -- Communication Layers", RFC 1122, ISI, October 1989.
- [16] Almquist, P., Editor, "Requirements for IP Routers", Work in Progress, October 1991.

Appendix

Throughput Test from UCL (sartre.cs.ucl.ac.uk)

Destination Host	test-noopt	test-opt
-----	-----	-----
oliver.cs.mcgill.ca	1.128756	1.285345
oliver.cs.mcgill.ca	1.063096	1.239709
bertha.cc.und.ac.za	0.094336	0.043917
bertha.cc.und.ac.za	0.075681	0.057120
vnet3.vub.ac.be	2.090622	2.228181
vnet3.vub.ac.be	1.781374	1.692740
itdsrv1.ul.ie	1.937596	2.062579
itdsrv1.ul.ie	1.928313	1.936784
sunic.sunet.se	11.064797	11.724055
sunic.sunet.se	10.861720	10.840306
pascal.acm.org	2.463790	0.810133
pascal.acm.org	1.619088	0.860198
iti.gov.sg	1.565320	1.983795
iti.gov.sg	1.564788	1.921803
rzusuntk.unizh.ch	9.903805	11.335920
rzusuntk.unizh.ch	9.597806	10.678098
funet.fi	9.897876	9.382925
funet.fi	10.487118	11.023745
odin.diku.dk	5.851407	5.482946
odin.diku.dk	5.992257	6.243283
cphkvx.cphk.hk	0.758044	0.844406
cphkvx.cphk.hk	0.784532	0.745606
bootes.cus.cam.ac.uk	28.341705	29.655824
bootes.cus.cam.ac.uk	24.804125	23.240990
pesach.jct.ac.il	1.045922	1.115802
pesach.jct.ac.il	1.330429	0.978184
sun1.sara.nl	10.546733	11.500778
sun1.sara.nl	9.624833	10.214136
cocos.fuw.edu.pl	1.747777	1.702324
cocos.fuw.edu.pl	1.676151	1.716435

RFC 1385

EIP

November 1992

apple.com	4.449559	4.145081
apple.com	6.431675	5.520443
gorgon.tf.tele.no	1.199810	1.374546
gorgon.tf.tele.no	0.508642	0.993261
kogwy.cc.keio.ac.jp	3.626448	3.249590
kogwy.cc.keio.ac.jp	3.913777	4.094849
exu.inf.puc-rio.br	1.913925	1.795235
exu.inf.puc-rio.br	1.154936	1.114775
inria.inria.fr	2.299561	0.599665
inria.inria.fr	1.219282	0.873672
kum.kaist.ac.kr	0.252704	0.254199
kum.kaist.ac.kr	0.236196	0.172367
sunipc1.labein.es	1.398777	1.243588
sunipc1.labein.es	0.876177	0.602964
wifosv.wsr.ac.at	0.531153	0.803387
wifosv.wsr.ac.at	0.773935	0.557798
uunet.uu.net	7.813556	6.764543
uunet.uu.net	7.969203	6.657325
infnsun.aquila.infn.it	2.321197	2.402477
infnsun.aquila.infn.it	2.400196	2.695016
muttley.fc.ul.pt	0.545775	0.434672
muttley.fc.ul.pt	0.284124	0.266017
dmssyd.syd.dms.csiro.au	2.734685	2.857545
dmssyd.syd.dms.csiro.au	1.168154	1.462789
hamlet.caltech.edu	2.552804	2.897286
hamlet.caltech.edu	3.839141	2.407945
sztaki.hu	0.294196	0.403697
sztaki.hu	0.236260	0.388755
menvax.restena.lu	0.465066	0.515361
menvax.restena.lu	0.358646	0.511985
nctu.edu.tw	0.484372	0.816722
nctu.edu.tw	0.705733	1.109228
xalapa.lania.mx	0.899529	0.822544
xalapa.lania.mx	1.150058	0.881713
truth.waikato.ac.nz	1.438481	1.993749
truth.waikato.ac.nz	1.325041	1.833999

Throughput Test from Bellcore (latour.bellcore.com)

Destination Host	test-noopt	test-opt
-----	-----	-----
oliver.cs.mcgill.ca	1.820014	2.128104
oliver.cs.mcgill.ca	1.979981	1.866815
bertha.cc.und.ac.za	0.099289	0.035877
bertha.cc.und.ac.za	0.118627	0.103763
vnet3.vub.ac.be	0.368476	0.694463
vnet3.vub.ac.be	0.443269	0.644050
itdsrv1.ul.ie	0.721444	0.960068
itdsrv1.ul.ie	0.713952	0.953275
sunic.sunet.se	2.989907	2.956766
sunic.sunet.se	2.100563	2.010292
pascal.acm.org	2.487185	3.896253
pascal.acm.org	1.944085	4.269323
iti.gov.sg	2.401733	2.735445
iti.gov.sg	2.950990	2.793121
rzusuntk.unizh.ch	4.094820	3.618023
rzusuntk.unizh.ch	2.952650	2.245001
funet.fi	6.703408	5.928008
funet.fi	7.389722	5.815122
odin.diku.dk	2.094152	2.450695
odin.diku.dk	5.362362	4.690722
cphkvx.cphk.hk	0.092698	0.106880
cphkvx.cphk.hk	0.496394	0.681994
bootes.cus.cam.ac.uk	2.632951	2.631322
bootes.cus.cam.ac.uk	3.717170	1.335914
pesach.jct.ac.il	0.684029	0.921621
pesach.jct.ac.il	0.390263	1.095265
sun1.sara.nl	3.186035	2.325166
sun1.sara.nl	3.053797	3.081236
cocos.fuw.edu.pl	0.154405	0.124795
cocos.fuw.edu.pl	0.120283	0.105825
apple.com	12.588979	12.957880
apple.com	13.861733	12.211125
gorgon.tf.tele.no	1.280217	1.112675
gorgon.tf.tele.no	0.243205	0.631096
kogwy.cc.keio.ac.jp	6.249789	5.075968
kogwy.cc.keio.ac.jp	3.387490	4.583511
exu.inf.puc-rio.br	2.089536	2.233711
exu.inf.puc-rio.br	2.476758	2.249439
inria.inria.fr	0.653974	0.866246
inria.inria.fr	0.739127	1.130521
kum.kaist.ac.kr	1.541682	1.312546
kum.kaist.ac.kr	0.906632	1.042793
sunipcl.labein.es	0.101496	0.091456
sunipcl.labein.es	0.054245	0.101585

RFC 1385

EIP

November 1992

wifosv.wsr.ac.at	1.044443	0.369528
wifosv.wsr.ac.at	0.596935	0.870377
uunet.uu.net	9.530348	8.999789
uunet.uu.net	8.941888	6.075660
infnsun.aquila.infn.it	1.619418	1.569645
infnsun.aquila.infn.it	1.156780	1.158000
muttley.fc.ul.pt	0.353632	0.416126
muttley.fc.ul.pt	0.221522	0.155505
dmssyd.syd.dms.csiro.au	3.433901	3.272839
dmssyd.syd.dms.csiro.au	3.408975	3.130188
hamlet.caltech.edu	5.367756	6.325031
hamlet.caltech.edu	4.828718	5.676571
sztaki.hu	0.301120	0.362481
sztaki.hu	0.253222	0.519892
menvax.restena.lu	0.364221	0.480789
menvax.restena.lu	0.456882	0.580778
nctu.edu.tw	0.246523	1.199412
nctu.edu.tw	0.423476	0.630833
xalapa.lania.mx	0.748642	0.607284
xalapa.lania.mx	0.716781	0.643030
truth.waikato.ac.nz	2.197595	2.072601
truth.waikato.ac.nz	2.489748	2.186684

Throughput Test from Cam U (cus.cam.ac.uk)

Destination Host	test-noopt	test-opt
-----	-----	-----
oliver.cs.mcgill.ca	1.128756	1.285345
oliver.cs.mcgill.ca	1.063096	1.239709
bertha.cc.und.ac.za	0.031218	0.031221
bertha.cc.und.ac.za	0.034405	0.034925
vnet3.vub.ac.be	0.568487	0.731320
vnet3.vub.ac.be	0.558238	0.581415
itdsrv1.ul.ie	1.064302	1.284707
itdsrv1.ul.ie	0.852089	1.025779
sunic.sunet.se	7.179942	6.270326
sunic.sunet.se	5.772485	6.689160
pascal.acm.org	1.661248	1.726725
pascal.acm.org	1.557839	1.428193
iti.gov.sg	0.600616	0.926690
iti.gov.sg	0.772887	0.956636
rzusuntk.unizh.ch	3.645913	4.504969
rzusuntk.unizh.ch	1.853503	2.671272
funet.fi	4.190147	3.421110
funet.fi	2.270988	3.789678
odin.diku.dk	1.361227	0.993901
odin.diku.dk	1.977774	2.415716
cphkvx.cphk.hk	1.173451	1.298421
cphkvx.cphk.hk	1.151376	1.184210
bootes.cus.cam.ac.uk	269.589141	238.920081
bootes.cus.cam.ac.uk	331.203020	293.556436
pesach.jct.ac.il	0.343598	0.492202
pesach.jct.ac.il	0.582809	0.930958
sun1.sara.nl	1.529277	1.470571
sun1.sara.nl	0.896041	0.894923
cocos.fuw.edu.pl	0.131180	0.142239
cocos.fuw.edu.pl	0.137697	0.148895
apple.com	1.330794	0.453590
apple.com	0.856476	0.714661
gorgon.tf.tele.no	0.094793	0.099981
gorgon.tf.tele.no	0.167257	0.151625
kogwy.cc.keio.ac.jp	0.154681	0.178868
kogwy.cc.keio.ac.jp	1.095814	0.871496
exu.inf.puc-rio.br	0.454272	0.384484
exu.inf.puc-rio.br	0.705198	0.690708
inria.inria.fr	0.149511	0.150021
inria.inria.fr	0.071125	0.077257
kum.kaist.ac.kr	0.721184	0.549511
kum.kaist.ac.kr	0.250285	0.296195
sunipcl.labein.es	0.519284	0.491745
sunipcl.labein.es	0.990174	1.009475

wifosv.wsr.ac.at	0.360751	0.418554
wifosv.wsr.ac.at	0.344268	0.326605
uunet.uu.net	4.247430	3.305592
uunet.uu.net	3.139251	2.945469
infnsun.aquila.infn.it	0.480731	0.782631
infnsun.aquila.infn.it	0.230471	0.292273
muttley.fc.ul.pt	0.239624	0.334286
muttley.fc.ul.pt	0.586156	0.419485
dmssyd.syd.dms.csiro.au	3.630623	3.607504
dmssyd.syd.dms.csiro.au	1.743162	2.994665
hamlet.caltech.edu	5.897946	4.650703
hamlet.caltech.edu	5.118200	5.622022
sztaki.hu	0.338358	0.225206
sztaki.hu	0.113328	0.112637
menvax.restena.lu	0.224967	0.359237
menvax.restena.lu	0.452945	0.472345
nctu.edu.tw	2.549709	2.037245
nctu.edu.tw	2.229093	2.469851
xalapa.lania.mx	0.713586	0.810107
xalapa.lania.mx	0.612278	0.731705
truth.waikato.ac.nz	1.438481	1.993749
truth.waikato.ac.nz	1.325041	1.833999

Security Considerations

Security issues are not discussed in this memo.

Author's Address

Zheng Wang
 Dept of Computer Science
 University College London
 London WC1E 6BT, UK

EEmail: z.wang@cs.ucl.ac.uk

US005442633A

United States Patent [19]

[11] **Patent Number:** 5,442,633

Perkins et al.

[45] **Date of Patent:** Aug. 15, 1995

- [54] **SHORTCUT NETWORK LAYER ROUTING FOR MOBILE HOSTS**
- [75] Inventors: **Charles E. Perkins**, Ossining; **Jacob Y. Rekhter**, Putnam Valley, both of N.Y.
- [73] Assignee: **International Business Machines Corporation**, Armonk, N.Y.
- [21] Appl. No.: **910,701**
- [22] Filed: **Jul. 8, 1992**
- [51] Int. Cl.⁶ **H04B 7/26; H04J 3/26**
- [52] U.S. Cl. **370/94.1; 455/54.1; 455/56.1**
- [58] **Field of Search** **370/85.1, 85.2, 85.3, 370/85.7, 85.13, 85.14, 94.1, 94.2, 94.3, 95.1, 95.3, 95.2; 340/825.5, 825.51; 455/39, 68, 53.1, 54.1, 54.2, 55.1, 56.1, 33.1, 33.2; 379/58, 59, 63**

[56] **References Cited**
U.S. PATENT DOCUMENTS

4,644,461	2/1987	Jennings	364/200
4,665,519	5/1987	Kirchner et al.	370/94.1
4,706,081	11/1987	Hart et al.	340/825.03
4,750,109	6/1988	Kita	364/200
4,807,222	2/1989	Amitay	370/85.15
4,809,257	2/1989	Gentenbein et al.	
4,893,307	1/1990	McKay et al.	370/94.1
4,914,652	4/1990	Nguyen	370/85.5
5,040,175	8/1991	Tuch et al.	370/94.1
5,046,066	9/1991	Messenger	370/94.1
5,068,916	11/1991	Harrison et al.	455/39
5,210,753	5/1993	Natarajan	370/95.1

FOREIGN PATENT DOCUMENTS

0182417	5/1986	European Pat. Off.	.
0328100	8/1989	European Pat. Off.	.
WO88/07794	10/1988	WIPO	.

OTHER PUBLICATIONS

Data Communications, vol. 16, No. 12, Nov. 1987, New York US, pp. 209-225; D. Retz: "TCP/IP: DOD suite marches into the business world".
"Internet Protocol DARPA Internet Program Protocol Specification", Sep. 1981, Information Sciences Insti-

tute, University of Southern CA, Marina del Rey, Calif. 90291.

"Infrared Microbroadcasting Network For In-House Data Communication" F. Gfeller, *IBM Technical Disclosure Bulletin*, vol. 24, No. 8, Jan. 1982.

IEEE Transactions on Communications, vol. 38, No. 8, Aug. 1990, New York, pp. 1272-1280; D. J. Goodwin: "Cellular Packet Communications".

10th Conference On Local Computer Networks, Oct. 1985, New York US pp. 149-157 W. M. Loucks et al.: "Implementation Of A Dynamic Address Assignment Protocol In A Local Area Network" RFC 1122, (Oct. 1989) R. Braden (pp. 35,36).

Primary Examiner—Wellington Chin
Attorney, Agent, or Firm—Perman & Green

[57] **ABSTRACT**

A method for routing a packet of information between two hosts that are coupled to a network. Each of the hosts have a unique network address, and at least one of the hosts is a mobile host (10) that does not have a fixed network coupling location. The method includes a first step of (a) transmitting a packet from the mobile host to a second, destination host on the network through a wireless link that is established between the mobile host and a base access station (12) that serves a current physical location of the mobile host. The base access station is coupled to the network via a subnetwork (LAN) (14), and the packet includes a first Internet Protocol (IP) Loose Source Routing (LSR) option that includes a network address of the base access station. A second step (b) receives with the destination host the packet that includes the first IP LSR option. A third step (c) transmits a further, reply packet from the second host to the mobile host via the base access station in accordance with a path reversal technique wherein the reply packet includes a second IP LSR option that specifies as a first Routing address the network address of the base access station. As a result, the reply packet is directed through the network to the base access station that serves the current physical location of the mobile host, and an optimal, fast routing of the packet is achieved without involving intermediate gateways (16, 18).

20 Claims, 4 Drawing Sheets

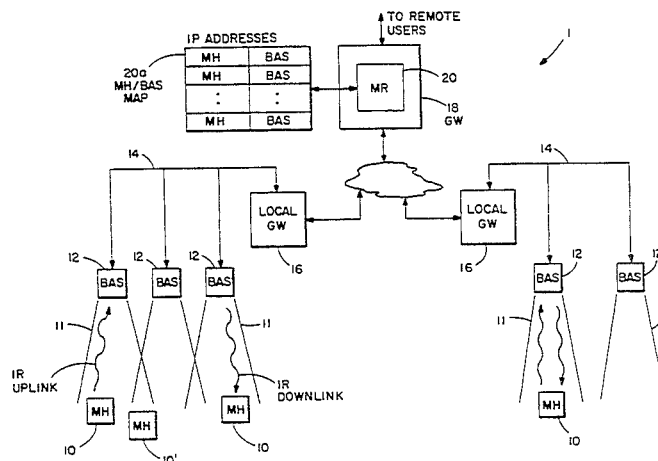


FIG. 1

PRIOR ART

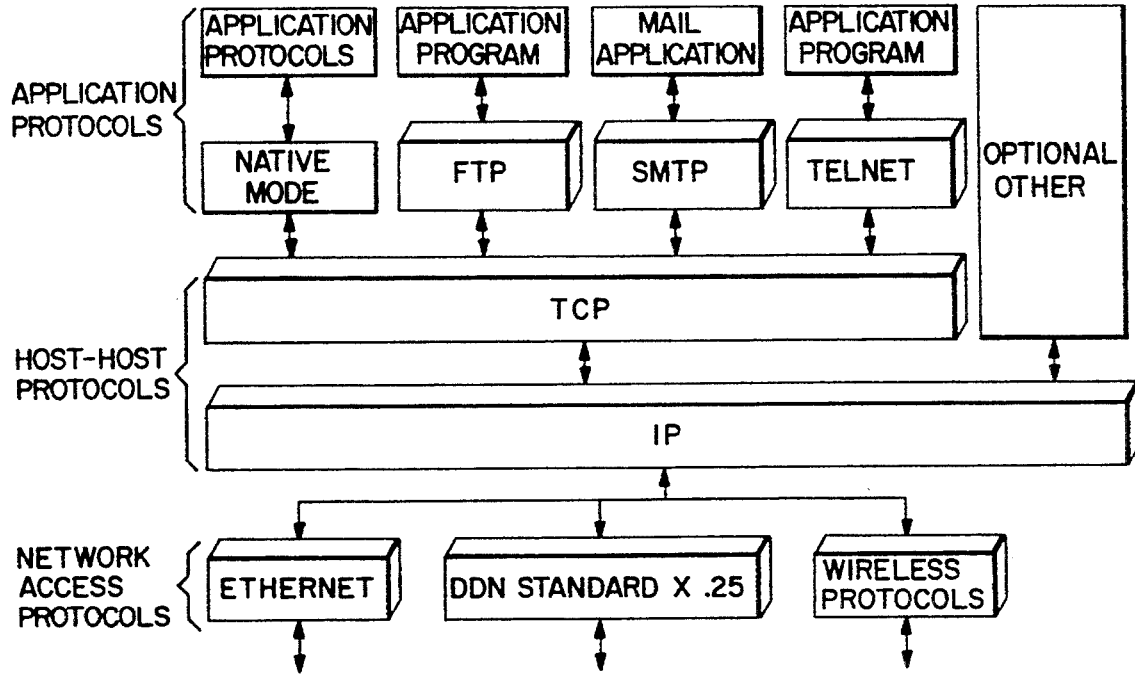


FIG. 3A

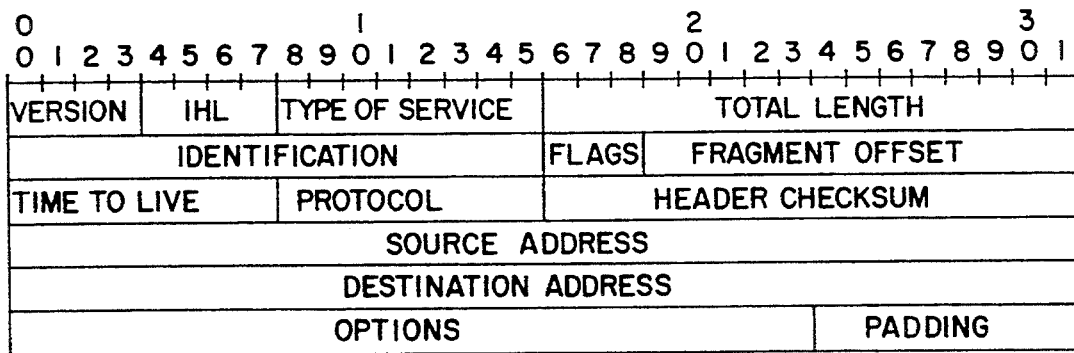
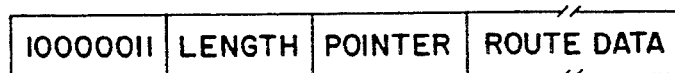


FIG. 3B

LSR OPTIONS



TYPE = 131

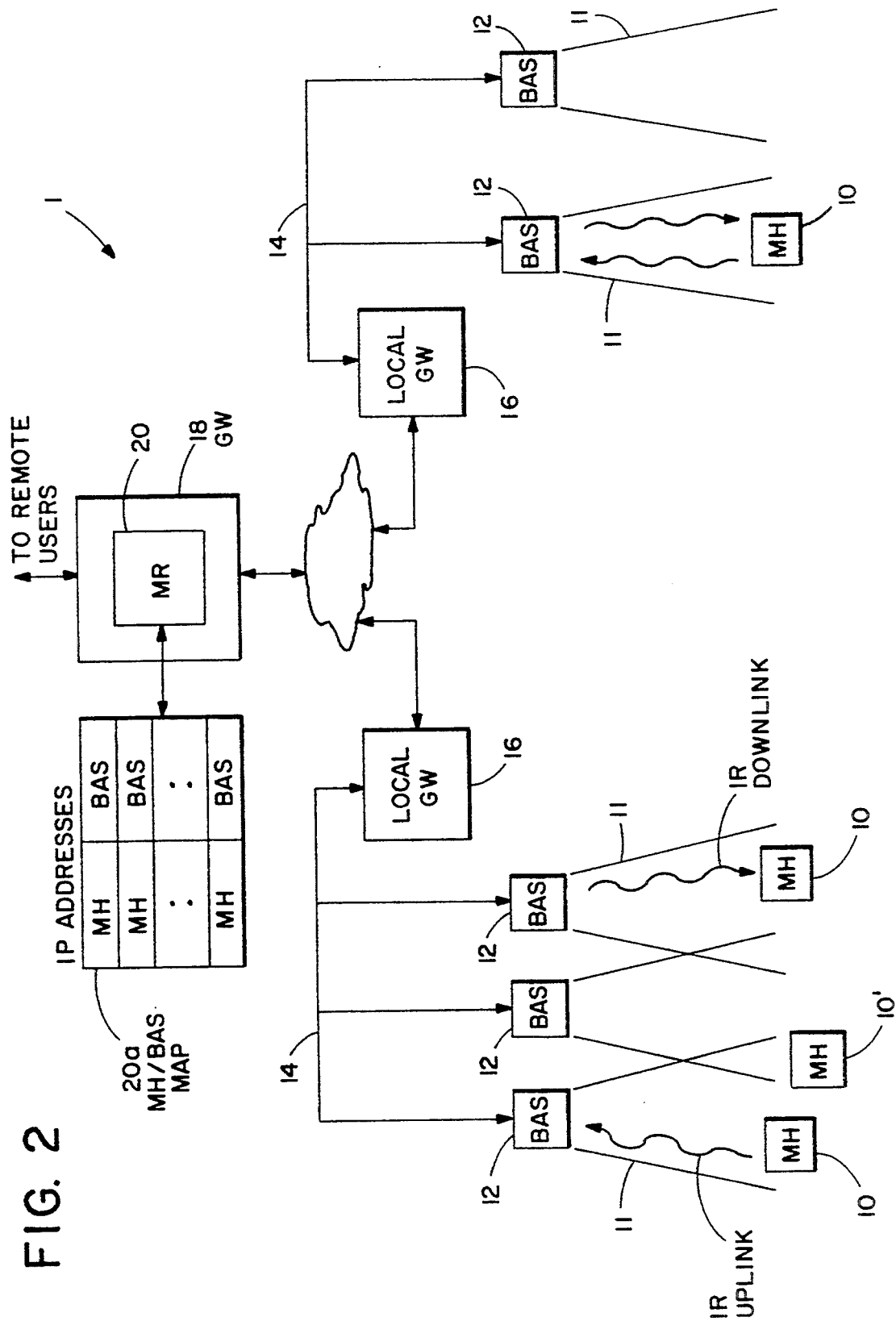


FIG. 2

FIG. 4

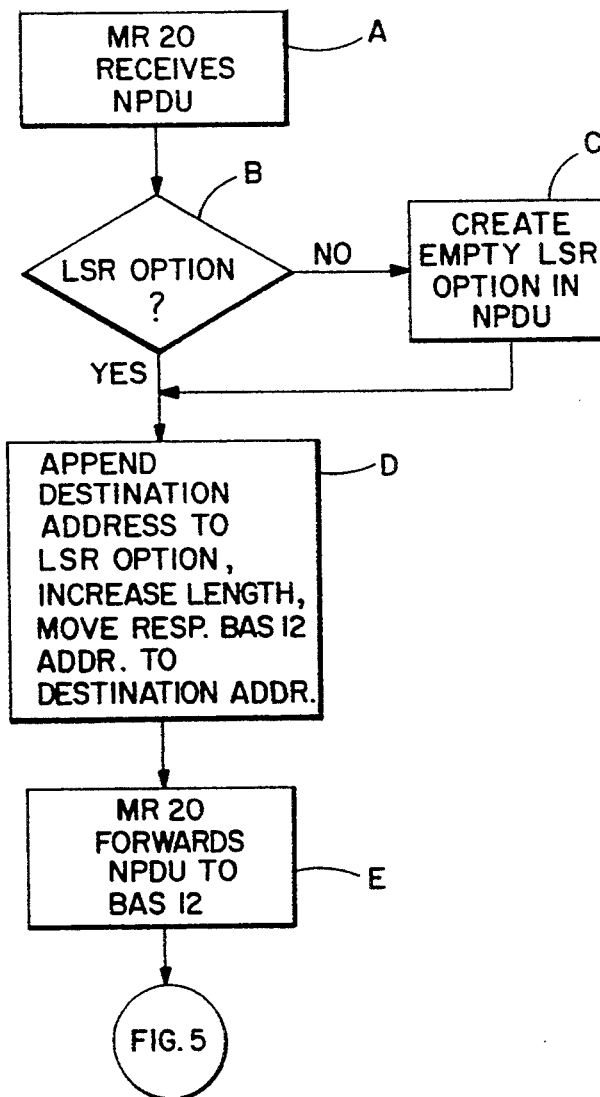
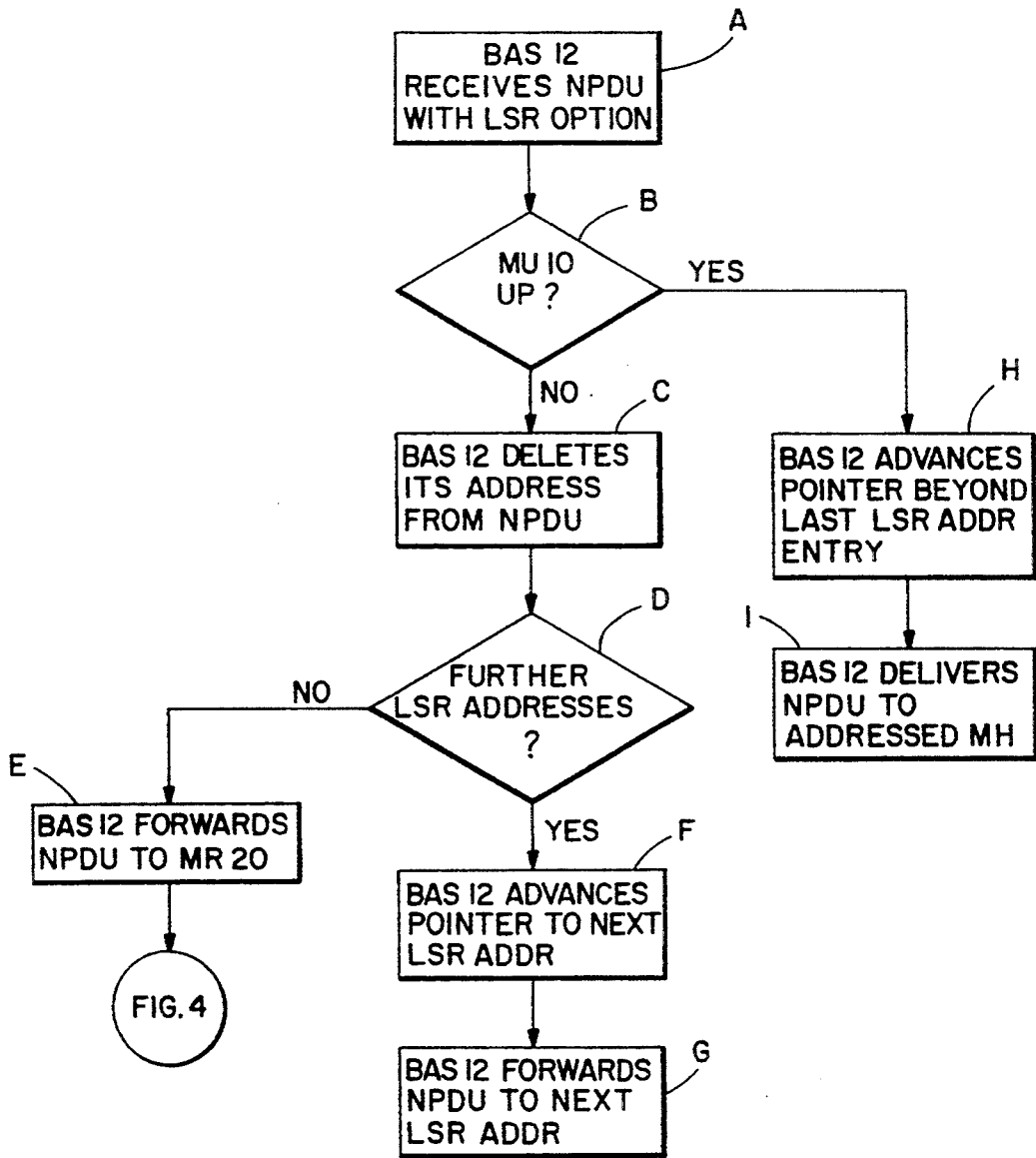


FIG. 5



SHORTCUT NETWORK LAYER ROUTING FOR MOBILE HOSTS

CROSS-REFERENCE TO A RELATED PATENT APPLICATION

This patent application is related to a commonly assigned U.S. patent application entitled "Network Address Management for a Wired Network Supporting Wireless Communication to a Plurality of Mobile Users", Ser. No. 07/605,592, filed Oct. 29, 1990, now U.S. Pat. No. 5,159,592, by C. E. Perkins.

FIELD OF THE INVENTION

This invention relates generally to communication methods and apparatus and, in particular, to methods and apparatus for managing network datagram routing in a network that includes mobile users.

BACKGROUND OF THE INVENTION

Commonly assigned U.S. Pat. No. 4,893,307, issued Jan. 9, 1990, "Method and Apparatus for Linking SNA Terminals to an SNA Host Over a Packet Switched Communications Network", D. B. McKay, R. M. Morten and M. P. Marsili, describes an architectural model of the Department of Defense (DoD) protocol suite.

Referring to FIG. 1, the architecture is said to be similar to, but not identical with, the International Standards Organization (ISO) Open Systems Interconnection (OSI) architecture.

A Defense Data Network (DDN) standard establishes criteria for an Internet Protocol (IP) which supports the interconnection of communication LANs.

It introduces the Internet Protocol's role and purpose, defines the services provided to users, and specifies the mechanisms needed to support those services. The standard also defines the services required of the lower protocol layer, describes the upper and lower interfaces, and outlines the execution environment services need for implementation.

A Transmission Control Protocol (TCP) is a transport protocol providing connection-oriented, end-to-end reliable data transmission in packet-switched computer LANs and internetworks.

The Internet Protocol (IP) and the Transmission Control Protocol (TCP) are mandatory for use in all DoD packet switching networks which connect or have the potential for utilizing connectivity across network or subnetwork boundaries. Network elements, such as hosts, front-ends, gateways, etc., within such networks which are to be used for internetting must implement TCP/IP.

The Internet Protocol is designed to interconnect packet-switched communication LANs to form an internetwork. The IP transmits blocks of data, called internet datagrams, from sources to destinations throughout the internet. Sources and destinations are hosts located on either the same subnetwork or on connected LANs. The IP is intentionally limited in scope to provide the basic functions necessary to deliver a block of data. Each internet datagram is an independent entity unrelated to any other internet datagrams. The IP does not create connections or logical circuits and has no mechanisms to promote data reliability, flow control, sequencing, or other services commonly found in virtual circuit protocols.

The DDN standard specifies a host IP. As defined in the DoD architectural model, the Internet Protocol

resides in the internetwork layer. Thus, the IP provides services to transport layer protocols and relies on the services of the lower network protocol. In each gateway, a system interconnecting two or more LANs, an IP resides above two or more LAN's protocol entities. Gateways implement IP to forward datagrams between LANs. Gateways also implement a routing protocol to coordinate signalling and other internet control information.

Various Network Access Protocols reside below the IP and may include, by example, an Ethernet protocol, an X.25 protocol, and, of particular interest herein, a wireless network protocol.

The Internet protocols were originally developed with an assumption that users, each of which is assigned a unique Internet address, would be connected to the network at fixed locations. However, for portable and handheld computers employing a wireless protocol the movement, or migration, of users about the network is typically the rule rather than the exception. As a result, a problem is created in that the implicit design assumptions of the Internet protocol are violated by this type of usage.

The problem that arises thus relates to providing optimal network layer routing with a mobile host, when network layer address(es) assigned to a host may not bear any network topological significance. The problem arises because of a requirement for a host to have an identifier that remains fixed, even as the host moves, while at the same time providing sufficient information in the network layer to make network layer routing feasible.

It is thus an object of this invention to provide a method for optimizing network layer routing between a pair of hosts, where at least one of the hosts is mobile and, as a result, does not have a fixed connection location with respect to the network.

It is another object of the invention to provide a method for optimizing network layer routing between a pair of hosts, where at least one of the hosts is mobile, in the context of a network that operates in accordance with the Internet or an Internet-type protocol.

SUMMARY OF THE INVENTION

The foregoing and other problems are overcome and the objects of the invention are realized by a method for routing a packet of information between two hosts that are coupled to a network. Each of the hosts have a unique network address, and at least one of the hosts is a mobile host that does not have a fixed network coupling location. The method includes a first step of (a) transmitting a packet from the mobile host to a second, destination host on the network through a wireless link that is established between the mobile host and a base access station that serves a current physical location of the mobile host. The base access station is coupled to the network via a subnetwork (LAN), and the packet includes, in a presently preferred embodiment of the invention, an Internet Protocol (IP) Loose Source Routing (LSR) option that includes a network address of the base access station.

A second step (b) receives with the destination host the packet that includes the first LSR option, and a third step (c) transmits a further packet, typically a reply packet, from the second host to the mobile host via the base access station in accordance with a path reversal technique.

In accordance with the path reversal technique, if a host receives a datagram containing a completed source route, i.e. the pointer points beyond the last address field, the datagram is considered to have reached its final destination. The source route option (recorded route), as received, is passed up to a Transport layer, or to ICMP message processing. The recorded route is reversed and is used to form a return source route for reply datagrams. When the return source route is built, it is correctly formed even if the the recorded route included the source host.

As a result, the reply packet (datagram) is directed through the network to the base access station that serves the current physical location of the mobile host, and an optimal, fast routing of the packet is achieved without requiring the involvement a mobile router.

In response to the mobile host establishing a wireless link with a second base access station on the same or a different subnetwork, the method includes the steps of determining the network address of the second base access station with the mobile host; transmitting the network address of the second base access station from the mobile host to a mobile router that is coupled between the subnetwork and the network, the transmission including the network address of the mobile host; and maintaining the network address of the mobile host and the network address of the second base access station with the mobile router.

The mobile router advertises to the network the network address of the mobile router and also the network address of network associated with the mobile hosts.

The step of transmitting the reply packet includes the steps of receiving the reply packet at the base access station; and determining if the mobile host is currently located within the physical area served by the base access station.

If it is determined that the mobile host is currently located within the physical area served by the base access station, the method forwards the reply packet from the base access station over the wireless link to the mobile host.

If, instead, it is determined that the mobile host is not currently located within the physical area served by the base access station, the method forwards the reply packet from the base access station over the network to the mobile router, if there are no further intermediate addresses within the LSR option. The mobile router then forwards the reply packet to a second base access station that serves a physical location within which the mobile host is currently located.

In response to a reception of an packet by the mobile router from the network, the packet not including a LSR option and having the network address of the mobile host as a destination address, the method includes the steps of (a) converting the received packet into an packet that includes a LSR option, the LSR option being provided by the mobile router with the network address of the base access station that is serving the physical location within which the mobile host is currently located; (b) forwarding the converted packet from the mobile router to the base access station that serves the physical location within which the mobile host is currently located; and (c) receiving the converted packet and forwarding the received converted packet from the base access station, over the wireless link, to the mobile host.

BRIEF DESCRIPTION OF THE DRAWING

The above set forth and other features of the invention are made more apparent in the ensuing Detailed Description of the Invention when read in conjunction with the attached Drawing, wherein:

FIG. 1 is a prior art architectural diagram of the Defense Data Network;

FIG. 2 is a block diagram showing a plurality of mobile hosts in bidirectional wireless communication with two LANS via a plurality of base access stations;

FIG. 3a illustrates a format of an Internet Datagram Header;

FIG. 3b illustrates a format of a Loose Source and Record Route (LSSR) option employed in the OPTION field of FIG. 3a; and

FIGS. 4 and 5 are each a flowchart that illustrate an aspect of the method of the invention.

DETAILED DESCRIPTION OF THE INVENTION

It is noted that the method described in detail below is applicable to hosts that use either the IP, or a protocol known as the Connectionless-Mode Network Service Protocol (CLNP), as the network layer protocol. IP is described in detail in a document entitled "Internet Protocol Darpa Internet Program Protocol Specification", September 1981, RFC:791. CLNP is described in a document entitled "Protocol for providing connectionless-mode network service", ISO 8473. Although the ensuing description is made specifically with reference to the IP, it should be realized that support for CLNP may be accomplished using the same techniques. As such, the teaching of the invention is not intended to be limited in scope to only networks that employ IP.

Reference is made to FIG. 2 where there is illustrated a communications area network 1. The network 1 includes one or more local area networks (LANs) 14. Each LAN 14 includes a wireless network comprised of at least one Mobile Host (MH) 10 in wireless communication with one or more Base Access Stations (BAS) 12. Each of the BASs 12 is bidirectionally coupled to one of the wired LANs 14, also referred to herein as Level 2 subnetworks. In the presently preferred embodiment of the invention the wireless medium is comprised of infrared (IR) radiation, although other embodiments may employ an RF wireless medium. Each of the BASs 12 has associated therewith a communications coverage area, or cell 11, which may or may not overlap one another. Communication between MHs 10, and from a MH 10 to other entities coupled to the network, is through the BASs 12, via the LAN 14. Communication between the BASs 12 is primarily via the LAN 14.

One suitable embodiment for the BASs 12 and the MHs 10 is disclosed in commonly assigned U.S. Pat. No. 5,068,916, issued Nov. 26, 1991, entitled "Coordination of Wireless Medium Among A Plurality Of Base Stations", by C. G. Harrison and Peter D. Hortensius. It should be realized, however, that the teaching of the invention may be embodied within a number of different types of wireless network embodiments.

If there is more than one Level 2 subnetwork (LAN), then each of the LANs 14 preferably includes at least one local gateway (GW) 16 for coupling the MH 10, via the BAS 12 and the LAN 14, to a gateway 18. The connection between the local gateways 16 and the gateway 18 may be via some arbitrary number of additional gateways. The gateway 18 is also coupled to remote

network users who may be dispersed over a wide geographic area. The local gateways **16** may each be an “intelligent” BAS, or may be a separate dedicated network entity as shown. The gateway **18** is preferably a data processor having suitable network adapters and an archival facility.

An IP address consists of four bytes, or 32 bits, that are partitioned into a LAN identification and a Host identification. By example, an IP address may have the form 123.45.67.12. In the absence of a subnet mask, the first one, two, or three bytes encode a LAN address. For example, the LAN address may be encoded as 123 (byte 1) and 45 (byte 2). The remaining bits generally encode Host address information. In the example provided Host (**12**) may have up to 64K IP addresses associated therewith, as encoded in the third and fourth bytes.

It is assumed in the ensuing description that each MH **10** has a Network Layer Address (IP address or an NSAP). This address is referred to as a “permanent address” that does not change as the MH **10** migrates between Level 2 subnetworks. However, it is within the scope of the invention to provide dynamic creation of such a “permanent” address for the MH **10**, as is described in the aforementioned commonly assigned U.S. patent application entitled “Network Address Management for a Wired Network Supporting Wireless Communication to a Plurality of Mobile Users”, Ser. No. 07/605,592, filed Oct. 29, 1990, by C. E. Perkins. As such, the gateway **18** may include components for maintaining and allocating pseudo-IP addresses to the MHs **10**, as described in the above referenced commonly assigned U.S. patent application Ser. No. 07/605,592.

It is also assumed that each MH **10** has at least one server, referred to herein as a Mobile Router (MR) **20**. The functionality of the MR **20** is included within the gateway **18**, and is shown as such in FIG. 2. The MR **20** serves two functions.

Firstly, the MR **20** is used as a “proxy” for “permanent addresses” assigned to MHs **10**. In other words, the MR **20** advertises reachability (via routing protocols) to the addresses of the MHs **10** for which it acts as a proxy. It should be noted that such an advertisement has no implications on the actual Level 2 subnetworks that the MR **20** is attached to.

Secondly, as a MH **10** moves between different Level 2 subnetworks, the MR **20** that is currently acting as a proxy for that MH **10** is informed of the MHs **10** location, via the BAS **12**, as described in detail below.

For redundancy, a MH **10** may have more than one MR **20** associated therewith. Thus, there is no restriction implied or intended as to the number of MRs **20** associated with a single MH **10**, although the MH **10** will own but a single IP address.

It is also assumed herein that a given MH **10** can ascertain a Network Layer address of the MR **20** that serves as its proxy by way of, by example only, a MH system management function, or by a static configuration.

From a network layer routing point of view, a given MH **10** is always considered to be attached to the one of the Level 2 subnetworks as defined by the permanent IP address or pseudo-IP address of the MH **10**. An MR **20** associated with a given MH **10** functions as the closest network layer gateway to the MH **10**.

Each MH **10** maintains its network connectivity by accepting service from the BAS **12** that transmits data to and receives data from the mobile host during the

time that the MH **10** is within a coverage area served by the BAS **12**. For example, if the wireless communication occurs in the infrared (IR) frequency spectrum, a BAS **12** would be limited to servicing mobile hosts within infrared range of the BAS **12**. As was noted above, the area served by a BAS **12** is referred to as a “cell” and cells served by different BASs **12** may overlap. As seen in FIG. 2, a given MH **10** may be positioned within an overlap between two or more cells **11** and, as a result, in some embodiments of the invention may be serviced by each of the associated BASs **12**. However, service by more than one BAS **12** is not required and, thus, the teaching of the invention may also be practiced in a wireless network wherein only one BAS **12** may be assigned responsibility, at any given time, for a MH **10**.

It is also assumed that a given MH **10** can ascertain the network layer address of a BAS **12** (or one of the interfaces of the BAS **12**) serving the current cell wherein the MH **10** is located, and that a BAS **12** can ascertain network layer addresses of all operational MHs **10** within the cell served by the BAS **12**. Both of these assumptions are based on the bidirectional communications capability that exists between a BAS **12** and the MH(s) **10** with the associated cell **11**.

As the MH **10** moves or migrates, the set of the BAS(s) **12** that the MH **10** can reach within a Level 2 subnetwork (within a single network layer hop) may change. As employed herein, a network layer hop is considered to be a transmission between two network entities without involving intermediate network entities. The method of the invention requires that a MH **10** notify the associated MR(s) **20** of such changes by supplying the MR(s) **20** with the address(es) of the BAS(s) **12** that are currently reachable by the MH **10**. The MR **20** maintains this information within a MH/BAS MAP **20a**, wherein the IP address of each MH **10** is associated with one or more IP addresses of the BASs **12**.

An aspect of this invention is the use of an IP feature known as a Loose Source Record Routing, or Loose Source Routing (LSR) option. By exploiting the LSR option in a novel fashion within the context of a wireless network having migrating MHs **10**, the invention enables a packet from a source host to bypass the MR **20** and to be routed instead directly to the BAS **12** that serves the MH **10** that is the destination for the packet.

As seen in FIG. 3a, the contents of an internet datagram header includes an OPTIONS field. Options may or may not appear in a datagram. What is optional is their transmission in any particular datagram, not their implementation. That is, the OPTIONS must be implemented by all IP modules (hosts and gateways). The option type of particular relevance herein is the LSR option, which is used to route an internet datagram based on information supplied by the source of the datagram.

Referring to FIG. 3b, the LSR option provides a means for the source of an internet datagram to supply routing information to be used by the gateways in forwarding the datagram to the destination, and to record the route information.

The LSR option begins with the option type code (**131**). The second octet is the option length, the length including the option type code, the length octet, a Pointer octet, and length-3 octets of Routing Data. The third octet is the Pointer into the Route Data, and which indicates the octet which begins the next source address to be processed. The Pointer is relative to this

option, and the smallest legal value for the Pointer is four.

The Route Data is composed, typically, of a series of internet addresses. Each internet address is 32 bits, or four octets. If the Pointer is greater than the length, that is the Pointer points beyond the last address entry in the Route Data, the source route is empty (and the recorded route full) and the routing is then based on the datagram Destination Address Field (FIG. 3a).

If the address in the Destination Address Field has been reached, and the Pointer is not greater than the length, the next address in the source route replaces the address in the Destination Address Field, the recorded route address replaces the Source Address just used, and the Pointer is increased by four.

The recorded route address is the internet module's own internet address, as known in the environment into which this datagram is being forwarded.

The procedure of replacing the source route with the recorded route (though it is in the reverse of the order required to be used as a source route) means that the LSR option (and the IP header as a whole) remains a constant length as the datagram progresses through the internet.

This option is referred to as a Loose Source Route because the gateway 18 (MR 20) or host IP is allowed to use any route of any number of intermediate gateways to reach the next address in the route.

As employed by the invention, the LSR option includes a list of network layer addresses of the BAS(s) 12 serving the current location of the MH 10, and employs the Pointer to point to one of the addresses. It is a specified requirement of the Network Layer Protocol that responses to packets that use the LSR option be delivered along the reverse of the path specified by the packet initiating the response (e.g. the packet from the MH 10). Thus, it follows that forwarding of the response packets is determined by the addresses specified by the LSR option, i.e. the addresses of the BAS(s) 12 of the cell(s) presently serving the MH 10. As a result of the use of the LSR option, a single IP header conveys information both about the identity of the MH 10, in the form of the Destination Address of the packet, as well as about the topological location of the MH 10, in the form of the BAS 12 address entry or entries of the LSR option Route Data fields.

When one source host sends a Network Layer Protocol Data Unit (NPDU) to another host (destination), and if the destination is the MH 10, then there exists only one of two possible conditions:

- a) the source host is initiating a data stream to the destination, with no previous knowledge of the destination host's whereabouts; or
- b) the source host is sending data to the destination in response to some previous transmission by that destination host.

In the case (a), and as seen in the flow chart of FIG. 4, the NPDU is delivered in accordance with conventional IP routing from the source host to the MR 20 which has previously advertised reachability to the destination address (Block A). Upon receipt of the NPDU, the MR 20 makes a determination if the NPDU has a LSR option (Block B). If NO, and in that the MR 20 maintains a record of the network layer address(es) of the BAS(s) 12 that serve the cell(s) that the destination MH 10 is presently located (BAS/MH MAP 20a), the MR 20 creates an empty LSR option in the NPDU (Block C). That is, an LSR option is created wherein

the Route Data is empty, and wherein the Pointer points to the first (null) Route Data address.

Control then passes to Block D, as it also does if the determination at Block B is YES. That is, if YES at Block B, the incoming NPDU already contains a Loose Source Route option.

The MR 20 operates at Block D to first append the content of the IP header Destination Address field to the end of the LSR option Route Data. At this point, the Destination Address is the address of the MH 10 to which the NPDU is directed. The MR 20 then increases the LSR option Length accordingly. Next, the MR 20 accesses the MH/BAS Map 20a to retrieve the address of a BAS 12 that is responsible for the MH 10 to which the NPDU is directed. The address of the responsible BAS 12 is then inserted into the Destination Address of the NPDU.

At Block E the MR 20 forwards the NPDU, in accordance with the address in the Destination Address field, to the BAS 12 that is currently responsible for the MH 10 to which the NPDU is directed.

It is noted that for condition (a) the LSR option will typically not be on. However, if the datagram is first sourced by another MH 10, and assuming that, in accordance with the invention, all MH 10 sourced datagrams are always sent with the LSR option turned on, then the YES condition at Block B will be satisfied.

In the case (b) the source host will have previously received a packet from the destination MH 10 specifying the return path to the destination MH 10, using the Loose Source Routing option of the Network Layer Protocol. The network will (as required) attempt to deliver the packet to the destination, which was taken from the next address in the list of addresses specified with the LSR option. The destination address will, in accordance with the invention, correspond to a BAS 12 that is presently serving the destination MH 10.

As a MH 10 moves from one Level 2 subnetwork to another, its current cell and, thus, BAS 12, also changes. If the traffic between a pair of hosts is bidirectional, then the recipient of an NPDU directly notifies the originator of the NPDU of its movement by inserting the address of the new BAS 12 as the Destination Address. This occurs when the recipient host sends NPDUs back to the originating host. According to the "Host Requirements", specified by "Requirements for Internet Hosts Communications Layers", by R. Braden, RFC 1122 (10/89), when a host receives a Network Layer NPDU destined to it with the LSR option in the NPDU, the host is required to reverse the Source Route and use it to form the return source route for reply datagrams. This operation is specified in detail in Section 3.2.1.8 of the above referenced RFC 1122.

That is, and in accordance with this path reversal technique that is employed by the teaching of the invention, if a host receives a datagram containing a completed source route, i.e. the pointer points beyond the last address field, the datagram is considered to have reached its final destination. The source route option (recorded route), as received, is passed up to the Transport layer, or to ICMP message processing. The recorded route is reversed and is used to form a return source route for reply datagrams. When the return source route is built, it is correctly formed even if the recorded route included the source host.

Furthermore, and as is specified in Section 3.2.1.8 of the RFC 1122, if a source-routed datagram is fragmented, each fragment will contain a copy of the source

route. Since the processing of IP options (including a source route) must precede reassembly, the original datagram will not be reassembled until the final destination is reached.

For example, a source routed datagram is to be routed from a source (S) host to a destination (D) host via gateways G1, G2, . . . Gn. An ambiguity may exist in the specification as to whether the source route option in a datagram sent out by S should be (A) or (B): (A): {>>G2, G3, . . . Gn, D} (correct), or (B): {S, >>G2, G3, . . . Gn, D} (incorrect), where >> represents the Pointer. If (A) is sent, the datagram received at D will contain the option: {G1, G2, . . . Gn >>}, with S and D as the IP source and destination addresses, respectively. If (B) were sent, the datagram received at D would again contain S and D as the IP source and destination addresses, but the option would be: (S, G1, . . . Gn >>), i.e., the originating host would be the first hop in the route.

In accordance with the teaching of the invention, when a MH 10 moves from one Level 2 subnetwork to another, while communicating with another host, only the first NPDU sent to the MH 10 would go through the MR 20 that acts as the proxy for the MH 10. The remainder of the NPDUs then flow instead to the MH 10 directly through the BAS 12 currently servicing the MH 10, as indicated by the LSR Option.

In accordance with the invention the use of the LSR Option, in effect, produces a "shortcut routing" between a pair of hosts. The shortcut routing significantly improves the optimality of routing between hosts, when at least one of the hosts is mobile and does not have a fixed point of connection to the network.

As the MH 10 moves, the cell and BASs 12 to which it is connected through, by example, the bidirectional IR wireless link, also changes. In the process of moving from one cell to another it is possible for a MH 10 to be located in more than one cell, such as the MH 10' of FIG. 2, and to thus be able to receive service from more than one BAS 12. To facilitate shortcut routing the MH 10 may, but is not required to, insert the addresses of all the relevant BASs into the Loose Source Route option. The order in which these addresses are specified in the Loose Source Route option may be irrelevant, or may be based on criteria such as signal strength, with the BAS 12 having the highest signal strength being listed first.

As a result, it is typically unnecessary to involve the MR(s) 20 that act as proxies for the MH 10 as the MH 10 moves. This technique yields even a greater optimality in routing. It should be noted, however, that the MH 10 must inform the MR 20 of any changes in the identities of the BASs that are serving the MH 10, in that such changes amount to changes to the topology of the Level 2 subnetwork containing the "permanent" address of the MH 10.

By supplying the associated BAS 12 address in each reply NPDU, a MH 10 informs the originator of the packet traffic about its most current network location. By using the address of the BAS 12 recipient (supplied in the reply NPDUs), the originator of the traffic is enabled, in effect, to track the most current location of the recipient of the traffic. Such tracking enables the construction of optimal routes between a pair of hosts, using the facilities of the IP, while at the same time minimizing the involvement of the MR(s) 20.

As seen in the flowchart of FIG. 5, when a BAS 12 receives an NPDU with the LSR option (Block A), the BAS 12 first examines the last LSR option address data entry, i.e. the address of the mobile host 10. The BAS 12 then determines at Block B whether the addressed MH 10 is "up" (i.e. currently located within the cell served by the BAS 12). If the MH 10 is not up, the BAS 12 deletes its own IP address from the Destination Address field (Block C). The BAS 12 then determines at Block D if there are any intermediate LSR option addresses before the last LSR option address of the MH 10. If NO, the BAS 12 forwards the NPDU to the MR 20, and processing continues in accordance with the flowchart of FIG. 4. If YES at Block D, the BAS 12 advances the Pointer in the LSR option to the next address entry (Block F) and proceeds to forward the NPDU to the next entry in the LSR option (Block G).

If the MH 10 is up (YES at Block B), the BAS 12 processes the LSR option by inserting its Destination Address into the LSR option Route Data at the current Pointer location. The BAS 12 also takes the last Route Data address, that is the address of the MH 10, and puts the MH 10 address into the Destination Address. The BAS 12 advances the Pointer beyond the end of the LSR option (Block H), and delivers the NPDU directly to the MH 10 that is specified in the datagram header Destination Address (Block I). If the BAS 12 advances the pointer beyond the last entry in the LSR, the forwarding of the NPDU is directly to that MH 10; otherwise the NPDU will be forwarded to the MR 20 associated with the destination MH 10. This occurs because the BAS 12 forwards the NPDU to the appropriate gateway, just as any agent would forward the packet to the MR 20, and all gateways have a record of the routing information advertised by the MR 20 for the address of the MH(s) served by the MR 20.

Receiving an NPDU with a MH 10 does not require any special processing, other than advancing the pointer beyond the end of the LSR option (if required) prior to any further processing so as to conform to the LSR option specification. In addition, for every incoming packet the MH 10 stores the portion of the LSR Option that does not have addresses of the BAS(s) 12 directly reachable by the MH 10. Such a LSR Option fragment may indicate, for example, the BAS(s) 12 that serve the originator of the packet if the originator is also a MH 10, or any host using the LSR option for any reason.

Sending an IP packet to another host involves constructing a LSR option as a concatenation of the BAS(s) 12 directly reachable by the MH 10, followed by the LSR extracted from the packet(s) received from that host (if any). The LSR pointer is set to point beyond the list of the BAS(s) 12 directly reachable by the MH 10, in that there is no need for the MH 10 to route the packet to the BAS(s) that serve the MH 10. This constructed LSR option is then inserted within the outgoing packet.

Some previous approaches for forwarding NPDUs between a pair of hosts attached to different Level 2 subnetworks involve datagram encapsulation by the MRs 20 and BASs 12. Moreover, at least one of these approaches ("IP-based Protocols for Mobile Internetworking", Ioannidis, J., Duchamp, D., Maguire, G., Proceedings SIGCOMM 1991) requires ubiquitous knowledge of the actual location of all the mobile hosts within a routing domain. That knowledge, in turn, may require significant information exchange between all the BASs 12 within the routing domain. For inter-

domain connectivity the routing always involves a BAS 12 located in the "home" domain of a mobile host.

The present invention provides advantages over these previous approaches. Firstly, it does not require flooding the network with information, about actual 5 locations of mobile hosts, to all the BASs 12 within a domain, as in the immediately above referenced proposal, in that for each MH 10 only the associated MR 20 has to maintain the location information, in the form of a MH 10, BAS 12 mapping. 10

Secondly, in many cases routes between MH 10s that involve inter-domain mobility are likely to be shorter than the routes obtained with above referenced proposal.

Thirdly, by restricting the knowledge of MH 10 15 movements to only the MH 10 itself, and the MR 20 of that MH 10, the teaching of the invention significantly simplifies security implications and authentication requirements.

Fourthly, by avoiding datagram encapsulation the invention avoids potentially detrimental performance implications associated with the fragmentation that may be necessary to accomplish encapsulation. 20

Fifthly, by avoiding datagram encapsulation the teaching of the invention reduces the amount of protocol information carried by NPDUs. 25

Finally, by avoiding datagram encapsulation the method of the invention reduces the overhead that otherwise would be imposed on BASs 12. That is, for most NPDUs the BASs 12 function as pure IP routers, 30 without any knowledge of whether the NPDUs are destined to mobile or non-mobile hosts.

While the invention has been particularly shown and described with respect to a preferred embodiment thereof, it will be understood by those skilled in the art 35 that changes in form and details may be made therein without departing from the scope and spirit of the invention.

Having thus described our invention, what we claim as new, and desire to secure by Letters Patent is:

1. A method for routing a packet of information between a first host and a second host that are coupled to a network, each of the hosts having a unique network address, wherein at least one of the hosts is a mobile host that does not have a fixed network coupling location, comprising the steps of: 40

transmitting a packet from the mobile host to the second host on the network through a wireless link that is established between the mobile host and a first base access station that serves a current physical location of the mobile host, the first base access station being coupled to the network via a subnetwork, the packet including a first Loose Source Routing LSR option that includes a network address of the first base access station and possibly at least one further network address; 50

receiving with the second host the packet that includes the first LSR option; and

transmitting a further packet from the second host to the mobile host, via the first base access station and the wireless link, in accordance with a path reversal technique such that the further packet includes a second LSR option that includes the network address of the first base access station that was included within the first LSR option, whereby the further packet is directed through the network to the first base access station that serves the current physical location of the mobile host. 65

2. A method as set forth in claim 1 and further comprising the steps of:

in response to the mobile host establishing a wireless link with a second base access station on the same or a different subnetwork,

transmitting the network address of the second base access station from the mobile host to a mobile router that is coupled between the subnetwork and the network, the transmission including the network address of the mobile host; and

maintaining, with the mobile router, the network address of the mobile host and the network address of the second base access station.

3. A method as set forth in claim 2 and including a step of transmitting to the network from the mobile router the network address of the mobile router and the network address of the mobile host.

4. A method as set forth in claim 2 wherein, in response to a reception of a packet by the mobile router from the network, the packet not including an LSR option and having the network address of the mobile host as a destination address, the method includes the steps of:

converting the received packet into a packet that includes a newly created LSR option, the newly created LSR option being provided by the mobile router with the network address of the mobile host, the received packet further being provided with a destination address of a base access station that is serving the physical location within which the mobile host is currently located;

forwarding the converted packet from the mobile router to the base access station that serves the physical location within which the mobile host is currently located; and

receiving the converted packet and forwarding the received converted packet from the base access station over the wireless link to the mobile host.

5. A method as set forth in claim 1 wherein the step of transmitting the further packet includes the steps of:

receiving the further packet with the first base access station; and

determining if the mobile host is currently located within the physical area served by the first base access station;

if it is determined that the mobile host is currently located within the physical area served by the first base access station, forwarding the further packet from the first base access station over the wireless link to the mobile host; else

if it is determined that the mobile host is not currently located within the physical area served by the first base access station, and if it is determined that the second LSR option does not include a network address other than the network address of the mobile host, forwarding the further packet from the first base access station over the subnetwork to a mobile router that is coupled between the subnetwork and the network; and

forwarding the further packet from the mobile router to a second base access station that serves a physical location within which the mobile host is currently located.

6. A method as set forth in claim 5 wherein, if it is determined that the mobile host is not currently located within the physical area served by the first base access station, the step of forwarding the further packet from the first base access station over the subnetwork to the

mobile router includes an initial step of deleting the address of the first base access station from the further packet.

7. A method as set forth in claim 5 wherein, if it is determined that the mobile host is currently located within the physical area served by the first base access station, the step of forwarding the further packet from the first base access station over the wireless link to the mobile host includes an initial step of advancing an LSR option address pointer beyond a last LSR option address.

8. A method as set forth in claim 5 wherein, if it is determined that the mobile host is not currently located within the physical area served by the first base access station, the method includes a step of:

deleting the network address of the first base access station from the further packet; and, if it determined that the LSR option includes a further network address other than the network address of the mobile host, the method further includes the steps of:

advancing a LSR option address pointer to the next, further network address; and

forwarding the packet to the next network address.

9. A method as set forth in claim 1 wherein the wireless link is an infrared radiation link.

10. A method for routing a packet of information between two hosts that are coupled to a network, each of the hosts having a unique network address, wherein at least one of the hosts is a mobile host that does not have a fixed network coupling location, comprising the steps of:

in response to a reception of a packet from the network by a mobile router, the packet not including an LSR option and having the network address of the mobile host as a destination address,

converting the received packet into a packet that includes a LSR option, the LSR option being provided by the mobile router with the network address of the mobile host, the received packet further being provided with a destination address of a base access station that is serving, with a wireless communication link, a physical location within which the mobile host is currently located;

forwarding the converted packet from the mobile router to the base access station that serves the physical location within which the mobile host is currently located; and

receiving the converted packet and forwarding the received converted packet from the base access station over the wireless link to the mobile host.

11. A method as set forth in claim 10 wherein the step of receiving the converted packet includes the steps of: determining if the mobile host is currently located within the physical area served by the base access station;

if it is determined that the mobile host is currently located within the physical area served by the base access station, forwarding the converted packet from the base access station over the wireless link to the mobile host; else

if it is determined that the mobile host is not currently located within the physical area served by the base access station, and if it is determined that the LSR option does not include a network address other than the network address of the mobile host, forwarding the converted packet from the base access station to the mobile router; and

forwarding the converted packet from the mobile router to another base access station that serves a physical location within which the mobile host is currently located.

12. A method as set forth in claim 10 wherein the wireless link is an infrared radiation link.

13. A method as set forth in claim 10 wherein the network address and the destination address are Internet addresses.

14. A method as set forth in claim 10 wherein the steps of receiving and forwarding are accomplished using an Internet Protocol.

15. A mobile host having a network address, the mobile host comprising:

means for bidirectionally communicating with a data communications network through a first base access station over a wireless communications link, the first base access station serving a wireless communications cell that encompasses a current physical location of the mobile host; and

means for transmitting an information packet to the data communications network, the information packet being transmitted over the wireless communications link to the first base access station, the information packet including a first Loose Source Routing LSR option that includes a network address of the first base access station.

16. A mobile host as set forth in claim 15 and further comprising means for receiving an information packet from the data communications network, the information packet being received from the first base access station over the wireless communications link, the received information packet including a second LSR option that includes the network address of the first base access station.

17. A mobile host as set forth in claim 15 and further comprising means, responsive to the mobile host establishing a wireless communications link with a second base access station, for transmitting the network address of the second base access station from the mobile host to a mobile router that is coupled to the network, the transmission including the network address of the mobile host.

18. A mobile host as set forth in claim 15 wherein said means for bidirectionally communicating includes means for transmitting and receiving infrared radiation signals.

19. A data communications system for routing a packet of information over a network, comprising:

at least one mobile host having a network address, said at least one mobile host not having a fixed network coupling location, said at least one mobile host including means for bidirectionally communicating with a wireless communications link;

at least one base access station that is coupled to the network and that serves a communication cell by means of the wireless communication link; and

a mobile router having means for transmitting information packets to the network and means for receiving information packets from the network, said mobile router further including means, responsive to a reception of an information packet from the network, the packet not including a Loose Source Routing LSR option and having a network address of a mobile host as a destination address, for converting the received information packet into a converted packet that includes a LSR option, said converting means providing the LSR option with

5,442,633

15

the network address of the mobile host and also providing the converted packet with a destination address of a selected base access station that is serving, through the wireless communication link, a communication cell within which the mobile host was last known to be located.

20. A data communications system as set forth in claim 19 wherein said mobile router further includes

16

means for forwarding the converted packet to said selected base access station having the destination address, and wherein said selected base access station includes means for receiving the converted packet and means for transmitting the received packet over the wireless communication link to the mobile host.

* * * * *

10

15

20

25

30

35

40

45

50

55

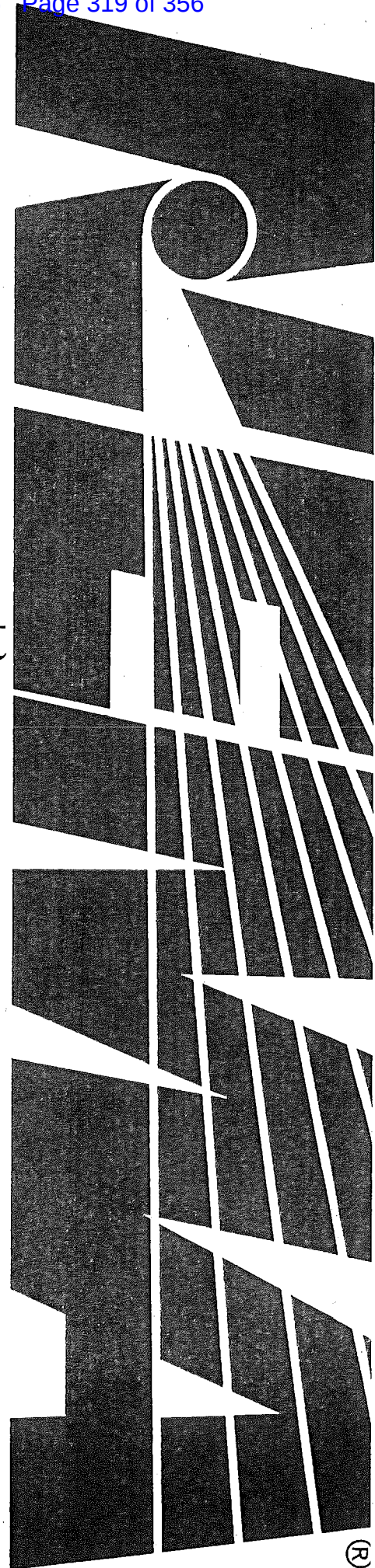
60

65

NEMA Standards Publication

ANSI C12.18-1996 (R2002)

Protocol Specification for ANSI Type 2 Optical Port





ANSI C12.18-1996 (R2002)

American National Standard
Protocol Specification for ANSI Type 2 Optical Port

Secretariat:

National Electrical Manufacturers Association

Approved as an
American National Standards Institute, Inc.

NOTICE AND DISCLAIMER

The information in this publication was considered technically sound by the consensus of persons engaged in the development and approval of the document at the time it was developed. Consensus does not necessarily mean that there is unanimous agreement among every person participating in the development of this document.

The National Electrical Manufacturers Association (NEMA) standards and guideline publications, of which the document contained herein is one, are developed through a voluntary consensus standards development process. This process brings together volunteers and/or seeks out the views of persons who have an interest in the topic covered by this publication. While NEMA administers the process and establishes rules to promote fairness in the development of consensus, it does not write the document and it does not independently test, evaluate, or verify the accuracy or completeness of any information or the soundness of any judgments contained in its standards and guideline publications.

NEMA disclaims liability for any personal injury, property, or other damages of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, application, or reliance on this document. NEMA disclaims and makes no guaranty or warranty, express or implied, as to the accuracy or completeness of any information published herein, and disclaims and makes no warranty that the information in this document will fulfill any of your particular purposes or needs. NEMA does not undertake to guarantee the performance of any individual manufacturer or seller's products or services by virtue of this standard or guide.

In publishing and making this document available, NEMA is not undertaking to render professional or other services for or on behalf of any person or entity, nor is NEMA undertaking to perform any duty owed by any person or entity to someone else. Anyone using this document should rely on his or her own independent judgment or, as appropriate, seek the advice of a competent professional in determining the exercise of reasonable care in any given circumstances. Information and other standards on the topic covered by this publication may be available from other sources, which the user may wish to consult for additional views or information not covered by this publication.

NEMA has no power, nor does it undertake to police or enforce compliance with the contents of this document. NEMA does not certify, test, or inspect products, designs, or installations for safety or health purposes. Any certification or other statement of compliance with any health or safety-related information in this document shall not be attributable to NEMA and is solely the responsibility of the certifier or maker of the statement.

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer.

Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that a concerted effort be made toward their resolution.

The use of American National Standards is completely voluntary; their existence does not in any respect preclude anyone, whether he has approved the standards or not, from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the standards.

The American National Standards Institute does not develop standards and will in no circumstances give an interpretation of any American National Standard. Moreover, no person shall have the right or authority to issue an interpretation of an American National Standard in the name of the American National Standards Institute. Requests for interpretations should be addressed to the secretariat or sponsor whose name appears on the title page of this standard.

CAUTION NOTICE: This American National Standard may be revised or withdrawn at any time. The procedures of the American National Standards Institute require that action be taken periodically to reaffirm, revise, or withdraw this standard. Purchasers of American National Standards may receive current information on all standards by calling or writing the American National Standards Institute.

Published by

National Electrical Manufacturers Association
1300 N. 17th Street, Rosslyn, Virginia 22209

Approved by ANSI, April 8, 1996

Copyright © 1996 National Electrical Manufacturers Association
All rights reserved

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without prior written permission of the publisher.

Printed in the United States of America

CONTENTS

Page

TABLE OF CONTENTS

1	Scope	1
2	References.....	1
3	Definitions	1
	3.1 Point-to-point communications.....	1
	3.2 Table	1
	3.3 Document syntax	1
4	Protocol details	2
	4.1 Order of transmission.....	3
	4.2 Layer 7—Application layer	3
	4.3 Layer 6—presentation layer.....	11
	4.4 Layer 5—session layer.....	11
	4.5 Layer 4—transport	12
	4.6 Layer 3—network layer	12
	4.7 Layer 2—data link	12
	4.8 Layer 1—physical.....	15
	ANNEX A—PROTOCOL SYNTAX Informative Protocol Syntax.....	21
	A.1 Order of transmission syntax	21
	A.2 Layer 7 syntax.....	21
	A.3 Layer 2 syntax.....	25
	ANNEX B—Communication example (Layer 7 & Layer 2)	27
	ANNEX C—Packet transmission example.....	29
	ANNEX D—Service sequence state control	31

ANSI C12.18-1996

Foreword

(This foreword is not part of American National Standard for Protocol Specification for ANSI Type 2 Optical Port, ANSI C12.18-1995.)

This American National Standard provides an open platform communications protocol for two-way communication with an electronic metering device or an electromechanical metering device with an added electronic board. The communications is specified to pass through an ANSI Type 2 Optical Port. The protocol is written to conform to the OSI seven layer stack.

Suggestions for improvement to this standard are welcome. They should be sent to:

National Electrical Manufacturers Association
Vice President of Engineering
1300 North 17th Street
Suite 1847
Rosslyn, VA 22209

This standard was processed and approved for submittal to ANSI by Accredited Standards Committee for Electricity Metering, C12. At the time the committee approved this standard, the C12 Committee had the following members:

R. S. Turgel, **Chairman**
Christopher F. Merther, **Secretary**

- Cruz Gomez
- H. Jones
- Lauren Pananen
- Timothy Vahlstrom
- Clark Smith
- James Mining
- Joe Blackmer
- John McEvoy
- Dan McAuliff
- Herman Millican
- Tom Drew
- Ray Stevens
- Francis Marta
- James Schlatter
- John Lauletta
- Warren Germer
- Edmund Hoffman
- Ahn Mai
- Ralph Fahmy

Subcommittee 17 that developed the standard consisted of:

- Lynnda K. Ell, **Chairwoman**
- John Lauletta, **Vice Chairman**
- Michael Anderson
- Ellen Edge
- Lynnda Ell
- Bill Gibson
- Bruce Johnson
- Larry Kotewa
- Tempe Lampe
- Avy Moise
- Terry Penn

Subcommittee 17 (continued)

Wes Ray
Clark Smith
Chris Stanfield
George Stephens
Paul Taylor
Kurt Wiechert
Michelle Veillette

STD-NEMA C12.18-ENGL 1996 ■ 6470247 0514328 211 ■

ANSI C12.18-1996

Protocol Specification for ANSI Type 2 Optical Port**1 Scope**

This document details the criteria required for communications with an electric power metering device by another device via an optical port. The other device could be a handheld reader, a laptop or portable computer, a master station system, a power metering device, or some other electronic communications device.

This document provides details for a complete implementation of an OSI 7-layer model.

The protocol specified in this document was designed to transport data in table format. The table definitions are referenced in Clause 2.

2 References

ISO 7498/1, *OSI Reference Model*

ISO 3309-1991: *Information technology—Telecommunications and information exchange between systems - High-level data link control (HDLC) procedures*

3 Definitions

For the purposes of this document, the following definitions are made for terms and syntax used throughout this document.

3.1 Point-to-point communications

Point-to-point communications is defined as communication between two devices through a single optical interface.

3.2 Table

Functionally related data elements, grouped together into a single data structure for transport

3.3 Document syntax

Describing data definitions is usually accomplished within the confines of a given language and the grammar rules of that language. Since the data definitions embodied within this document are meant to be independent of a specific language and, hopefully, capable of being implemented within the confines of any language, a method for describing the data definitions must be developed. A modified form of the Backus-Naur Format (BNF) will serve as the basis for building the descriptions used to construct the data definitions.

ANSI C12.18-1996

The modified form of BNF has the following definitions:

Symbol	Meaning
< >	A string contained inside angle brackets is called a non-terminal. That is, while it may be viewed as a single unit it can and should be redefined as consisting of one or more simpler elements.
::=	This symbol is read as "is defined as." The non-terminal which occurs on the left hand side (LHS) of this symbol consists of the elements (non-terminals, terminals, or a combination of the two) found on the right hand side (RHS). A line containing an LHS, ::=, and an RHS is known as a production rule.
	The vertical bar is an "OR" symbol. The OR symbol always occurs on the right hand side of a production where the left hand side can be defined in more than one way. The OR bar separates valid alternative right hand sides.
[]	A symbol enclosed in square brackets is optional. The production is valid whether or not it is included.
•	The superscript asterisk is known as the Kleene star. A symbol followed by the Kleene star may occur zero or more times without violating the grammar.
+	The superscript plus sign is known as the Kleene cross. A symbol followed by the Kleene cross must occur one or more times.
+n	A symbol followed by the Kleene cross and any number "n" represents "n" occurrences of the symbol.
{ }	The curly braces are used to enclose comments within the descriptions. Comments have no impact on the productions.

4 Protocol details

Following the guidelines established by the OSI seven layer model, the protocol described in this document provides three functions. These are:

- 1) modification of the communication channel
- 2) transport of information to and from the metering device
- 3) orderly closure of the communication channel when communications are complete

Three layers are used to provide these communication capabilities. These are the Physical, Data Link and Application layers.

With the default conditions established by this document, the communication channel is considered available once the physical connection has been completed. The identification service is required to establish the protocol version and revision in use. Certain communication parameters may be modified through the use of the negotiate service in the Application layer. This service is optional and if not implemented in the metering device or not used during actual communications, the communication channel parameters will remain at the default settings specified by this document. Device implementors are strongly encouraged to implement this optional Application service.

Once the data link is established, the application layer provides logon, security and logoff services for session activation, access control and deactivation; read and write services for issuing data transmission requests; terminate service for shutdown of the communication channel; and a response structure that provides information regarding the success or failure of the service requests.

An example of a typical communications session would consist of the following services with appropriate responses, in the order listed: identification; negotiate; logon; security; read (zero or more); write (zero or more); and terminate. Note that this brief example does not detail the packet structure nor other aspects of the protocol. For a more detailed example of a typical communications session reference Annex B, Communication example.

4.1 Order of transmission

Within the syntax definitions, multiple parameters shall be transmitted in the order as shown, from left to right.

Parameters in all layers within the protocol definition are transmitted most significant byte first. The order of transmission of data field parameters within tables are dictated by the table structure.

Bytes are transmitted in frames. Each frame consists of a start bit, followed by a <byte>, and ending with a stop bit. The start bit is transmitted first.

Bits within each byte are transmitted least significant bit first with the least significant bit identified as bit 0 and most significant bit as bit 7.

```

<word24> ::= <msbyte> <byte> <lsbyte>
<word16> ::= <msbyte> <lsbyte>

<msbyte> ::= <byte> {most significant byte}
<lsbyte> ::= <byte> {least significant byte}

<byte> ::= <bit0> <bit1> <bit2> <bit3> <bit4> <bit5> <bit6> <bit7>
    
```

4.2 Layer 7—application layer

The application layer provides a minimal set of services and data structures required to support electronic metering devices for purposes of configuration, programming and information retrieval.

4.2.1 Data structure

This protocol shall transport table structures. The table specifications this standard was designed to transport are referenced in Clause 2.

4.2.2 Language—protocol specifications for electric metering (“PSEM”)

The language PSEM has been designed to provide an interface between the metering device and any other device over a point-to-point communication medium. The PSEM language consists of 9 services. Each service consists of a request and a response. Each of these requests and responses is described in following service clauses.

```

<requests> ::= <ident> | { Identification request }
              <read> | { Read request }
              <write> | { Write request }
              <logon> | { Logon request }
              <security> | { Security request }
    
```

ANSI C12.18-1996

	<logoff>	{ Logoff request }
	<negotiate>	{ Negotiate request }
	<wait>	{ Wait request }
	<terminate>	{ Terminate request }
<responses>	::= <ident_r>	{ Identification response }
	<read_r>	{ Read response }
	<write_r>	{ Write response }
	<logon_r>	{ Logon response }
	<security_r>	{ Security response }
	<logoff_r>	{ Logoff response }
	<negotiate_r>	{ Negotiate response }
	<wait_r>	{ Wait response }
	<terminate_r>	{ Terminate response }

4.2.2.1 Request codes

PSEM requests always include a one byte request code. Code numbers are assigned as follows:

00H-1FH	Codes shall not be used to avoid confusion with response codes
20H-7FH	Codes are available for use within optical port protocol
80H-FFH	Codes shall be reserved for protocol extensions

4.2.2.2 Response codes

PSEM responses always include a one byte response code. These codes are defined as follows:

<nok>	::= <err> <sns> <isc> <onp> <iar> <bsy> <dnr> <dlk> <no> <isss>	
<ok>	::= 00H	{Acknowledge—No problems, request accepted.}
<err>	::= 01H	{Error—This code is used to indicate rejection of the received service request. The reason for the rejection is not provided.}
<sns>	::= 02H	{Service Not Supported—This application level error response will be sent to the device when the requested service is not supported. This error indicates that the message was valid, but the request could not be honored.}
<isc>	::= 03H	{Insufficient Security Clearance—This application level error indicates that the current authorization level is insufficient to complete the request.}
<onp>	::= 04H	{Operation Not Possible—This application level error will be sent to the device which requested an action that is not possible. This error indicates that the message was valid, but the message could not be processed. Covers

conditions such as: Invalid Length, Invalid Offset}

<code><iar></code>	<code>::= 05H</code>	{Inappropriate Action Requested—This application level error indicates that the action requested was inappropriate. Covers conditions such as write request to a read only table or an invalid table id.}
<code><bsy></code>	<code>::= 06H</code>	{Device Busy—This application level error indicates that the request was not acted upon because the device was busy doing something else.}
<code><dnr></code>	<code>::= 07H</code>	{Data Not Ready—This application level error indicates that request was unsuccessful because the requested data is not ready to be accessed.}
<code><dlk></code>	<code>::= 08H</code>	{Data Locked—This application level error indicates that the request was unsuccessful because the data cannot be accessed.}
<code><rno></code>	<code>::= 09H</code>	{Renegotiate Request—This application level error indicates that the responding device wishes to return to the ID or base state and re-negotiate communication parameters.}
<code><isss></code>	<code>::= 0AH</code>	{Invalid Service Sequence State—This application level error indicates that the request is not accepted at the current service sequence state. For more information on service sequence states, see Annex D.}
	<code>0BH-1FH</code>	{Codes are currently undefined, but are available for use within optical port protocol}
	<code>20H-7FH</code>	{Codes shall not be used to avoid confusion with request codes}
	<code>80H-FFH</code>	{Codes shall be reserved for protocol extensions}

4.2.2.3 Identification service

This service shall be the first service issued once the physical connection is established. The service returns the version and revision of the protocol. The version is positioned left of the decimal indicating a major change. The revision is positioned right of the decimal indicating a minor change.

Request:

`<ident>` ::= `20H`

Response:

The responses `<err>`, `<bsy>`, and `<isss>` indicate a problem with the received service request.

ANSI C12.18-1996

The response <ok> indicates the identification service request was accepted and the version and revision are included in the response.

```

<ident_r> ::= <err> | <bsy> | <isss> | <ok> <std> <ver> <rev> <rsvd>

<std> ::= <byte> {Code identifying reference standard. The codes
                are defined as follows:
                00H = ANSI C12.18
                01H = For use by Industry Canada
                02H-FFH = Reserved }

<ver> ::= <byte> {Referenced standard version number}

<rev> ::= <byte> {Referenced standard revision number}

<rsvd> ::= <byte> {Reserved for future use. This byte shall be set
                to 00H until further defined.}
    
```

4.2.2.4 Read service

The read service is used to cause a transfer of table data to the requesting device.

Request:

The read request allows both complete and partial table transfers. The retrieval of a portion of a table is possible through the use of both offset-count and index-count methods.

Request codes 30–39, 3E and 3F give access to all possible methods used for table transfer. Request code 30 specifies a complete table transfer. Request codes 31 through 39 specify a partial table transfer using 1 to 9 indices. Request code 3E specifies a default table transfer. Request code 3F specifies a partial table transfer using the offset-count method.

```

<read> ::= <full_read> | <pread_index> | <pread_offset> |
            <pread_default>

<full_read> ::= 30H <tableid>

<pread_index> ::= <3jH> <tableid> <index>+ <count>

<3jH> ::= 31H | { 1 <index> included in request }
          32H | { 2 <index> included in request }
          33H | { 3 <index> included in request }
          34H | { 4 <index> included in request }
          35H | { 5 <index> included in request }
          36H | { 6 <index> included in request }
          37H | { 7 <index> included in request }
          38H | { 8 <index> included in request }
          39H | { 9 <index> included in request }

<pread_default> ::= 3EH { Transfer default table }

<pread_offset> ::= 3FH <tableid> <offset> <count>

<tableid> ::= <word16> { Table identifier }
    
```

<offset> ::= <word24> { Offset into data table in bytes }
 <index> ::= <word16> { Index value used to locate start of data }
 <count> ::= <word16> { Length of table data requested, in bytes }

Response:

Responses of type <nok> indicate a problem with the received service request.

The response <ok> indicates the read service was accepted and the data is part of the response.

<read_r> ::= <nok> | <ok> <table_data>
 <table_data> ::= <count> <data> <cksum>
 <count> ::= <word16> { Length of <data> returned, in bytes }
 <data> ::= <byte>+
 <cksum> ::= <byte> { 2's compliment checksum computed only on the <data> portion of <table_data>. The checksum is computed by summing the bytes (ignoring overflow) and negating the result. }

4.2.2.5 Write service

The write service is issued to transfer table data to the target device.

Request:

The write request allows both complete and partial table transfers. The modification of a portion of a table is possible through the use of both offset-count and index-count methods.

Request codes 40–49 and 4F give access to all possible methods used for table transfer. Request code 40 specifies a complete table transfer. Request codes 41 through 49 specify a partial table transfer using 1 to 9 indices. Request code 4F specifies a partial table transfer using the offset-count method.

<write> ::= <full_write> | <pwrite_index> | <pwrite_offset>
 <full_write> ::= 40H <tableid> <table_data>
 <pwrite_index> ::= <4jH> <tableid> <index>+ <table_data>
 <4jH> ::= 41H | { 1 <index> included in request }
 42H | { 2 <index> included in request }
 43H | { 3 <index> included in request }
 44H | { 4 <index> included in request }
 45H | { 5 <index> included in request }
 46H | { 6 <index> included in request }
 47H | { 7 <index> included in request }
 48H | { 8 <index> included in request }
 49H | { 9 <index> included in request }
 <pwrite_offset> ::= 4FH <tableid> <offset> <table_data>
 <tableid> ::= <word16> { Table identifier }

STD.NEMA C12.18-ENGL 1996 ■ 6470247 0514336 396 ■

ANSI C12.18-1996

<offset>	::=	<word24>	{ Offset into data table in bytes }
<index>	::=	<word16>	{ Index value used to locate start of data }
<table_data>	::=	<count> <data> <cksum>	
<count>	::=	<word16>	{ Length of <data> to be written, in bytes }
<data>	::=	<byte>+	
<cksum>	::=	<byte>	{ 2's compliment checksum computed only on the <data> portion of <table_data>. The checksum is computed by summing the bytes (ignoring overflow) and negating the result. }

Response:

Responses of type <nok> indicate a problem with the received service request.

The response <ok> indicates the write service was successfully completed and the data was successfully transmitted to the device.

```
<write_r> ::= <nok> | <ok>
```

4.2.2.6 Logon service

Logon service establishes a session without establishing access permissions.

Request:

The <user_id> parameter is a code, optionally stored by the metering device, indicating a utility supplied identity of the operator requesting the creation of a session. The <user_id> would be inserted in the Event and History Logs of the tables specification referenced in Clause 2, References, if the logs are supported by the metering device. The <user> field provides a utility supplied name of the operator requesting the access to the device. The logon service is required.

The logon service has the following format:

```
<logon> ::= 50H <user_id> <user>
<user_id> ::= <word16> {User identification code}
<user> ::= <byte>+10 {10 bytes containing user identification}
```

Response:

The responses <err>, <bsy>, <iar>, and <isss> indicate a problem with the received service request.

The response <ok> indicates the logon service was successfully completed and the session was established.

```
<logon_r> ::= <err> | <bsy> | <iar> | <isss> | <ok>
```

4.2.2.7 Security service

The security service is provided for setting access permissions.

Request:

A password is used as a means to select access permissions. This service request may only be sent during a session. The <password> field will be compared with those in the password table of the table specifications referenced in Clause 2, References, if the password tables are supported by the metering device.

```
<security> ::= 51H <password>
<password> ::= <byte>+20 {20 byte field containing password}
```

Response:

The responses <err> <bsy>, and <iss> indicate a problem with the received service request.

The response <ok> indicates the security service was successfully completed and the access permissions associated with the password were granted.

```
<security_r> ::= <err> | <bsy> | <iss> | <ok>
```

4.2.2.8 Logoff service

The logoff service provides for an orderly shutdown of the session established by the logon service.

Request:

Following a logoff service request the communication channel will retain the parameters previously established. The communication channel is terminated by either physical disruption of the channel or by the terminate service.

```
<logoff> ::= 52H
```

Response:

The responses <err> <bsy>, and <iss> indicate a problem with the received service request.

The response <ok> indicates the acceptance of the logoff service and the cessation of the session established by the logon service. Prior to further data transfers with the metering device, the logon service must be reissued.

```
<logoff_r> ::= <err> | <bsy> | <iss> | <ok>
```

4.2.2.9 Negotiate service

The negotiate service provides the mechanism for reconfiguring the communication channel for communication parameters differing from the default values specified in this document.

Request:

ANSI C12.18-1996

This service is initiated by the device communicating with the metering device. It is optional and, if not used, the communication channel operates with the default parameters established by this document. Device implementers are strongly encouraged to provide this service.

<negotiate>	::=	<6jH> <packet_size> <nbr_packet> <baud_rate>*	
<6jH>	::=	60H	{ No <baud rate> included in request. Stay at default baud rate }
		61H	{ 1 <baud rate> included in request }
		62H	{ 2 <baud rate> included in request }
		63H	{ 3 <baud rate> included in request }
		64H	{ 4 <baud rate> included in request }
		65H	{ 5 <baud rate> included in request }
		66H	{ 6 <baud rate> included in request }
		67H	{ 7 <baud rate> included in request }
		68H	{ 8 <baud rate> included in request }
		69H	{ 9 <baud rate> included in request }
		6AH	{ 10 <baud rate> included in request }
		6BH	{ 11 <baud rate> included in request }
<packet_size>	::=	<word16>	{ Maximum packet size supported, in bytes. This value shall not be greater than 8192 bytes. }
<nbr_packet>	::=	<byte>	{ Maximum number of packets this layer is able to reassemble into an upper layer data structure at one time. }
<baud_rate>	::=	00H	{ Externally defined }
		01H	{ 300 baud }
		02H	{ 600 baud }
		03H	{ 1200 baud }
		04H	{ 2400 baud }
		05H	{ 4800 baud }
		06H	{ 9600 baud }
		07H	{ 14400 baud }
		08H	{ 19200 baud }
		09H	{ 28800 baud }
		0AH	{ 57600baud }
			{ 0BH -FFH reserved }

Response:

The responses <err>, <sns>, <bsy>, and <isss> indicate a problem with the received service request and that the communication channel will maintain its current settings.

The response <ok> indicates the service request was accepted and the new settings now apply. The data following the <ok> indicates the setting that will apply upon positive acknowledgement. If the target cannot accept the negotiate request baud rates, the original baud rate will be echoed in the response.

<negotiate_r> ::= <err> | <sns> | <bsy> | <isss> | <ok> <packet_size> <nbr_packet> <baud_rate>

4.2.2.10 Wait service

The wait service is used by the communicating devices to maintain an established communication channel during idle periods thus preventing automatic termination. This temporarily extends the channel traffic time-out to the <time> specified in the request upon acknowledgement of the wait service request. The channel traffic time-out will be reset to the default value once the next valid packet is received by the target.

Request:

<wait> ::= 70H <time>
 <time> ::= <byte> { Suggested wait period in seconds. }

Response:

The responses <err>, <sns>, <bsy>, and <isss> indicate a problem with the received service request and time-out is not extended.

The response <ok> indicates the service request was accepted and the time-out is extended to the value requested.

<wait_r> ::= <err> | <sns> | <bsy> | <isss> | <ok>

4.2.2.11 Terminate service

The terminate service provides for an immediate cessation of the communication channel.

Request:

This service should be used to terminate either partially or fully established communication channels for reasons such as excessive errors, security issues, internal error conditions, end of session, or other reasons as set by the device manufacturer.

<terminate> ::= 21H

Response:

The response <err> indicates a problem with the received service request and the channel remains open.

The response <ok> indicates the service request was accepted and the channel will return to default settings as stated in Clause 4.7.1.1, Default Settings, upon receipt of a positive acknowledgment.

<terminate_r> ::= <err> | <ok>

4.3 Layer 6—presentation layer

4.3.1 Null layer

The end device will not provide queuing capabilities for service requests.

4.4 Layer 5—session layer

4.4.1 Null layer

ANSI C12.18-1996

Communications with the end device over the optical port communications path will be connection oriented and consist of a single session. The session is defined to begin with the acceptance of the logon service and terminates with the acceptance of either the logoff service or the terminate service.

4.5 Layer 4—transport**4.5.1 Null layer****4.6 Layer 3—network layer****4.6.1 Null layer****4.7 Layer 2—data link**

Services of upper layers are transported in one or many packets. Each packet is variable in length but cannot exceed a maximum packet size. The maximum packet size has a default value when the communication channel is opened. The maximum packet size can be changed through the use of the negotiate service.

For each packet received, a positive or negative acknowledgment is sent by the target. This acknowledgment consists of a single byte transmitted outside of the packet structure. If the requester does not receive an acknowledgment before the Response Time-Out, or a negative acknowledgment is received, the same packet is re-transmitted up to 3 times. After the 3rd retry attempt, the requester should assume termination has occurred.

4.7.1 Basic data

Communication channel settings are specified below. The baud rate, number of packets, and packet size each has a default setting which applies at the initiation of communication. These settings may be changed through the negotiate service. Negotiated channel settings will return to defaults as a result of the terminate service or channel traffic time-out.

DATA TYPE: Asynchronous, serial bit (start - stop), half duplex

DATA FORMAT: 8 data bits, 1 start bit, 1 stop bit, no parity

DATA POLARITY: LED on, start bit, space, logical 0

LED off, stop bit, mark, logical 1, quiescent state

DATA RATE: The maximum transmitting speed shall be at least 9600 baud. Selection codes have been arranged for the following baud rates: 300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 56200, or externally defined. Additional selection codes have been reserved for future assignment.

NUMBER OF PACKETS: At least one (1) packet is required although more are negotiable.

PACKET SIZE: Default packet size is 64 bytes, although a larger size can be negotiated.

CHANNEL TRAFFIC TIME-OUT: 6 seconds

INTER-CHARACTER TIME-OUT: 500 milliseconds

RESPONSE TIME-OUT: 2 seconds

TURN-AROUND DELAY: 175 microseconds

In the event of a collision (communicating device and meter are transmitting at the same time), the meter shall cease transmission and wait for the transmission from the communicating device.

4.7.1.1 Default settings

DATA RATE: 9600 baud

NUMBER OF PACKETS: 1

PACKET SIZE: 64 bytes

4.7.2 Packet

<packet>	::=	<stp> <reserved> <ctrl> <seq_nbr> <length> <data> <crc>	
<stp>	::=	EE _H	{ Start of packet character. }
<reserved>	::=	<byte>	{ This field reserved for manufacturer or utility use. Value of the byte should be zero (00 _H) if the field is not used. }
<ctrl>	::=	<byte>	{ Control field. Bit 7. If true (1 _H) then this packet is part of a multiple packet transmission. Bit 6. If true (1 _H) then this packet is the first packet of a multiple packet transmission. Bit 5. Represents a toggle bit to reject duplicate packets. This bit is toggled for each new packet sent. Retransmitted packets keep the same state as the original packet sent. Bits 0 to 4, Reserved.}
<seq_nbr>	::=	<byte>	{Number that is decremented by one for each new packet sent. The first packet in a multiple packet transmission shall have a <seq_nbr> equal to the total number of packets minus one. A value of zero in this field indicates that this packet is the last packet of a multiple packet transmission. If only one packet in a transmission, this field shall be set to zero.}
<length>	::=	<word16>	{ Number of bytes of data in packet. }
<data>	::=	<byte>+	{ <length> number of bytes of actual packet data. This value is limited by the maximum packet size minus the packet overhead of 8 bytes. This value can never exceed 8183. }
<crc>	::=	<word16>	{ CCITT CRC standard polynomial $X^{16} + X^{12} + X^5 + 1$ }

4.7.3 CRC selection

The CRC selected for implementation is the CCITT CRC standard polynomial $X^{16} + X^{12} + X^5 + 1$. The method for calculation and insertion is the HDLC standard per ISO 3309-1979(E) Annex A.

In the PSEM frame, there is no opening flag as referenced in ISO 3309-1979 Annex A. The PSEM start of packet character EE is included in the CRC calculation. The result of the CRC calculation shall be transmitted least significant byte first per the ISO 3309 Annex.

4.7.4 Acknowledgment

A positive or negative acknowledgment is used by the communicating devices to indicate either acceptance or rejection of a packet.

An <ack> is issued by the receiving device to the transmitting device for each packet received that meets the constraints established by this clause.

ANSI C12.18-1996

<ack> ::= 06H

A <ack> is issued by the receiving device to the transmitting device for each packet received that does not meet the constraints established by this clause. Examples of problems with received packets indicated by a <nak> response include CRC errors, packet structure errors, incorrect bit patterns and missing characters.

<nak> ::= 15H

4.7.5 Retransmission

The same packet shall be transmitted if a <nak> is received, an invalid character is received, or the acknowledge time-out elapses. After 3 consecutive negative acknowledgments, automatic termination shall occur.

If a duplicate packet is received by the target, the target should disregard the duplicate packet and return an <ack>.

4.7.6 Time-out

4.7.6.1 Channel traffic time-out

The metering device will terminate communications after waiting some period of time for a valid packet or acknowledgement. This period of time, defined as the channel traffic timeout, shall be 6 seconds.

4.7.6.2 Inter-character time-out

The recipient of the packet must handle the case when the end of a packet is lost. Inter-character time-out is defined as the minimum amount of time the recipient shall wait between characters within a packet when receiving data over the communication channel. The recipient shall wait at least this amount of time before it ceases to wait for the remaining packet and sends a <nak>. The inter-character time-out shall be 500 milliseconds.

4.7.6.3 Response time-out

The sender of the packet must handle the case when the acknowledgment is lost. Response time-out is defined as the minimum amount of time the sender shall wait after a packet is sent to receive an acknowledgment over the communication channel. If this amount of time elapses, the sender will send the packet again. The response time-out shall be 2 seconds.

4.7.7 Delays

4.7.7.1 Turn around delay

The turn around delay is the minimum delay the recipient must wait after receiving a packet and before sending a positive or negative acknowledge. The turn around delay shall be 175 microseconds.

4.8 Layer 1—physical

4.8.1 Physical

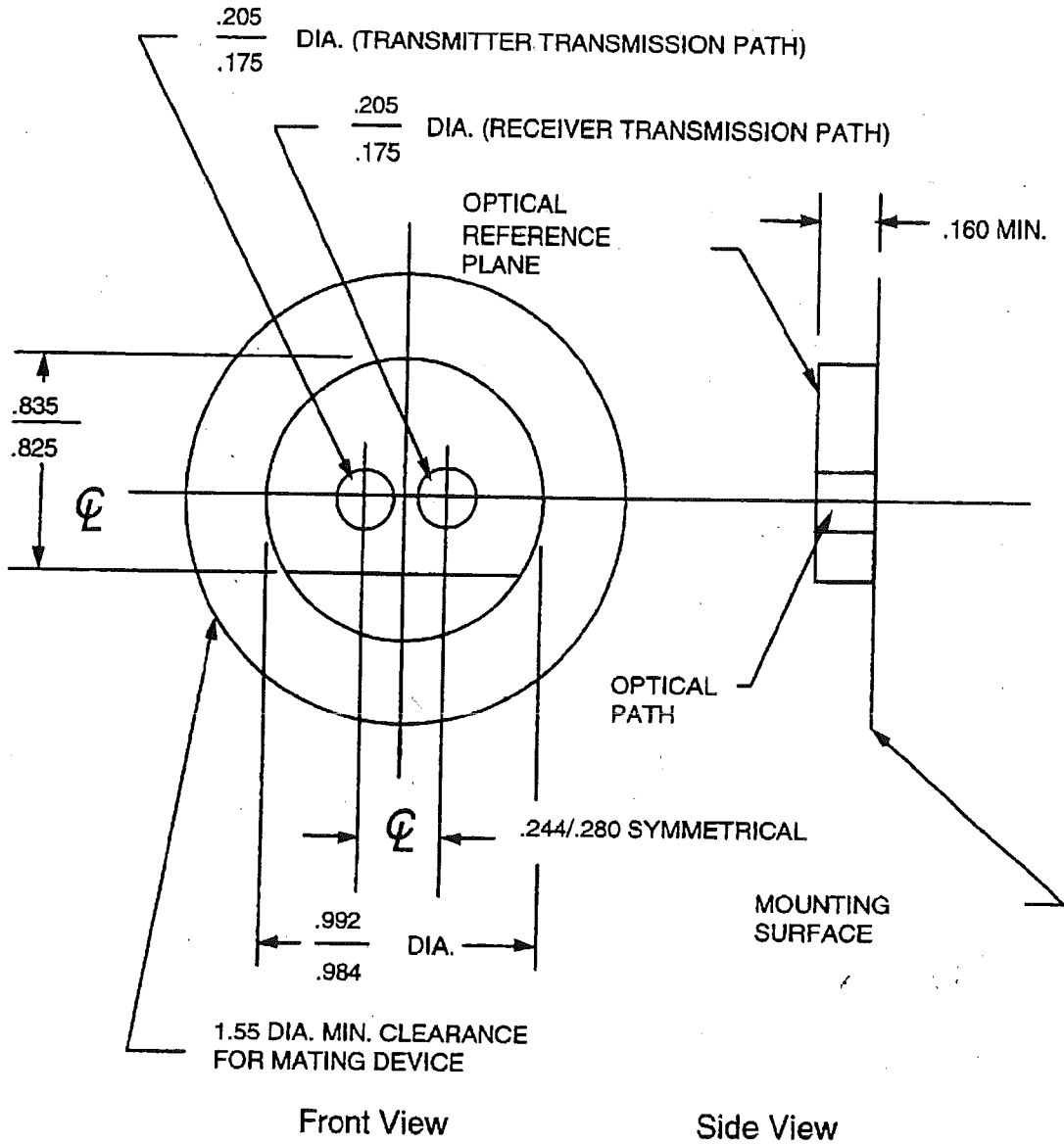


Figure 4-1—External view of port optical port—ANSI type

NOTE: All dimensions are in inches.

ANSI C12.18-1996

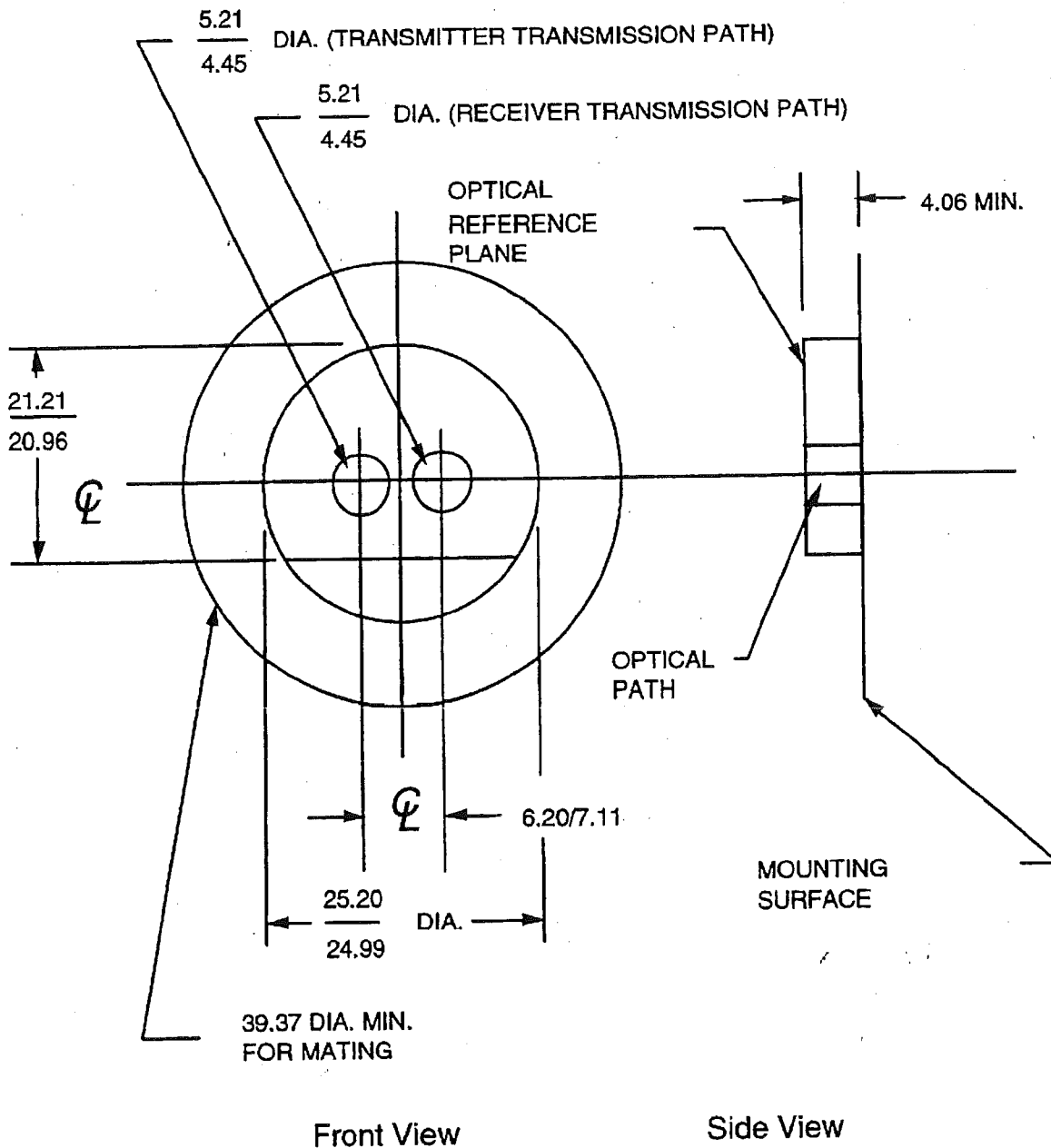


Figure 4-2—External view of port optical port—ANSI type 2

NOTE: All dimensions are in mm.

4.8.2 Basic data

The physical layer data setting is specified below.

DATA POLARITY: LED on, start bit, space, logical 0
 LED off, stop bit, mark, logical 1, quiescent state

4.8.3 Light levels

4.8.3.1 Optical characteristics

The wavelength of the radiated signals in both directions is between 800 nm and 1,000 nm (infrared).

4.8.3.2 Transmitter characteristics

With reference to figure 4-3, the transmitter in the metering device generates a signal with a radiation strength $E_{\theta/T}$ over a defined circular reference surface (optically active area) of diameter d_1 . The test receiver is on the optical axis of the transmitter at a distance d_2 from the optical reference plane on the metering device.

The following limiting values apply:

The reference temperature is 23°C ($\pm 2^\circ\text{C}$).

$d_1 = 5 \text{ mm } (\pm 1 \text{ mm})$

Both conditions shall be met:

	ON-condition	OFF-condition
$d_2 = 10 \text{ mm } (\pm 1 \text{ mm})$	$250 < E_{\theta/T} < 7500 \text{ } \mu\text{W}/\text{cm}^2$	$E_{\theta/T} < 10 \text{ } \mu\text{W}/\text{cm}^2$
$d_2 = 25 \text{ mm } (\pm 1 \text{ mm})$	$85 < E_{\theta/T} < 7500 \text{ } \mu\text{W}/\text{cm}^2$	$E_{\theta/T} < 10 \text{ } \mu\text{W}/\text{cm}^2$

ANSI C12.18-1996

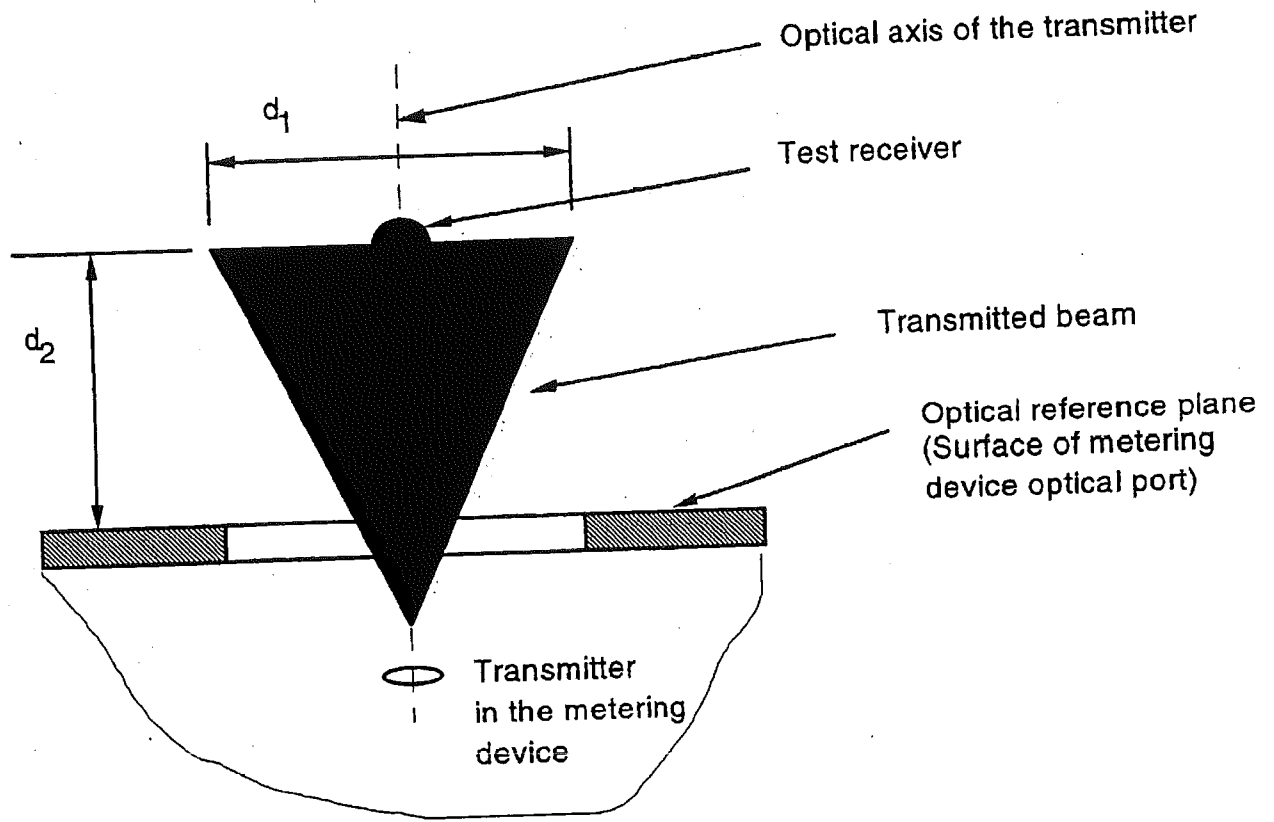


Figure 4-3—Test arrangement for the transmitter

4.8.3.3 Receiver characteristics

With reference to figure 4-4, a test transmitter, located on the optical axis, and positioned at a distance d_2 from the optical reference plane of the metering device, generates a signal with a radiation strength $E_{\theta/R}$ over a defined circular reference surface (optically active area) with a diameter d_1 at the optical reference plane. The receiver shall respond to test signals as follows:

The following limiting values apply:

The reference temperature is $23^{\circ}\text{C} (\pm 2^{\circ}\text{C})$.

$d_1 = 5 \text{ mm} (\pm 1 \text{ mm})$

Both conditions shall be met:

	ON-condition	OFF-condition
$d_2 = 10 \text{ mm} (\pm 1 \text{ mm})$	$1000 < E_{\theta/R} < 7500 \mu\text{W}/\text{cm}^2$	$E_{\theta/T} < 10 \mu\text{W}/\text{cm}^2$
$d_2 = 25 \text{ mm} (\pm 1 \text{ mm})$	$1000 < E_{\theta/R} < 7500 \mu\text{W}/\text{cm}^2$	$E_{\theta/T} < 10 \mu\text{W}/\text{cm}^2$

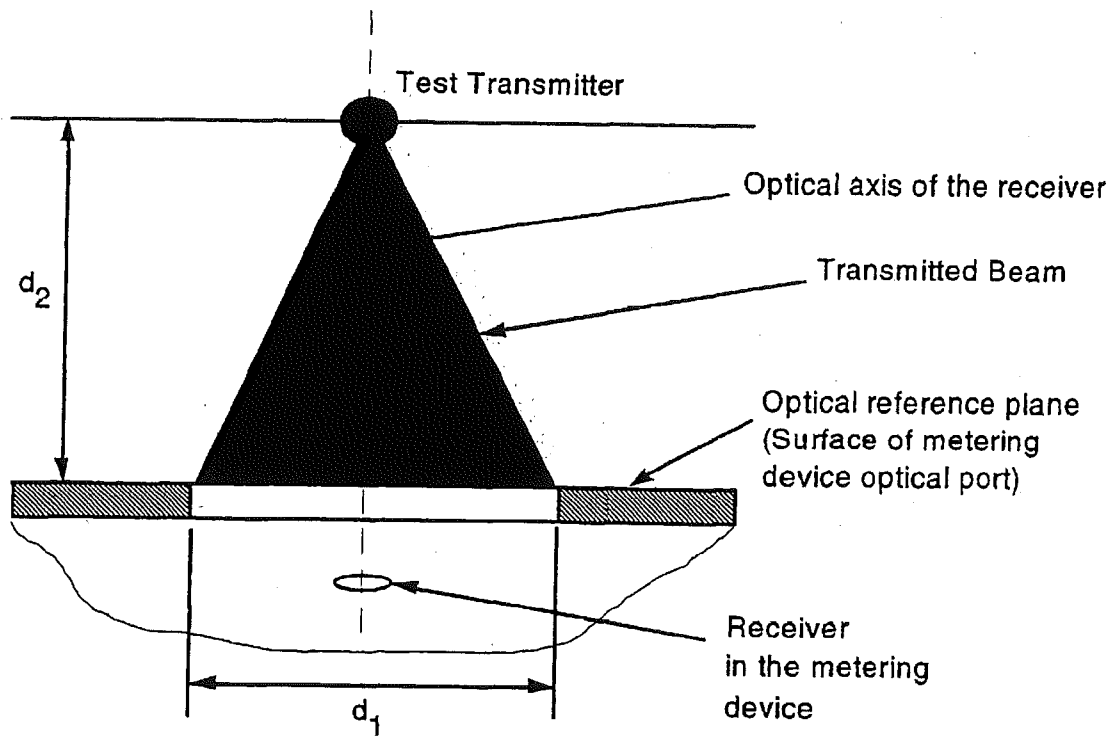


Figure 4-4—Test arrangement for the receiver

4.8.3.4 Environmental lighting condition

The optical path (data transmission) shall not be affected by surrounding light with an intensity of up to 16,000 lux (light composition comparable with daylight, including fluorescent light).

STD. MEDIA C12.18 ENGL 1996 ■ 8470247 0504348 000 ■

ANSI C12.18-1996

ANNEX A
(Informative)
Protocol syntax

A.1 Order of transmission syntax

<bit 0> ::= Least significant bit
 <bit 1> ::= Second least significant bit
 <bit 2> ::= Third least significant bit
 <bit 3> ::= Fourth least significant bit
 <bit 4> ::= Fourth most significant bit
 <bit 5> ::= Third most significant bit
 <bit 6> ::= Second most significant bit
 <bit 7> ::= Most significant bit

<byte> ::= <bit0> <bit1> <bit2> <bit3> <bit4> <bit5> <bit6> <bit7>

<lsbyte> ::= <byte> {least significant byte}

<msbyte> ::= <byte> {most significant byte}

<word16> ::= <msbyte> <lsbyte>

<word24> ::= <msbyte> <byte> <lsbyte>

A.2 Layer 7 syntax

<3jH> ::= 31H | { 1 <index> included in request }
 32H | { 2 <index> included in request }
 33H | { 3 <index> included in request }
 34H | { 4 <index> included in request }
 35H | { 5 <index> included in request }
 36H | { 6 <index> included in request }
 37H | { 7 <index> included in request }
 38H | { 8 <index> included in request }
 39H | { 9 <index> included in request }

<4jH> ::= 41H | { 1 <index> included in request }
 42H | { 2 <index> included in request }
 43H | { 3 <index> included in request }
 44H | { 4 <index> included in request }
 45H | { 5 <index> included in request }
 46H | { 6 <index> included in request }
 47H | { 7 <index> included in request }
 48H | { 8 <index> included in request }
 49H | { 9 <index> included in request }

<6jH> ::= 60H | { No <baud rate> included in request. Stay at
 default baud rate }
 61H | { 1 <baud rate> included in request }
 62H | { 2 <baud rate> included in request }
 63H | { 3 <baud rate> included in request }
 64H | { 4 <baud rate> included in request }
 65H | { 5 <baud rate> included in request }

ANSI C12.18-1996

		66H	{ 6 <baud rate> included in request }
		67H	{ 7 <baud rate> included in request }
		68H	{ 8 <baud rate> included in request }
		69H	{ 9 <baud rate> included in request }
		6AH	{ 10 <baud rate> included in request }
		6BH	{ 11 <baud rate> included in request }
<baud_rate>	::=	00H	{ Externally defined }
		01H	{ 300 baud }
		02H	{ 600 baud }
		03H	{ 1200 baud }
		04H	{ 2400 baud }
		05H	{ 4800 baud }
		06H	{ 9600 baud }
		07H	{ 14400 baud }
		08H	{ 19200 baud }
		09H	{ 28800 baud }
		0AH	{ 57600 baud }
<bsy>	::=	06H	{ Device Busy—This application level error indicates that the request was not acted upon because the device was busy doing something else. }
<cksum>	::=	<byte>	{ 2's compliment checksum computed only on the <data> portion of <table_data>. The checksum is computed by summing the bytes (ignoring overflow) and negating the result. }
<count>	::=	<word16>	{ Length of table data in bytes }
<data>	::=	<byte>+	
<dлк>	::=	08H	{ Data Locked—This application level error indicates that the request was unsuccessful because the data cannot be accessed. }
<dnr>	::=	07H	{ Data Not Ready—This application level error indicates that request was unsuccessful because the requested data is not ready to be accessed. }
<err>	::=	01H	{ Error—This code is used to indicate rejection of the received service request. The reason for the rejection is NOT provided. }
<full_read>	::=	30H <tableid>	
<full_write>	::=	40H <tableid> <table_data>	

<iar>	::=	05H	{Inappropriate Action Requested—This application level error indicates that the action requested was inappropriate. Covers conditions such as write request to a read only table or an invalid table id.}
<ident>	::=	20H	
<ident_r>	::=	<err> <bsy> <iyss> <ok> <std> <ver> <rev> <rsvd>	
<index>	::=	<word16>	{Index value used to locate start of data}
<isc>	::=	03H	{Insufficient Security Clearance—This application level error indicates that the current authorization level is insufficient to complete the request.}
<iyss>	::=	0AH	{Invalid Service Sequence State—This application level error indicates that the request is not accepted at the current service sequence state. For more information on service sequence states, see Annex D.}
<logoff>	::=	52H	
<logoff_r>	::=	<err> <bsy> <iyss> <ok>	
<logon>	::=	50H <user_id> <user>	
<logon_r>	::=	<err> <bsy> <iar> <iyss> <ok>	
<negotiate>	::=	<6jH> <packet_size> <nbr_packet> <baud_rate>*	
<negotiate_r>	::=	<err> <sns> <bsy> <iyss> <ok> <packet_size> <nbr_packet> <baud_rate>	
<nbr_packet>	::=	<byte>	{Maximum number of packets that this layer is able to reassemble into an upper layer data structure at one time.}
<nok>	::=	<err> <sns> <isc> <onp> <iar> <bsy> <dnr> <dlk> <rno> <iyss>	
<offset>	::=	<word24>	{Offset into data table in bytes }
<ok>	::=	00H	{Acknowledge—No problems, request accepted.}
<onp>	::=	04H	{Operation Not Possible—This application level error will be sent to the device which requested an action that is not possible. This error indicates that the message was valid, but the message could not be processed. Covers conditions such as: Invalid Length, Invalid Offset}
<packet_size>	::=	<word16>	{ Maximum packet size supported, in bytes. This value shall not be greater than 8192 bytes.}

ANSI C12.18-1996

<password>	::=	<byte>+	{20 byte field containing password}
<pread_default>	::=	3EH	{ Transfer default table }
<pread_index>	::=	<3jH> <tableid> <index>+ <count>	
<pread_offset>	::=	3FH <tableid> <offset> <count>	
<pwrite_index>	::=	<4jH> <tableid> <index>+ <table_data>	
<pwrite_offset>	::=	4FH <tableid> <offset> <table_data>	
<read>	::=	<full_read> <pread_index> <pread_offset> <pread_default>	
<read_r>	::=	<nok> <ok> <table_data>	
<responses>	::=	<ident_r> { Identification response } <read_r> { Read response } <write_r> { Write response } <logon_r> { Logon response } <security_r> { Security response } <logoff_r> { Logoff response } <negotiate_r> { Negotiate response } <wait_r> { Wait response } <terminate_r> { Terminate response }	
<requests>	::=	<ident> { Identification request } <read> { Read request } <write> { Write request } <logon> { Logon request } <security> { Security request } <logoff> { Logoff request } <negotiate> { Negotiate request } <wait> { Wait request } <terminate> { Terminate request }	
<rev>	::=	<byte>	{Referenced standard revision number}
<rno>	::=	09H	{Renegotiate request—This application level error indicates that the responding device wishes to return to the ID or base state and re-negotiate communication parameters.}
<rsvd>	::=	<byte>	{Reserved for future use. This byte shall be set to 00H until further defined.}
<security>	::=	51H <password>	
<security_r>	::=	<err> <bsy> <i:sss> <ok>	
<sns>	::=	02H	{Service Not Supported—This application level error response will be sent to the device when the requested service is not supported. This

error indicates that the message was valid, but the request could not be honored.)

<std>	::=	<byte>	{Code identifying reference standard. The codes are defined as follows: 00H = ANSI C12.18 01H = Industry Canada 02H-FFH = Reserved }
<tableid>	::=	<word16>	{Table identifier}
<table_data>	::=	<count> <data> <cksum>	
<terminate>	::=	21H	
<terminate_r>	::=	<err> <ok>	
<time>	::=	<byte>	{Suggested wait period in seconds}
<user_id>	::=	<word16>	{User identification code}
<user>	::=	<byte>+	{10 bytes containing user identification}
<ver>	::=	<byte>	{Referenced standard version number}
<wait>	::=	70H <time>	
<wait_r>	::=	<err> <sns> <bsy> <isss> <ok>	
<write>	::=	<full_write> <pwrite_index> <pwrite_offset>	
<write_r>	::=	<nok> <ok>	

A.3 Layer 2 syntax

<ack>	::=	06H	
<crc>	::=	<word16>	{ CCITT CRC standard polynomial $X^{16} + X^{12} + X^5 + 1$ }
<ctrl>	::=	<byte>	{ Control field. Bit 7. If true (1H) then this packet is part of a multiple packet transmission. Bit 6. If true (1H) then this packet is the first packet of a multiple packet transmission. Bit 5. Represents a toggle bit to reject duplicate packets. This bit is toggled for each new packet sent. Retransmitted packets keep the same state as the original packet sent. Bits 0 to 4, Reserved.
<data>	::=	<byte>+	{ <length> number of bytes of actual packet data. This value is limited by the maximum packet size minus the packet overhead of 8 bytes. This value can never exceed 8183. }

Thus the issue is HOW SHOULD ACCESS BE CONTROLLED, AT NETWORK ENTRY OR AT SOME MORE CENTRALIZED POINT? A distributed access control system has some problems of concurrency but minimizes resource utilization in the access process. A centralized one suffers from single-point vulnerability and needs to have some means to prevent someone from tying up the network with access requests.

While the above deals only with single network access it is assumed that access from the internet is also possible. Gateways may then hold the same access role as the collection of hosts, depending on how important access control to the network itself is viewed.

3) *Addressing and Naming*: WHAT ADDRESSING AND NAMING CONVENTIONS ARE APPROPRIATE FOR PACKET RADIO NETWORK INTERNAL TRAFFIC? Increasingly it is assumed that addressing in a packet radio network should be compatible with some internet addressing convention. If so then the question naturally arises as to whether purely internal packet radio network traffic should carry the internet address (header). This is principally a question of efficiency. If internal packet radio nodes carry full internet addresses then each packet has extraordinary overhead. On the other hand, if it is desirable to load or debug a given node from an internet host not on the radio net, then internet addressing may be very convenient.

While an address of a switching node can also serve as the name of an attached host, there is, in mobile networks, some advantage to keeping name and address separate. A host that moves topologically can retain a constant name while having its address change. As mentioned earlier, DARPA's packet radio network dynamically binds name and address as part of its continuous assessment of topology. Whether dynamic name-address binding is needed, then, is an operational issue.

A very useful service to have on any network is a directory of users and hosts. These network servers, among other things, may give the name and address of the entity sought. They may serve the user community or even the internal switches as nodes come and go. Therefore, an issue is whether to use name and address servers, where should they be located, and how are they to be maintained?

4) *Security—Compatibility and Constraints*: Because of the flexible routing of packets in packet switched networks, such networks are more suited to end-to-end rather than link encryption. End-to-end encryption means the user-supplied contents of a given packet are not available at any intermediate switching node. But, for the packets to be routed correctly, the header must be readable by all switching nodes. Thus one of the interesting issues in packet radio networks is SHOULD THE HEADER REMAIN IN THE CLEAR EVEN IF THE TEXT OF THE PACKET IS ENCRYPTED? If the security demands on network operation are severe, then the entire packet must be encrypted, with corresponding requirements of header decryption in each node. There are many advantages, of course, to keeping the interior of the network, the subnet, free of encryption and decryption equipment. Not only is it then unnecessary to deal with the movement of keys, but a very complex network management system for encryption can be avoided as well.

Another encryption strategy for packet radio networks that might be interesting to explore is the use of public key encryption for the packet header only. In fact, a sequential

set of normal addresses, as in a route, could be encrypted in the corresponding public key of each relevant node. This technique could thus define a unique route that could not be entirely decoded, even by the participating nodes. Neighbor nodes could still assist in the routing if congestion arose but only if the source that specified the route enabled that feature. Distributed routing decisions, in other words, would not be possible.

C. Impact on the Radio Spectrum

Electromagnetic compatibility is a two-way street. Like all radio systems, packet radio networks are vulnerable to intentional or accidental impact by external signals or noise and they, in turn, can harm other networks. The radio spectrum is a carefully regulated resource in most countries and therefore the radio frequency, the bandwidth, the power, modulation, and sometimes the direction and directivity of the signal are subject to review and approval.

1) *Electromagnetic Compatibility*: While not necessary, some packet radio systems use a large time-bandwidth product through the use of techniques such as spread spectrum; that is, the modulation bandwidth is much greater than the information bandwidth. There are several reasons for this: the opportunity for encoding for code division multiple access (CDMA), less fading because of the ability to resolve and process multipath components, a lower probability of interference from narrow-band signals within the bandwidth occupied by decorrelating them, and for low radio emission visibility. However, this consideration of the spectral distribution of energy is but a part of one of the most important issues in any packet radio system: HOW SHOULD A GIVEN PACKET RADIO SYSTEM BE ACCOMMODATED IN THE EXISTING RADIO SPECTRUM? This issue is not one of simply being compatible with the allocations of the spectrum imposed by some governing body. The choice of frequency also has to do with whether the propagation characteristics that determine coverage and reliability are suitable and if the cost, portability, and placement requirements of the radio frequency components are likewise acceptable, as discussed in Section II.

2) *Electronic Counter-Countermeasures and Noise Immunity*: Because of the highly capable network management in a packet radio network, it is possible to give the network some countermeasures of its own, ones intrinsically different from those offered by individual nodes. For example, the knowledge of a jammer targeting one or two nodes can be used to direct traffic around these threatened parts of the network. Power control on a link-by-link basis to hide the presence of the network from a known enemy is another example. But as with any of the other overhead functions of the network the major issue ultimately becomes THE AMOUNT OF NETWORK-BASED COUNTERMEASURES VERSUS THE MAINTENANCE OF THROUGHPUT. Since the value of countermeasures is a purely military question, it cannot be answered apart from some specific mission statements.

While this issue surrounding countermeasures is military-oriented, there are also some similar but less hostile threats on the domestic scene. Ignition noise and other varieties of noise are inimical to digital data radio systems and signal processing is one of the best means to cope with them. The coherent spreading of the radio frequency wave-

form, a widely practiced ECCM technique, is one effective means of combating impulsive interference. Likewise, adaptive routing techniques can be used to route around an area of the network experiencing a high degree of interference.

3) *Efficiency*: Several efficiency factors may be defined for a network, but ultimately one must consider the number of users that can effectively share a given set of resources, say bandwidth and area. (The sharing of time is intrinsic to packet switching.) A typical measure might be the number of users accommodated in a given area over a given time for a given bandwidth. Unfortunately, the shared collective resource in packet radio has many dimensions, including not only time-bandwidth but space, antenna directivity, and conceivably polarization. There are other aspects of network functionality that must also be considered in any examination of efficiency. These functions deal mostly with topological change: the use of resource, including bandwidth, to measure, assess, and control the network and to maintain individual connections, even under the threat of countermeasures.

Perhaps the most important issue regarding efficiency in a mobile packet radio network is the value of this new network functionality. More quantitatively WHAT FRACTION OF TOTAL AVAILABLE RESOURCES SHOULD BE ALLOCATED FOR NETWORK OVERHEAD FUNCTIONS? Obviously, there is no one answer to this question but perhaps more importantly, there is no general means to answer it. The use of network resources for other than pure data transport varies from desirable to critical and evaluating the network on transport alone does not recognize that value. A well-designed network will allocate the maximum resource to meeting transport criteria consistent with connection establishment and maintenance. Conceivably, network overhead could take more time-bandwidth than user traffic and still be justified. In the final analysis, then, performance can only be judged in light of the overall mission of the network.

V. NETWORK PERFORMANCE EVALUATION AND COSTS

There is a great deal of complexity in the design of a packet radio network, but no aspect is more difficult than the gauging of network performance. Choosing the metrics and forming the criteria for optimum performance has not been accomplished except for the simplest of topologies. It is not only the topologies that make the analysis complicated, but the fact that the numerous functions the networks perform have not been given relative value. Internal network monitoring and diagnostics, dynamic hand-off, route establishment, alternate routing, delay minimization, and access control are just some of the features other than simple transport that impact network performance. How well a network performs these various functions are components of an overall figure of merit yet to be devised.

A. Metrics for Performance Evaluation

1) *Metrics from a User Perspective*: While most analytical expressions of performance deal only with throughput and delay, the user is generally interested in several more performance metrics. Table 1 lists some of these.

Obviously, these disparate performance measures cannot easily be reconciled. The major performance and eval-

Table 1 User-Oriented Performance Metrics

Availability	Fraction of time that a connection is available.
Delay	The end-to-end transit time through the network.
Priority	Differential handling of a packet based on its relative value in the user community.
Throughput	Bits per second accepted and correctly delivered by the network.
Coverage	The average area from which packets can be transmitted or received.
Mobility	The maximum speed a terminal can travel and still open and maintain a connection with useful throughput.
Accuracy	The overall error rate as seen by the user terminal equipment.

uation issue is determining WHAT IS THE RELATIVE IMPORTANCE OF EACH PERFORMANCE METRIC TO THE COMMUNITY OF USERS OF THE NETWORK. Once their requirements are known, then the designer can emphasize various functions accordingly.

2) *Metrics from a Network Operator Perspective*: To the extent that the operator is providing for the explicit needs of the user, their decisions and criteria about performance will be similar. Usually, however, the operator must also be concerned with different elements such as cost, tolerance to faults, spectrum efficiency, and environmental compatibility. Important parameters such as bandwidth and power are enforced by governmental bodies, may not be negotiable, and thus set hard limits on some of the user performance metrics mentioned above.

B. Costs

The costs of a packet radio network, as with any network, can be grouped into design, implementation, and operating costs. Cost cutting in the design phase may have financial repercussions on operations. In the case of the DARPA packet radio network the extra time and cost taken to provide remote diagnostics and maintenance have repaid themselves many times over. Ultimately, the cost issue is the tradeoff of THE COST OF CREATING THE NETWORK VERSUS THE COST OF OPERATING IT. The outcome is often defined by the amount of each type of money available (capitalization costs versus operating costs) and by such issues as cash flow and payback. In both a military and commercial setting, advantage will accrue from keeping network operation from being very labor-intensive.

Another perspective on network costs comes from looking at the cost of components. In any large network the costs will be dominated by the unit cost of the switch, even if some network control or monitoring equipment is to be used. Another dominant equipment cost is the means of attaching terminal equipment to the network: interface devices and so-called digital termination equipment. So ultimately, the important cost issue becomes HOW TO MINIMIZE THE COST OF THE HIGHLY REPLICATED COMPONENTS OF THE NETWORK.

VI. SUMMARY AND CONCLUSIONS

We have presented in this paper a discussion of a number of issues that arise in the design, development, and deployment of a packet radio network. It is clear from this discussion that there is a large collection of design choices to be made. A number of packet radio networks have already

been developed and tested, and it is expected that more will be deployed in the future as the technology matures. It is hoped that this paper provides the context in which the lessons of past research can be applied to the systems of the future.

ACKNOWLEDGMENT

The list of contributors to packet radio technology, and the ideas in this paper, is clearly too long to include here. However, a special note of acknowledgment is due R. Kahn [8] for his vision in initiating the DARPA packet radio network program, upon which much of the other packet radio efforts have been based.

REFERENCES

- [1] F. Tobagi, R. Binder, and B. M. Leiner, "Packet radio and satellite networks," *IEEE Commun. Mag.*, Nov. 1984.
- [2] B. H. Davies and T. R. Davies, "The application of packet switching techniques to combat net radio," this issue, pp. 43-55.
- [3] R. E. Kahn *et al.*, "Advances in packet radio technology," *Proc. IEEE*, vol. 66, pp. 1468-1496, Nov. 1978.
- [4] J. Jubin and J. Tornow, "The DARPA packet radio network protocols," this issue, pp. 21-32.
- [5] D. Connors, "Amateur packet radio," presented at INFOCOM'83, panel session, Apr. 1983.
- [6] A. Ephremides, J. E. Wieselthier, and D. J. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling," this issue, pp. 56-73.
- [7] R. Binder, S. D. Huffman, I. Gurantz, and P. A. Vena, "Cross-link architectures for a multiple satellite system," this issue, pp. 74-82.
- [8] R. E. Kahn, "The organization of computer resources into a packet radio network," *IEEE Trans. Commun.*, vol. COM-25, pp. 169-178, Jan. 1977.
- [9] H. Zimmerman, "OSI reference model—the ISO model of architecture for open systems interconnection," *IEEE Trans. Commun.*, vol. COM-28, pp. 425-432, Apr. 1980.
- [10] L. Kleinrock and J. Silvester, "On the behavior of multi-hop packet radio networks," this issue, pp. 156-167.
- [11] F. Tobagi, "Multi-access link control," in *Computer Network Architectures and Protocols*, P. E. Green, Ed. New York: Plenum, 1982.
- [12] —, "Modeling and performance analysis of multihop packet radio networks," this issue, pp. 135-155.
- [13] N. Abramson, "The ALOHA system—Another alternative for computer communications," in *1970 Fall Joint Comput. Conf., AFIPS Conf. Proc.*, vol. 37, pp. 281-285 (AFIPS Press), 1970.
- [14] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: Part I—Carrier sense multiple access modes and their throughput delay characteristics," *IEEE Trans. Commun.*, vol. COM-24, pp. 1400-1416, Dec. 1975.
- [15] F. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part II—The hidden terminal problem in carrier sense multiple access and the busy tone solution," *IEEE Trans. Commun.*, vol. COM-23, pp. 1417-1433, Dec. 1975.
- [16] M. B. Pursley, "The role of spread spectrum in packet radio networks," this issue, pp. 116-134.
- [17] W. C. Fifer and F. J. Bruno, "The low-cost packet radio," this issue, pp. 33-42.
- [18] D. Behrman and W. C. Fifer, "A low-cost spread-spectrum packet radio," in *Proc. MILCOM82*, paper 10.4, 1982.
- [19] J. Jubin, "Current packet radio protocols," in *INFOCOM85 Proc.*, Mar. 1985.
- [20] D. L. Nielson, "Microwave propagation measurements for mobile digital radio applications," in *Proc. EASCON-77*, pp. 14-2A-14-2L, Sept. 1977.
- [21] J. McQuillan, I. Richer, and E. Rosen, "The new routing algorithm for the Arpanet," *IEEE Trans. Commun.*, vol. COM-28, pp. 711-719, May 1980.
- [22] V. G. Cerf and E. Cain, "The DoD internet architecture model," *Comput. Net.*, pp. 307-318, July 1983.
- [23] B. M. Leiner, J. Postel, R. Cole, and D. Mills, "The DARPA internet protocol suite," *IEEE Commun. Mag.*, vol. 23, pp. 29-34, Mar. 1985.
- [24] N. Shacham and J. Westcott, "Future directions in packet radio architectures and protocols," this issue, pp. 83-99.
- [25] A. S. Tanenbaum, *Computer Networks*. Englewood Cliffs, NJ: Prentice Hall, 1981.
- [26] J. Westcott and J. Jubin, "A distributing routing design for a broadcast environment," in *Proc. MILCOM82*, paper 10.3, 1982.
- [27] J. Jubin and N. Gower, "Congestion control using pacing in a PRNET," in *Proc. MILCOM82*, Oct. 1982.

Barry M. Leiner (Senior Member, IEEE), for a photograph and biography please see page 4 of this issue.

Donald L. Nielson (Member, IEEE), for a photograph and biography please see page 5 of this issue.

Fouad A. Tobagi (Fellow, IEEE), for a photograph and biography please see page 5 of this issue.

A Routing Protocol for Packet Radio Networks

Shree Murthy and J.J. Garcia-Luna-Aceves

Computer Engineering

University of California

Santa Cruz, CA 95064

shree, jj@cse.ucsc.edu

Abstract

We present a new distance-vector routing protocol for a packet radio network. The new distributed routing protocol, WRP, works on the notion of second-to-last hop node to a destination. WRP reduces the number of cases in which a temporary routing loop can occur and also provides a mechanism for the reliable transmission of update messages. The performance of WRP has been compared quantitatively by simulations with that of distributed Bellman-Ford (DBF), DUAL (a loop-free distance-vector algorithm) and an ideal link-state algorithm (ILS) which represents the state of the art of Internet routing, in a highly dynamic environment. The simulation results indicate that WRP is the most efficient of the algorithms simulated in a wireless environment.

1 Introduction

With the recent proliferation of laptop and portable computers, and the development of wireless network interfaces, host mobility is becoming an important issue. An efficient routing protocol is necessary to communicate directly with the participating computers in a highly dynamic environment in which the hosts are mobile.

The routing protocols used in multihop packet-radio networks implemented in the past [1, 2, 11] were based on shortest-path

⁰This work was supported in part by the Advanced Research Projects Agency (ARPA) under contract F19628-93-C-0175 and by the Office of Naval Research under Contract No. N-00014-92-J-1807.

Permission to make digital/hard copies of all or part of this material for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication and its date appear, and notice is given that copyright is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires specific permission and/or fee.

MOBICOM 95 Berkeley CA USA

© 1995 ACM 0-89791-814-2/95/10..\$3.50

routing algorithms that have been typically based on the distributed Bellman-Ford algorithm (DBF) [3]. According to DBF, a node knows the length of the shortest path from each neighbor node to every network destination and this information is used to compute the shortest path and next node in the path to each destination. An update message contains a vector of one or more entries, each of which specifies as a minimum, the distance to a given destination. A major performance problem with DBF is that it takes a very long time to update the routing tables of network nodes after network partitions, node failures, or increases in network congestion. This performance problem of DBF stems from the fact that it has no inherent mechanism to determine when a network node should stop incrementing its distance to a given destination.

Because of DBF's performance problems, most router manufacturers have opted for routing protocols based on topology broadcast such as OSPF [12]. However, there are significant differences between the wired Internet over which standard routing protocols are used today, and wireless networks. Today's networks have relatively high bandwidth and topologies that change infrequently; in contrast, wireless networks have mobile nodes and have limited bandwidth for network control. Mobility management algorithms which involve user location and hand-off management, essentially require reliable and efficient routing algorithms. For scalability with base stations and mobile nodes, these algorithms need to be distributed. Also, the routing algorithms should be flexible enough to serve as a template to carry mobility management information. In addition, since the bandwidth is limited in a wireless environment, the routing algorithms should not suffer from looping problems.

The flooding techniques used in link-state or topology broadcast protocols create excessive traffic in a multihop radio network with dynamic topology. On the other hand, the routing protocols based on DBF or modifications of DBF take a long time to converge and the frequent topology changes in a wireless network with mobile nodes make the looping problem of DBF unacceptable. Therefore, there is a need for a new routing protocol which is devoid of all these drawbacks.

In the recent past, a number of efforts have been made to address the limitation of DBF and topology broadcast in mobile wireless networks. One such effort is the DSDV protocol [5]. In this protocol, each mobile host which is a specialized router, periodically advertises its view of the interconnection topology with other mo-

bile hosts within the network thereby maintaining an up to date information about the status of the network. But, DSDV suffers from the inherent problem of synchronization. In DSDV, a node has to wait until it receives the next update message originated by the destination to update its distance table entries. Also, it uses both periodic and triggered updates for updating routing information. This could cause excessive overhead.

A distributed routing algorithm for mobile wireless networks based on diffusing computations has been proposed by Corson and Ephremides [6]. This protocol relies on the exchange of short control packets forming a *query-reply* process. It also has the ability to maintain multiple paths to a given destination. This is a destination-oriented protocol in which separate versions of the algorithm run independently for each destination. Routing is source-initiated in which routes are maintained by those sources which actually desire routes. Even though this algorithm provides multiple paths to the destination, because of the query-based synchronization approach to achieve loop-free paths, the communication complexity could be high.

Recently, a number of distributed shortest-path algorithms have been proposed [4, 7, 9, 10, 14] that utilize information regarding the length and second-to-last hop (predecessor) of the shortest path to each destination to eliminate the counting-to-infinity problem of DBF. We call this type of algorithms as *path-finding algorithms*. According to these algorithms, each node maintains the shortest-path spanning tree reported by its neighbors. A node uses this information along with the cost of adjacent links to generate its own shortest-path spanning tree. An update message exchanged among neighbors consists of a vector of entries that report updates to the sender's spanning tree; each update entry contains a destination identifier, the distance to the destination, and the second-to-last hop of the shortest path to the destination.

Path-finding algorithms are an attractive approach for wireless networks. Although they eliminate counting-to-infinity problem, they can still incur substantial temporary loops in the paths specified by the predecessor before they converge. This can lead to slow convergence, or incur substantial processing by requiring a node to update its entire routing table for each input event. To address these problems, we have proposed a path-finding algorithm, PFA, that substantially reduces the temporary looping situations [13].

The rest of this paper describes a routing protocol for a packet radio network based on PFA, which we call *wireless routing protocol* (WRP). Section 2 gives a detailed description of WRP, illustrating the key aspects of the protocol's operation. Section 3 compares the performance of WRP with that of DBF, DUAL and ILS. Finally, Section 4 presents our conclusions.

2 Wireless Routing Protocol

2.1 Overview

To describe WRP, we model a network as an undirected graph represented as $G(V, E)$, where V is the set of nodes and E is the set of links (or edges) connecting the nodes. Each node represents a router and is a computing unit involving a processor, local memory and input and output queues with unlimited capacity. A functional

bidirectional link connecting the nodes is assigned a positive weight in each direction. All messages received (transmitted) by a node are put in the input (output) queue and are processed in FIFO order. The communication links in the network are such that all update messages transmitted over an operational link are received in the order in which they were transmitted within a finite time.

A link is assumed to exist between two nodes only if there is radio connectivity between the two nodes and they can exchange update messages reliably with a certain probability of success. When a link fails, the corresponding distance entries in a node's distance and routing tables are marked as infinity. A node failure is modeled as all links incident on that node failing at the same time.

WRP is designed to run on top of the link-level protocol of a wireless network. Update messages may be lost or corrupted due to changes in radio connectivity or jamming. Reliable transmission of update messages is implemented by means of retransmissions. After receiving an update message free of errors, a node is required to send a positive acknowledgment (ACK) indicating that it has a good radio connectivity and has processed the update message. Because of the broadcast nature of the radio channel, a node can send a single update message to inform all its neighbors about changes in its routing table; however, each such neighbor sends an ACK to the originator node.

In addition to ACKs, the connectivity can also be ascertained with the receipt of any message from a neighbor (which need not be an update message). To ensure that connectivity with a neighbor still exists when there are no recent transmissions of routing table updates or ACKs, periodic update messages without any routing table changes (null update messages) are sent to the neighbors. The time interval between two such null update messages is the *HelloInterval*.

If a node fails to receive any type of message from a neighbor for a specified amount of time (e.g., three or four times the *HelloInterval* known as the *RouterDeadInterval*), the node must assume that connectivity with that neighbor has been lost.

2.2 Information Maintained at Each Node

For the purpose of routing, each node maintains a *distance table*, a *routing table*, a *link-cost table*, a *message retransmission list* and an *ack-status table*.

The distance table of node i is a matrix containing, for each destination j and each neighbor of i (say k), the distance to j (D_{jk}^i) and the predecessor (p_{jk}^i) reported by k .

The routing table of a node i is a vector with an entry for each known destination j which specifies:

- The destination's identifier
- The distance to the destination (D_j^i)
- The predecessor of the shortest path chosen toward j (p_j^i)
- The successor (s_j^i) of the shortest path chosen for j
- A marker (tag_j^i) used to update routing table; it specifies whether the entry corresponds to a simple path ($tag_j^i = \text{correct}$), a loop ($tag_j^i = \text{error}$) or a destination that has not been marked ($tag_j^i = \text{null}$).