Distributed Queue Wireless Arbiter (DQWA)

BACKGROUND

Traditional Controller Area Network (CAN) protocol utilizes a Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) technique similar to that of Ethernet but with frames that are relatively small by networking standards in that the largest possible frame may be around 128-bits (i.e. 16-Bytes, including the maximum of 8-bytes for the payload), whereas the Ethernet Frame varies between 64-bytes and 1,536-bytes. Unlike Ethernet however, there is no loss of data as a result of collisions. This is because of CAN's unique non-destructive message arbitration methodology that guarantees high priority messages access to the CAN bus with no fear of collision or loss of data; hence, no need for retransmission. However, the same feature that is CAN's strength (its non-destructive collision resolution methodology) is also its weakness in that as a CAN bus approaches its utilization capacity so does its propensity for indefinite starvation of lower priority messages. Given that a CAN message cannot arbitrarily change its priority; the CAN protocol is completely inflexible under heavy loads for successfully ensuring that lower-priority messages reach their destination. The traditional methodology as known in the art for resolving this problem has been in the separation of CAN nodes into multiple CAN sub-networks. However, such delineation can often be the source of frustration when attempting to discern the most efficient means for dividing the devices into disparate CAN networks while still affording cross network communication through various backhaul communication technologies. Embodiments presently disclosed provide security and reliability within a network, while maintaining CAN's distributed network communication methodology and implicit avoidance of single points of failure within the network.

DETAILED DESCRIPTION

In the following description, numerous specific details are set forth to clearly describe various specific embodiments disclosed herein. One skilled in the art, however, will understand that the presently claimed invention may be practiced without all of the specific details discussed below. In other instances, well known features have not been described so as not to obscure the invention. In addition, it should be understood that embodiments of the invention include both hardware and electronic components or modules that, for purposes of discussion, may be illustrated and described as if the majority of the components were implemented solely in hardware. However, one of ordinary skill in the art, and based on a reading of this detailed description, would recognize that, in at least one embodiment, the electronic based aspects of the invention may be implemented in software. As such, it should be noted that a plurality of hardware and software-based devices, as well as a plurality of different structural components may be utilized to implement the invention. Furthermore, and as described in subsequent paragraphs, the specific mechanical configurations illustrated in the drawings are intended to exemplify embodiments of the invention and that other alternative mechanical configurations are possible.

A Distributed Queuing Wireless Arbiter (DQWA) Protocol is based on the Distributed Queue Switch Architecture (DQSA) developed at the Illinois Institute of Technology. The DQSA was originally designed as Layers One (1) and Two (2) broadcast network architecture for cable TV networks that provided deterministic access to the transmission queue while simultaneously limiting collisions to a finite window within the DQ Transmission Frame. The DQSA may be extended into the wireless arena by focusing mostly on the Link Layer (i.e. Layer Two (2)) with only minimal direction regarding the Physical Layer (i.e. layer two (2)). The wireless nature of DQ may be defined in Distributed Queuing Wireless Arbiter (DQWA) with most of the specification dealing with the Link Layer while also providing only minimal direction for the Physical Layer.

The DQWA is a hybrid of a traditional “hub and spoke” network architecture with that of a peer-to-peer MESH network architecture. The primary area of focus of the DQWA specification is that of the Link Layer, although a key and critical aspect of it successful implementation, the Contention Window and associated Min-Slots, is heavily dependent upon the Physical Layer in that successful implementation of a unique Collision Detection mechanism may be implemented.
The heart of DQWA technology is a Medium Access Control (MAC) layer that allows an arbitrary number of stations to share a common communications channel over any distance and operating at any data rate.

DQSA can operate over virtually any topology and will also provide a Quality of Service (QoS) at the MAC layer that includes the ability to temporarily elevate priorities in order to prevent starvation (as can occur in traditional CAN).

DQWA may be a distributed architecture with respect to communication. However, for control, DQWA is static for a given point in time; specifically, it is static for the duration of a DQ Transmission Frame. The designated central control point may transition to other nodes upon completion of the current DQ Transition Frame; which is why DQWA can be viewed as a hybrid between a pure MESH ad-hoc architecture and that of a traditional Hub-and-Spoke architecture.

The hybrid nature of the DQWA network architecture provides flexibility for adaptation to a CAN Wireless Extension in that communication is distributed while enabling a central authority to elevate priorities of messages as needed providing a QoS aspect to DQWA that CAN severely lacks. Also, because the central authority may shift from DQ node to DQ node if desired (i.e. enabled to do so), traffic patterns may be localized with respect to control. Thus, reducing latency when and where needed; according to the traffic pattern. Because all communication can be encrypted at the MAC layer, including the headers; security may be maintained at all times in spite of the fact that all traffic is broadcast wirelessly.
The key feature of DQSA is that all control resides in the stations, no central control is required. The network state is maintained at all times by each station in just two (2) binary counters per DQSS, providing it with all the information necessary to make decisions as to when to transmit for that specific DQSS. A DQ Transmission Frame may be divided into three separate time periods/segments listed below:

- 1) Referring to FIG. 35, Contention Window (CW), utilized as part of the Access Request Sequence (ARS) **10** to the Transmission Queue with three (3) control mini-slots **15**, **20** and **25** acting as a finite sized Contention Queue;
- 2) Data and Control Window consisting of a single DQ Data and Control Frame; and,
- 3) Feedback Window, consisting of the DQ Feedback Frame with Synchronization Beacon.

A synchronization beacon may be transmitted to all stations prior to the start of each segment from which all stations must synchronize with for every transmission frame so that they may participate in the DQSS. The DQ Feedback Frame and associated Synchronization Beacon can come from any node within the DQSS, but is always sent by a single node at any given time and from which the node is typically chosen as one of a set of nodes designated for accessing gateways beyond the DQSS. Within a wireless environment, this central point would normally be referred to as the Base Station, Access Point, or Hub; the DQWA nomenclature for this central authority may be Cluster Head.
Variable length DQ Messages may be segmented into multiple data slots without requiring any further overhead. Qualities of Service (QoS) Priorities are available and it may be possible for a higher priority DQ Data & Control Frame to preempt a lower priority DQ Data & Control Frame during transmission within a period of one DQ Message. Segments may be allocated to a specific station thus providing time-division-multiplex (TDM) channels, commingled with normal DQ Frame traffic. The overall utilization within a wireless environment, i.e., ratio of data content to the channel capacity ranges from over 95% down to 80%; depending upon frame size and overall network utilization.
As mentioned in above, because access to communication within a DQSS consists solely of member nodes, the entire contents within a MAC layer frame, including the header, may be encrypted; thus ensuring the both security and privacy. The purpose of the CW's ARS is twofold:

- 1. To afford current members of the DQSS with an opportunity to request communication privileges with one or more of the other nodes (including the Cluster Head) within the network; and
- 2. To simultaneously mitigate the potential for MAC & Data Payload collisions and hence, dropped frames resulting from corruption.

The latter is achieved by limiting the contention for access to the channel to a finite and predictable period of time. With the exception of the Cluster Head, all nodes may utilize this mechanism in order to access the MAC & Data Payload segment of the DQ Transmission Sequence. The ARS Segment **10** may be divided into three (3) sub-parts, termed, Mini-Slots (MS) **15**, **20** and **25** as shown in FIG. 35.
The collision resolution process referenced above may utilize unique patterns transmitted by each soliciting device and a summation of those patterns in the event of a collision as a means for detecting collisions. The operation of DQWA is based on the m-ternary feedback information on the state of each of the mini-slots **15**, **20** and **25**. The Cluster Head may be able to distinguish between the three states: Idle, Success, and, Collision, for each mini-slot; as this information may provide protocol rules at the end of each frame. Each node may be assigned a unique bit pattern that has the property that when two or more ARS **10** collide, the pattern of the overlapping signal is distinguishable from the original pattern of any single ARS **10**; hence, the Cluster Head can detect the collision.
In one exemplary embodiment, patterns are binomial coefficients; however, this number may be modified to meet the requirements of the targeted environment. Each node accepted into the network is assigned both a Node Address **30** and a constant size Code Word **35** of constant Hamming Weight as shown in FIG. 36.

When a collision does occur, and particularly within an RF environment, it may be possible to determine that a collision has occurred since the collision may make the interpretation of the combined signal unintelligible. Further, even if the resultant collided signal does result in an intelligible result, the resulting Hamming Weight may be something other than the selected constant value. When taking into account that the correct associated DQSS node address must accompany the code word of constant hamming weight, the detection of a collision is possible.
DQWA may have an additional validation mechanism within the DQ Feedback Frame that protects against the unlikely occurrence of an illegitimate, but valid Code Word and DQSS Node Address combination resulting from a collision.

The aforementioned ternary decision described above may be subsequently determined as follows: Idle (i.e. no signal in ARS Mini-Slot)—Received Signal is below the RSSI (Noise) Threshold; Success—A demodulation resulting in the correct hamming weight and correct code word value and node address combination and again validated within the DQ Feedback Frame; Collision—Any signal detected above the noise (RSSI) threshold not resulting in a translation into the digital domain of a code word with the correct hamming weight and correct code word value and node address combination.

The Cluster Head may respond with the collision results as part of the DQSS Management Segment in order to clarify any potential ambiguities. Standard DQSS Network addresses may be 12-bits in length, with the lower 10-bits assigned for the dynamic portion of a valid address; as the upper two bits have special meaning. Both bits along with the rest of the DQSS Network Address are shown in FIG. 37.

Referring to FIG. 37, a DQSS Node Cluster Bit **45** may be set to zero during the ARS.
The Most Significant Bit (MSB) of the address is reserved for the Cluster Head. This is particularly helpful if the Network Topology moves and the Cluster Head moves with it. Thus, allowing any node to maintain its original identity both before and after assuming the duties of the Cluster Head. In this way, the DQSS table maintains consistency regardless of which node is currently in charge of the network.
A DQSS Node Join Request Bit **45** may be used by nodes wishing to join the network. In order for an unknown node to be considered for admittance to the DQSS, it may be configured to satisfy the following two conditions:

-
  - 1) The “Join Request” Bit **45** as shown in FIG. 37 must be set within the DQSS Node Address Field. The Join Request Bit **45** allows for parts to be installed within a particular network architecture with little to any actual configuration in that “newly” installed parts can automatically request for inclusion in the desired vehicle's DQSS.
  - 2) The “DQSS Mini-Cluster” Sub-Field **50** must set ‘7’ (i.e. “111 b”).

The “DQSS Individual Address” Sub-Field **55** may be a value between ‘0’ and “127” (i.e. a span of 128-values). The DQSS Mini-Cluster Sub-Field **50**, this is an important field in that it explicitly affords specific portions of a DQSS to be segmented into individual address spaces for the purpose of multi-cast addressing as well as enabling CAN sub-networks within a specific DQSS. The addition of a Message Bit to the DQSS Node Address Field (as alluded to in the previous section) would enable further enforcement of messages being restricted to specific CAN sub-networks.

The DQSS Individual Address Sub-Field **55**, these seven bits are used for assigning individual addresses, with any value between ‘0’ and “126” assignable for an individual DQSS Network Address. The only time “127” may be used during the ARS is during a “Join Request.” As “127” is otherwise set aside for “Directed Broadcasts” and regular “Broadcasts” for all Mini-Cluster Sub-Field values except for ‘7’ (i.e.“111b”).

A key component of the DQ Service Set concept is network security and the rules by which nodes may become members of a specific DQ Service Set. A DQSS can operate in one of three operational modes listed below the operational modes listed in decreasing order of centralized membership control: Static Association Mode; Semi-Manual Association Mode; Promiscuous Mode. Each of the modes will now be individually discussed in detail.

In Static Association Mode, the DQSS is completely pre-configured. New nodes may not request to join and can only become part of the DQSS either by directly adding nodes to an existing DQSS Configuration Database or by installing a completely new DQSS Configuration Database containing the desired nodes. In response to the fact that a DQSS configured in Static Association Mode cannot add nodes in real time (doing so only through configuration); any attempt to submit a DQSS Membership Request Code Word during the ARS segment will be ignored.

A DQSS configured to be in Semi-Manual Mode has all of the capabilities of a Static Association Mode DQSS as well as the additional ability to add nodes in real time. There are two methods for which a node may acquire inclusion within a DQSS configured in DQ Semi-Manual Association Mode. The first method for acceptance for a given node into a DQSS while in DQSS Semi-Manual Association Mode is via manual configuration as part of a DQSS Configuration Database. The second method utilizes a two-step process for any node outside of the current DQSS membership and described below:

-
  - 1) First, the Candidate Node must issue a request for DQSS Inclusion.
  - 2) Second, an external confirmation of the request from either an operator (i.e. service technician or factory installation personnel) or configuration robot utility must explicitly accept the Candidate Node into the DQSS; presumably based upon some criteria established for admission.

It is the latter act that serves as the basis for the moniker, “DQSS Semi-Manual Association Mode” since confirmation of inclusion requires an explicit action from an external source.
A DQSS configured to be in Promiscuous Association Mode has two methods for DQSS membership inclusion. As with all modes, the first method for inclusion into a DQSS is through configuration. The second method for inclusion into an existing DQSS is similar to the second inclusion method listed for DQSS Semi-Manual Association Mode; however, no operator intervention is required except for the case of an operator explicitly desiring to exclude a node from the DQSS.

Thus, the only time external intervention occurs during a DQSS operating in Promiscuous Association Mode is when an operator wishes to explicitly “blacklist” a candidate node; adding it to either a permanent blacklist or a blacklist that can be aged out. An example of a situation in which permanent blacklisting may be desired would be if a paid subscriber for XM Radio or other paid electronic subscription service was delinquent in paying their subscriber fees and/or had exceeded their usage. The subscriber could then be explicitly blacklisted until they brought their account current again and/or purchases additional time. An example of temporary blacklisting could occur as a result of a background task monitoring

network usage. If there was a limit as to the daily network activity for a particular subscriber and that subscriber had exceeded their limit, the Candidate Node of the subscriber could be placed on a blacklist that expired whenever their “lease” renewed again. While there are certainly other, potentially more cogent examples, each of the above examples sufficiently illustrates the viability of the blacklist exclusion capability.

Encryption may be used in any mode and can be implemented such that there is little, if any affect, as to how each Association Mode operates. There are two different types of encryption used within DQWA: Encrypted Private Key Mode; and Encrypted Public Key Mode. Both of these encryption methodologies will now be discussed in relation to their effects on operating modes. A DQSS configured to be in Encrypted Private Key Mode utilizes a symmetric encryption methodology with respect to both encrypting outgoing messages and decrypting incoming messages. Because both sides know what the decryption algorithm is, both sides may transmit the entire message encrypted, including the header. The clear implication with this mode is that the encryption/decryption algorithms must be done within the PHY in hardware in order for the three operating modes (Static, Semi-Manual, and Promiscuous) to operate oblivious to the effects of encryption performed on the encapsulated data.

A DQSS configured to be in Encrypted Public Key Mode utilizes an asymmetric encryption methodology with respect to the encryption of outgoing messages and decrypting incoming messages. Specifically, the shared (i.e. private) key is used for decrypting messages, but the public key must be utilized for encrypting messages. In this way, the entire message may be encrypted (as is done with Private (Shared) Key Mode), but the public key must be known in order to encrypt an outgoing message. Thus, nodes wishing to “join” the network, regardless of the configuration must “listen” to the Feedback Packet in order to get the Public Key before they can transmit. The cogent point here is that although the public key is broadcast, it is done so in encrypted form using the “Private” key; thus adding an additional layer of security to this process.

One of the advantages to this encryption mode to the automotive industry is that the public key could be provided to all legitimate parts vendors without sacrifice of security. The designated Cluster Head within a specific vehicle could then validate the part as valid or invalid according to the default configuration within the vehicle database. Not only would this serve the purpose of providing security to the vehicle insofar as normal traffic is concerned, it also ensures that only authorized parts may be used for a given vehicle type.

DQ supports Dynamic Clustering for the Control Point of DQ Network Topology. If Dynamic Clustering is disabled, the Cluster Head serves as the static control point of the vehicle DQSS network. Thus, if the static DQSS Cluster Head goes down, so does the DQ Network. However, if Dynamic Clustering is enabled, the Dynamic Cluster Head Designation Order will be included within the DQSS and updated separately on a periodic basis. There are multiple events that may trigger a Cluster Head Transition including traffic loading, hardware and/or power failures, energy consumption fairness criteria, or simply user discretion are a few of the more prominent events. Therefore, in order to support the various types of event triggers, there are multiple selections for the type of Cluster Topology configuration. The different Cluster Topology configuration types are listed below:

-
  - Clustering Disabled—The network is complete static, with one and only one node designated as the central control and arbitration point. Thus, if the Cluster Head fails, then the overall network connectivity also fails.
  - Clustering Enabled for Backup Only—So long as the network is operating normally, the network is completely static; with a single node designated as the Cluster Head. However, in the event the designated Cluster Head fails, a succession of backup Cluster Heads have been previously identified within the DQSS Table and thus assume the role of the Cluster Head according to their priority order and online status (i.e. the node that is both “online” and has the highest designated priority status becomes the Cluster Head if the current Cluster Head fails; if the highest designated priority status node is not online then the duty falls to the next lower designated priority status node). In the event there

are no nodes that are online and have been designated as a backup Cluster Head, the network connectivity fails.
- Limited Clustering Enabled—Normal Clustering is enabled for the network with this setting; however, only a limited set of designated nodes may participate as Cluster Heads.
- Clustering Enabled—Normal Clustering is enabled for the network, with all nodes eligible for Cluster Head designation.

As alluded to above, for clustering to occur within a DQSS not only must the overall Cluster Topology be specified, but so must the Clustering Methodology.
At present there are three distinct Clustering Methodologies: 1. Static Clustering; 2. Traffic Flow Clustering; and 3. Traffic Flow with Topology Coverage Clustering.
1) Static Clustering

-
- Regardless of the setting of the Cluster Topology for a given DQSS, if the Cluster Methodology is set to “Static Clustering”, then Dynamic Cluster is completely disabled. This is the only setting allowed for the “Clustering Disabled” and “Clustering Enabled for Backup Only” Cluster Topologies. If this setting is used for either the “Limited Clustering Enabled” or “Clustering Enabled” topologies, then the net effect is to force the overall network topology into that of “Clustering Enabled for Backup Only”.

2) Traffic Flow Clustering

-
- Traffic Flow Clustering enables the Cluster Head to be located at the node providing the most efficiency with respect to being a “gate keeper” of the traffic flow. Because all communication and control is distributed and is not routed through a central spoke in order to communicate with other nodes within the DQSS, the only real advantage to the Cluster Head moving as the flow moves would be if the gateway can move with it. Meaning, the Cluster Head nodes have dual functionality with one port servicing the DQSS and other ports servicing one or more gateways.

3) Traffic Flow with Topology Coverage Clustering

-
- Traffic Flow with Topology Coverage Clustering enables the Cluster Head to be located at the node providing the greatest coverage for the current traffic flow. The distinction between this mode and standard “Traffic Flow Clustering” is that the former does not take into account the overall range of coverage of the client nodes within the DQSS.

Similar to standard “Traffic Flow Clustering”, because all communication and control is distributed and is not routed through a central spoke in order to communicate with other nodes within the DQSS, the only real advantage to the Cluster Head moving as the flow moves would be if the gateway can move with it. Thus, as above, in order for this mode to be effective, Cluster Head nodes must have dual functionality with one port servicing the DQSS and other ports servicing one or more gateways. The Cluster Head distributes the DQSS table on a periodic basis. No node may communicate with another node unless both nodes are contained within the same DQSS.

Because of the strict adherence to this policy, in order for a node to join and subsequently communicate with other nodes, including the Cluster Head, within the DQSS, the following sequence of events may occur:

-

- a) The Cluster Head may explicitly acknowledge and admit a node for inclusion into the DQSS;
- b) The Cluster Head may then add it to the DQSS and perform either a complete or partial DQSS update of the DQSS Table to the nodes within the DQSS.

The Cluster Head may first admit the node in the network and then secondarily inform the other nodes in the DQSS of the joining node's admission into the DQSS. The format of the DQSS Table includes the following:

- 1) DQSS Configuration Data; providing information specifying the functional and operational makeup of the DQSS. Information included would be the DQSS Mode (i.e. Static, Manual, Promiscuous, Promiscuous-Shared Key), Encryption Indication, DQ Gateway Information, Maximum DQ Frame and DQ Packet Sizes;
- 2) 48-Bit MAC Address of every Node within the DQSS;
- 3) 12-Bit DQSS Address; this address is assigned by the Cluster Head to the individual nodes within the DQSS as a means of reducing the amount of overhead within the transmission stream;
- 4) Static Sized Code Word, assigned by the Cluster Head, and used for Access Requests to the

Transmission Queue. This value is coupled with the DQSS Address on all access requests;

- 5) Active or Inactive Indicators for Every DQ Member.

Given that the primary purpose of the DQSS Table is to maintain the integrity of the network, a DQSS Table should be viewed as an Object Oriented Encapsulation of a specific DQ Network.
The bandwidth in DQWA may be divided into fixed-size segments and groups of contiguous segments are allocated to each DQ Frame but many applications, such as a fuel injection module would be better served with the equivalent of a TDM channel. DQWA supports this feature; a node requests that a segment be allocated on a recurring basis resulting in an isochronous (TDM) channel of the desired bandwidth. This feature is of true significance since it means that DQWA can satisfy with equal facility both packet and fixed-bandwidth requirements.

Each DQ Data & Control Frame contains the total number of bytes within the frame at the beginning of the header; thus non-essential devices may go into a power save sleep mode for the period of the DQ Data & Control Frame transmission; awaking in time for the DQ Feedback Frame and inclusive DQ Transmission Beacon.

There is no congestion in a DQSA network thus networks may be designed for average loading of 90%. The surges over 100% that cause chaos in conventional routers just mean that the distributed queues get longer, temporarily.

There are no lost packets except for those lost due to Line Error. If only a single node has packets to send, that node can utilize 100% of the available capacity, when a second node desires to transmit, the available capacity is split automatically without any central control input, evenly between the two stations. And so on for an arbitrary number of stations. Priorities can be utilized to negate this inherent fairness.

The distributive and non-static control aspect of DQWA affords DQWA to be used “As Is” within environments requiring mission critical and/or fail-safe architectures and without any additional redundancies in the network. Unlike conventional Hub-and-Spoke architectures, the current DQWA control node within a given DQWA network may fail without affecting the communication abilities of the remaining nodes within the DQSA network. In short, DQWA eliminates the single point of failure, which is common in all commercial network architectures deployed today. This is huge benefit that Mission and

Safety Critical applications a built-in mechanism within the network architecture for supporting their specific application. A DQWA network becomes part of the Mission and/or Safety Critical Solution and not another problem for which a work-around must be found (usually involving duplicate and/or alternative hardware and communication paths).

The distributive and non-static (i.e. transitional) control aspect of DQWA affords DQWA to be used “As Is” within environments requiring mission critical and/or fail-safe architectures (like that necessitated within the automotive domain) and without any additional redundancies in the network. Further, given the increasing security needs of automotive onboard network devices and the ubiquitous and pervasive nature of CAN; DQWA would be an excellent complimentary technology for wireless CAN networks; particularly as a wireless CAN backhaul topology.

Distributed Queuing Wireless Arbiter (DQWA)
Referring to FIGS. 38 and 39, DQWA is a broadcast medium MAC Layer Protocol and PHY Interface that is carrier independent and is specifically designed to be a wireless back-haul solution for the transportation of both mobile telephony data and TCP/IP network data.
DQWA may provide the following advantages over the systems known in the art:

-
  - 1) Non-LoS Support (requires Dual Antenna)
  - 2) Increased Bandwidth Utilization—bandwidth efficiency up to 95%.
  - 3) Organic Network Organization (capability to assemble and grow automatically)
  - 4) Built-in Redundancy of Network Control Mechanisms
  - 5) Direct Peer-to-Peer communication for nodes within same service set (i.e. local network); meaning no retransmission by central control required.
  - 6) Built-in capability for energy efficiency.
  - 7) No physical network size restriction (can be adapted for any number of nodes).
  - 8) Carrier and Modulation independent—designed for adaptation to virtually any carrier, modulation, and data rate.

Referring to FIGS. 40 and 41, DQWA may allow backhaul providers to quickly augment existing infrastructure with equipment that is easy to install and configure (self-configuring if enabled) while being more efficient than other comparable solutions (such as Wi-Max).

DQWA Backhaul Technology may be an alternative to both traditional Point-to-Point (P2P) backhaul and Star Topology solutions. With a DQWA system, the data moving between a Micro-cell Aggregation Point (termed, Cluster Node) and the Macro Cell Aggregation Point (termed, Cluster Head Node) may pass through a neighbor Micro-cell Aggregation Point before reaching the Macro Cell Aggregation Point. This ‘multi-hop’ function provides an extended array of data routing options to overcome LoS restrictions from that of a traditional P2P or even Star Topology Solution.

The advantages of a DQWA backhaul solution are numerous and sizable. Of primary importance is the potential ability to deploy Pico-cells wherever and however the carrier desires without concern for LoS limitations or fiber/copper run cost considerations. With Siting and backhaul comprising the large majority of pico-cell deployment costs, DQWA may bring a key CAPEX reduction to the operator. DQWA systems may reduce the average RF link distances; hence reducing the radio & antenna costs and further reducing backhaul CAPEX. And as DQWA systems select the ‘best-path’ route, network reliability is increased and OPEX is reduced. Reliability may also be gained through the flexible nature of DQWA as a result of the fact that the Micro Cell Base Station does not need to be a single fixed node and may in fact transition from node to node within the Pico-Cell. Thus, allowing automatic recovery if the primary Micro Cell Base station should fail.

FIG. 42 depicts an exemplary embodiment of DQ Transmission sequence according to the present application.
DQWA—Common Terms:

-
  - Access Request Sequence (ARS)—The ARS occurs within the Contention Window Segment and consists of three mini-slots within the segment acting as elements of the Contention Queue.
  - Cluster Head—The Cluster Head is the central and only arbiter for a specific DQSS.
  - Cluster Head Master—The preferred Cluster Head within a given DQSS.
  - Cluster Head Priority—The predefined priority of nodes that may assume the role as Cluster Head.
  - Cluster Node—Any node within the DQSS that is NOT the Cluster Head.
  - Contention Queue—FIFO Queue used by DQSS for candidacy into Transmission Queue.
  - Contention Window Segment—The Cluster Head is the central and only arbiter for a specific DQSS.
  - Distributed Queuing Service Set (DQSS)—Collection of nodes that are defined to be within a specific DQ Network.
  - DQ Payload & Control Packet Segment—This segment encapsulates Data and optional Control Information.
  - Feedback Packet (FBP) Segment—This segment encapsulates Data and optional Control Information from the Cluster Head and serves as a Transmission Beacon for the DQSS.
  - Transmission Queue—FIFO Queue with Optional Priorities used by DQSS to maintain order of scheduled transmissions.
  - Transmission Sequence—Term describing the complete sequence of the standard periodic transmission that occurs within a DQSS network. The Transmission Sequence is delineated into three separate and contiguous segments (listed below in the order of their appearance):
    - Contention Window Segment
    - Payload & Control Packet Segment
    - Feedback Packet Segment
  - ARS Contention Window—Refers to the period of time within the DQ Transmission Sequence in which nodes may contend for access to the DQ Transmission Queue.
  - ARS Mini-Slot—Refers to the period of time within the DQ Transmission Sequence in which nodes may contend for access to the DQ Transmission Queue.
  - DQSS ARS Segment—Refers to the first segment within the DQ Transmission Sequence, which is when nodes may request access to the DQSS' Transmission Queue.
  - DQSS Feedback Packet Segment—Refers to third and final segment within the Transmission Sequence, which is where the node acting as the Cluster Head provides feedback to the nodes within the DQSS. This is also where it may preempt both ongoing transmissions as well as upcoming and previously scheduled transmissions in favor of higher priority transmissions.
  - DQ Frame—Refers to collection of one or more DQ Control and Payload Packets; when application data is included within the collection of packets, the DQ Frame represents a single complete logical unit of encapsulated application data.
  - DQ Control and Data Payload Packet Segment—Refers to the middle segment within the DQWA Transmission Sequence. This segment can carry both control and payload information within.
  - DQ Segment—Refers to one of three logically distinct delineations within a DQ Transmission Sequence (listed as follows):
    - Access Request Sequence Segment;
    - DQ Control and Payload Packet Segment;
    - Feedback Packet Segment.
  - DQ Service Set (DQSS)—Refers to a set of nodes within a DQ Network that share a common peer-to-peer communication medium and are managed by a single authority that utilizes queues to control access to the DQ Network.
  - DQ Transmission Sequence—Refers the complete sequence of the three DQ Segments (i.e. ARS, Payload, Feedback) repeatedly, consistently, and always occurring in every DQWA transmission.

- Feedback Window—Refers to the period of time within the DQ Transmission Sequence in which the Access Point or Cluster Head provides feedback to the nodes within the DQSS.
- Queue Transmission Window—Refers to the period of time within the DQ Transmission Sequence in which the node at the top of the Transmission Queue is afforded the opportunity to transmit.

ACK Acknowledgment
ACK_AL Acknowledgment in Active Listening
AL Active Listening
AP Access Point
ARQ Automatic Retransmission/Repeat Request
ARS Access Request Sequence
BEB Binary Exponential Back-off
C-ARQ Cooperative ARQ
CCA Clear Channel Assessment
CDMA Code Division Multiple Access
CFC Call for Cooperation
CRC Cyclic Redundancy Code
CRQ Collision Resolution Queue
CSMA Carrier Sensing Multiple Access
CSMA/CA Carrier Sensing Multiple Access with Collision Avoidance
CTS Clear to Send
DBE Detailed Balance Equations
DCF Distributed Coordination Function
DIFS DCF Inter Frame Space
DPCF Distributed Point Coordination Function
DQ Distributed Queuing
DQCOOP DQMAN for Cooperative ARQ
DQMAN Distributed Queuing MAC protocol for Ad Hoc Networks
DQWA Distributed Queue Wireless Arbiter
DSSS Direct Sequence Spread Spectrum
DTQ Data Transmission Queue
ED Error Detection
FBP Feed-Back Packet
FCS Frame Check Sequence
FEC Forward Error Correction
GUI Graphic User Interface
IMSI Initial Master Sensing Interval
IEEE Institute of Electrical and Electronics Engineers
ISM Industrial, Scientific, and Medical free-license band
ISO International Standards Organization
LAN Local Area Network
MAC Medium Access Control
MACSWIN The MAC Simulator for Wireless Networks
MCR Master Cooperation Request
MCS Message Check Sequence
MIFS Maximum Inter Frame Space
MIMO Multiple Input Multiple Output
MRAC Multiple Relay Access Control
MSP Master Selection Phase
MSS Master Service Set
MSSI Master Sensing Selection Interval
MTO Master Time-Out
NAV Network Allocation Vector
OSI Open System Interconnection
PAN Personal Area Network

- PCF Point Coordination Function
- PDA Personal Digital Agenda
- PLCP PHY Layer Convergence Procedure
- PHY Physical Layer
- PIFS PCF Inter Frame Space
- QoS Quality of Service
- RTS Request to Send
- SIFS Short Inter Frame Space
- SNIR Signal to Noise plus Interference Ratio
- SNR Signal to Noise Ratio
- STC Space-Time Codes
- TDMA Time Division Multiple Access
- WLAN Wireless LAN
- WWRF Wireless World Research Forum

Automotive Industry:
In one exemplary embodiment, the DQWA may be applied in the automotive industry. The DQWA is ideal for applications requiring distributed communication and control, of which the automotive world certainly falls into that category. In short, DQWA adds the ability to simplify intra-vehicle connectivity while expanding overall communication capabilities.

The CAN protocol has served the automotive and related industries well for over twenty-five (25) years; with the original CAN protocol officially released in 1986 followed by the release of CAN 2.0 in 1991. Since then many variants and improvements in CAN combined with the proliferation of automotive onboard microprocessor based sensors and controllers have resulted in CAN establishing itself as the dominant network architecture for automotive onboard communication in layers one (1) and two (2). Going forward however, the almost exponential growth of automotive onboard computing and the associated devices necessary for supporting said growth will unfortunately necessitate an equivalent growth in the already crowded wired physical infrastructure unless a suitable wireless alternative can be provided.

While a wireless implementation of CAN has been produced, it has never obtained real traction within the automotive world. Other alternative methodologies for providing wireless connectivity have been much more pervasive and accepted, but none of them provide anything more to CAN interfaces than a CAN-to-Wireless Bridge; with Wi-Fi, Blue Tooth, and GSM being the primary wireless network architectures bridging to CAN.

Contrary to prior art, present application provides more than simply a wireless extension of CAN in that it does more than extend CAN into the wireless domain (as was the case with CANRF). As pure wireless CAN with no accommodations for heavy utilization would only exacerbate CAN's primary deficiency of starving out lower priority messages; since there would be no way to isolate devices in sub-networks as could be done with a wired infrastructure.

Embodiment presently disclosed remove CAN's deficiency by modifying the newly defined wireless network protocol and architecture, DQWA (Distributed Queuing Wireless Arbiter) to not only extend CAN into the wireless domain, but also addresses CAN's more prominent shortcomings.
Recognizing the proliferation of devices with network connectivity within vehicles is going to continue escalating; it is logical to look for a means to facilitate this expansion without an equivalent expansion in wired infrastructure. Anyone who has looked under the hood of a vehicle from the 70's and then compared that to what is under the hood today must wonder where the space for any additional infrastructure is going to come from.

The same is true for under the dashboard and/or in the trunk with respect to entertainment systems. Consumers want more space, not less; they want their technologic advances without paying the price in either comfort or cost. The only foreseeable path to that end is a wireless one. It is this path that brings fewer wires; lower costs; easier installation; greater capabilities for expansion. DQWA is a solution that

provides both security and reliability within a wireless framework, while maintaining CAN's distributed network communication methodology and implicit avoidance of single points of failure within the network. Given the proliferation of network devices in people's daily lives, it is only logical to deduce a similar growth pattern within vehicles. As that growth pattern continues, it will become increasingly difficult to depend so heavily on a wired infrastructure for providing communication connectivity within the vehicle. Of greater significance will be the proliferation of automotive onboard devices that will be expected to communicate externally; particularly with respect to both personal data derived from the human passengers as well as vehicular data exchanged with vehicular traffic management technology both fixed and potentially with other vehicles. It is clear for many reasons, both because of the physical limitations, difficulty, and expense of installing and maintaining wired bus infrastructures that the necessity of a wireless alternative is inevitable.

The primary weakness in attempting to utilize CAN within a heavily utilized bus is the propensity for lower priority messages to be starved out and hence never sent; or sent too late to be of any use. Obviously, if CAN is to be deployed within a wireless environment then this weakness becomes a severe problem given that it will become difficult for CAN nodes to form a sub-network within the same vehicle; not to mention potential interference from external sources, including CAN nodes broadcasting on the same frequency in nearby vehicles. Even if adequate RF shielding and filtering techniques are utilized within the vehicle chassis to maintain successful RF communication; given the limited number of available frequencies, a methodology would still need to be employed that would facilitate coexistence with other nodes broadcasting on the same frequency within the vehicle; particularly with respect to access to the bus' transmission queue. Also, given the real-time, mission and safety critical nature of automotive communication, reliability and robustness must be key considerations in any deployed networking methodology supporting automotive communication.

Given that by definition wireless communication is ubiquitously broadcast, security becomes a crucial concern. Examples of such concern consists both of those from listening in violating both privacy and network security as well as those attempting to gain unwanted access over the network devices within the network (ex. either by either directly manipulation of the devices or by indirect manipulation via the spoofing of existing devices within the network). Additionally, as more and more automotive modules require intra-vehicle network connectivity, wireless becomes the only viable alternative. The challenge is to enable the transition to wireless connectivity, reliably, safely, and most of all securely. DQWA provides the answer to this increasingly important and difficult problem.
An exemplary Distributed Queuing Wireless Arbiter (DQWA) PHY and MAC Protocol Specification according to the present application is provided next.

“Distributed Queuing Wireless Arbiter (DQWA) PHY & MAC Protocol Specification”1.0 Objective and Scope

The philosophical premise of this document is to take the DQ Protocol MAC & PHY beyond the theoretical realm and move it squarely into the application and development reality. Resulting from that directive, there are two stated primary objectives for this document.

1.1 Define the Distributed Queuing Wireless Arbiter (DQWA) Protocol

The first objective is to describe and specify the DQWA Protocol MAC & PHY in sufficient enough detail so that any two implementations resulting from the aforementioned specification are 100% interoperable. In fulfilling the first objective, much of this document is spent in fully defining the DQWA Protocol. While DQWA is designed to outperform most, if not all, current wireless environments including 802.11 based technologies; particular attention is given to honing the DQWA protocol for its initial target market as a Wireless Mobil Backhaul Technology primarily servicing countries without significant copper and/or fiber communication infrastructure. It is with that in mind that the first full draft of DQWA has been designed.

Additionally, the reader will note that while Wireless Mobile Backhaul is the primary target, DQWA also has features specifically designed in to work with and as a replacement for mobile last mile solutions. The

premise of such thinking is that deploying a technology that can be used across a broad spectrum of applications (i.e. mobile backbone, last mile, and even WLAN if desired) means lower cost, easier deployment, and greater bandwidth.

Lastly, so much of the world is moving towards automating environments that require mission and/or safety critical applications. It just so happens that the primary concern within these environments is eliminating single points of failure. Whenever a network is involved in such an environment, the only mechanism for achieving that is duplication. Fortunately, because the Cluster Head within a DQWA Network can move, with any node being capable of assuming Cluster Head responsibilities; very little needed to be added in order to take advantage and utilize the distributive nature of the DQWA network for this purpose.

1.2 Provide Technology Plan and Associated Implementation Outline
The first objective is to provide a Technology Plan and associated Implementation Schedule outline that:

-
  - Specify the individual features to be implemented.
  - Specify the implementation order of those features.
  - Specify which features are not covered within the scope of this document.
  - Specify the future direction of the DQWA MAC & PHY Technology.

1.3 Objective and Scope Conclusion
The second objective is to provide a Technology Plan and associated Implementation. The expectation of achieving both stated objectives (i.e. defining the protocol and Technology Plan) will enable consistency for both implementers and users alike.

2.0 Background and Related Information

The Distributed Queuing Wireless Arbiter (DQWA) Protocol is based on the Distributed Queue Switch Architecture (DQSA) developed at the Illinois Institute of Technology. The heart of this technology is a medium access control (MAC) that allows an arbitrary number of stations to share a common communications channel over any distance and operating at any data rate. DQSA can operate over virtually any topology and will also provide a Quality of Service (QoS) superior to any currently available. The key feature of DQSA is that all control resides in the stations, no central control is required. The network state is maintained at all times by each station in just two (2) binary counters per DQ Service Set (DQSS), providing it with all the information necessary to make decisions as to when to transmit for that specific DQSS. A DQ Transmission Frame is divided into three separate time periods or segments; with the three segments listed below:

-
  - 1) Contention Window, utilized as part of the Access Request Sequence (ARS) to the Transmission Queue with three (3) control mini-slots acting as a finite sized Contention Queue;
  - 2) Data and Control Window consisting of a single DQ Data and Control Frame; and,
  - 3) Feedback Window, consisting of the DQ Feedback Frame with Synchronization Beacon.

The only “central” control required is that a synchronization beacon must be transmitted to all stations prior to the start of each segment from which all stations must synchronize with for every transmission frame so that they may participate in the DQSS. The Feedback Packet and associated Synchronization Beacon can come from any node within the DQSS, but is always sent by a single node at any given time and from which the node is typically chosen as one of a set of nodes designated for accessing gateways beyond the DQSS. Within a wireless environment, this central point would normally be referred to as the Base Station, Access Point, or Hub.

Variable length packets may be segmented into multiple data slots without requiring any further overhead. Qualities of Service (QoS) Priorities are available and it is possible for a higher priority packet to preempt a lower priority packet during transmission within a period of one Transmission Sequence.
Segments can be allocated to a specific station thus providing time-division-multiplex (TDM) channels, commingled with packet traffic. The overall utilization within a wireless environment, i.e., ratio of data slot content to capacity of channel will range from over 95% down to 80%, depending upon frame size and overall network utilization.

Lastly, because access to communication within a DQSA service set consists solely of member nodes, the entire contents within a MAC layer frame, including the header, may be encrypted; thus ensuring the utmost of both security and privacy.

In addition to the original work done by Graham Campbell, Ph.D., as referenced in [1] and [2], acknowledgements and credit should be given to Luis Alonso, PhD, Jesús Alonso Zárate, PhD, and their research team at the Polytechnic University of Catalonia, Spain. In addition to heavy dependence on their many papers, many of which are published by the IEEE; they have also provided a significant amount of time, feedback, and guidance in defining the DQWA Protocol discussed and detailed within this document. Thus, while every instance is not cited, all relevant documents used as research material have been cited within the index section of this document; with much attention given to the documents directly focused on the protocol, Distributed Queuing with Collision Avoidance (DQCA) (i.e. [4], [5]).

3.0 Glossary of Terms and Acronyms

3.1 Acronyms

-
  - ACK Acknowledgment
  - ACK_AL Acknowledgment in Active Listening
  - AL Active Listening
  - AP Access Point
  - ARQ Automatic Retransmission/Repeat Request
  - ARS Access Request Sequence
  - BEB Binary Exponential Back-off
  - C-ARQ Cooperative ARQ
  - CCA Clear Channel Assessment
  - CDMA Code Division Multiple Access
  - CFC Call for Cooperation
  - CRC Cyclic Redundancy Code
  - CRQ Collision Resolution Queue
  - CSMA Carrier Sensing Multiple Access
  - CSMA/CA Carrier Sensing Multiple Access with Collision Avoidance
  - CTS Clear to Send
  - DBE Detailed Balance Equations
  - DCF Distributed Coordination Function
  - DIFS DCF Inter Frame Space
  - DPCF Distributed Point Coordination Function
  - DQ Distributed Queuing
  - DQCOOP DQMAN for Cooperative ARQ
  - DQMAN Distributed Queuing MAC protocol for Ad Hoc Networks
  - DQWA Distributed Queue Wireless Arbiter
  - DSSS Direct Sequence Spread Spectrum
  - DTQ Data Transmission Queue
  - ED Error Detection
  - FBP Feed-Back Packet
  - FCS Frame Check Sequence
  - FEC Forward Error Correction

- GUI Graphic User Interface
- IMSI Initial Master Sensing Interval
- IEEE Institute of Electrical and Electronics Engineers
- ISM Industrial, Scientific, and Medical free-license band
- ISO International Standards Organization
- LAN Local Area Network
- MAC Medium Access Control
- MACSWIN The MAC Simulator for Wireless Networks
- MCR Master Cooperation Request
- MCS Message Check Sequence
- MIFS Maximum Inter Frame Space
- MIMO Multiple Input Multiple Output
- MRAC Multiple Relay Access Control
- MSP Master Selection Phase
- MSS Master Service Set
- MSSI Master Sensing Selection Interval
- MTO Master Time-Out
- NAV Network Allocation Vector
- OSI Open System Interconnection
- PAN Personal Area Network
- PCF Point Coordination Function
- PDA Personal Digital Agenda
- PLCP PHY Layer Convergence Procedure
- PHY Physical Layer
- PIFS PCF Inter Frame Space
- QoS Quality of Service
- RTS Request to Send
- SIFS Short Inter Frame Space
- SNIR Signal to Noise plus Interference Ratio
- SNR Signal to Noise Ratio
- STC S pace-Time Codes
- TDMA Time Division Multiple Access
- WLAN Wireless LAN
- WWRF Wireless World Research Forum

3.2 Terms

- ARS Contention Window—Refers to the period of time within the DQ Transmission Sequence in which nodes may contend for access to the DQ Transmission Queue.
- ARS Mini-Slot—Refers to the period of time within the DQ Transmission Sequence in which nodes may contend for access to the DQ Transmission Queue.
- DQSS ARS Segment—Refers to the first segment within the DQ Transmission Sequence, which is when nodes may request access to the DQSS' Transmission Queue.
- DQSSFeedback Packet Segment—Refers to third and final segment within the Transmission Sequence, which is where the node acting as the Cluster Head provides feedback to the nodes within the DQSS. This is also where it may preempt both ongoing transmissions as well as upcoming and previously scheduled transmissions in favor of higher priority transmissions.
- DQ Frame—Refers to collection of one or more DQ Control and Payload Packets; when application data is included within the collection of packets, the DQ Frame represents a single complete logical unit of encapsulated application data.
- DQ Control and Data Payload Packet Segment—Refers to the middle segment within the DQWA Transmission Sequence. This segment can carry both control and payload information within it.
- DQ Segment—Refers to one of three logically distinct delineations within a DQ Transmission Sequence (listed as follows):
    - Access Request Sequence Segment;

    - DQ Control and Payload Packet Segment;
    - Feedback Packet Segment.
- DQ Service Set (DQSS)—Refers to a set of nodes within a DQ Network that share a common peer-to-peer communication medium and are managed by a single authority that utilizes queues to control access to the DQ Network.
- DQ Transmission Sequence—Refers the complete sequence of the three DQ Segments (i.e. ARS, Payload, Feedback) repeatedly, consistently, and always occurring in every DQWA transmission.
- Feedback Window—Refers to the period of time within the DQ Transmission Sequence in which the Access Point or Cluster Head provides feedback to the nodes within the DQSS.
- Queue Transmission Window—Refers to the period of time within the DQ Transmission Sequence in which the node at the top of the Transmission Queue is afforded the opportunity to transmit.

4.0 Introduction to Distributed Queuing

Distributed Queuing as defined within this document describes a Layer 2 Protocol and PHY Transmission scheme that is agnostic to the underlying carrier. The initial and primary technology medium reaping the largest benefit from this technology is in the wireless realm; although there is no reason that it could not be equally applicable in a wire line based medium as well. The initial targeted benefit is as a Wireless Mobile Backhaul solution as well as a potential alternative to the entire series of wireless **802** based technologies, with specific attention to 802.11; while still being able to maintain coexistence with one of the very technology targets it is designed to replace.

Coexistence is not automatic; an implementer of DQ would have to design their product with coexistence explicitly set out as a goal. Essentially, some portion of the time would be spent processing DQ frames and the remainder of the time would be spent processing the 802.11 (or whatever other MAC it was replacing) for the remainder of the time.

The packet and frame formats have been specifically designed to take advantage of the relative collision free environment in the data content portion of the packet segment. Thus, there are two basic types of record keeping header formats:

-
    - Those that are sent during every transmission sequence, otherwise known as packet segments.
    - Those that are sent only for an entire frame, which can and often does span multiple packet segments.

The DQ Frame Header contains information normally found within an 802.11 type frame, but with one additional address in the event forwarding is necessary by either an address within the Distributed Queuing Service Set (DQSS) or to the greater network cloud beyond the Cluster Head.

The address types are listed below:

-
    - Immediate DestinationDQ Network Address;
    - Immediate SourceDQ Network Address;
    - Cluster HeadDQ MAC Address;
    - Actual Destination DQ MAC Address;
    - Original Source DQ MAC Address.

Only the first three addresses are required within normal DQ Frames; with the latter two addresses only necessary whenever forwarding is required beyond the current Distributed Queue Service Set (DQSS). A DQ Transmission Sequence, is depicted in FIG. 1.

The DQ Transmission Sequence is divided into three separate segments (not counting the interval spacing):

- 1) the DQSS Access Request Sequence (ARS) Segment (also known as the “ARS Contention Window);
- 2) the DQ Control & Payload Segment (also known as the Queue Transmission Window);
- 3) and, the DQSS Feedback Packet Segment (also known as the Feedback Window).

Below is a brief overview of each segment:

- DQSS ARS Segment—This segment, which is actually divided into three (3) subsegments, enables nodes within the DQSS with the ability to request permission for exchanging data with other nodes, including the Cluster Head.
- DQ Control& Payload Segment—This segment represents both the addressing of the affected nodes exchanging data as well as the actual data itself. DQ Management Commands, Replies, and Requests are also communicated within this segment.
- DQSS Feedback Packet Segment—This segment provides feedback representing DQSS Management & Record Keeping that is almost always in direct response to information contained in the immediate prior two (2) segments. It also has the intended side-effect of serving as a beacon, as it is transmitted at the end of every frame and should be used for synchronization purposes.

Up to five different nodes can successfully participate within a single transmission sequence; three within the ARS Segment with one per mini-slot, a fourth one within the DQ Control & Payload Segment, and finally a fifth from the Cluster Head within the Feedback Packet segment. FIG. 2 depicts an example of a successful Transmission Sequence with five disparate transmitters.
The Protocol, MAC, and other operational aspects will now be explained in more detail.

5.0 Distributed Queuing Operational Methodology

Like, the Basic Service Set in 802.11, DQ has a similar methodology in that a DQ Service Set can be viewed as a set of nodes within a network that communicate with each other while sharing a common distributed network that is managed by a central controlling authority, either an Access Point or a Cluster Head. NOTE: Because DQ is by definition a distributed architecture, communication is therefore peer-to-peer even though “control” is centralized. What this means in practice is that the Cluster Head dictates which nodes have access to the queue; but all communication within the network is peer-to-peer.

5.1 DQ Service Set Modes

A key component of the DQ Service Set concept is network security and the rules by which nodes may become members of a specific DQ Service Set. A DQSS can operate in one of three operational modes listed below the operational modes listed in decreasing order of centralized membership control:

- Static Association Mode;
- Semi-Manual Association Mode;
- Promiscuous Mode;

Each of the modes will now be individually discussed in detail.

5.1.1 DQSS Static Association Mode

In Static Association Mode, the DQ Service Set is completely pre-configured. New nodes may not request to join and can only become part of the DQSS either by directly adding nodes to an existing DQSS Configuration Database or by installing a completely new DQSS Configuration Database containing the desired nodes.
In response to the fact that a DQSS configured in Static Association Mode cannot add nodes in real time (doing so only through configuration); any attempt to submit a DQSS Membership Request Code Word during the ARS segment will be ignored.

5.1.2 D CMS Semi-Manual Association Mode

A DQSS configured to be in Semi-Manual Association Mode has all of the capabilities of a Static Association Mode DQSS as well as the additional ability to add nodes in real time. There are two methods for which a node may acquire inclusion within a DQSS configured in DQ Semi-Manual Association Mode.
The first method for acceptance for a given node into a DQSS while in DQSS Semi-Manual Association Mode is via manual configuration as part of a DQSS Configuration Database. The second method utilizes a two-step process for any node outside of the current DQSS membership and described below:

-
  - 1) First, the Candidate Node must issue a request for DQSS Inclusion.
  - 2) Second, an external confirmation of the request from either an operator or configuration robot utility must explicitly accept the Candidate Node into the DQSS; presumably based upon some criteria established for admission. It is the latter act that serves as the basis for the moniker, “DQSS Semi-Manual Association Mode” since confirmation of inclusion requires an explicit action from an external source; presumably an operator or configuration robot utility.

5.1.3 DQSS Promiscuous Association Mode
A DQSS configured to be in Promiscuous Association Mode has two methods for DQSS membership inclusion. As with all modes, the first method for inclusion into a DQSS is through configuration.
The second method for inclusion into an existing DQSS is similar to the second inclusion method listed for DQSS Semi-Manual Association Mode; however, no operator intervention is required except for the case of an operator explicitly desiring to exclude a node from the DQSS.
Thus, the only time operator intervention occurs during a DQSS operating in Promiscuous Association Mode is when an operator wishes to explicitly “blacklist” a candidate node; adding it to either a permanent blacklist or a blacklist that can be aged out.
An example of a situation in which permanent blacklisting may be desired would be if a paid subscriber within a physical locality like an Internet Café was delinquent in paying their subscriber fees and/or had exceeded their usage. The subscriber could then be explicitly blacklisted until they brought their account current again and/or purchases additional time.
An example of temporary blacklisting could occur as a result of a background task monitoring network usage. If there was a limit as to the daily network activity for a particular subscriber and that subscriber had exceeded their limit, the Candidate Node of the subscriber could be placed on a blacklist that expired whenever their “lease” renewed again.
While there are certainly other, potentially more cogent examples, each of the above examples sufficiently illustrates the viability of the blacklist exclusion capability.
5.2 DQSS Encryption Modes
Encryption may be used in any mode and can be implemented such that there is little, if any affect, as to how each Association Mode operates. There are two different types of encryption used within DQWA:

-
  - Encrypted Private Key Mode.
  - Encrypted Public Key Mode.
    Both of these encryption methodologies will now be discussed in relation to their effects on operating modes.

5.2.1 DQSS Encrypted Private (Shared) Key Mode
A DQSS configured to be in Encrypted Private Key Mode utilizes a symmetric encryption methodology with respect to both encrypting outgoing messages and decrypting incoming messages. Because both sides know what the decryption algorithm is, both sides may transmit the entire message encrypted, including the header.
The clear implication with this mode is that the encryption/decryption algorithms must be done within the PHY in hardware in order for the three operating modes (Static, Semi-Manual, and Promiscuous) to operate oblivious to the effects of encryption performed on the encapsulated data.

5.2.2 DQSS Encrypted Public Key Mode

A DQSS configured to be in Encrypted Public Key Mode utilizes an asymmetric encryption methodology with respect to the encryption of outgoing messages and decrypting incoming messages. Specifically, the shared (i.e. private) key is used for decrypting messages, but the public key must be utilized for encrypting messages. In this way, the entire message may be encrypted (as is done with Private (Shared) Key Mode), but the public key must be known in order to encrypt an outgoing message. Thus, nodes wishing to "join" the network, regardless of the configuration must "listen" to the Feedback Packet in order to get the Public Key before they can transmit. The cogent point here is that although the public key is broadcast, it is done so in encrypted form using the "Private" key; thus adding an additional layer of security to this process.

5.3 Dynamic Clustering

DQ supports Dynamic Clustering for the Control Point of DQNetwork Topology. If Dynamic Clustering is disabled, the Cluster Head serves as the static control point. Thus, if the Access Point goes down, so does the DQ Network. However, if Dynamic Clustering is enabled, the Dynamic Cluster Head Designation Order will be included within the DQSS and updated separately on a periodic basis.

There are multiple events that may trigger a Cluster Head Transition including traffic loading, hardware and/or power failures, energy consumption fairness criteria, or simply user discretion are a few of the more prominent events. Therefore, in order to support the various types of event triggers, there are multiple selections for the type of Cluster Topology configuration. The different Cluster Topology configuration types are listed below:

-
  - Clustering Disabled—The network is complete static, with one and only one node designated as the Access Point. Thus, if the Access Point fails, then so does the network connectivity.
  - Clustering Enabled for Backup Only—So long as the network is operating normally, the network is completely static; with a single node designated as the Access Point. However, in the event the designated Access Point fails, a succession of backup Access Points has been previously identified within the DQSS Table and thus assume the role of the Access Point according to their priority order and online status (i.e. the node that is both "online" and has the highest designated priority status becomes the Access Point if the current Access Point fails; if the highest designated priority status node is not online then the duty falls to the next lower designated priority status node). In the event there are no nodes that are online and have been designated as a backup Access Point, the network connectivity fails.
  - Limited Clustering Enabled—Normal Clustering is enabled for the network with this setting; however, only a limited set of designated nodes may participate as Cluster Heads.
  - Clustering Enabled—Normal Clustering is enabled for the network, with all nodes eligible for Cluster Head designation.

As alluded to above, for clustering to occur within a DQSS not only must the overall Cluster Topology be specified, but so must the Clustering Methodology.

5.3.1 Clustering Methodologies

At present there are three distinct Clustering Methodologies:

-
  - 1. Static Clustering;
  - 2. Traffic Flow Clustering; and,
  - 3. Traffic Flow with Topology Coverage Clustering.

More Clustering Methodologies may be added over time; but these three represent the initial set. Each of the three Clustering Methodologies will now be discussed.

5.3.1.1 Static Clustering

Regardless of the setting of the Cluster Topology for a given DQSS, if the Cluster Methodology is set to “Static Clustering”, then Dynamic Cluster is completely disabled. This is the only setting allowed for the “Clustering Disabled” and “Clustering Enabled for Backup Only” Cluster Topologies. If this setting is used for either the “Limited Clustering Enabled” or “Clustering Enabled” topologies, then the net effect is to force the overall network topology into that of “Clustering Enabled for Backup Only”.

5.3.1.2 Traffic Flow Clustering
Traffic Flow Clustering enables the Cluster Head to be located at the node providing the most efficiency with respect to being a “gate keeper” of the traffic flow. Because all communication and control is distributed and is not routed through a central spoke in order to communicate with other nodes within the DQSS, the only real advantage to the Cluster Head moving as the flow moves would be if the gateway can move with it. Meaning, the Cluster Head nodes have dual functionality with one port servicing the DQSS and other ports servicing one or more gateways.

5.3.1.3 Traffic Flow with Topology Coverage Clustering
Traffic Flow with Topology Coverage Clustering enables the Cluster Head to be located at the node providing the greatest coverage for the current traffic flow. The distinction between this mode and standard “Traffic Flow Clustering” is that the former does not take into account the overall range of coverage of the client nodes within the DQSS.
Similar to standard “Traffic Flow Clustering”, because all communication and control is distributed and is not routed through a central spoke in order to communicate with other nodes within the DQSS, the only real advantage to the Cluster Head moving as the flow moves would be if the gateway can move with it. Thus, as above, in order for this mode to be effective, Cluster Head nodes must have dual functionality with one port servicing the DQSS and other ports servicing one or more gateways.

5.4 Additional DQ Service Set Rules
The Access Point or Cluster Head distributes the DQ Service Set on a periodic basis. No node may communicate with another node unless both nodes are contained within the same service set. Because of the strict adherence to this policy, in order for a node to join and subsequently communicate with other nodes, including the Cluster Head, within the DQSS, the following sequence of events must occur:

-
  - a) The Access Point or Cluster Head must explicitly acknowledge and admit a node for inclusion into the DQSS;
  - b) The Access Point or Cluster Head must then add it to the DQSS and perform either a complete or partial DQSS update of the DQSS Table to the nodes within the DQSS.

When possible, the Cluster Head will update the DQSS Table through update distributions as a means of saving time and bandwidth. There are few instances in which a complete DQSS distribution will occur, with the nominal occurrence being during initialization and start-up of the DQSS.
In short, the Cluster Head must first admit the node in the network and then secondarily inform the other nodes in the DQSS of the joining node's admission into the DQSS. The format of the DQSS Table is defined in section 9.1 on the “Distribute DQ Service Set Table (0x01)” command and includes the following:

-
  - DQSS Configuration Data; providing information specifying the functional and operational makeup of the DQSS. Information included would be the DQSS Mode (i.e. Static, Manual, Promiscuous, Promiscuous-Shared Key), Encryption Indication, DQ Gateway Information, Maximum DQ Frame and DQ Packet Sizes,
  - 48-Bit MAC Address of every Node within the DQSS.
  - 12-Bit DQSS Address; this address is assigned by the Cluster Head to the individual nodes within the DQSS as a means of reducing the amount of overhead within the transmission stream.

- 20-bit Code Word, assigned by the Cluster Head, and used for Access Requests to the Transmission Queue. This value is coupled with the DQSS Address on all access requests.
- Active or Inactive Indicators for Every DQ Member Given that the primary purpose of the DQSS Table is to maintain the integrity of the network, a DQSS Table should be viewed as an Object Oriented Encapsulation of a specific DQ Network.

6.0. THE ACCESS REQUEST SEQUENCE

The purpose of the Access Request Sequence (ARS) is twofold:

- 1. To afford current members of the DQSS with an opportunity to request communication privileges with one or more of the other nodes (including the Cluster Head) within the network.
- 2. To simultaneously mitigate the potential for MAC & Data Payload collisions and hence, dropped frames resulting from corruption.

The latter is achieved by limiting the contention for access to the channel to a finite and predictable period of time. With the exception of the Cluster Head, all nodes must utilize this mechanism in order to access the MAC & Data Payload segment of the DQ Transmission Sequence.

6.1 ARS Mechanics

The ARS Segment is divided into three (3) sub-parts, termed, Mini-Slots (MS) (as shown in FIG. 3). This number was initially chosen based upon research[1] (i.e. Xu & Campbell, 1992) showing that the collision resolution process can be made to work faster than the data transmission process when the number of MS is restricted to three (3). Increasing the number of MS beyond three (3) may introduce additional delay as well as adding increased overhead to the overall protocol resulting from the added delay.
The collision resolution process referenced above utilizes unique patterns transmitted by each soliciting device and a summation of those patterns in the event of a collision as a means for detecting collisions. The operation of DQWA is based on the m-ternary feedback information on the state of each of the mini-slots. The Cluster Head must be able to distinguish between the three states:

- Idle,
- Success,
- Collision,

for each mini-slot; as this information is crucial for the application of the protocol rules at the end of each frame. Adopting a patented technology [2] (i.e. Campbell & Xu, 2001) each node is assigned a unique bit pattern that has the property that when two or more ARS collide, the pattern of the overlapping signal is distinguishable from the original pattern of any single ARS; hence, the Cluster Head can detect the collision.

The preferred example patterns referenced in the paper are binomial coefficients; however, DQWA uses an increased hamming weight of four (4) in order to support a significantly increased number of unique code words than can otherwise be supported with a constant hamming weight of two (2). For instance, within a 32-bit word, there exists only 496-Code Words with a Hamming Weight of two; as compared to 35,960 Code Words having a Hamming Weight of four within the same 32-bits (almost two orders of magnitude more).

Given that DQWA is targeting potential MESH networks much larger than 496 nodes, larger Hamming Weights are necessitated for real-world implementation with (as mentioned above) four (4) being the current selected Hamming Weight.

Each node accepted into the network is assigned both a 12-bit Node Address and a 20-bit Code Word with a Hamming Weight of four (4) (as shown in FIG. 4).
When a collision does occur, it is a relatively straightforward process to determine since the Hamming Weight will be greater than four (4). There are 4,845 4-Bit Code Words within a 20-bit binary string; thus, the worst case probability that a collision could occur and result in a valid Code Word is less than ½ of a percent (0.46%). However, since the Code Word is also coupled with the Node Address, there is an additional safeguard procedure to ensure that any anomalous undetected collision is immediately detected.

The aforementioned ternary decision can be subsequently determined as follows:

-
  - Idle (i.e. no signal in ARS Mini-Slot)—Received Signal is below the RSSI (Noise) Threshold.
  - Success—A demodulation resulting in a precise hamming weight of four (4) and a correlated (i.e. correct) code word value and node address combination.
  - Collision—Any signal detected above the noise (RSSI) threshold not resulting in a translation into the digital domain of a code word with a hamming weight of four (4) and/or not having a correlated (i.e. correct) code word value and node address combination.

The Cluster Head will respond with the collision results as part of the DQSS Management Segment in order to clarify any potential ambiguities.
6.2 ARS QoS Support
It is presumed that in most cases, DQWA will be utilized with some level of QoS enabled; if so, two additional fields are added to the ARS Mini-Slot structure so that the feedback packet can adequately determine the queuing order for each node:

-
  - Requested Message Payload Limit, and
  - Requested Message Priority;

Each field is 4-bits, which in turn expands each ARS Mini-Slot to a Preamble plus 40-bits of information. FIG. 5 depicts the expanded ARS Segment with QoS support. FIG. 6 depicts the expanded version of an individual Mini-Slot.
The contents of each field will now be detailed; although a more complete explanation can be found in section 10 on “The DQSS Management Segment (Feedback Packet (FP)).”
6.2.1 QoS Requested Message Payload Limit
Table 1 specifies each setting and corresponding reservation amount:

TABLE 1

ARS QoS Requested Message Payload Limit Settings

QoS Requested Message Payload Setting (in binary)	QoS Message Payload Value (in bytes)
0000	4,096
0001	8,192
0010	12,288
0011	16,384
0100	20,480

0101	24,576
0110	28,672
0111	32,768
1000	36,864
1001	40,960
1010	45,056
1011	49,152
1100	53,248
1101	57,344
1110	61440
1111	65,536

The implied value specified by the QoS Requested Message Payload setting is used by the Cluster Head to determine the relative placement in the distribution queue of the requesting station.

6.2.2 QoS Requested Message Priority

The values used for the QoS Requested Message Priority field are the same values used within a frame, as detailed in section 8.1.1.1.9 on “Quality of Service (QoS) Level-111b.”

There are eight priority levels, thus only three bits are required, leaving the uppermost bit unused and reserved (as shown in FIG. 7). The priority levels increase linearly, thus a priority level of ‘0’ is of the lowest priority and a priority level of ‘7’ is of the highest priority. DQWA does not define what the individual priority levels mean, leaving that up to the network layer protocols sitting on top of DQWA.

6.3 DQSS Node Addressing within the ARS

DQSS Network addresses are 12-bits in length, however, only the lower 10-bits are assignable for the dynamic portion of a valid address; as the upper two bits have special meaning. Both bits along with the rest of the DQSS Network Address are shown in FIG. 8:

The DQSS Node Addressing will now be explained within the context of the ARS; addition detail of the DQSS Node Address field is found in subsequent sections.

6.3.1 DQSS Node Address Field

6.3.1.1 DQSS Node Cluster Bit

NOTE: This bit is NOT used within the ARS; but will be explained here since this bit is part of the DQSS Node Address Field. This bit should ALWAYS be zero during the ARS; as the Cluster Head may preempt the Transmit Queue any time it deems necessary to do so and is not restricted to the transmit request process as the rest of the nodes within the DQSS are.

The MSB of the address is reserved for the Cluster Head. This is particularly helpful if the Network Topology moves and the Cluster Head moves with it. Thus, allowing any node to maintain its original identity both before and after assuming the duties of the Cluster Head. In this way, the DQSS table maintains consistency regardless of which node is currently in charge of the network.

6.3.1.2 DQSS Node Join Request Bit

The next most significant bit (bit **1**) is used by nodes wishing to join the network. In order for an unknown node to be considered for admittance to the DQSS, it must satisfy two conditions:

-
  - 1) The “Join Request” Bit shown in Error! Reference source not found. must be set within the DQSS Node Address Field.
  - 2) The “DQSS Mini-Cluster” Sub-Field must set ‘7’ (i.e. “111b”).

The “DQSS Individual Address” Sub-Field may be any value between ‘0’ and “127” (i.e. a span of 128-values). The complete list of predefined Hamming Weights and DQSS Network Addresses may be found in Appendix A.

6.3.1.3 DQSS MiniCluster SubField
These three bits are used to allow the network administrator to organize nodes in accordance to their own internal policies. Assignable values are between ‘0’ (“000b”) and ‘6’ (“110b”), with ‘7’ (“111b”) reserved for “Join Requests” and “Broadcasts”.

6.3.1.4 DQSS Individual Address SubField
These seven bits are used for assigning individual addresses, with any value between ‘0’ and “126” assignable for an individual DQSS Network Address. The only time “127” may be used during the ARS is during a “Join Request.” As “127” is otherwise set aside for “Directed Broadcasts” and regular “Broadcasts” for all Mini-Cluster Sub-Field values except for ‘7’ (i.e. “111b”).

6.4 ARS Join Requests
As outlined in the prior section, “Join Requests” may choose between any one of 128 values for the DQSS Individual Address Sub-Field and any one of 17-values for the Code Word. So long as predefined values are selected for those fields as well as the “Join Request” bit being set; the Join Request will be considered valid.

7.0 DQ Message
A DQ Message is what is presented as the interface between the MAC and Network layers and consists of the below fields:

-
  - 1) Address Fields;
  - 2) Frame Length Field;
  - 3) Data Payload area;
  - 4) and a Frame Check Sequence (FCS) Field.

FIG. 9 depicts a complete DQ Frame:
Each of the above four logical divisions of the DQ Frame Structure will now be detailed.

7.1 DQ Frame Address Fields
The DQ Frame has two variants for addressing:

-
  - Internal DQSS Network Addresses;
  - External DQ MAC Address.

A DQSS Network address is a 12-bit address that uniquely identifies the DQ Node within a specific DQSS Network and was explained in detail in section 6.3 and depicted in Error! Reference source not found.
A DQ Network Address is at most 12-bits, with the uppermost 4-bits of each DQ Network Address set aside and reserved for future expansion. Thus, the maximum number of nodes potentially supported within a given DQSS is 4,096; minus selected addresses set aside for explicit functionalities. However, as explained in sections 6.3.1.1 and 6.3.1.2, the uppermost two bits have special significance; thus preventing them from being used as normal address bits. Meaning, the number of stations that can actually be delineated is 210 (i.e. 1,024).

The DQ MAC Address adheres to standard IEEE 802 MAC-48/EUI-48 formatting and structure with the intent it eventually be adopted into the overall 802 standard.

7.1.1 The Standard Addressing DQ Frame Header

With few exceptions (Application Data intermediate frames being noted as the most common exception) most DQ Frames include the DQ Network Address of both the destination and sender along with the DQ MAC Address of the DQ Cluster Head/Access Point. This is known as the “Standard Addressing DQ Frame Header” and is shown in FIG. 10.
The Standard DQ Address Header contains the three Address Fields:

- 1) The Immediate Destination DQ Network Address;
- 2) The Immediate Source DQ Network Address;
- 3) The Cluster Head DQ MAC Address
  with the first two addresses being internal DQ Network Addresses and the Cluster Head being a standard DQ MAC Address.

7.1.2 The Extended Addressing DQ Frame Header
The Extended Addressing DQ Frame Header extends the Standard Addressing DQ Frame Header by adding the DQ MAC Addresses of the original sender and final destination nodes (as shown in FIG. 11). This frame is only required if the Final Destination and Original Source Nodes are not part of the same DQSS. In this case, the “Destination DQ Network Address” is set to that of the Access Point or Cluster Head.
Therefore, with one exception, any time the Access Point or Cluster Head is specified as the “Destination DQ Network Address”, the Extended Address DQ Frame Header is used. The lone exception is whenever the Access Point or Cluster Head is also the final destination; in which case only the Standard DQ Frame Header is utilized.

7.2 DQ Frame Payload Length Field
As the name implies, the length contained here specifies the number of bytes within the frame payload and must be a number between 256 and 4,096 bytes. Meaning, 256-bytes is the minimum size Frame Payload and 4,096-bytes is the maximum size Frame Payload.

7.3 DQ Payload Field
This field carries the data payload of the frame. Other than length, there are no restrictions to the contents of this field. If there are not sufficient bytes to fill the minimum size DQ Payload field, the missing bytes will be zero filled.

7.4 DQ Frame Check Sequence (FCS) Field
The FCS is a 32-Bit CRC located immediately following the last byte transmitted for a given frame and covers the entire frame contents, including the four bytes of the FCS.

8.0 DQ Data & Control Window

The DQ Data & Control Window is the portion of the Transmission Sequence in which application data is communicated and is the most complex of the three segments comprising the Transmission Sequence. The three segments are: The DQSS ARS Segment, the DQ Control & Payload Segment, and the DQSS Feedback Packet Segment.
All DQ Packet Segments are comprised of:

- 1) A DQ Packet Segment Pre-Header;
- 2) An optional Management Information Sub-Header and Directives;
- 3) An optional Frame Data Payload section;
- 4) A 4-Byte Packet Check Sequence (PCS).
  NOTE: Although both (2) and (3) above are optional, all DQ Packet Segments must contain one or both of them.

The most basic DQ Packet is one in which the entire frame is contained within the packet and has no MI Directives. However, DQ Packet Segments may also contain Management Information Directives, Frame Check Sequence (if the entire frame is not contained within one Packet Segment), and may even exclude a Data Payload portion altogether if only MI Directives are required for a given Packet Segment.
The individual elements of the above Basic DQ Packet Segment will now be detailed in order to provide the framework for the more complex Packet Segments discussed later in this section.

8.1 the Basic DQ Packet Segment with No MI Directive
The Basic DQ Packet is shown in FIG. 12. The Basic DQ Packet Segment may be between 278 and 4,134 bytes in length and is comprised (at a minimum) of the DQ Packet Segment Pre-header, the DQ Frame Header, the DQ Frame Data Payload, and the Packet Check Sequence (PCS); but also may include a Frame Length Field and Frame Check Sequence (FCS) depending upon the type of packet, as discussed throughout this section.

8.1.1 The DQ Packet Segment PreHeader
FIG. 13 depicts the physical layout of the DQ MAC Basic Pre-Header. All DQ Packets have a DQ Packet Segment Pre-Header and have the following three fields as listed below:

-
  - The Packet Segment Control Field,
  - The Packet Segment Length Field, and
  - The Sequence Control Field.
    These three fields provide the majority of the information required for describing the Packet Segment's content.

The first field, the Packet Segment Control field, provides detailed information about both the packet itself as well as the current configuration of the network. This is most helpful to nodes listening in that may need to adjust their own configuration prior to attempting to enter into the DQSS. The settings within the Packet Segment Control field detail the contents of the packet, including whether or not the DQ Frame portion of the packet is an entire frame or one in a series of fragmented frame segments.
The remaining next two fields, the Packet Segment Length and Sequence Control fields will now be detailed.

8.1.1.1 Packet Segment Control Field
The contents of the Packet Segment Control bits determine the size and content of the rest of the frame and therefore are the most interesting portion of this segment. The fields and meanings are shown in FIG. 14.

8.1.1.1.1 DQ Protocol Version
The DQ Protocol Version is initially set to “0000b” and is set aside as a backwards compatibility measure in anticipation that future use of DQ will expand beyond what is currently envisioned and hence require structure and format changes.

8.1.1.1.2 Data Fragment Management
The Data Fragment Management field provides information to the recipient node enabling the receiving station to discern if this frame is part of a larger fragmented frame or not. If so, these settings directly determine whether or not the packet contains a Frame Length field as is the case with completely encapsulated frames, the initial segment of a fragmented frame, and the initial segment of a fragmented resumed frame. Additionally, the settings contained within determine if the DQ Packet contains Application Data and/or if the packet simply contains DQSS Management Information. The settings and associated meanings are provided in Table 2.

TABLE 2

Data Fragment Management Field Settings

Bits 4 5 6	Description
0 0 0	Management Packet with no Application Data
0 0 1	First Data Packet of Frame
0 1 0	First Resumed Data Packet of Frame
0 1 1	Resumed Frame with Final Data Packet of Frame
1 0 0	Final Data Packet of Frame
1 0 1	Intermediate Data Packet of Frame
1 1 0	Complete Frame within Data Packet
1 1 1	Reserved

8.1.1.1.2.1 Management Frame—000b
This field indicates that there is no Application Data within this packet. Therefore, the packet is strictly for management and control purposes.

8.1.1.1.2.2 First Data Packet of Frame—001b
This value indicates the frame is fragmented and that the packet is the initial packet in a sequence of packets comprising the overall frame. All necessary address fields for the frame are included with this packet as well as a frame length field.
FIG. 15 depicts the header part of this frame, including the DQ Packet Segment Pre-Header. NOTE: There is no FCS within this packet since the FCS does not occur until the final Packet representing the Frame.

8.1.1.1.2.3 First Resumed Data Packet of Frame—010b
This value indicates the frame transmission sequence was previously preempted by higher priority traffic and that the packet is the first packet in the resumption of the frame transmission sequence; but is NOT the last packet within the sequence. There is a separate delineation for an occurrence of the latter (see section 8.1.1.1.2.4 below).

All necessary address fields for the frame are repeated within this packet including the frame length field with one minor exception, the length contained with the frame length field specifies the number of bytes left within the resumed frame including the bytes within the current packet.
The DQ Packet Segment Pre-Header and the Resumed DQ Fragmented Frame Header showing all of the DQ Frame fields repeated are shown in FIG. 16. NOTE: The figure is an example of a Standard Addressing DQ Frame.

8.1.1.1.2.4 Resumed Frame with Final Data Packet of Frame—011b
This value indicates the frame is fragmented and that this is the first packet following a pause in the packet sequence transmissions for that frame, as the transmission sequence was previously preempted by a higher priority form of traffic. It also indicates that this is the final fragment within the sequence.
The Frame Address fields are again repeated for this final packet; however, the frame length field is not included since it is superfluous given that the DQ Packet Segment Pre-Header contains the length of the entire packet and hence the payload length can be easily calculated from it.

FIG. 17 depicts the DQ Packet Segment Pre-Header and DQ Frame header of a Resumed Frame that occurs as the Final Data Packet of the Frame: NOTE: The figure is an example of a Standard Addressing DQ Frame. An Extended Addressing DQ Frame would have additional addresses, as detailed in section

7.1.2, “The Extended Addressing DQ Frame Header”.

Another consequence of a multi-packet frames is that in addition to an Packet Control Sequence (PCS) validating the contents of the overall packet; there is a Frame Check Sequence, validating the contents of the overall frame.

FIG. 18 depicts the complete structure of this type of packet, including the FCS and PCS. NOTE: The figure is an example of a Standard Addressing DQ Frame. An Extended Addressing DQ Frame would have additional addresses, as detailed in section 7.1.2, “The Extended Addressing DQ Frame Header”.

8.1.1.1.2.5 Last Data Packet of Frame—100b
This value indicates that the data segment contains the last segment of a larger message. There are no Frame Address fields following the DQ Packet Segment Pre-Header for this case; but there is an FCS as well as PCS (see Error! Reference source not found.).

FIG. 19 depicts the complete structure of this type of packet, including the FCS and PCS. NOTE: There are NO address fields within this packet.

8.1.1.1.2.6 Intermediate Data Packet of Frame—110b
This value indicates that the data segment contains an intermediate segment of a larger message. There are noFrame Address fields following the DQ Packet Segment Pre-Header for this case; nor is there an FCS.

FIG. 20 depicts the complete packet of this type of packet, including the FCS and PCS. NOTE: There are NO address fields within this packet.

8.1.1.1.2.7 Complete Frame within Data Packet—011b
This value indicates that the DQ Packet contains the entire DQ Frame. The Frame address fields immediately follow the DQ Packet Segment Pre-Header; however, there is neither a Frame Length field nor a Frame Check Sequence (FCS) field, as both would be redundant if included.
FIG. 21 depicts the complete structure of this type of packet, including the FCS and PCS. NOTE: FIG. 21 is an example of a Standard Addressing DQ Frame. An Extended Addressing DQ Frame would have additional addresses, as detailed in section 7.1.2, “The Extended Addressing DQ Frame Header”.

8.1.1.1.2.8 Reserved—111b
This field is reserved for future use.

8.1.1.1.3 Management Directive (MD) Bit (Bit **7**)
If set, this bit indicates that there is Management Information (MI) Header within the packet and that the MI Sub-Header is located immediately following the DQ Packet Segment Pre-Header and before the Address and/or Payload fields if any.

8.1.1.1.4 Retransmission Bit (Bit **8**), RB
If set, this bit indicates that the packet is a retransmission of a previously transmitted packet. This can be used by the receiver station to determine that this may be a duplicate transmission of prior frames as result of an Acknowledgement being lost.

8.1.1.1.5 Dynamic Clustering Enable Bit, DC
If set, this bit indicates that the Cluster Head is Dynamic; thus the Cluster Head will change in real time according to predefined rules.

8.1.1.1.6 Power Management Bit, PM
If set, this bit indicates the Power Management mode that the station will be in after the transmission of the frame; this bit is used by stations that are changing state from Power Save to Active or vice-versa.

8.1.1.1.7 Encryption Bit, EE
This bit indicates encryption is enabled.

8.1.1.1.8 Priority Queuing Enable Bit, PQ
If set, this bit indicates priority queuing is enabled

8.1.1.1.9 Quality of Service (QoS) Level—111b
This field only has meaning if the Priority Queuing Enable Bit is set and there is Application Data within the payload; otherwise these bits are unused. There are eight levels of priority, with the level of priority increasing linearly with the value of the QoS bits:

-
  - Lowest Priority: “000b”
    - □
    - □
    - □
  - Highest Priority: “111b”

8.1.1.2 DQ Frame Length Field
The Frame Length field provides the length of the entire DQ Frame, including the FCS.

8.1.1.3 DQ Sequence Control Fields
The Sequence Control Fields keep maintain control of the application data exchanged between two DQSS nodes.

8.1.1.3.1 DQ Sequence Number Field
The Sequence Number identifies the last packet the sending station sent to the destination station. The Sequence Number is checked at the receiver for missing or duplicated packet. A station receiving numbered information packet advances its Nr count if the packet received is in sequence and does not have errors. The receiving station's Nr count will be equal to the Ns in the next expected information packet or one greater than the Ns in the last packet received. The receiver confirms accepted numbered information packet by returning its Nr count to the transmitting station.
If the incoming Ns does not agree with the receiving station's Nr count, the packet is out of sequence and Nr does not advance. The Nr in the out-of-sequence packet is still valid for confirming transmitted packets.
The count range for Ns and Nr is 256, using the digits 0 through 255. Once the sequence number 255 is reached, the count wraps back around to 0. The Nr and Ns counts are initialized to 0.

8.1.1.3.2 DQ Acknowledgment Number Field
The Acknowledgement Number identifies the last packet the sending station has received from the destination station.
The Acknowledgment Number is checked at the destination for missing or duplicated packets. If the incoming Nr does not agree with the receiving stations Ns, the receiving station must reset its Ns to match the incoming Nr and resend any missing packets not received by the sending station the next time it gains control of the queue.
The count range for Ns and Nr is 256, using the digits 0 through 255. Once the sequence number 255 is reached, the count wraps back around to 0. The Nr and Ns counts are initialized to 0.

8.1.2 Frame Address Fields
DQ Packets utilize the same addresses as do DQ Frames; however, because DQ Packets can and often are much smaller; these frames are NOT repeated for multi-packet frames unless otherwise explicitly noted (such as in the case of a “resumed” frame packet sequence).

8.1.3 Frame Length Field
As mentioned in section 4, 8, 8.1.1, and subsections within 8.1.1.1.2, a DQ Frame can be encapsulated either within one single DQ Packet (as detailed in section 8.1.1.1.2.7 above) or divided across multiple packets.

If the frame is to be divided across multiple packets, it will always contain a length field prior to the data payload area within the initial packet of the frame sequence and will also contain a Frame Check Sequence following the data payload area within the last packet of the frame sequence. Otherwise, if the entire frame is encapsulated within a single DQ Packet, neither of these fields is required since both can be deduced from similar fields within the DQ Packet structure (i.e. the Packet Length Field in lieu of the Frame Length Field and Packet Check Sequence in lieu of the Frame Check Sequence).

8.1.4 Packet Data Payload
This segment contains the actual data or body data that is the intended communication.

8.1.5 Frame Check Sequence (FCS)
The FCS is a 32-Bit CRC located immediately following the last byte transmitted for a given frame. The only time the FCS is included within an actual DQ Packet is immediately following the last packet of a multi-packet Frame sequence (see Error! Reference source not found. for an example).

8.1.6 Packet Check Sequence (PCS)
The PCS is a 32-Bit CRC located immediately following the last byte transmitted for a given packet and occurs in every single packet. The structure of a typical packet and PCS is shown in FIG. 22. NOTE: The PCS is applied to the entire packet plus the four bytes of the PCS.

8.2 The Basic DQ Packet Segment with MI Directive
The basic DQ Packet with an MI Directive area is shown in FIG. 23. The basic DQ Packet is between 276 and 4,130 bytes in length and is comprised of the DQ Packet Segment Pre-header, the DQ Frame Header, the DQ Management Information Sub-Header and Associated MI Payload (if any), the DQ Frame Data Payload, and the Packet Check Sequence (PCS).

8.2.1 The Management Information (MI) Directive SubHeader
The MI Sub-Header provides a mechanism for Communication and Control Directives and associated data between DQSS Nodes and has only one mandatory field, the DQSS Management Information Directive Field (as shown in FIG. 24).
Any additional fields within the MI Sub-Header are MI Directive dependent. The below list details the current list and associated values of all the DQSS MI Directives:

-
  - 0x00: Reserved
  - 0x01: Distribute DQ Service Set Table Command (no acknowledgement. See details in Section 8.2.2.1)
  - 0x02: Mandatory Disconnect Command (no acknowledgement)
  - 0x03: Disconnect Request (from Station to Cluster Head)
  - 0x04: Disconnect Confirmed Response (from Cluster Head to Station)
  - 0x05: Join Request (from Station to Cluster Head)
  - 0x06: Join Accepted Response (from Cluster Head to Station)
  - 0x07: Re-cluster Command (from NEW Cluster Head)
  - 0x08: Re-cluster Acknowledge Response (from each individual station within cluster)
  - 0x09: Link Quality SNR Exchange Request (from Cluster Head to Station)
  - 0x0A: Link Quality SNR Exchange Response (from Station to Cluster Head)
  - 0x0B: Bandwidth Management Command (from Cluster Head to Station)
  - 0x0C: Bandwidth Management Acknowledge Response (from Station to Cluster Head)
  - 0x0D: Maximum Frame Size Command (no acknowledgement) (from Cluster Head to Stations)
  - 0x0E: Switch Queue Command (no response)
  - 0x0F: Pause Queue Command (no response)
  - 0x10: Pause Queue, Enable Join Request for Mini-Slot **1**
  - 0x11: Pause Queue, Enable Join Request for Mini-Slot **2**
  - 0x12: Pause Queue, Enable Join Request for Mini-Slot **3**
  - 0x13: Resume Queue Command

However, while the above list enumerates all of the possible Directives; those directives are divided between ones that are can be transmitted during the Feedback Packet segment and those that can be transmitted during the DQ Packet Segment.

8.2.2 Management Information (MI) Directive Used within DQ Packet Segment
The Directives discussed within this section can only occur within the DQ Packet Segment.

8.2.2.1 Distribute DQ Service Set Table (0x01)
This command is a minimum of bytes in length and can only be sent by the Access Point/Cluster. FIG. 25 depicts the global parameter area of the Distribute DQSS Table Command. NOTE: all Bits labeled as 'R' are unused and hence reserved. The DQ Service Set Table is divided into two sets of parameters:

-
  - 1) The first set of parameters are applicable to the entire table and shown below:
    - a. The Security Status of the DQSS:
      - i. No Encryption (1-Byte);
      - ii. Public Key Encryption (1-Byte);
      - iii. Private (Shared) Key Encryption (1-Byte).
    - b. Public Encryption Key (16-Bytes: this field only exists when Security Status is set to "Public Key Encryption"; otherwise, this field is not part of the DQSS Table).
    - c. Maximum Packet Payload Limitation of the DQSS (1-Byte).
    - d. Number of Configured DQSS Nodes (2-Bytes).
  - 2) The remaining parameters are applicable for all DQSS Nodes with one entry per Node within the DQSS including the Access Point or Cluster Head:
    - DQSS MAC Address.
    - Assigned DQSS Network Address.
    - Assigned DQSS Hamming Weight.
    - Assigned Cluster Head Priority.
    - Assigned QoS Node Priority. This field should be zero (0) for most networks and should only be used if there are specific nodes that need higher priority than others nodes. In those cases in which QoS is on and two nodes are in the queue with the same QoS priority traffic waiting to be sent, the one with the higher priority (if any) moves to the top of the queue for that specific priority setting. NOTE: This only effects traffic of equivalent QoS priorities. It does NOT affect higher priority traffic from a lower priority node. Higher priority traffic is always serviced before lower priority traffic regardless of the priority of the node.
    - Assigned Bandwidth Status:
      - No Bandwidth Guarantee;
      - Limited/Restricted bandwidth;
      - Guaranteed bandwidth.
    - Assigned Bandwidth Setting of each station (i.e. limit or minimum guaranteed value) when applicable.

8.2.2.1.1 DQSS Security Status & Public Encryption Key Fields
In order to provide the maximum degree of security within the network, the encryption switch is found within the first byte following the DQSS Table Command byte. Only the least significant lower two bits are valid. The remaining upper six bits are unused and hence reserved.

8.2.2.1.1.1 DQSS No Encryption ("00b")
Whenever the Security Status field is set to No Encryption, then there are no additional fields; thus the first two bytes of DQSS table are simply the command and security status fields as shown in FIG. 26. In this case, the remaining bytes within the DQSS Table are encrypted using the Shared Encrypted Key possessed by all of the DQSS member stations.

8.2.2.1.1.2 DQSS Public KeyEncryption (“01b”)
Whenever the Security Status field is set to Public Key Encryption, then the immediate subsequent field is the Public Key field as shown in FIG. 27. In this case, the remaining bytes within the DQSS Table are encrypted using the Public Key.

8.2.2.1.1.3 DQSS Shared KeyEncryption (“10b”)
Whenever the Security Status field is set to Shared Key Encryption, then there is no public key field. The DQSS Table Command and Security Status fields are shown in FIG. 28. In this case, the remaining bytes within the DQSS Table are encrypted using the Shared Encrypted Key possessed by all of the DQSS member stations.

8.2.2.2 Maximum Payload
The Maximum Payload field enables specifies the maximum number of bytes allowed within the payload portion of a DQSS Information Frame. The minimum allowed payload is 256-bytes and the maximum allowed payload is 4,096-bytes.
Only the lower four bits of this field are used with the remaining upper four bits being reserved. All payload specifications are given in 256-byte increments with “0000b” representing 256-bytes and “1111 b” representing 4,096-bytes.

8.2.2.3 Number of Configured DQSS Nodes
This field can never be zero, since the Access Point or Cluster Head always counts as part of the network. Hence, zero a setting of “00000000b” is considered invalid.

8.2.2.4 DQSS Table Entries per Node
The remaining fields within the DQSS Table are on a per node basis and in the following order for each entry:

-
  - Bytes: 0-5—DQSS MAC Address.
  - Bytes: 6-9:
    - Assigned DQSS Hamming Weight.
    - Assigned DQSS Network Address.
  - Bytes: 10-11—Assigned Cluster Head Priority.
  - Byte: 12—Assigned QoS Node Priority.
  - Byte: 13—Assigned Bandwidth Status:
    - No Bandwidth Guarantee;
    - Limited/Restricted bandwidth;
    - Guaranteed bandwidth.
  - Bytes: 14-15—Assigned Bandwidth Setting of each station.

FIG. 29 depicts a Single Table Record within the Distribute DQSS Table Command.

8.2.2.5 Mandatory Disconnect Command
This command is 5-bytes in length and can only be sent by the Cluster Head to a DQSS Client Node. It cannot be ignored by the DQSS Client Node. The format of the Mandatory Disconnect is shown in FIG. 30.

No response is expected or desired from the affected DQ Client Node. If the DQ Client Node attempts any further communication other than a request to “Join the DQSS”, the Cluster Head will in turn respond with another MD command.

-
  - Distribute DQ Service Set Table—
  - 0x14: Mandatory Disconnect (no acknowledgement)
  - 0x15: Disconnect Request (from Station to Cluster Head)

- 0x16: Disconnect Confirmed (from Cluster Head to Station)
- 0x17: Join Request (from Station to Cluster Head)
- 0x18: Join Accepted (from Cluster Head to Station)
- 0x19: Re-cluster Command (from NEW Cluster Head)
- 0x20: Re-cluster Acknowledge (from each individual station within cluster)
- 0x21: Link Quality SNR Exchange Request (from Cluster Head to Station)
- 0x0G: Link Quality SNR Exchange Response (from Station to Cluster Head)
- 0x0H: Bandwidth Management Command (from Cluster Head to Station)
- 0x0I: Bandwidth Management Acknowledge (from Station to Cluster Head)
- 0x0J: Maximum Frame Size Command (no acknowledgement) (from Cluster Head to Stations)
- 0x0K: Switch Queue
- 0x0L: Pause Queue
- 0x22: Pause Queue, Enable Join Request for Mini-Slot **1**
- 0x23: Pause Queue, Enable Join Request for Mini-Slot **2**
- 0x24: Pause Queue, Enable Join Request for Mini-Slot **3**
- 0x25: Resume Queue

The MI Sub-Header provides a mechanism for Communication and Control Directives and associated data between DQSS Nodes and has only one mandatory field, the DQSS.

8.3 Frame Control Sequence (FCS)
The MCS is a 32-Bit CRC located immediately following the last byte transmitted for a given message. This field is not part of a frame whose payload comprises a complete message. There are only two instances where this field would appear:

-
- When the Data Fragment Management field is set to “011 b”—Indicating the frame is a “Resumed Message with Final Data Segment” frame. Meaning, it is the last frame of previously interrupted sequence of frames for the associated message.
- When the Data Fragment Management field is set to “100b”—Indicating the frame is a “Final Data Segment” frame. Meaning, it is the last frame of sequence of frames for the associated message.

In these two instances, the format of the MAC & Data Payload Segment are shown in FIG. 31. NOTE: The MCS is only applied to the payload portion of the message plus the four bytes of the MCS.

9.0 Management Information (MI) Directives

The MI Directives are used to maintain and control the network. Directives initiated by the Access Point or Cluster Head are usually intended to maintain the order and integrity of the overall DQSS network. While directives initiated by DQSS Client Nodes are generally used for a specific service or action for that particular DQSS Client Node. Each MI Directive will now be individually detailed, including a complete description of its use, its structure, and intended actions resulting whenever it is used.

9.1 Distribute DQ Service Set Table (0x01)
9.2 Mandatory Disconnect (0x02)
9.3 Disconnect Request (0x03)
9.4 Disconnect Confirmed (0x04)
9.5 DQSS Join Request (0x05)
9.6 DQSS Join Confirmed (0x06)
9.7 Re-Cluster Command (0x07)
9.8 Re-Cluster Acknowledge (0x08)
9.9 Link Quality SNR Exchange Request (0x09)
9.10 Link Quality SNR Exchange Response (0x0A)
9.11 Bandwidth Management Command (0x0B)

9.12 Bandwidth Management Acknowledge (0x0C)
9.13 Maximum Frame Size Command (0x0D)
9.14 Maximum Frame Size Command (0x0E)
9.15 Switch Queue Command (0x0F)
9.16 Pause Queue Command (0x10)

This command can only occur within the Feedback Packet and causes the immediate cessation of application data for all subsequent transmission sequences pending further notice. This includes the case of the command being issued during the transmission of a multi-frame message.
If the command occurs within the sequence of a multi-frame message; the continuation of that message is paused effectively immediately and is not resumed until a “Resume Queue” command is later issued by the Cluster Head.

9.17 Pause Queue, Enable Join Request for ARS MiniSlot One (1) Command (0x11)
This command has the same effect as the Pause Queue Command (0x10), but with two additional side-effects:

-
  - 1) The ARS is eliminated during the immediate transmission sequence; thus this is the notification to all stations so that they may abide by it.
  - 2) The Station making the join request within ARS Mini-Slot One (1) of the prior ARS segment is directed to issue a Join Request Directive within the DQ Control & Data Payload Segment of the next transmission sequence.

Assuming successful transmission of this directive, the subsequent feedback packet will contain feedback as to the determination and resultant actions of the Join Request for ARS Mini-Slot One (1).
9.18 Pause Queue, Enable Join Request for ARS Mini-Slot Two (2) Command (0x12)
This command has the same effect as the Pause Queue Command (0x10), but with two additional side-effects:

-
  - 1) The ARS is eliminated during the immediate transmission sequence; thus this is the notification to all stations so that they may abide by it.
  - 2) The Station making the prior join request within ARS Mini-Slot Two (2) of the prior ARS segment is directed to issue a Join Request Directive within the DQ Control & Data Payload Segment of the next transmission sequence.

Assuming successful transmission of this directive, the subsequent feedback packet will contain feedback as to the determination and resultant actions of the Join Request for ARS Mini-Slot Two (2).
9.19 Pause Queue, Enable Join Request for ARS Mini-Slot Three (3) Command (0x13)
This command has the same effect as the Pause Queue Command (0x10), but with two additional side-effects:

-
  - 1) The ARS is eliminated during the immediate transmission sequence; thus this is the notification to all stations so that they may abide by it.
  - 2) The Station making the prior join request within ARS Mini-Slot Three (3) of the prior ARS segment is directed to issue a Join Request Directive within the DQ Control & Data Payload Segment of the next transmission sequence.

Assuming successful transmission of this directive, the subsequent feedback packet will contain feedback as to the determination and resultant actions of the Join Request for ARS Mini-Slot Three (3).
9.20 Resume Queue Command (0x0E)

10.0 the DQSS Management Segment (Feedback Packet (FP))

The DQSS Management Segment has three primary functions:

- 1) To provide the Cluster Head a means in which to manage the DQSS and associated nodes from the perspective of membership, Quality of Service (QoS), and both queues (i.e. Data Queue and Request Queue).
- 2) To provide feedback to the other nodes in the system for both data and control information.
- 3) To signify and thus mark the end of a single transmission sequence, therefore providing a beacon to all stations for synchronizations purposes.

FIG. 32 represents the structure of the FP and fulfills the above three requirements.
As shown above, the DQSS Management Segment or FP consists of five sections:

- Preamble
- ARS Response
- MI Command or Response
- Sequence Control
- Feedback Packet 8-Bit CRC.
  Other than the "Preamble," which is self-explanatory; each one will now be described in detail.

10.1 ARS Response

Similar to the actual ARS, which has three Mini-Slots, the response to the ARS contains a one-to-one correlation as shown in FIG. 33. With the precise contents for each ARS Mini-Slot Response divided into three separate sections as shown in FIG. 34.

10.2 FP MI Command/Response

10.3 Sequence Control

10.4 Feedback Packet CRC

Figure 1

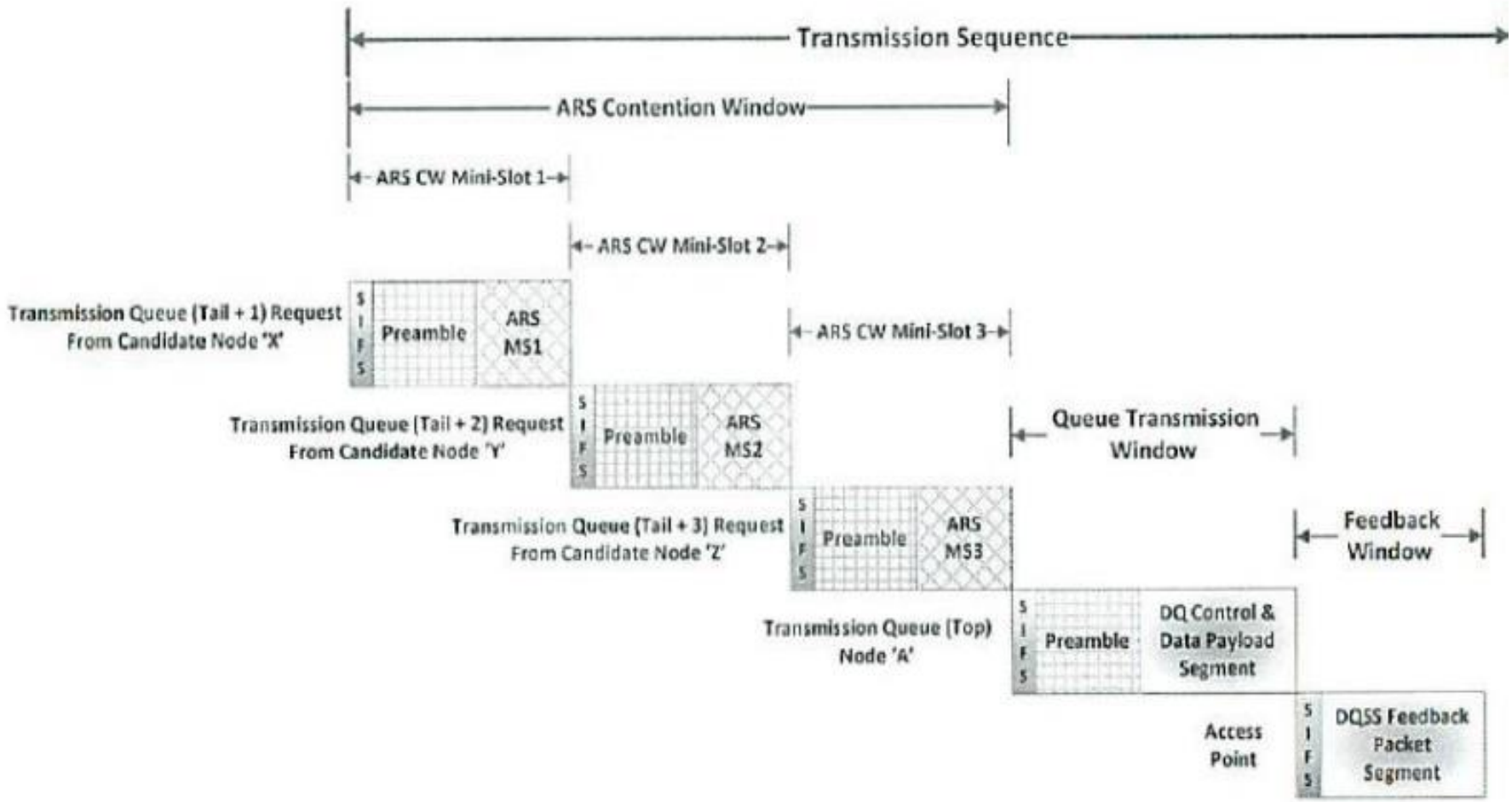
Transmission Sequence
ARS Contention Window
ARS CW Mini-Slot 1
ARS CW Mini-Slot 2
ARS CW Mini-Slot 3
Transmission Queue (Tail + 1) Request From Candidate Node 'X'
SIFS
Preamble
ARS MS1
Transmission Queue (Tail + 2) Request From Candidate Node 'Y'
SIFS
Preamble
ARS MS2
Transmission Queue (Tail + 3) Request From Candidate Node 'Z'
SIFS
Preamble
ARS MS3
Queue Transmission Window
Feedback Window
Transmission Queue (Top) Node 'A'
SIFS
Preamble
DQ Control & Data Payload Segment
Access Point
SIFS
DQSS Feedback Packet Segment

Figure 2

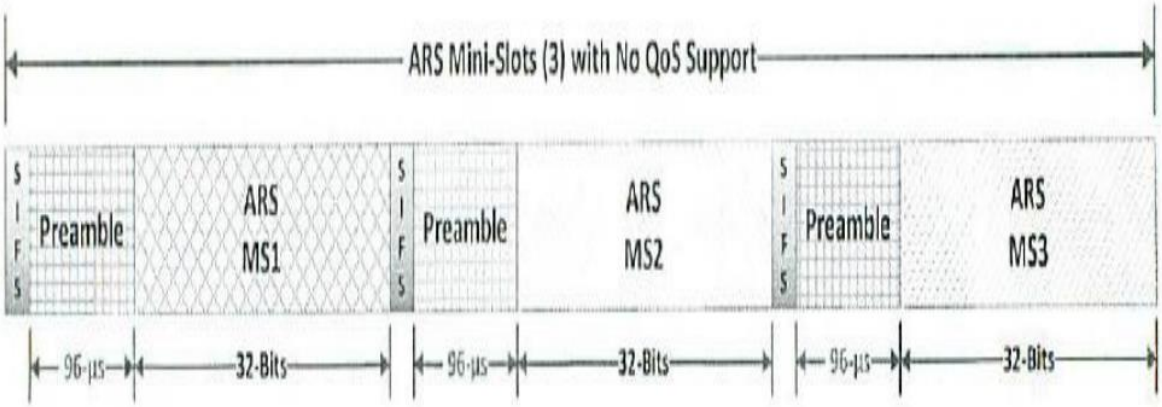

Figure 3

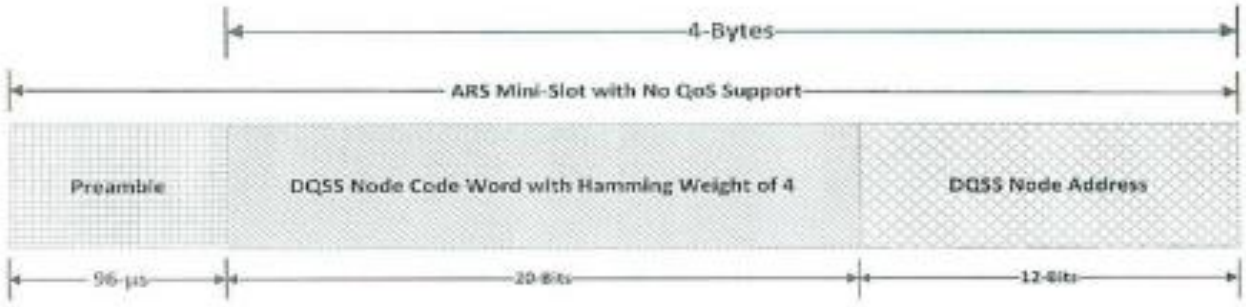

Figure 4

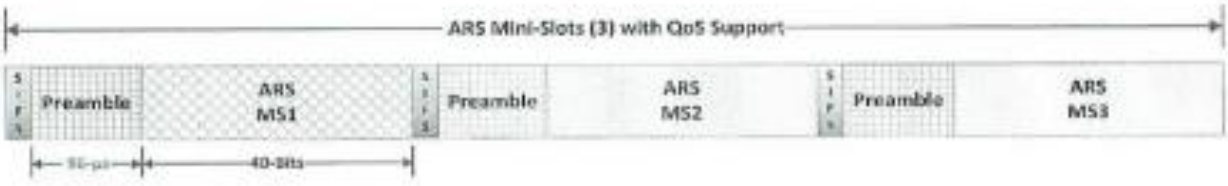

Figure 5

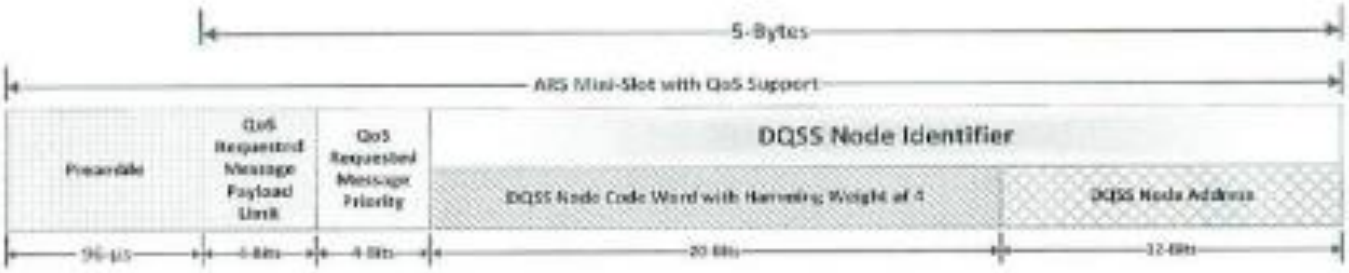

Figure 6

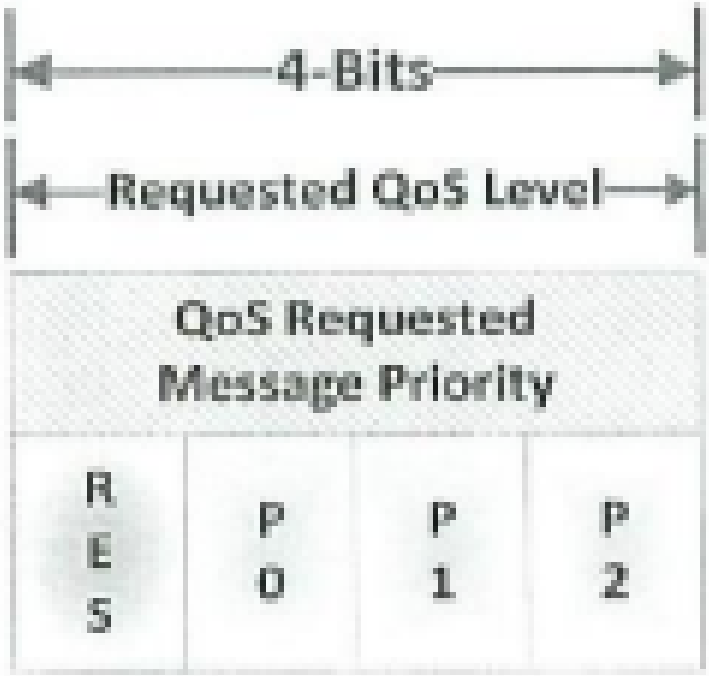

Figure 7

DQSS Node Address Field											
CLUSTER HEAD	JOIN REQUEST	DQSS Mini-Cluster Sub-Field			DQSS Individual Address Sub-Field						
BIT 00	BIT 01	BIT 02	BIT 03	BIT 04	BIT 05	BIT 06	BIT 07	BIT 08	BIT 09	BIT 10	BIT 11

Figure 8

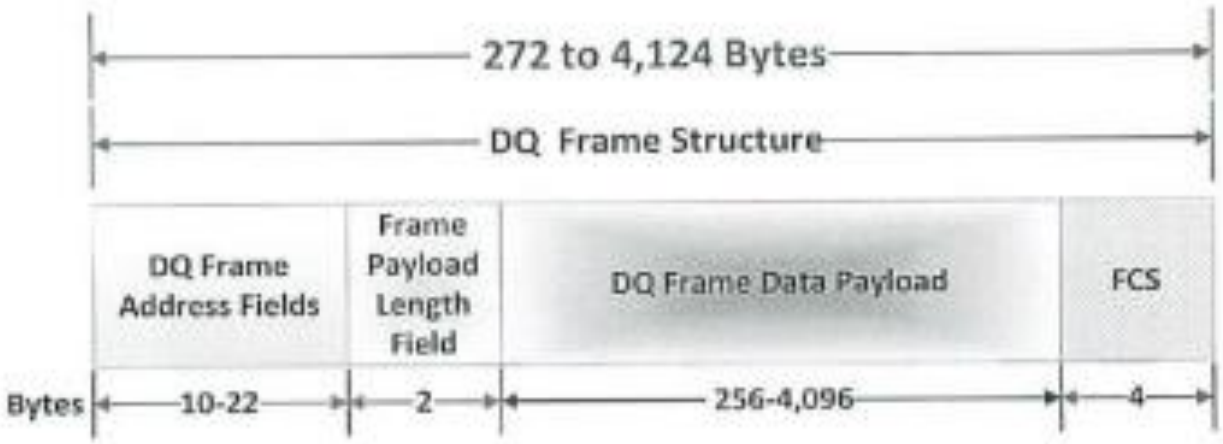

Figure 9

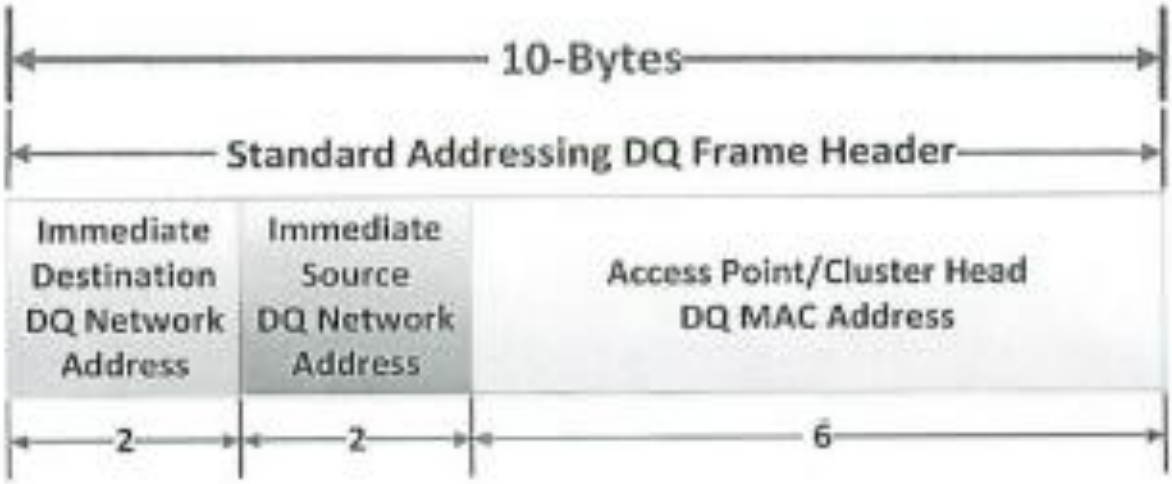

Figure 10

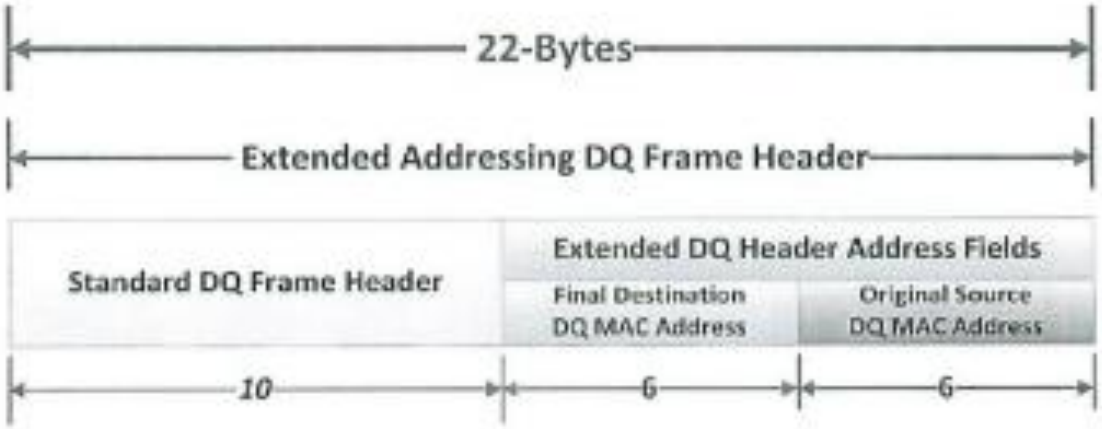

Figure 11

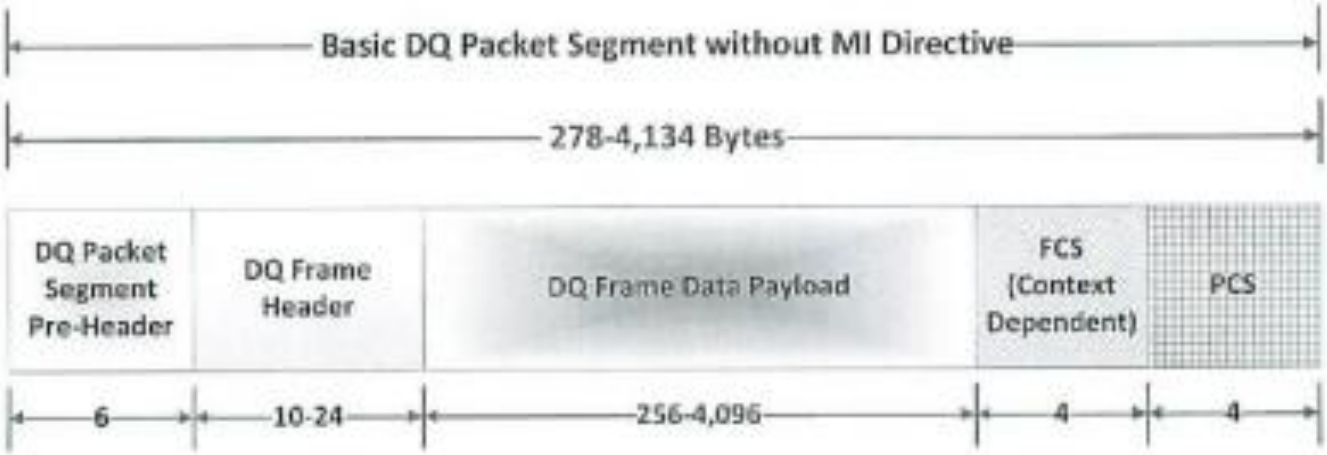

Figure 12

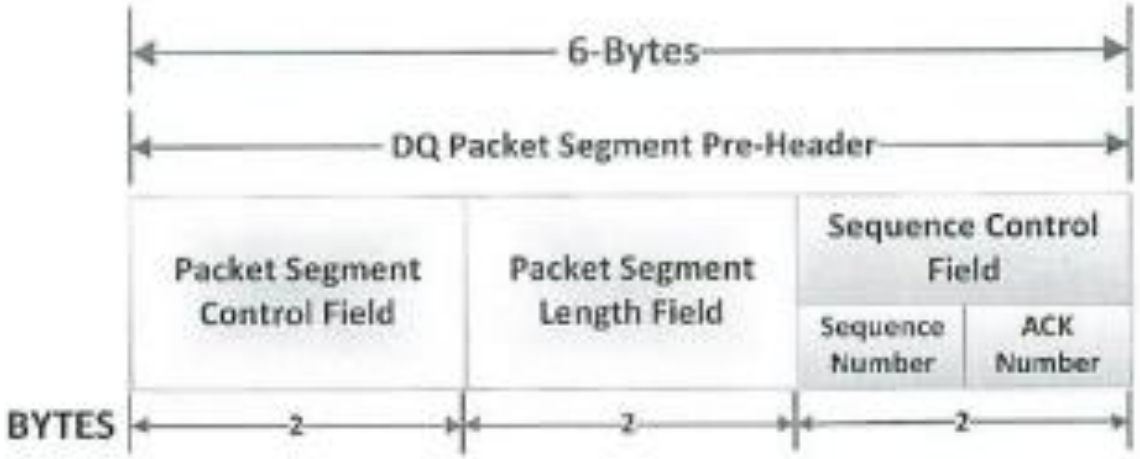

Figure 13

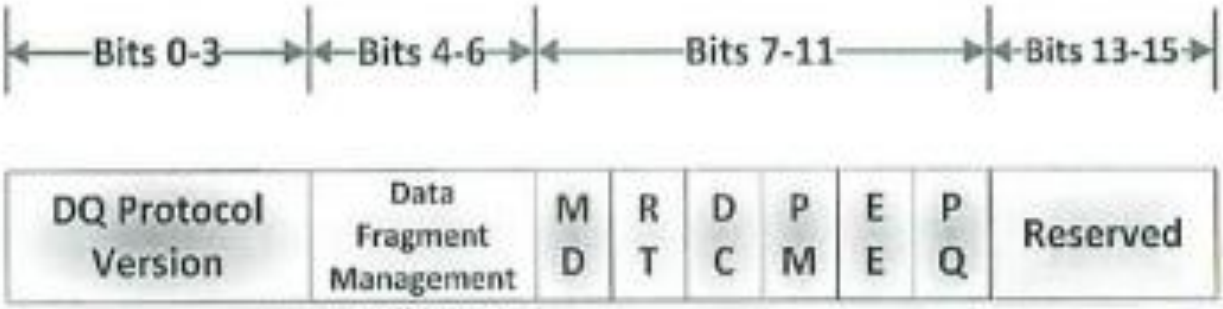

Figure 14

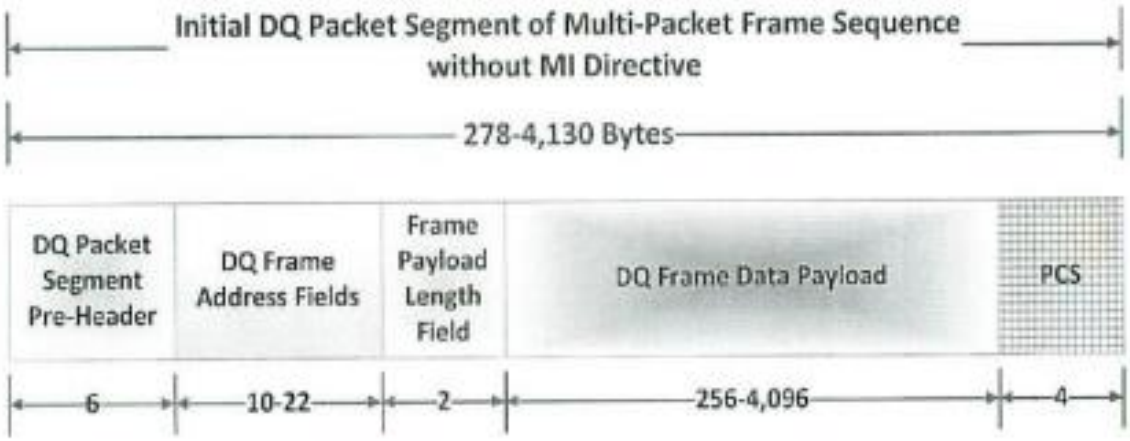

Figure 15

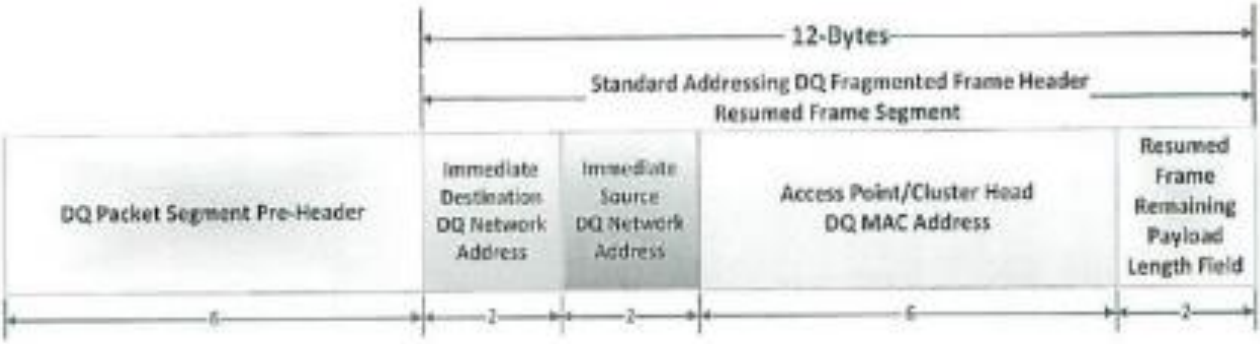

Figure 16

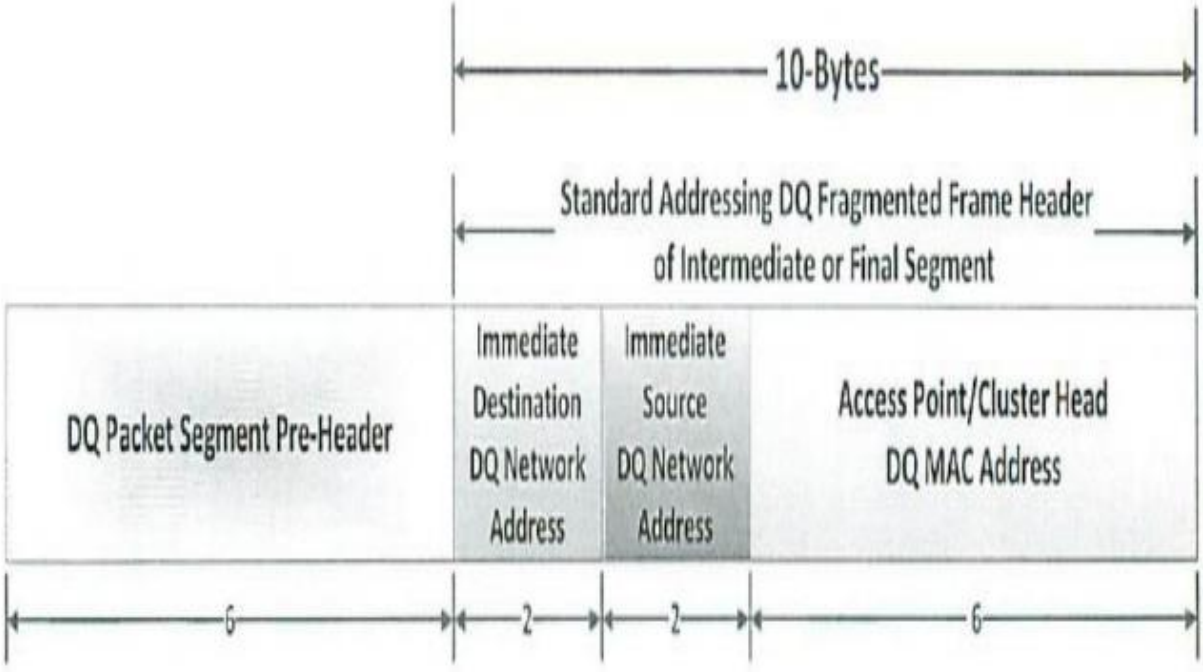

Figure 17

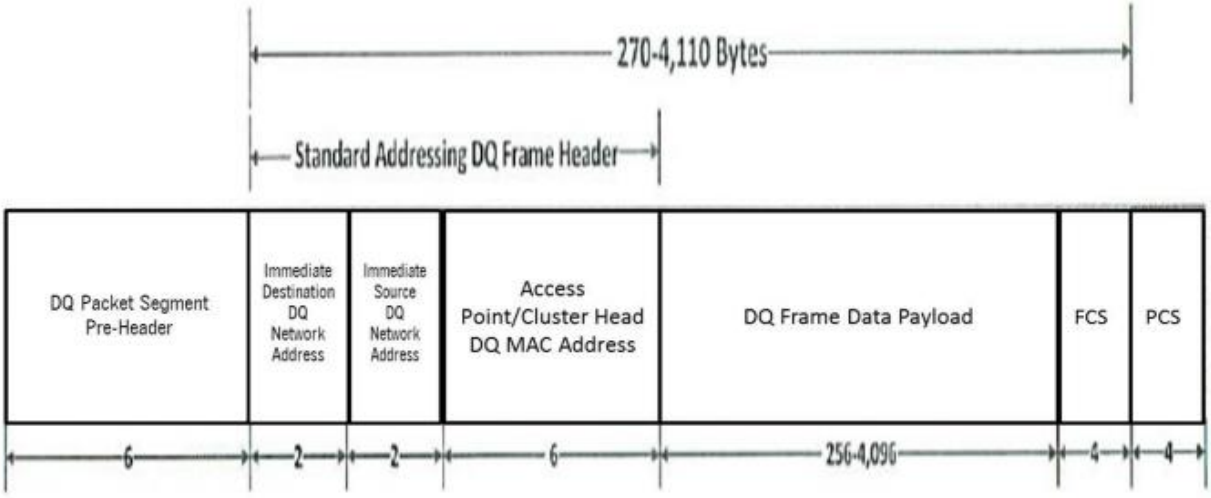

Figure 18

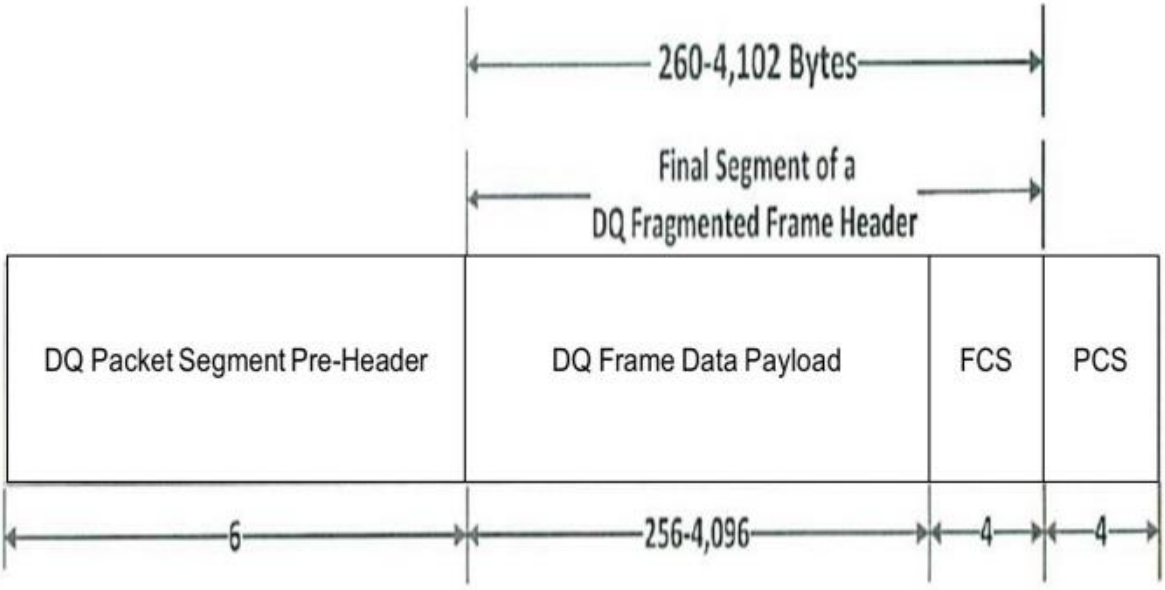

Figure 19

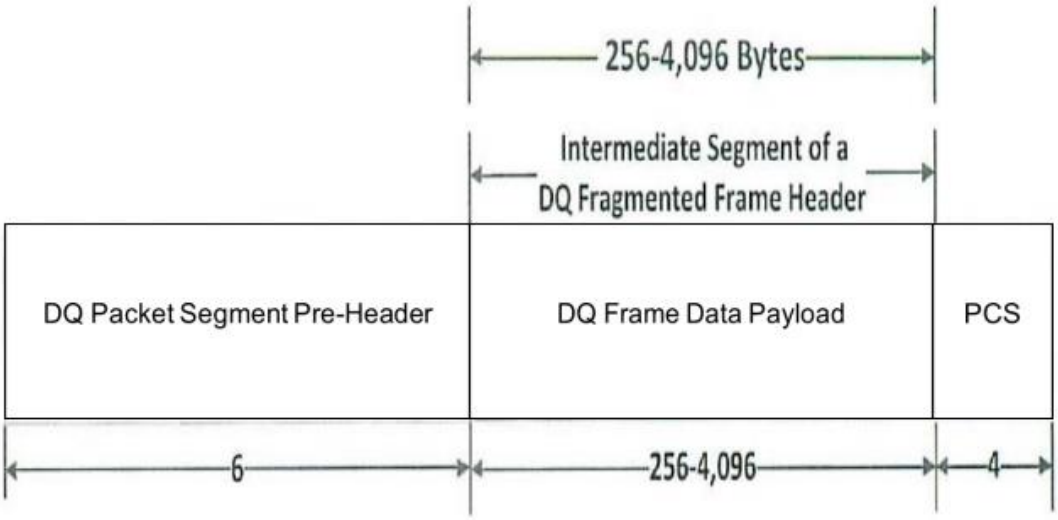

Figure 20

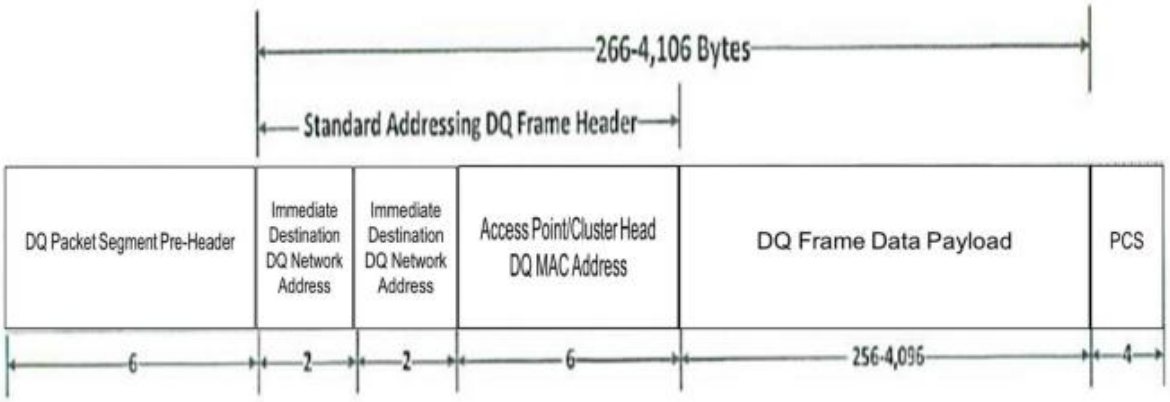

Figure 21

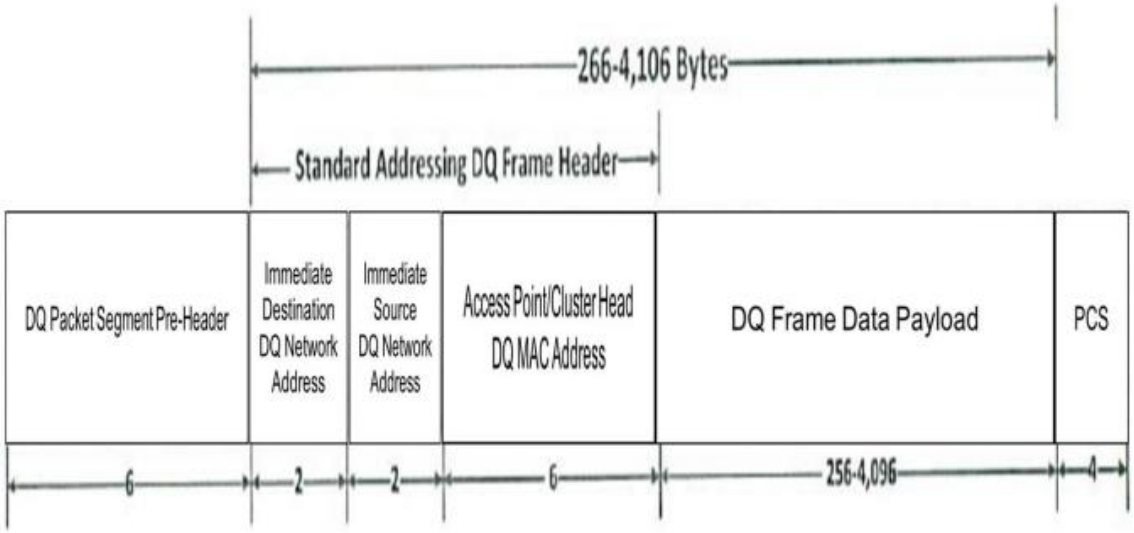

Figure 22

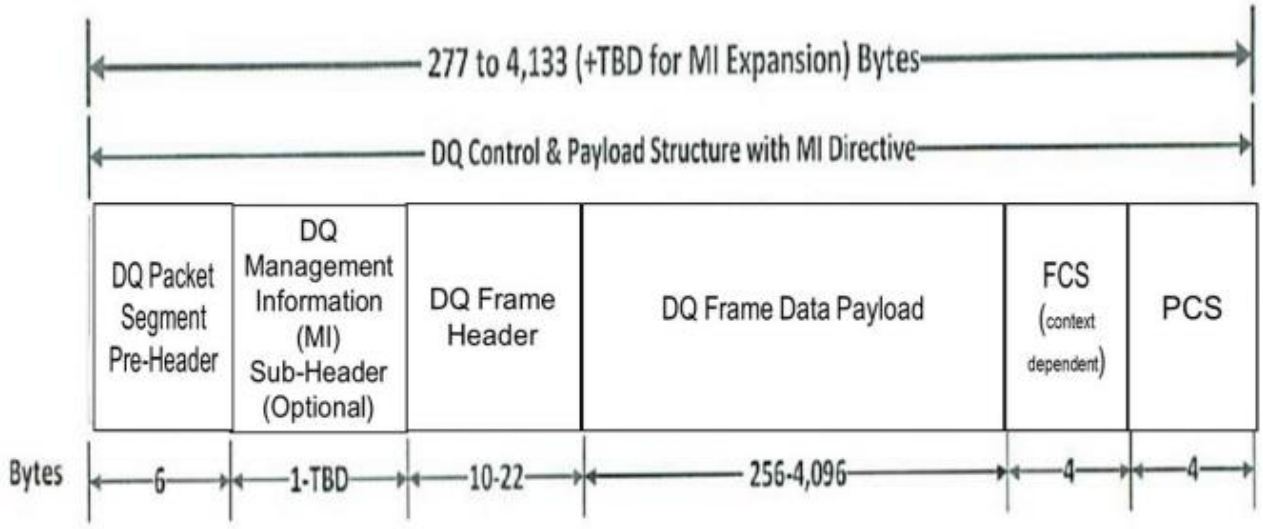

Figure 23

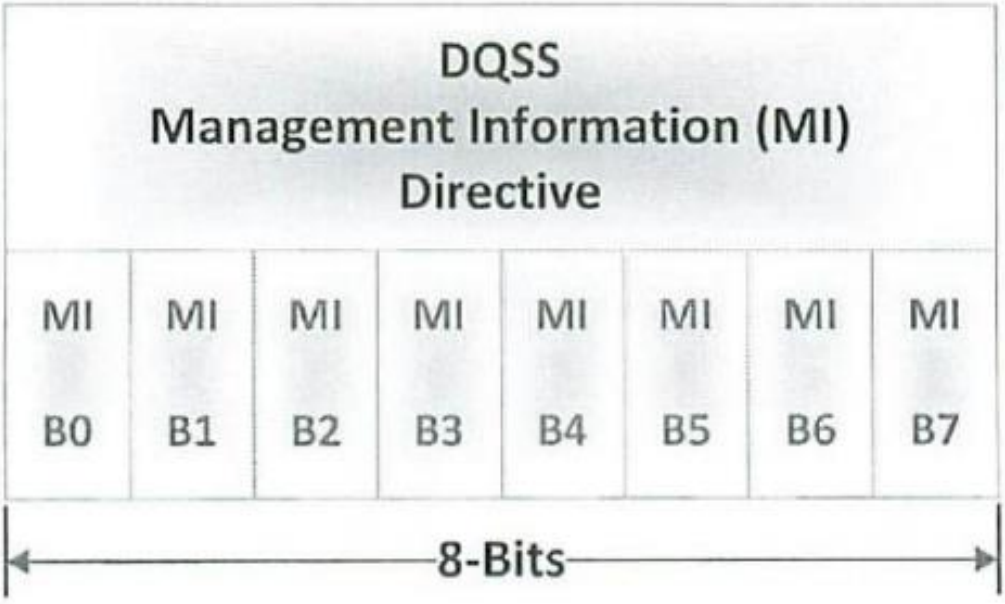

Figure 24

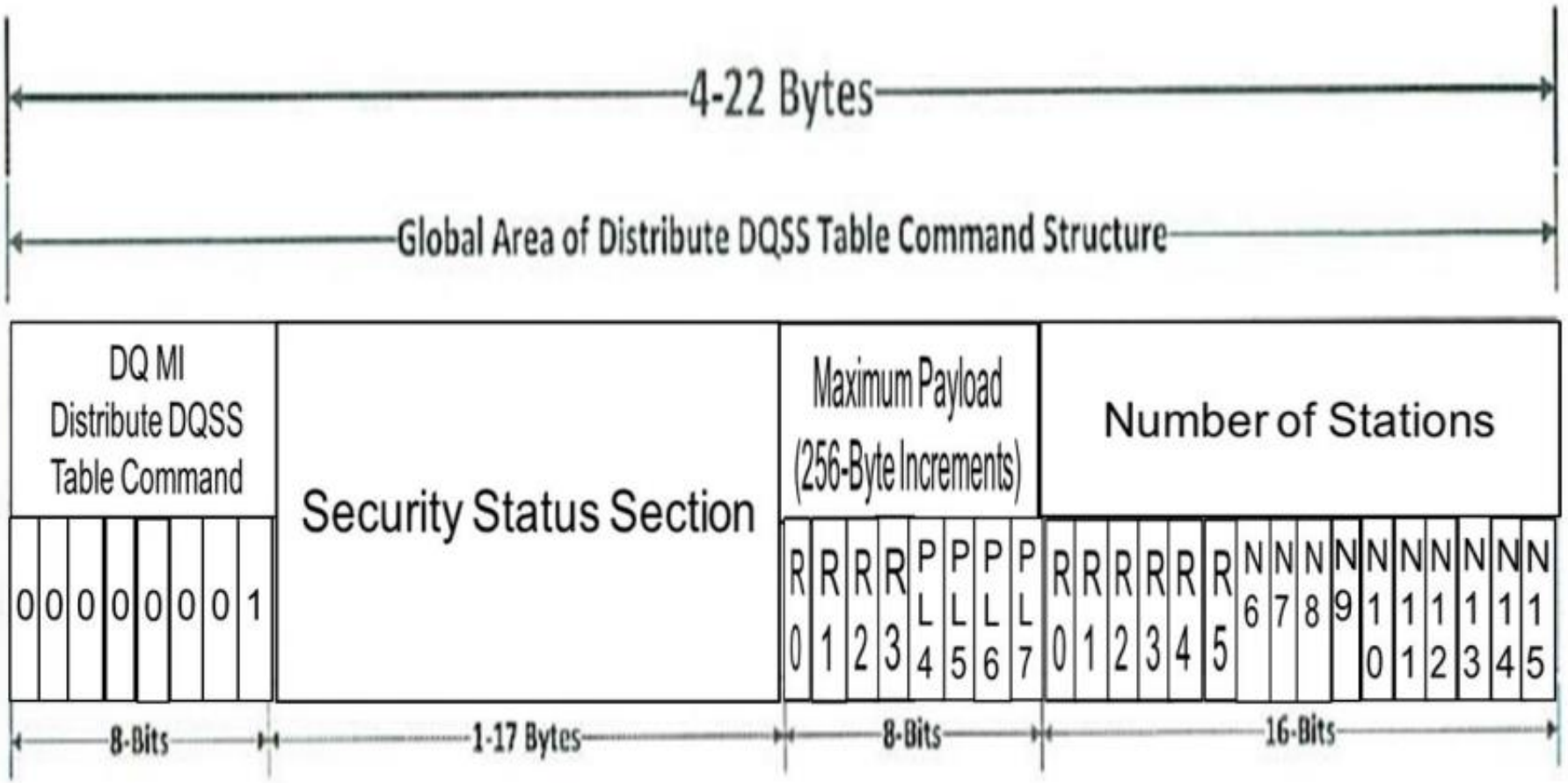
4-22 Bytes
Global Area of Distribute DQSS Table Command Structure
DQ MI Distribute DQSS Table Command
Security Status Section
Maximum Payload (256-Byte Increments)
Number of Stations
0 0 0 0 0 0 0 1
R0 R1 R2 R3 PL4 PL5 PL6 PL7
R0 R1 R2 R3 R4 R5 N6 N7 N8 N9 N10 N11 N12 N13 N14 N15
8-Bits
1-17 Bytes
8-Bits
16-Bits

Figure 25

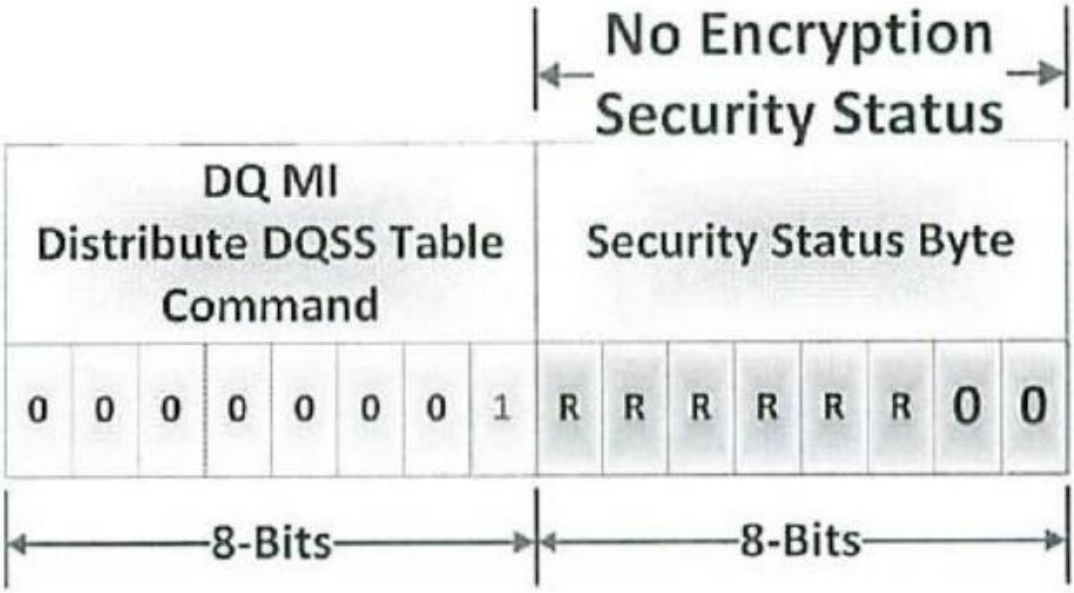

Figure 26

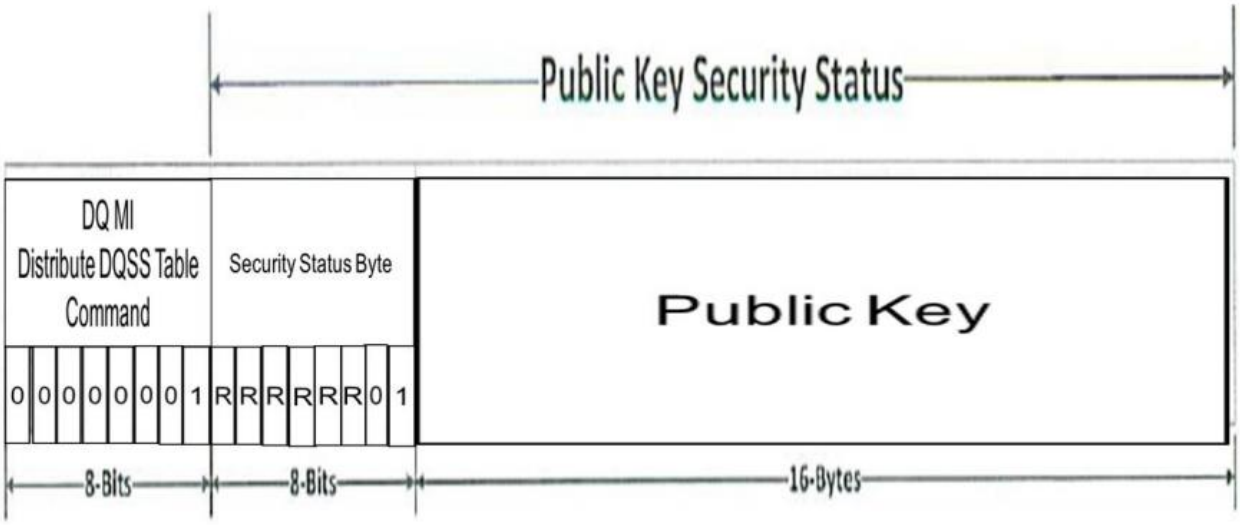

Figure 27

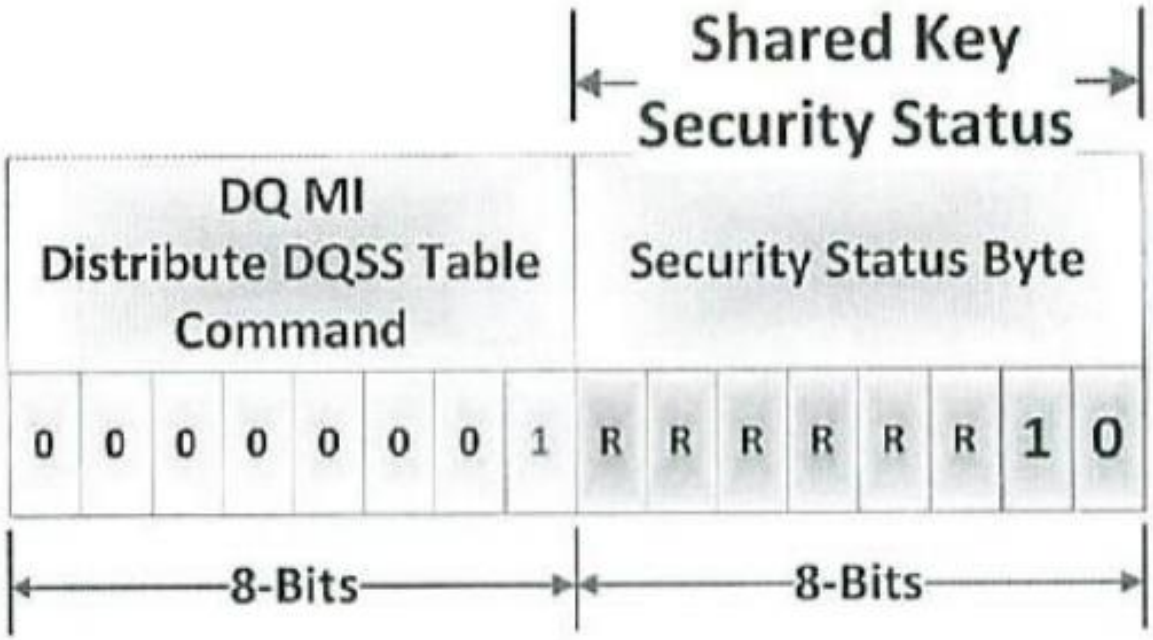
Shared Key
Security Status
DQ MI
Distribute DQSS Table
Command
Security Status Byte
0 0 0 0 0 0 0 1 R R R R R R 1 0
8-Bits
8-Bits

Figure 28

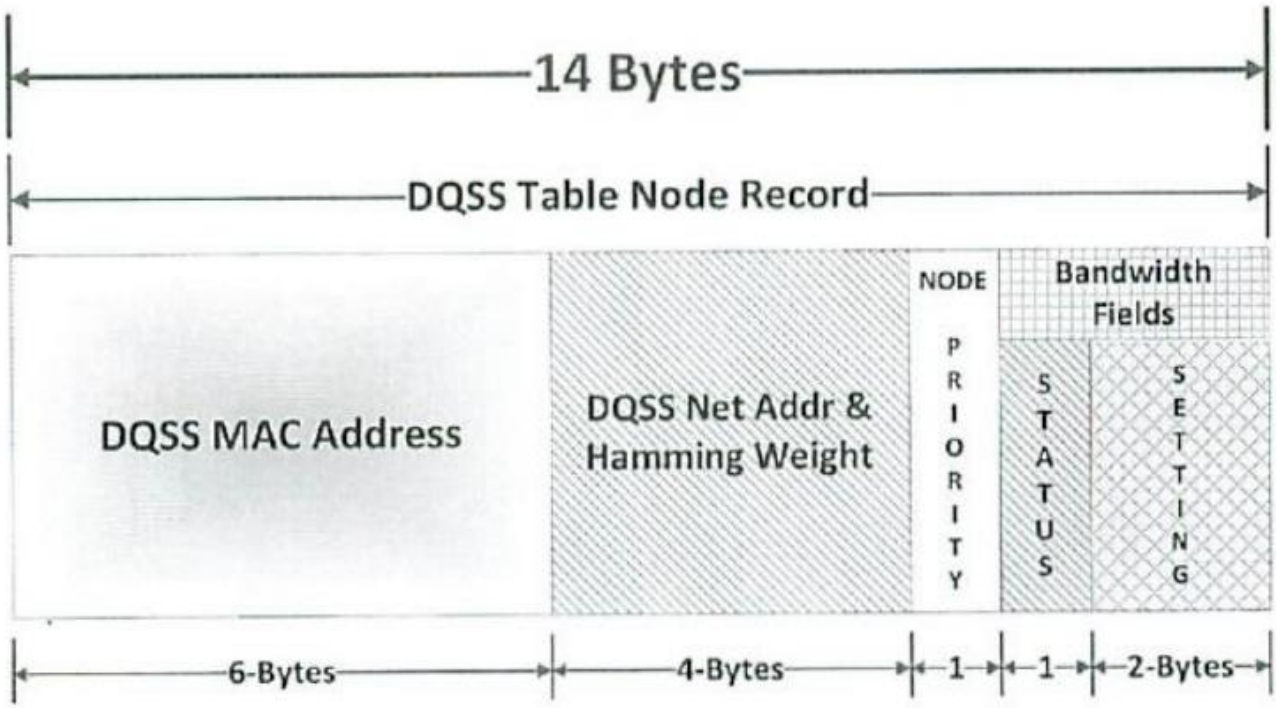

Figure 29

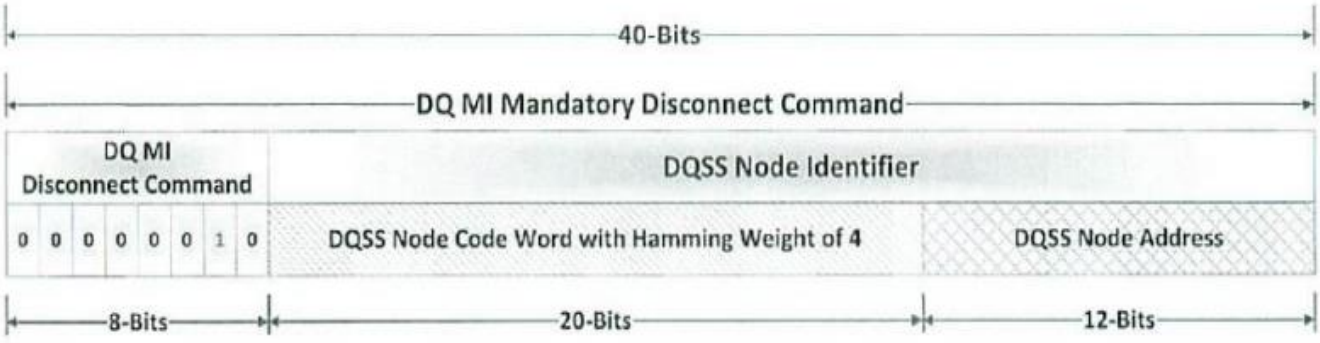

Figure 30

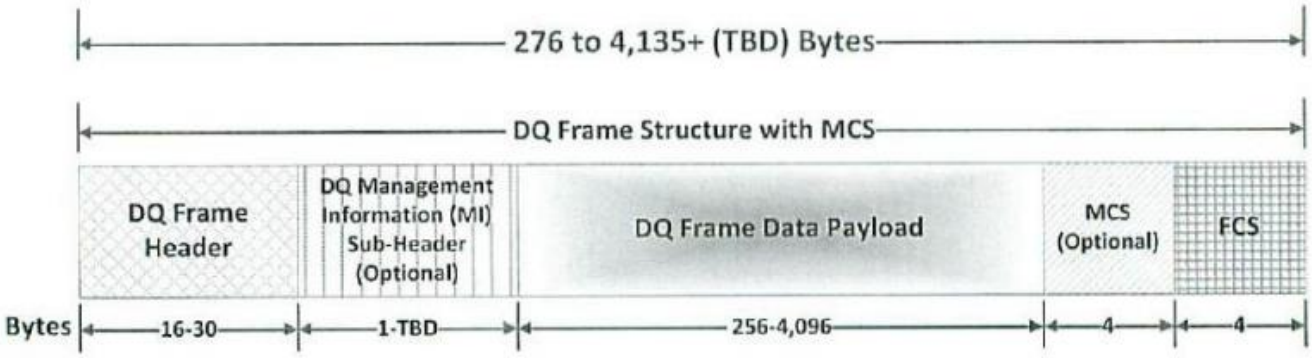

Figure 31

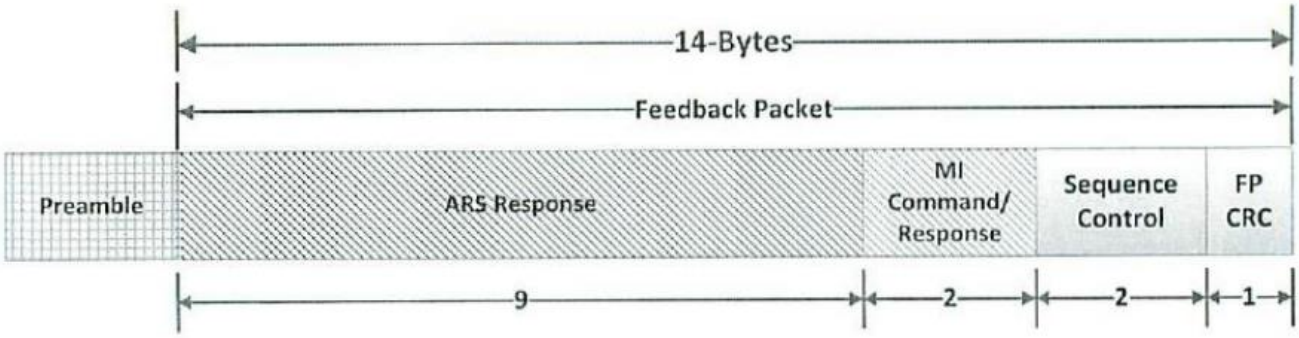

Figure 32

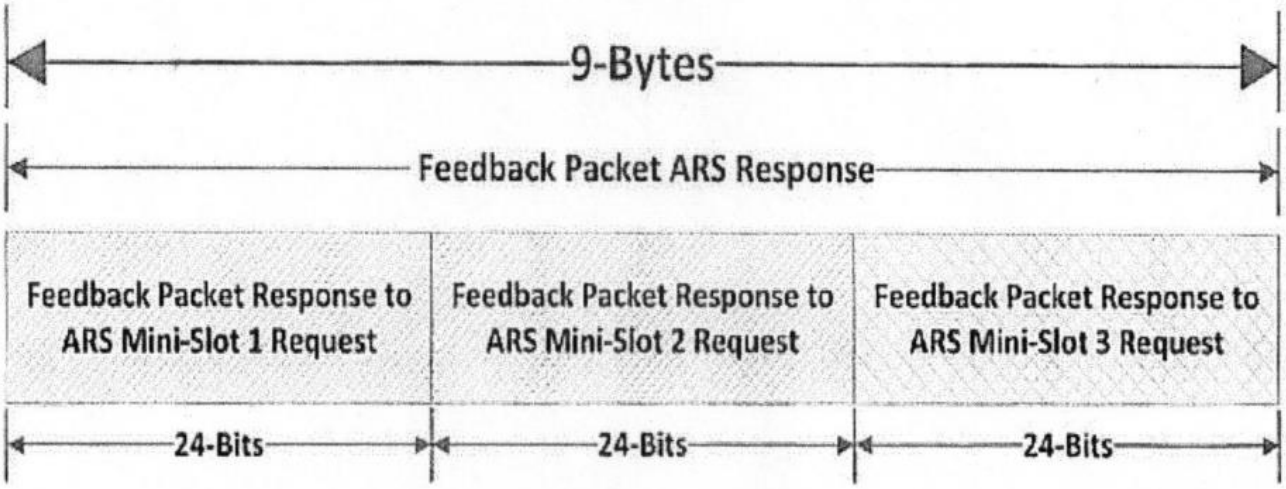

Figure 33

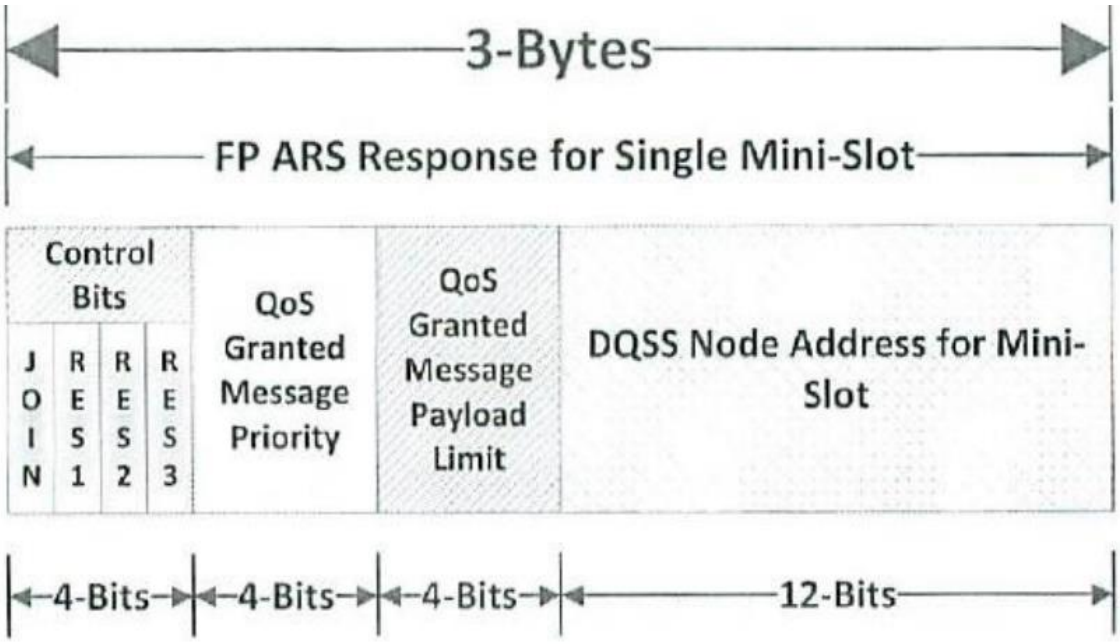

Figure 34

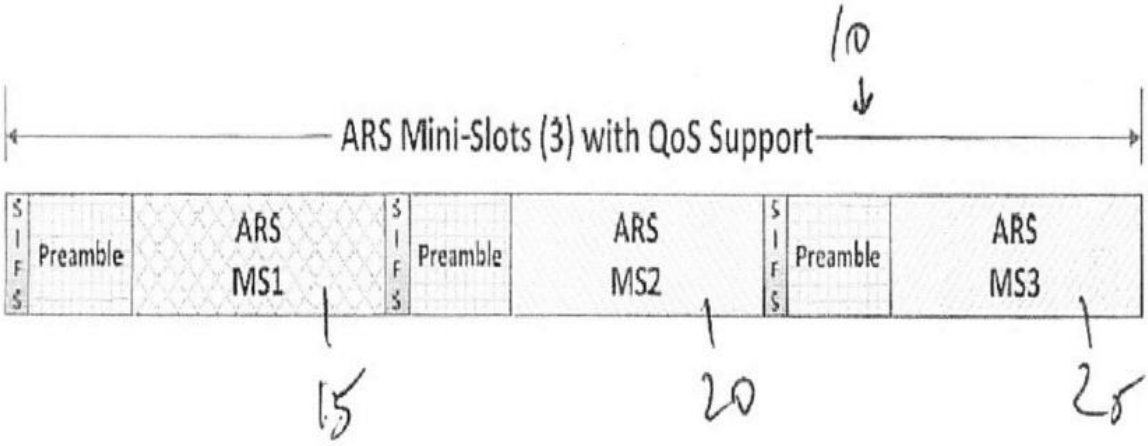

Figure 35

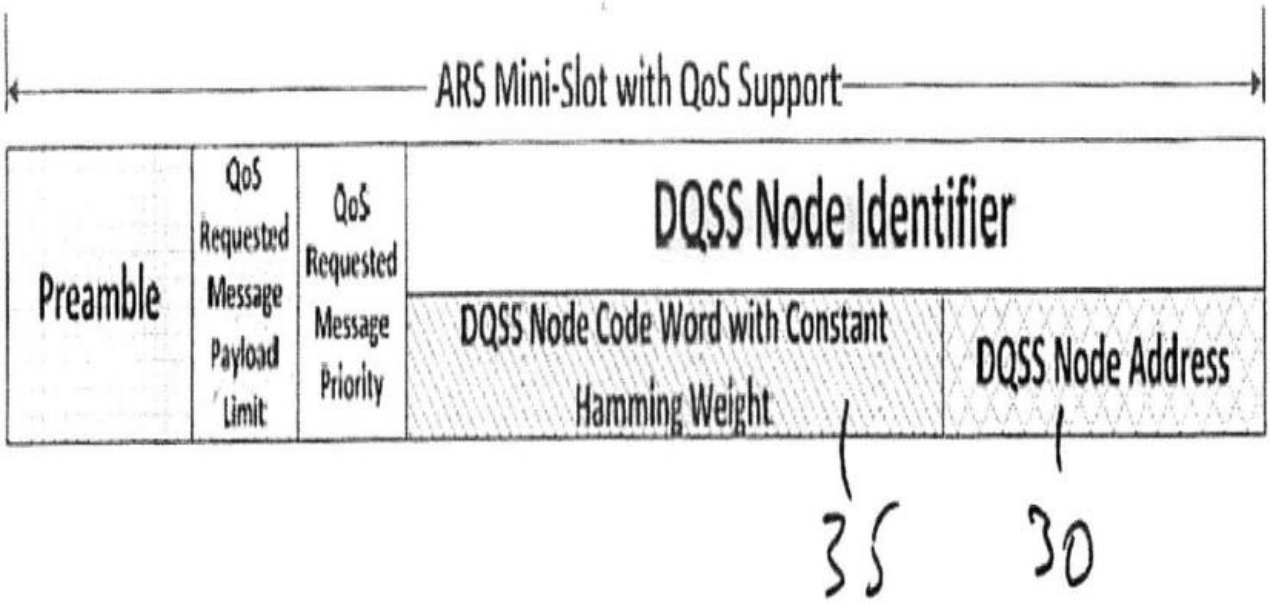

Figure 36

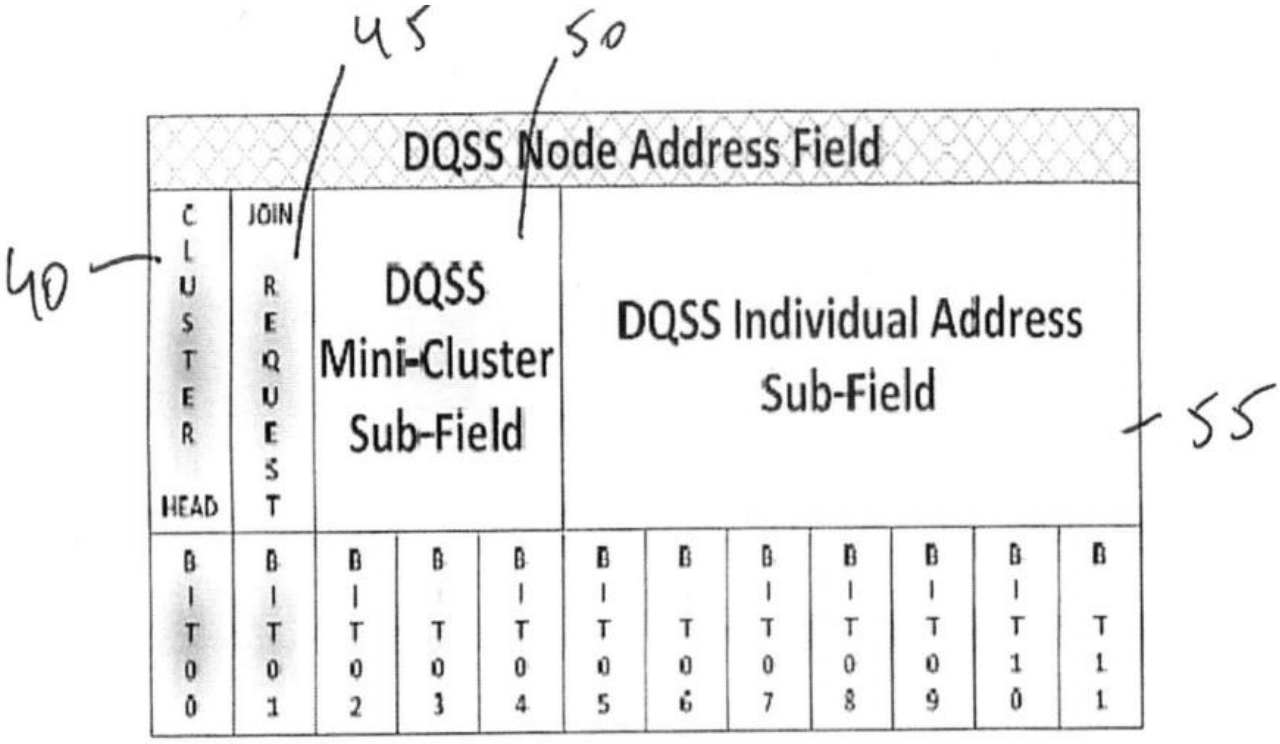
45
50
40
55
DQSS Node Address Field
CLUSTER HEAD
JOIN REQUEST
DQSS Mini-Cluster Sub-Field
DQSS Individual Address Sub-Field
BIT00
BIT01
BIT02
BIT03
BIT04
BIT05
BIT06
BIT07
BIT08
BIT09
BIT10
BIT11

Figure 37