International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.9954

(01/2007)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

Access networks – In premises networks

Home networking transceivers – Enhanced physical, media access, and link layer specifications

ITU-T Recommendation G.9954

ITU-T G-SERIES RECOMMENDATIONS

TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
ACCESS NETWORKS	G.9000–G.9999
In premises networks	**G.9950–G.9999**

For further details, please refer to the list of ITU-T Recommendations.

ITU-T Recommendation G.9954

Home networking transceivers – Enhanced physical, media access, and link layer specifications

Summary

ITU-T Recommendation G.9954 defines the PHY, MAC, LINK and CONVERGENCE protocol stack layers for the G.9954v2 system providing the following features:

- Operation over phoneline and/or over coax;
- PHY-layer payload transmission rates of 4 to 320 Mbit/s;
- Rate adaptive transceivers that optimize data rates and packet error rates for dynamically varying channel conditions on a per-packet basis;
- QAM modulation technique for communication over phone wire or coaxial cabling;
- Spectrum notching over phoneline for compatibility with amateur radio services;
- Synchronous MAC protocol controlled by a dynamically elected master employing a collision avoidance media access strategy;
- Support for constant and variable bit-rate data services;
- Peer-to-peer communication within a master-controlled network;
- Packet aggregation (packetization) performed within the G.9954v2 protocol stack layer up to latency limits of the service flow and available transmission bandwidth;
- Quality of service guarantees for bandwidth, jitter, latency and BER;
- QoS support for services with explicit traffic and rate specifications providing a link layer that is well suited for streaming audio and video;
- Protocol-specific convergence layers;
- Backward compatible with G.9951/G.9952 in mode A over phoneline using G.9951/G.9952 asynchronous MAC protocol;
- Coexistence and interoperability between G.9954v1 and G.9954v2 devices in a mixed network;
- Compatibility with other phoneline services such as POTS, V.90, ISDN and G.992.1, G.992.2, G.992.3, and G.992.4;
- Compatibility with other coaxial services such as VDSL, VDSL2 and cable-TV channels;
- Interoperability between phone and coax PHY layers using spectral modes A and B, thus allowing a mixed phoneline/coaxial network;
- Local and remote management of G.9954v2 devices;
- Provisions for future security extensions.

Source

ITU-T Recommendation G.9954 was approved on 9 January 2007 by ITU-T Study Group 15 (2005-2008) under the ITU-T Recommendation A.8 procedure.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

CONTENTS

ITU-T Recommendation G.9954

Home networking transceivers – Enhanced physical, media access, and link layer specifications

1 Scope

This Recommendation specifies the interoperability and compatibility for G.9954v2 stations. The requirements are written from the perspective of a compliant transmitter, although some minimum performance requirements are established for receivers. This Recommendation does not specify implementation.

A G.9954v2 station at a minimum shall be capable of transmitting and receiving over phoneline or over coax.

The structure of this Recommendation is as follows:

Clause 6: PHY layer specification over phoneline – This clause specifies the G.9954v2 PHY layer specification over phoneline.

Clause 7: PHY layer specification over coax – This clause specifies the G.9954v2 PHY layer specification over coax.

Clause 8: Media access protocol specification – This clause specifies the G.9954v2 media access protocol.

Clause 9: Compatibility specification – This clause describes the method by which backwards compatibility, coexistence and interoperability with G.9954 nodes is achieved in a mixed network of G.9954v2 and G.9954v1 nodes.

Clause 10: Quality of service – This clause describes the G.9954v2 quality of service framework.

Clause 11: Link-layer protocol specification – This clause specifies the required link layer control functionalities.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T G.9951] ITU-T Recommendation G.9951 (2001), *Phoneline networking transceivers – Foundation*.

[ITU-T G.9952] ITU-T Recommendation G.9952 (2001), *Phoneline networking transceivers – Payload format and link layer requirements*.

[ITU-T G.9953] ITU-T Recommendation G.9953 (2003), *Phoneline networking transceivers – Isolation function*.

[ITU-T G.9954] ITU-T Recommendation G.9954 (2005), *Phoneline networking transceivers – Enhanced physical, media access, and link layer specifications*.

[DSL TR-069] DSL Forum TR-069 (May 2004), *CPE WAN Management Protocol*.

3 Definitions

This Recommendation defines the following terms:

3.1 broadcast packet: A packet with the all-ones destination address (FF.FF.FF.FF.FF.FF).

3.2 capability and status announcement: A link-layer control protocol that is used to flood status information between stations with low overhead.

3.3 contention-free period: A media access period, allocated to a single network device, in which media access collisions should not (normally) occur.

3.4 convergence layer: A protocol-specific sublayer that maps transport layer protocols into the native primitives of the G.9954v2 link layer.

3.5 CS_IFG: The minimum amount of media silence that must be guaranteed between consecutive frame bursts.

3.6 device ID: A unique identifier allocated to a G.9954v2 device by the master after registration.

3.7 endpoint: A G.9954v2 device that is not the master.

3.8 EOF sequence: The 4-symbol sequence that is appended to the physical layer frame, consisting of the first four symbols of the TRN sequence.

3.9 flow: A unidirectional flow of data between network nodes characterized by traffic with well-defined QoS parameters for throughput, latency, jitter and BER.

3.10 flow ID: A unique identifier of a flow between a source and destination device.

3.11 flow signalling: A G.9954v2 link-layer protocol used to set up, modify and tear down flows.

3.12 flow specification: A specification of the characteristics of a flow in terms of its QoS traffic and rate parameters.

3.13 G.9951/G.9952: A general reference to HNT technology proposed in [ITU-T G.9951] and [ITU-T G.9952].

3.14 G.9954v1: A reference for the enhanced HNT technology proposed in [ITU-T G.9954].

3.15 G.9954v2: A reference for the enhanced HNT technology proposed in this version of the Recommendation (2007).

3.16 HNT: A general reference to home networking transceivers.

3.17 jitter: A measure of the latency variation above and below a mean latency value. The maximum jitter is defined as the maximum latency variation above and below the mean latency value and is expressed as (+Max/–Min).

3.18 latency: A measure of the delay from the point in time when a packet reaches the service access point of the HNT protocol stack until the last bit of the packet has been transmitted successfully on the wire. Mean and maximum latency measurements are assumed to be calculated over the 99th percentile of all latency measurements.

3.19 link integrity: A background process that derives a user indication that the interface is attached to the phoneline and can detect at least one other station.

3.20 link level priority: The software priority class associated with the link-layer packet. This value may be mapped when converting to/from PHY Priority.

3.21 MAC cycle: The media access period between two consecutive transmissions of the MAP control frame.

3.22 MAP: A control frame describing the media access plan for the following MAC cycle.

3.23 MAP_IFG: The amount of media silence between frame bursts used by the master in the media access planning and advertised in the MAP control frame.

3.24 master: A HNT device that has master-capabilities and was selected as the current active master. The master is responsible for planning media access timing on the network and periodically advertising the media access plan to all devices on the network.

3.25 master-controlled network: A network that contains a HNT device that is acting in the role of master.

3.26 packet aggregation: The concatenation of transport and link layer packets into a single PHY frame burst.

3.27 payload encoding: The baud and the constellation encoding (bits-per-symbol) of the payload bits.

3.28 PHY priority: The 3-bit absolute priority used by the G.9951/G.9952 media access control to rank preference to frames waiting to be transmitted on the channel. Priority 7 has preference over Priority 0.

3.29 PNT: A general reference to phoneline networking transceivers, and especially to the G.995.x and G.989.x series of ITU-T Recommendations.

3.30 preamble: The fixed signal sequence that is prepended to the physical layer frame. It consists of four copies of the TRN sequence.

3.31 priority group: A group of sub-burst priority slots with the same assigned priority.

3.32 priority slot: A sub-burst slot transmission opportunity, with an assigned priority, which is reserved for transmissions of data with a priority greater than or equal to the assigned priority.

3.33 QoS contract: A contract defining a set of negotiated QoS flow parameters between devices involved in a flow. A QoS contract is negotiated between devices at the endpoints of a flow in order to establish buffering and channel (BER/PER) constraints. A QoS contract is negotiated between flow source device and master in order to constrain bandwidth, latency and jitter requirements.

3.34 registration: The process used by a HNT network device to inform the active network master of its existence and its intention to negotiate future QoS contracts.

3.35 system margin: Set of values for impairment levels at which a receiver does not exceed a specified frame error rate on a given test loop.

3.36 sub-burst slot: A time-slot which is smaller than the size of a minimum-sized burst which represents an opportunity for the initiation of a data transmission by a network device.

3.37 transmission opportunity (TXOP): An interval of media time, with distinct start-time and length relative to the start of the MAP that can be used by an HNT device for the transmission of frames.

3.38 TRN16: The 16-symbol white, constant amplitude QPSK sequence which is used in the physical layer preamble.

3.39 TR-069: A CPE WAN management protocol (CWMP) managed and published within the DSL Forum.

3.40 valid CS frame: A description of the minimum transmitter signal which should be acceptable to implementations of carrier sense.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ADSL	Asymmetric Digital Subscriber Line
BER	Bit Error Ratio
BPS	Bits Per Symbol
CBR	Constant Bit Rate
CFTXOP	Contention-Free TXOP
CR	Collision Resolution
CS_IFG	Carrier-Sense IFG
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTXOP	Contention TXOP
DFPQ	Distributed Fair Priority Queuing (the G.9951/G.9952 enhanced method for collision resolution (see BEB))
DOCSIS	Data-Over-Cable System Interface Specification
FEC	Forward Error Correction
G.9951/G.9952	Device supporting the G.9951/G.9952 protocol
G.9954v1	Device supporting the 2005 version of the G.9954 protocol
G.9954v2	Device supporting the 2007 version of the G.9954 protocol
HCS	Header Check Sequence (a CRC-8 that covers portions of the header and Ethernet address fields)
HNT	Home Networking Transceiver
ICG	Inter-Cycle GAP
IFG	Inter-Frame GAP
LARQ	Limited Automatic Repeat reQuest (protocol for impulse noise error correction)
MAP_IFG	Media Access Plan IFG
MII	Media Independent Interface (defined by IEEE Std 802.3 Clause 22)
MPDU	MAC Protocol Data Unit
NEXT	Near-End Crosstalk
NID	Network Interface Device (a subscriber line protection device installed at the boundary between the subscriber loop and the in-premise wiring)
PAR	Peak-to-Average Ratio
PDU	Protocol Data Unit
PE	Payload Encoding
POTS	Plain Old Telephone Service (referring to telephony services using the 0-4 kHz spectrum on the phoneline)
QAM	Quadrature Amplitude Modulation
RG	Residential Gateway

RSVP	Resource Reservation Protocol
Self-NEXT	Near-End Crosstalk from other systems of the same type
SI	Scrambler Initialization
SP	Service Provider
TXOP	Transmission Opportunity
USB	Universal Serial Bus
UTXOP	Unallocated TXOP
VBR	Variable Bit Rate

5 Introduction

5.1 G.9954v2 protocol stack overview

The G.9954v2 protocol stack is an integrated protocol stack handling PHY, Data Link, Convergence and Management Layers. The G.9954v2 protocol stack supports synchronous collision-free media access method.

The media access method depends on the existence of a G.9954v2 device on the network that is able to assume the role of network master. Such a device is referred to as the master device or just master. A device that is able to assume the role of master on the network is referred to as a master-capable device. A master-capable device is a regular G.9954v2 device that also supports functional capabilities that allow it to assume the role of master, in the absence of an active master on the network.

The master is responsible for controlling media access by planning media access timing on the network and periodically advertising the media access plan to all devices on the network. The periodic timing is referred to as a MAC cycle. G.9954v2 nodes synchronize with the periodic MAC cycle and time their transmissions in accordance with the transmission timing described in the media access plan (MAP).

5.1.1 Compatibility and interoperability

The G.9954v2 PHY layer is composed of two PHY layers, where the first is specified for phonewire network and the other is specified for a coaxial cable-based network. Each maintains compatibility with existing services on the same network.

The PHY layer over phoneline maintains backward compatibility with G.9954v1 (HomePNA V3.0) using spectral mode A. It is also compatible with other phoneline services such as POTS, V.90, ISDN and G.992.1, G.992.2, G.992.3, and G.992.4.

Compatibility with amateur radio services is due to spectrum notching.

The PHY layer over coax is compatible with other coaxial services such as VDSL, VDSL2 & cable-TV channels.

The two PHY layers enable optional interoperability with each other using spectral modes A & B thus allowing a mixed phoneline/coaxial network.

A G.9954v2 device operating at mode A over phoneline will be backward compatible with a G.9951/G.9952 device. At this mode when a G.9954v2 device detects a G.9951/G.9952 device it will return to work as a G.9951/G.9952 device.

5.1.2 Media access method

The G.9954v2 MAC protocol is a synchronous MAC protocol that coordinates media access under master control. The protocol is synchronous in the sense that all G.9954v2 nodes on the network are synchronized to a periodic MAC cycle and transmissions are pre-planned and accurately timed.

The G.9954v2 MAC protocol is used to support different kinds of services including asynchronous best-effort data services and isochronous constant and variable bit-rate streaming services such as required by telephony, audio and video.

In a G.9954v2 network, media access is pre-planned and a collision avoidance (CA) strategy is used during normal data-transfer operations. Collision avoidance together with packet aggregation provides efficient use of the media and provides the infrastructure for supporting Quality of Service guarantees.

The G.9954v2 MAC protocol supports bridging to other synchronous protocols, such as IEEE 1394, USB, etc., and to broadband access protocols such as DOCSIS and IEEE 802.16, using the protocol convergence layer. Furthermore, the master-controlled network model, used in the G.9954v2 MAC, is a natural model for broadband access networks and is well suited to an architecture containing a residential gateway (RG).

5.1.3 Quality of Service

The G.9954v2 MAC supports both priority-based and parameter-based QoS methods.

Priority-based QoS supports priority classification using eight priority levels and provides a basic QoS mechanism for differentiating between different kinds of services. This mechanism is compatible with IEEE 802.1D recommendations and the VLAN priority tag (IEEE 802.1P) and the PRECEDENCE bits defined in the original interpretation of the type of service (TOS) field found in an IP packet using the differentiated services (Diffserv) protocol.

Parameter-based QoS supports traffic specifications defined in terms of rate, latency, jitter and other parameters. This provides controls to a finer level of detail than just a relative ordering of packets as supported by the priority-based scheme.

The G.9954v2 QoS mechanism is based on the concept of a flow, which represents a unidirectional flow of data between network nodes based on well-defined QoS parameters that allow strict control over network throughput, latency, jitter and BER parameters.

Flows are set up and torn down on a service-by-service basis. The G.9954v2 link-layer control (LLC) and MAC sublayers are responsible for scheduling the transmission of packets on flows in such a way so as to enforce respective traffic/QoS parameters. Bandwidth is reserved for a flow during its lifetime and this is reflected in the media access plan (MAP) prepared by the master G.9954v2 node. Bandwidth requirements for a flow may also be modified throughout its lifetime in order to support changing bandwidth requirements that are characteristic of "bursty" and variable bit-rate (VBR) data streams.

It is the responsibility of the convergence sublayer to map incoming data streams onto an appropriate flow in order to meet QoS requirements.

Flows may be set up automatically upon service invocation or they may be established at initialization time according to a predefined specification (e.g., part of the convergence layer) or configuration data. Flows may similarly be torn down automatically upon detection of inactivity in order to free network resources associated with the flow.

5.1.4 Performance

The G.9954v2 protocol improves on the performance of the G.9951/G.9952 and G.9954v1 asynchronous mode MAC protocol by using a pure collision avoidance media access method. In addition, the G.9954v2 MAC protocol improves on network utilization compared to G.9951/G.9952

by supporting aggregation of multiple MAC protocol data units (MPDUs) into a single PHY layer burst (frame).

The above performance gains are related to the G.9954v2 MAC protocol itself and further performance gains and advantages may be expected in implementations themselves.

5.1.5 External interfaces and protocols

The G.9954v2 protocol supports interfaces and bridging to external protocols through the convergence sublayer in the protocol stack.

It is the responsibility of the protocol convergence sublayer to map data packets arriving from a particular interface onto the *flows* appropriate for the particular data service.

The G.9954v2 protocol stack does not assume the existence of an external host processor and is able to directly interface, in hardware, to an external chip, possibly running a different protocol. In this configuration, the protocol convergence sublayer is assumed to co-reside with the MAC and Link Layers in an integrated G.9954v2 chip. Alternatively, the G.9954v2 convergence sublayer may actually execute, in part or entirely, on an external host processor.

The external protocols addressed explicitly in the G.9954v2 convergence sublayer description include the IEEE 802.3/Ethernet protocols and Internet protocol (IP). Additional protocols are also supported by the convergence layer through a generic packet classification mechanism.

Protocol mapping and convergence at an explicit level of the protocol stack supports synchronization between external and home networks. This is described in more detail in Appendix III. Furthermore, given QoS defined in terms that are similar to those of the external network, this further supports the extension of QoS methods from external networks into the home network.

5.1.6 Security and privacy

A G.9954v2 network node must register with the master in order to connect to the network and to initiate data transfer. This centrally controlled network model provides the necessary infrastructure for network admission control, security and privacy.

In the network admission control model, a device authorization list defines which devices are able to connect to the master and gain access to the network and its resources. Device identification is performed using the device's hardware MAC address. Network access may be denied on the basis of authorization information accessible to the master. The management of the device authorization list and its physical location are beyond the scope of this Recommendation.

Privacy of data is supported using shared private-key encryption methods. Link-layer privacy controls ensure that only devices with knowledge of the shared key are able to communicate and receive network data. The actual key management and distribution protocols are beyond the scope of this Recommendation.

Privacy may be required in the home in order to protect home content from exposure to unauthorized collection and monitoring caused by crosstalk. Encryption can be used to protect data in G.9954v2 transmissions rather than relying only on security mechanisms based on restricting receiver sensitivity.

Network admission controls, security and privacy support are all optional features.

5.1.7 Management support

Given a home network model based on a Residential Gateway through which services are delivered into the home, the need to be able to both locally and remotely manage, configure, monitor and troubleshoot home networks becomes critical. In this model, the RG is the primary means for collecting and reporting information about the state and health of the home network.

To support this functionality, G.9954v2 devices provide the following management functions:

- Configure, control and monitor all G.9954v2 devices on the network;
- Provide local and remote access to all devices;
- Access to all devices through the Gateway (Master);
- Support diagnostics using a standard message-based protocol (certification and diagnostics protocols – see clause 11.9);
- Support TR-069 data model;
- Support other higher-level management interfaces (e.g., SNMP, HTTP, etc.).

The management facilities are intended to provide access to the following management information:

- PHY information;
- Network information;
- Device statistics;
- QoS information;
- Device information;
- Configuration information;
- Authorization and security information;
- Version information.

5.2 Network reference model

This Recommendation defines base-level PHY, MAC, LINK and CONVERGENCE layer functionality.

The primary interface specified is the wire-side electrical and logical interface (W1) between a G.9954v2 station and the phone wire or the coaxial cable as shown in Figure 5-1. This Recommendation defines host-side interfaces in terms of example interfaces such as IEEE Std 802.3 logical link level frame formats, addressing and broadcast/multicast behaviour. Several options exist for host-side interfaces, with the MII interface recommendation described in Appendix II.

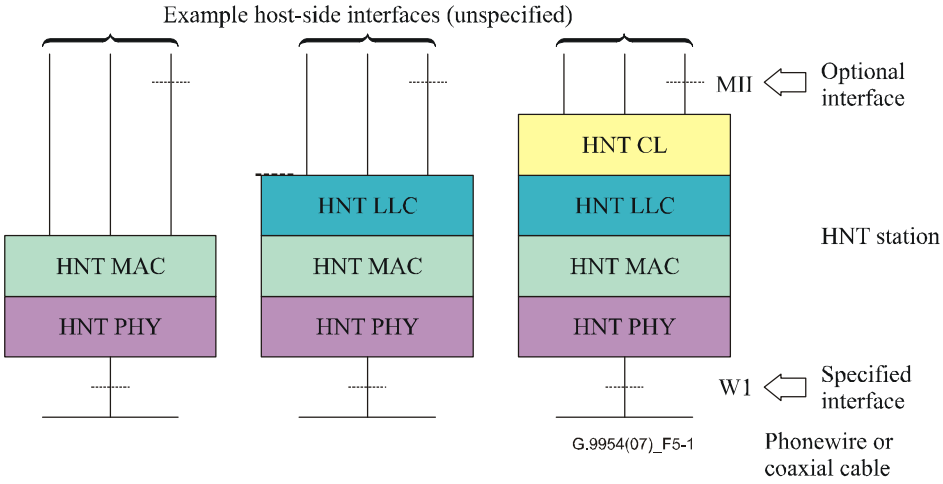

Figure 5-1 – Interfaces

Figure 5-2 illustrates the network reference model. The model assumes a home network connected to an external access network via a residential gateway (RG) or (Internet gateway device). The home network is composed of a number of network nodes all communicating over a shared media infrastructure within the home premises. In this model, management of the home network can be performed remotely, from the Broadband Service Management entity, using the Residential Gateway as the access point into the home network. In Figure 5-2, the RG assumes the role of network master and is responsible for coordinating media access within the home network. Although the RG is a natural candidate for being network master, it should be noted that this represents only one possible configuration since any device can assume the role of master without changing the network model. In the network model, each network device on the home network is assumed to be running an instance of the HNT protocol stack.

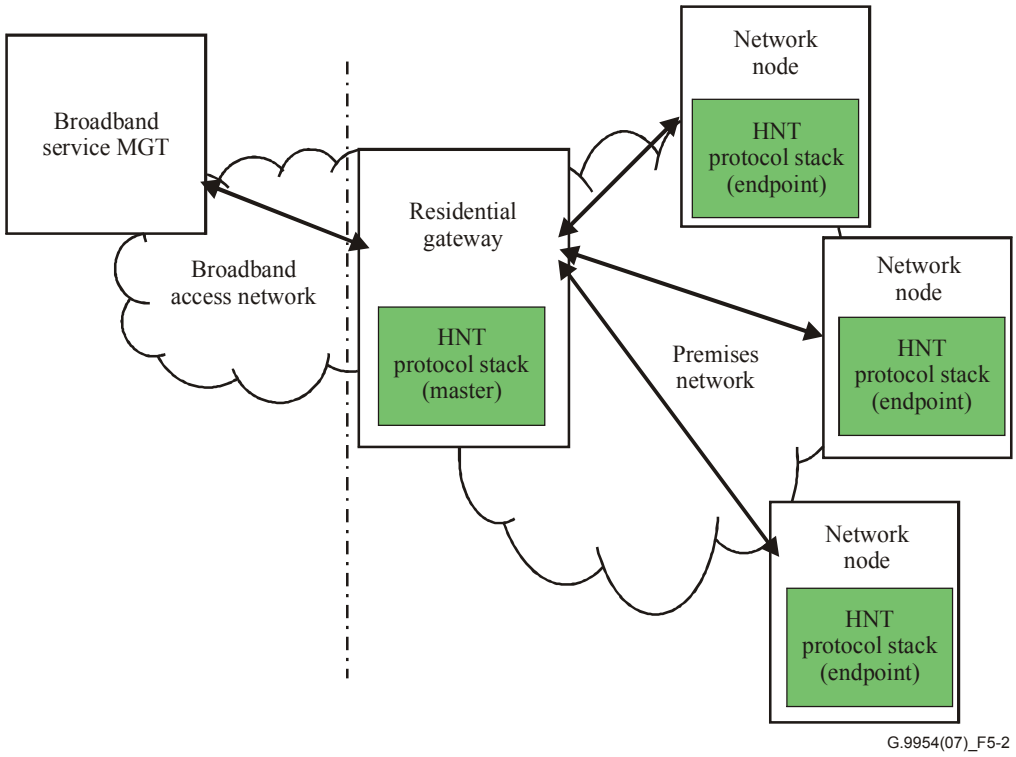

Figure 5-2 – Network reference model

The HNT system implements a *shared medium* single-segment network, as shown in Figure 5-3. All stations on a segment are logically connected to the same shared channel either on the phonewire or the coaxial cable or both. Multiple HNT network segments and other network links can be connected through ISO network Layer 2 (L2 or Data Link) or Layer 3 (L3 or IP) relays. Layer 1 relays (PHY layer repeaters) are not defined in this Recommendation.

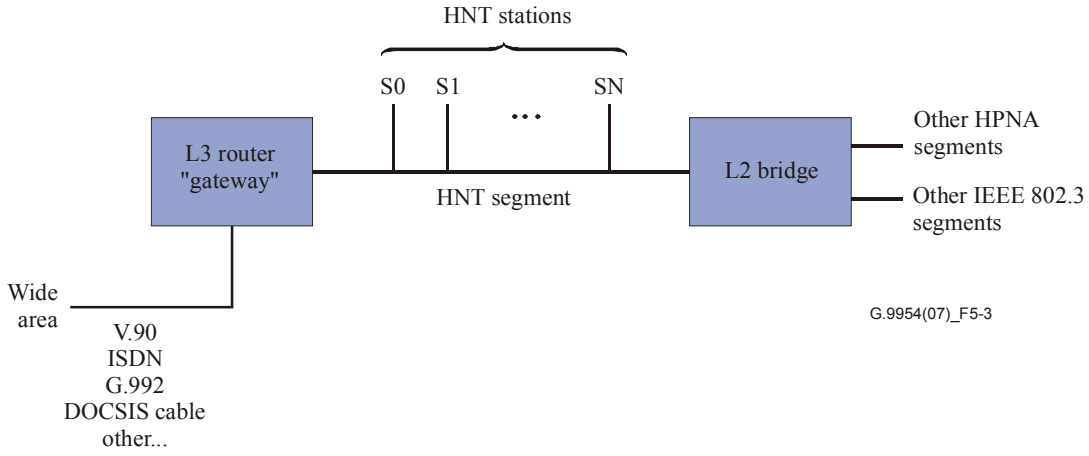

Figure 5-3 – HNT shared medium network segment on the coaxial cable

As seen in Figure 5-4, the G.9954v2 network model assumes a home network composed of a variety of types of network devices, connected to the shared media home phoneline and coaxial cable network backbone. It assumes a single broadband connection (e.g., using phoneline xDSL services), to an external access network, through an Internet gateway device and possible bridges to other home network segments, possibly based on other home networking technologies (e.g., wireless, power-line).

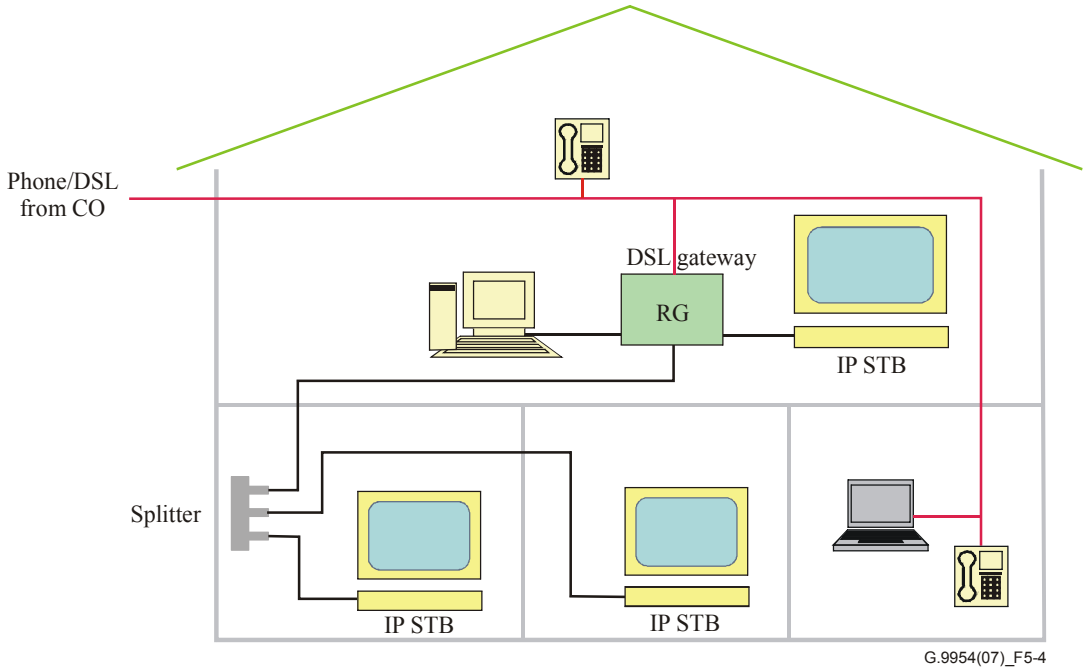

Figure 5-4 – G.9954v2 home network

In Figure 5-3, a Layer 3 router/gateway is shown which interconnects a wide-area network link to the in-house HNT network. Such wide-area link might be provided via subscriber line (V.90, ISDN, G.992.x), cable (DOCSIS) or wireless link. Also shown is a L2 bridge that interconnects the first HNT network with other HNT network segments or IEEE 802.3 (10BASE-T, 100BASE-T) networks. (Where multiple HNT network segments exist, they should not share the same cables infrastructure unless the segments are properly isolated in the HNT frequency band.)

Figure 5-5 shows the HNT standard relationship to the ISO/IEC Open Systems Interconnection reference model.

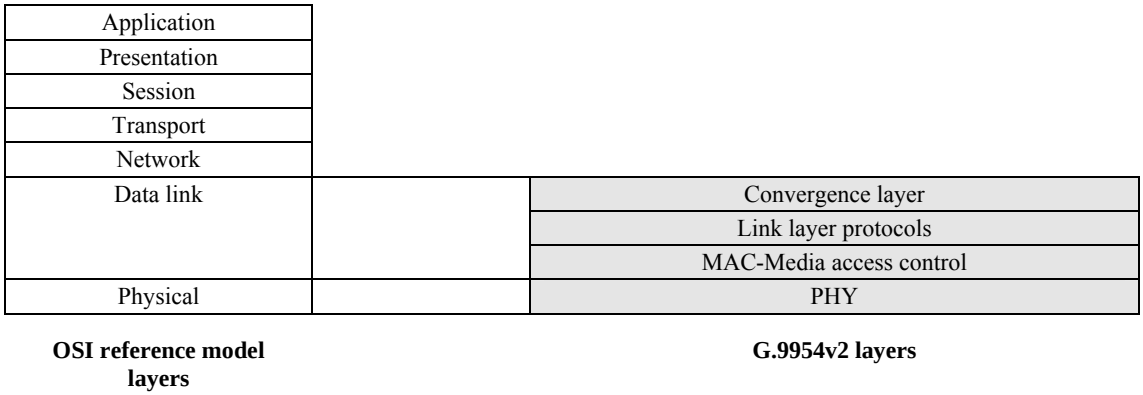

Figure 5-5 – Relationship to ISO/IEC open systems interconnection

The HNT network standard is designed to work over "as is" customer premise wiring or coaxial cabling.

The topologies anticipated on the phonewire are random combinations of star, tree and multipoint bus wiring: see Figure 5-6 for an example. Here, the "plain old telephone service" (POTS) network interface device (NID) is shown with the outside subscriber loop to the left, and the premises wiring splitting in a "star" from the NID to several wiring runs. Each run may have one or more modular connectors at wall plates, and variable length *extension* wires (shown as double lines) run from the wall plates to the attached POTS or HNT device. In the example, stations A and B are on one bus; station C is on a second bus, which is un-terminated at the end; station E is at the end of a direct run from the NID; and stations F and G share a single wall plate via a two-outlet adapter. Many other topologies are possible.

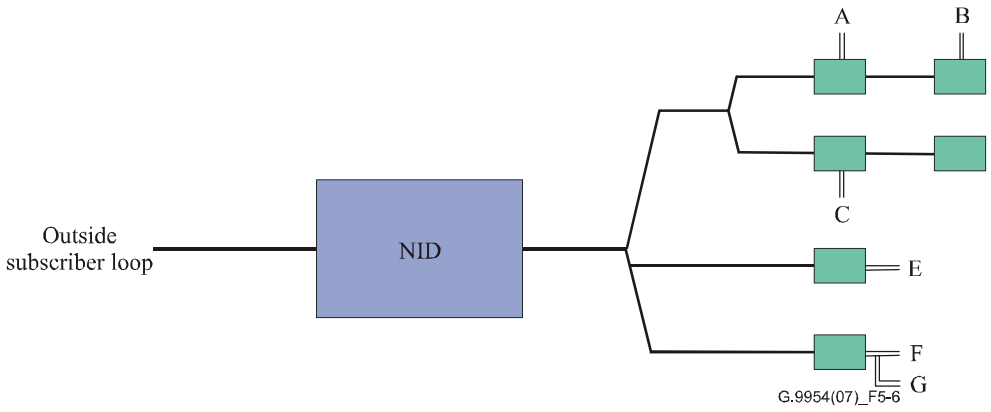

Figure 5-6 – Reference wiring topology over phonewire

The topologies anticipated on the coaxial cable are various combinations of tree and bridged-tap cabling, usually built for a distribution of signals from the outside cable to the coaxial outlets; see Figure 5-7 for an example. Here the outside subscriber cable is connected to the input of a main splitter with three output ports. The signal splits to three output paths where two of them are connected directly to outlets while a third path goes into a second splitter with two ports. Each output of the two-port splitter is connected to two bridge-taped outlets. The outlets close to the two-port splitter also serve as non-symmetric splitters since they split the signal between the local outlet and the other chained outlets in the bridge topology. In the example, stations A and B are on the same bridge-tap chain; station C is on a second chain; and station D is connected directly to one of the output ports of the main splitter. Many other topologies are possible.

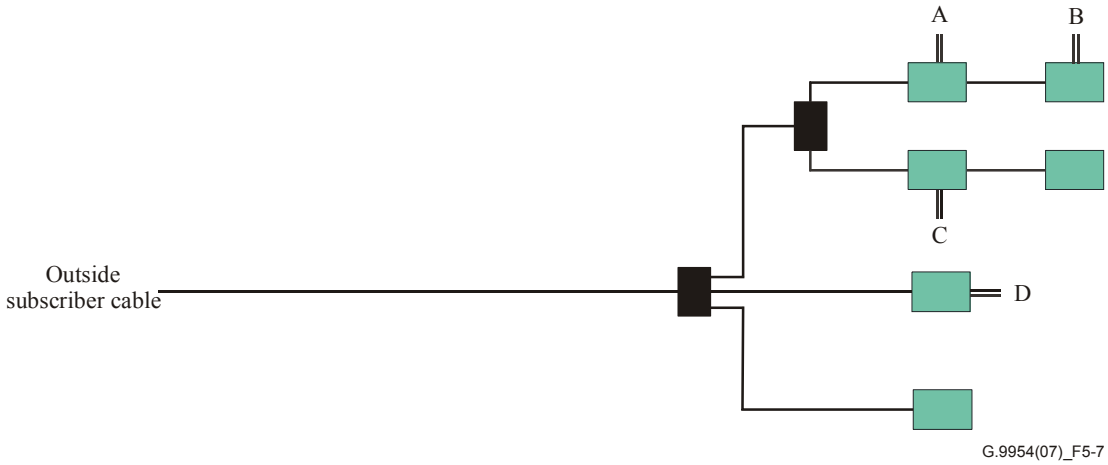

Figure 5-7 – Reference wiring topology over coaxial cabling

The G.9954v2 protocol is based on a master-controlled network model. The master-controlled network model assumes the existence of a master node that provides the timing on the network and synchronizes media access to all G.9954v2 network devices.

Although media access in a master-controlled network is controlled by the master, communication between two devices does not traverse the master – rather, devices communicate directly (peer-to-peer) at the master-designated time. Any device on the network can potentially act as the master, although it is a role most naturally assumed by a gateway or server device.

5.3 The protocol stack

The G.9954v2 protocol stack provides layer 1 (PHY) and layer 2 (Data-link) services for transmitting and receiving packets over a wired media using the G.9954v2 protocol. The protocol stack used by the G.9954v2 is illustrated in Figure 5-8.

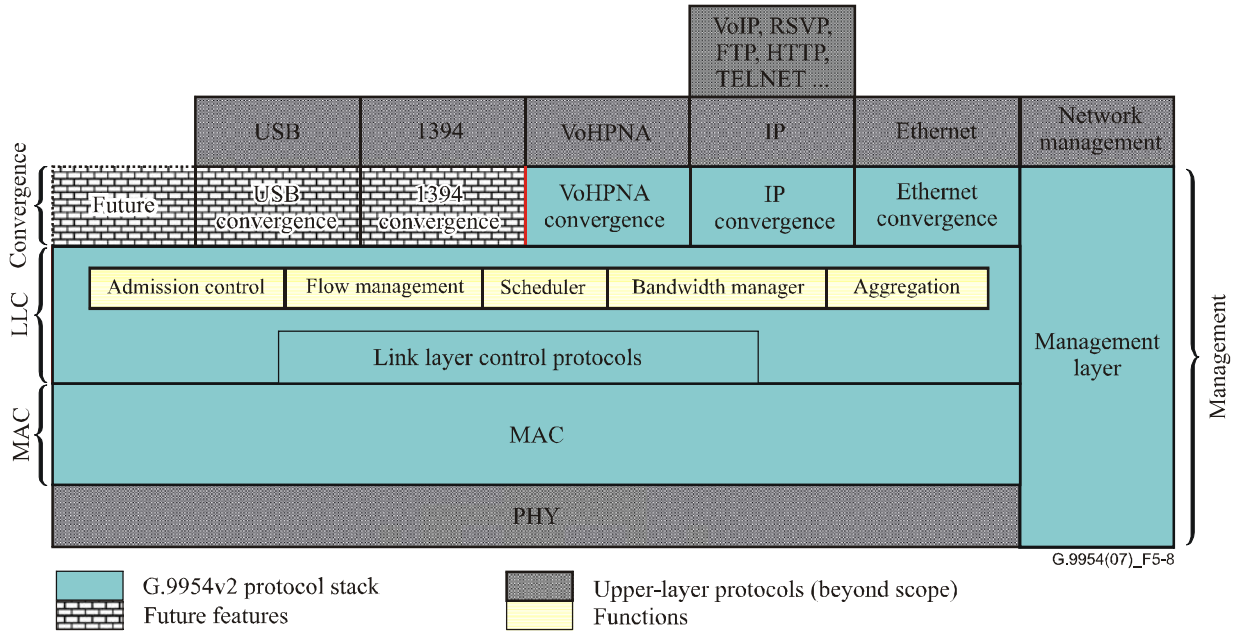

Figure 5-8 – G.9954v2 protocol stack

5.3.1 PHY layer

The PHY layer provides transmission and reception of physical layer frames using QAM modulation technique over phone wire media or coaxial cabling. The PHY layer over phone supports 2-, 4-, 8-, 16-Mbaud symbol rates with 2 to 10 bits-per-symbol constellation encoding. The PHY layer over coax supports 2-, 4-, 8-, 16- and 32-Mbaud symbol rates with 2 to 10 bits-per-symbol constellation encoding. This provides a PHY layer data rate in the range of 4-320 Mbit/s within an extended 4-36 MHz PSD mask supporting up to a 32-MHz bandwidth.

5.3.2 Data-link layer

The data-link layer is composed of three sublayers – the MAC, LLC and convergence layers.

5.3.2.1 G.9954v2 MAC sublayer

The MAC sublayer is responsible for managing access to the physical media using a Media Access protocol. It uses the PHY layer to schedule the transmission of MAC Protocol Data Units (MPDUs) over the physical media within PHY layer transport frames.

The G.9954v2 MAC sublayer uses a synchronous MAC protocol based CSMA/CA techniques to provide collision-free media access. Media access is performed under master control, and collisions are avoided by pre-planning the timing of all media-access.

The G.9954v2 MAC maintains a vector defining the media access timing planned by the master. Media access timing is planned according to quality of service (QoS) constraints for required network services and the plan is broadcast periodically to all G.9954v2 nodes. G.9954v2 MACs are responsible for guaranteeing that all media access is performed according to the plan by restricting transmissions only to transmission opportunities (TXOPs) allocated explicitly to it (or to its services) by the master or allocated to a group that it belongs to. The G.9954v2 master plans media access down to the service level and as such, a G.9954v2 MAC may schedule packets entirely using

the master plan. Alternatively, a G.9954v2 MAC is allowed to exercise some local QoS intelligence by making scheduling decisions itself within the constraints of the transmission opportunities TXOPs allocated to it in the MAP.

The MAC sublayer is further responsible for providing control information to the PHY layer in order to control the physical characteristics of the transmitted data.

5.3.2.2 G.9954v2 LLC sublayer

The LLC sublayer is responsible for performing link control functions. In particular, it is responsible for managing information concerning network connections, for enforcing quality of service (QoS) constraints defined for the various system data flows and for ensuring robust data transmission using rate negotiation, Reed-Solomon coding techniques and ARQ (automatic repeat request) techniques.

In addition, the G.9954v2 MAC protocol requires the support of link control protocols that manage network admission and flow setup and teardown procedures. These protocols are used to manage the information about connected devices and their associated service flows.

In addition to the link layer control protocols required by the G.9954v2 MAC, the following Link Layer control functions are required: Scheduling, Bandwidth management, Flow management, Network admission and Packet aggregation.

Packet aggregation is used to concatenate multiple MPDUs, within a single PHY layer frame. This concatenation technique is used to increase the size of the PHY frame in order to reduce the overall per-packet protocol overhead. However, the degree of aggregation performed is a function of the latency requirements of services and the size of the allocated transmission opportunity. The LLC sublayer is responsible for performing this framing and de-framing and for maximizing the size of a burst within the constraints defined by the media access plan.

5.3.2.3 Convergence layer

The convergence layer is a protocol-specific set of sublayers that map various transport layer protocols into the native primitives of the LLC sublayer. The LLC sublayer provides a protocol independent interface and a well-defined QoS framework. It is the responsibility of the convergence sublayer to translate the native protocol into this underlying framework.

The convergence sublayer may use protocol or configuration specific information to perform the translation.

5.3.2.4 Management layer

The management layers described in the protocol stack in Figure 5-8 includes both network layer management and G.9954v2 management facilities. Network layer management operates on network and transport layers, using higher-level management protocols and frameworks such as SNMP or TR-069 and as such is beyond the scope of this Recommendation.

G.9954v2 management includes all those facilities that are required in order to collect information from the PHY, MAC, link and convergence layers of the G.9954v2 device or remote devices and to exercise control over them. G.9954v2 management supports both local and remote management capabilities. This means that management operations may be performed from a local host interfacing to the G.9954v2 device from the host side or from a management entity interfacing with the G.9954v2 device from the network (wire) side using a peer management protocol.

6 PHY layer specification over phoneline

6.1 Overview

The G.9954v2 PHY layer over Phoneline supports two spectral modes. Each spectral mode supports a different band range while both supporting the same baud rates set:

- Spectral Mode A: 4-20 MHz; 2, 4, 8, 16 MBauds (similar to Mask #2 defined in [ITU-T G.9954]);
- Spectral Mode B: 12-28 MHz; 2, 4, 8, 16 MBauds.

The actual spectral mode to be used in the network is pre-configured according to considerations of coexistence with other services and line characteristics. The pre-configuration technique is implementation dependent and is out of the scope of this Recommendation. The network configuration is limited to a homogenous spectral mode for all devices.

Constellation sizes range from 2 to 10 bits per symbol, specifying PHY layer payload modulation rates that range from 4 Mbit/s to 160 Mbit/s.

Information is transmitted on the channel in bursts. Each burst or physical layer frame consists of PHY-layer payload information encapsulated with PHY preamble, header and postamble. The PHY-layer payload refers to the portion of the link level frame that is modulated at the payload rate, which is typically higher than the header rate. Hereafter, "payload" refers to the PHY-layer payload unless otherwise specified.

The following describes the physical layer formatting.

6.2 Transmitter reference model

The transmitter block diagram is shown in Figure 6-1. This consists of a frame processor, data scrambler, bit-to-symbol mapper, and QAM modulator, as defined in the following clauses.

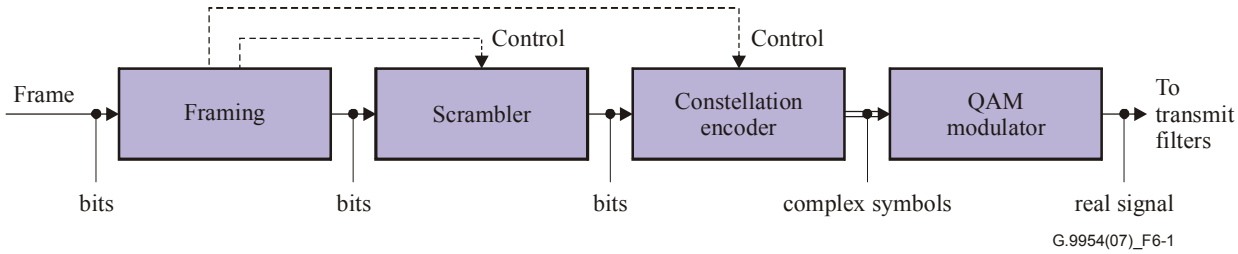

Figure 6-1 – Transmitter block diagram

6.3 Framing

The frame format is shown in Figure 6-2. This consists of a low-rate header section, a variable-rate payload section, and a low-rate trailer. Some parts of the frame are not scrambled, as described in clause 6.4.

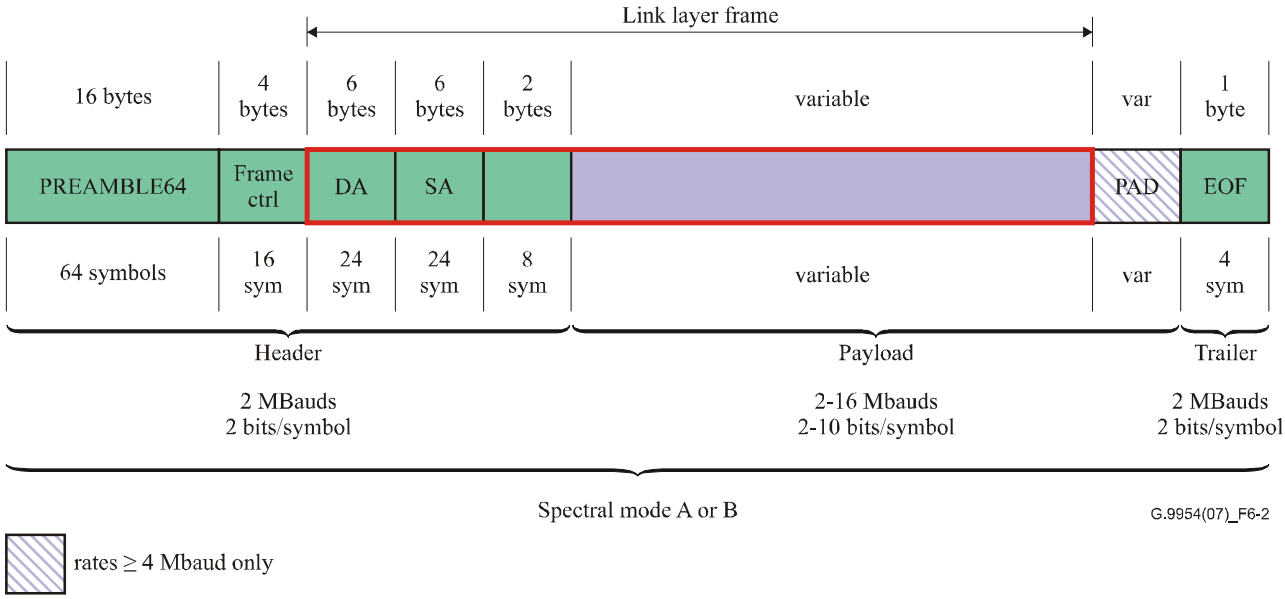

Figure 6-2 – PHY frame format

The interpretation of the two-byte field following the SA and the variable length field following it is given by the link layer frame format defined in clause 11.

6.3.1 Bit order

Except where otherwise stated, all fields are encoded most significant octet first, least significant bit first within each octet. Bit number 0 is the LSB within a field. Diagrams show MSB bits or octets to the left.

6.3.2 Preamble definition

The PREAMBLE64 is defined as a repetition of four 16-symbol sequences (TRN16) that result from encoding **0xfc483084** (in the order defined in clause 6.3.1) at 2 MBaud, 2 bits per symbol, with the scrambler disabled.

NOTE – The TRN16 is a white, constant amplitude QPSK sequence. The preamble was designed to facilitate:

- power estimation and gain control;
- Baud offset estimation;
- equalizer training;
- carrier sense.

6.3.3 Frame control definition

The frame control field is a 32-bit field defined in Table 6-1.

Table 6-1 – Frame control fields

Field	Bit number	Bits	Description
FT	31:28	4	Frame type
EID	27:25	3	Extended identifier
RSVD	24:24	1	Reserved. This field shall be set to zero by the transmitter and the receiver shall discard frames with non-zero values.
ID	23:20	4	Identifier
SI	19:16	4	Scrambler initialization
PE	15:8	8	Payload encoding
HCS	7:0	8	Header check sequence

Hence, with the bit-ordering defined in clause 6.3.1, the frame control fields are transmitted in the order shown in Figure 6-3.

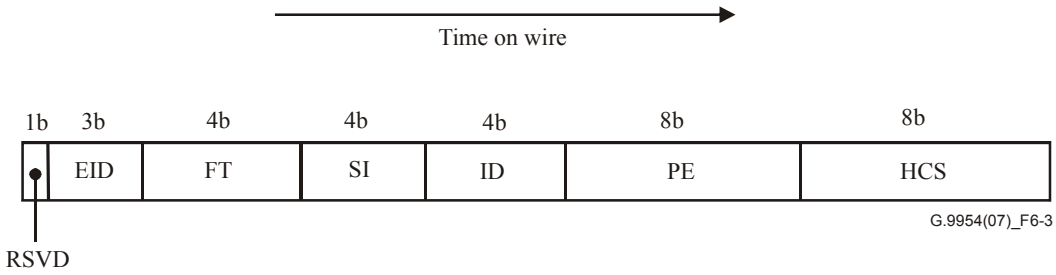

Figure 6-3 – Frame control field order

6.3.3.1 Frame type

The frame type (FT) is a four-bit field. The details of this field are defined in clause 8.13.1.

6.3.3.2 Reserved bits (RSVD)

This field shall be set to zero by the transmitter, and the receiver shall discard any frame where this field is non-zero.

6.3.3.3 Extended Identifier (EID)

This extended identifier is a 3-bit field. The details of this field are defined in clause 8.13.2.

6.3.3.4 Scrambler initialization bits

This 4-bit field shall be set to the value used to initialize the scrambler, as described in clause 6.4.

6.3.3.5 Identifier

The Identifier is a 4-bit field. The details of this field are defined in clause 8.13.3.

6.3.3.6 Payload encoding

This field determines the baud rate and the constellation encoding of the payload bits. This field is defined by the following sub-fields.

Table 6-2 – Payload encoding fields

Field	Bit number	Bits	Description
EBPS	7	1	Extended bits per symbol
Baud	6:3	4	Symbol rate
BPS	2:0	3	Bits per symbol

6.3.3.6.1 Extended Bits per Symbol bit

The EBPS is used to indicate an extended encoding of the BPS field. More specifically, it is used to extend the interpretation of the BPS field when EBPS = 1. This is described in detail in clause 6.3.3.6.3.

6.3.3.6.2 Symbol rate

This field indicates the symbol rate/baud rate of the payload bits:

Table 6-3 – Symbol rates

Baud value	Interpretation
0-3	Reserved on transmit, discard frame on receive
4	Symbol rate = 2 MHz
5	Symbol rate = 4 MHz
6	Symbol rate = 8 MHz
7	Symbol rate = 16 MHz
8-15	Reserved on transmit, discard frame on receive

6.3.3.6.3 Bits per symbol

The values are defined as follows:

Table 6-4 – Bits-per-symbol encoding

EBPS value	BPS value	Interpretation
0	0	Reserved on transmit, discard frame on receive
0	1	2 bits per symbol
0	2	3 bits per symbol
0	3	4 bits per symbol
0	4	5 bits per symbol
0	5	6 bits per symbol
0	6	7 bits per symbol
0	7	8 bits per symbol
1	0	8-round constellation; 8 bits per symbol
1	1	9-round constellation; 9 bits per symbol
1	2	10-round constellation; 10 bits per symbol
1	3-7	Reserved on transmit, discard frame on receive

6.3.3.7 Header check sequence (HCS)

An 8-bit cyclic redundancy check (CRC) is computed as a function of the 128-bit sequence in transmission order starting with the FT bits and ending with the Ethernet source address (SA) bits, with zeros substituted for the as-of-yet uncomputed HCS field. The encoding is defined by the following generating polynomial.

$$G(x) = x^8 + x^7 + x^6 + x^4 + x^2 + 1$$

Mathematically, the CRC value corresponding to a given frame is defined by the following procedure.

The first 8 bits of the input bit sequence in transmission order are complemented.

The 128 bits of the sequence in transmission order are then considered to be the coefficients of a polynomial $M(x)$ of degree 127. (The first bit of the FT field corresponds to the x^{127} term and the last bit of the SA field corresponds to the x^0 term.)

$M(x)$ is multiplied by x^8 and divided by $G(x)$, producing a remainder $R(x)$ of degree ≤ 7.

$R(x)$ is multiplied by $H(x)$ to produce $N(x)$, where $H(x)$ is defined as $H(x) = x^7 + x^6 + x^5 + x^4 + x^2 + x + 1$.

$N(x)$ is divided by $G(x)$, producing a remainder $R'(x)$ of degree ≤ 7.

The coefficients of $R'(x)$ are considered to be an 8-bit sequence.

The bit sequence is complemented and the result is the CRC'.

The 8 bits of the CRC' are placed in the HCS field so that x^7 is the least significant bit of the octet and x^0 term is the most significant bit of the octet. (The bits of the CRC' are thus transmitted in the order $x^7, x^6, .. x^1, x^0$.)

Although the HCS is embedded within the protected bit stream, it is calculated in such a way that the resulting 128-bit stream provides error-detection capabilities identical to those of a 120-bit stream with an 8-bit CRC appended. The resulting 128-bit sequence, considered as the coefficients of a polynomial of degree 127, when divided by $G(x)$, will always produce a remainder equal to $x^7 + x^6 + x + 1$.

The input bits are unscrambled.

Because all fields covered by the HCS are transmitted at 2 MBaud and 2 bits per symbol (as described in clause 6.5.1), these fields should be received correctly in many cases where the payload is received in error. The HCS may be used in conjunction with soft-decision error statistics to determine with high probability whether the header was received correctly. This knowledge may be useful for optimizing the performance of ARQ and/or rate negotiation algorithms.

6.3.4 Link layer frame

The bit fields following the frame control field and preceding the pad field are defined in the G.9954v2 link-layer specification in clause 11. The first 6 octets are the Destination Address and the next 6 octets are the Source Address.

The presence of the DA and SA in the low-rate header enables reliable error-detection, which is useful for rate selection.

6.3.5 Pad

For payloads encoded at rates greater than or equal to 4 MBaud, a variable-length *pad* field consisting of an integer number of octets shall be inserted. The last octet of the pad field (PAD_LENGTH) shall be 255 (0xff) or the number of zero octets (0x00) preceding PAD_LENGTH, whichever is less. The number of zero octets shall ensure that the minimum length of the transmission, from the first symbol of the PREAMBLE64 through the last symbol of the end of frame delimiter, is at least 92.5 µs. For 2-MBaud payloads, there shall not be a pad field.

An example of a compliant formula for generating PAD_LENGTH is:

$$\min\left\{255, \left\lceil \frac{(92.5\,\mu s - 68\,\mu s - 2\,\mu s) \times \mathrm{B}\frac{Msymbol}{\mathrm{second}} \times \mathrm{BPS}\frac{bit}{symbol}}{8\frac{bit}{octet}} \right\rceil - 1 - N\right\}$$

where the baud, B is either 4, 8, or 16, BPS is the bits per symbol, *N* is the number of octets in the part of the link layer frame transmitted in the payload-rate, 68 µs is the length of the header, and 2 µs is the length of the trailer. If the formula results in a negative quantity, it means that no pad is required.

6.3.6 End-of-Frame (EOF) delimiter

The End-of-Frame sequence consists of the first 4 symbols of the TRN sequence, or **0xfc** encoded as 2 bits per symbol at 2 MBaud.

This field is provided to facilitate accurate end-of-carrier-sensing in low-SNR conditions. A station demodulating a frame can use this field to determine exactly where the last payload symbol occurred.

6.4 Scrambler

The scrambler is the frame-synchronized scrambler shown in Figure 6-4, which uses the following generating polynomial.

$$G(x) = x^{23} + x^{18} + 1$$

Bits 15 through 18 of the shift register shall be initialized with a 4-bit pseudo-random number. This value shall be placed in the SI field defined in 6.3.3.4 in the order such that register position 15 is the MSB (bit 19 of frame control) and bit 18 is the LSB (bit 16 of frame control).

The scrambler shall be bypassed during the preamble bit field and the first 16 bits of frame control. The scrambler shall be initialized and enabled starting with the 17th bit of the frame control field.

The scrambler shall be bypassed after the last bit of the link layer frame, or the last bit of the PAD field, if present. The EOF sequence shall not be scrambled.

The use of a pseudo-random initial scrambler state results in a more uniform power spectral density (PSD) measured over multiple similar frames. This eliminates the problem of tones in the PSD from highly correlated successive packets.

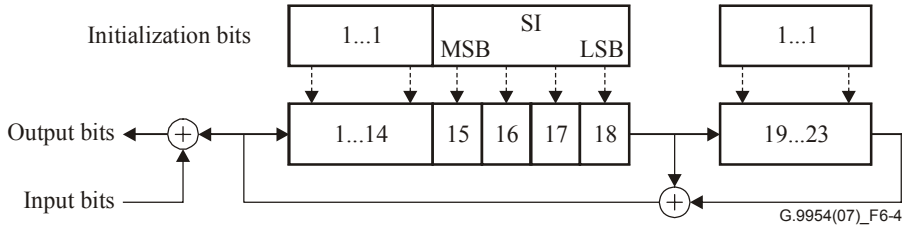

Figure 6-4 – Data scrambler

6.5 Constellation encoder

6.5.1 Constellation encoding control

All header bits up to and including the first two bytes following the SA field shall be encoded at 2 MBaud, 2 bits per symbol. The output symbols shall be modified as described in clause 6.5.6.

Starting with the 1st bit following the two bytes following the SA field, the bits shall be encoded according to the PE field (see Table 6-2) up to the last bit of the link-layer frame, or the last bit of PAD if it is present.

The EOF sequence shall be encoded at 2 MBaud, 2 bits per symbol. The output symbols shall be modified as described in clause 6.5.6.

6.5.2 Bit-to-symbol mapping

The incoming bits shall be grouped into N-bit symbols, where N is the number of bits per symbol specified in the PE field. The bit-to-symbol mapping is shown in Figures 6-5 through 6-14. The symbol values are shown with bits ordered such that the rightmost bit is the first bit received from the scrambler and the leftmost bit is the last bit received from the scrambler.

All constellations except for 3 bits per symbol lie on a uniform square grid, and all constellations are symmetric about the real and imaginary axes.

For the round constellations, only the 1st quadrant is shown and the 2 leftmost bits are omitted from the figures. For these cases, the 2 leftmost bits are specified in Figure 6-5.

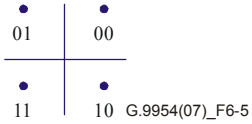

Figure 6-5 – 2 bits per symbol

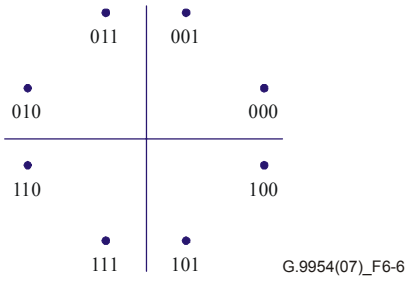

Figure 6-6 – 3 bits per symbol

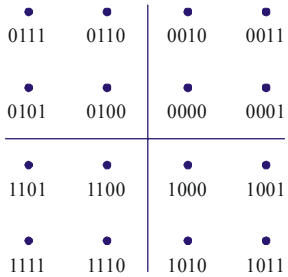

Figure 6-7 – 4 bits per symbol

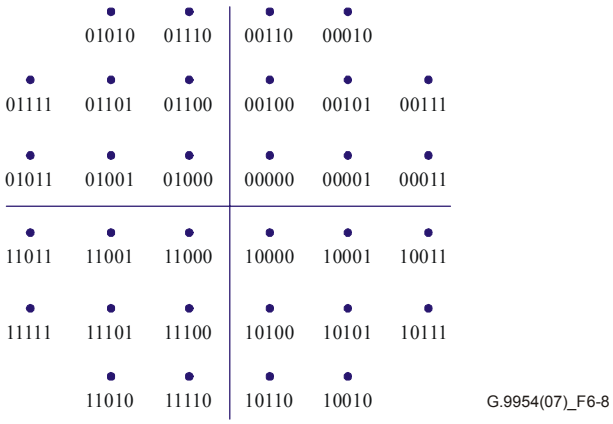

Figure 6-8 – 5 bits per symbol

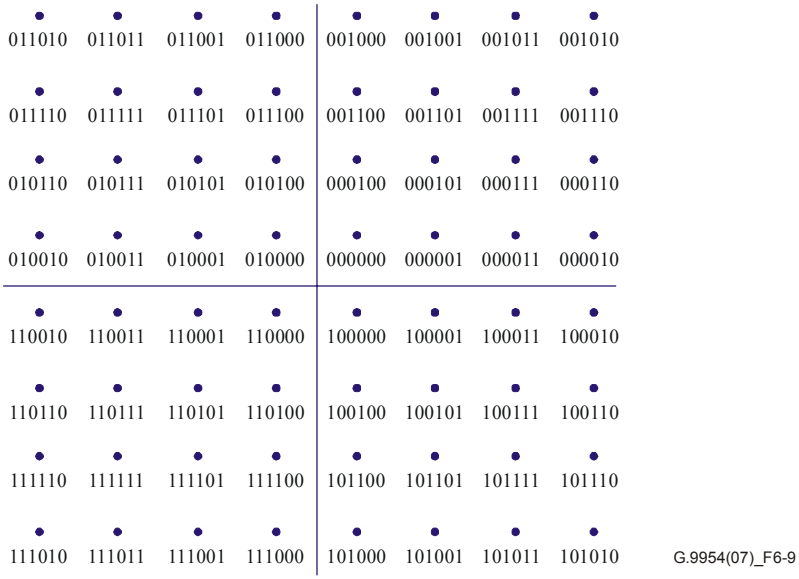

Figure 6-9 – 6 bits per symbol

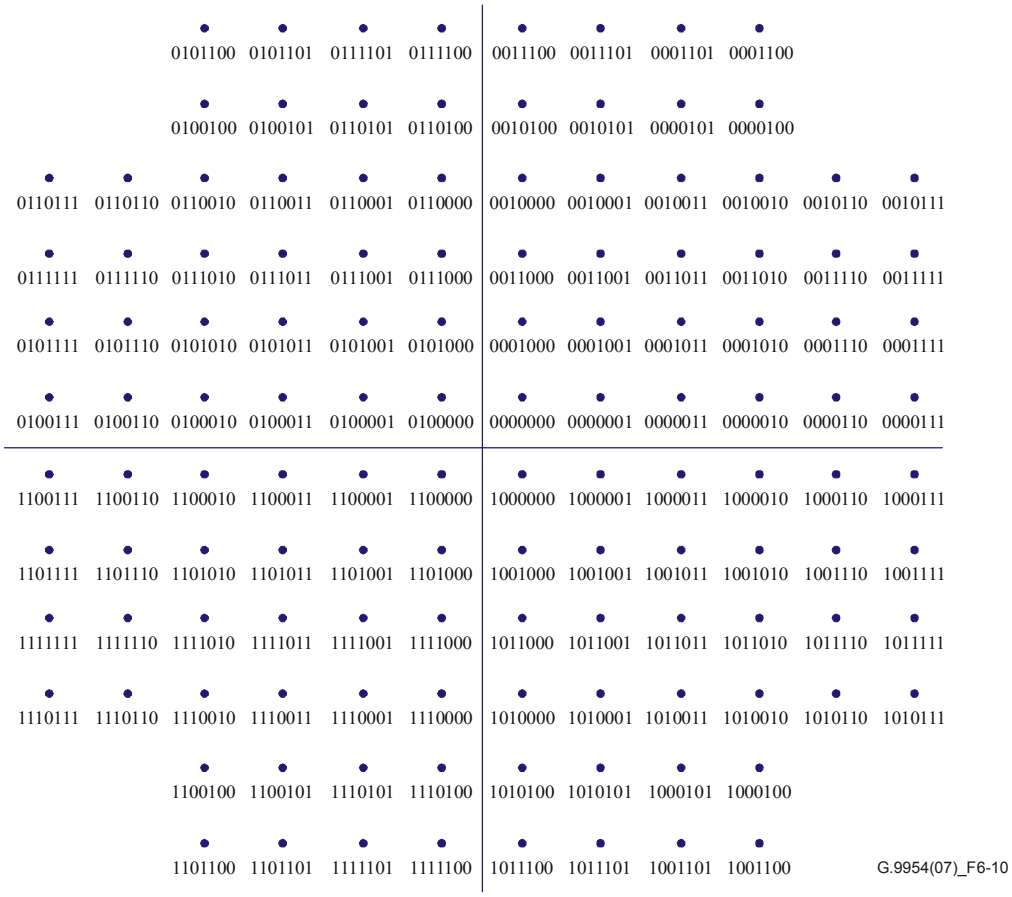

Figure 6-10 – 7 bits per symbol

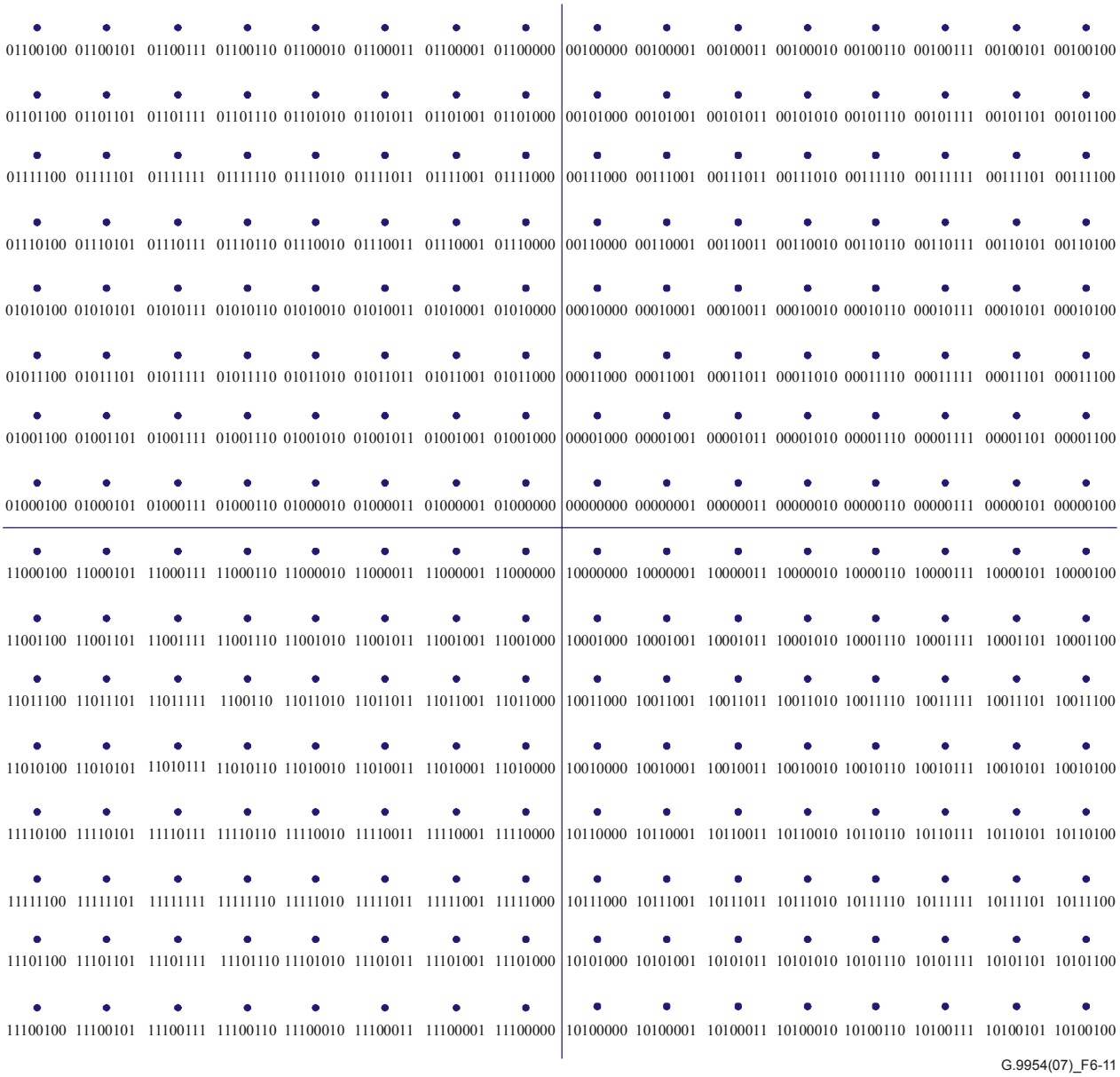

Figure 6-11 – 8 bits per symbol

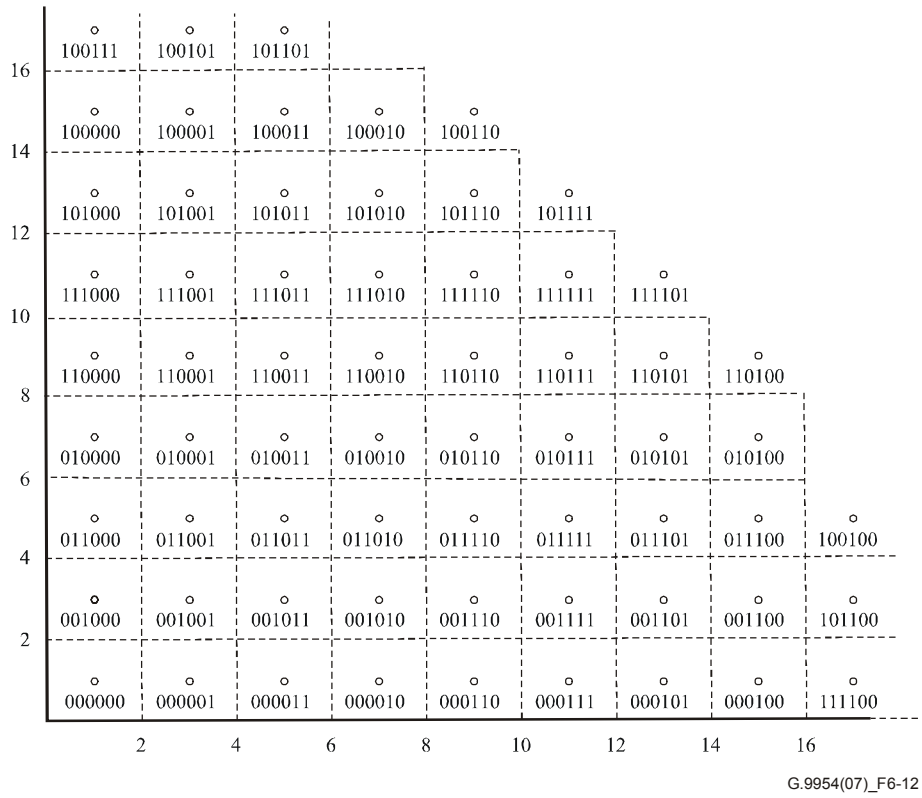

Figure 6-12 – 8 bits per symbol round constellation

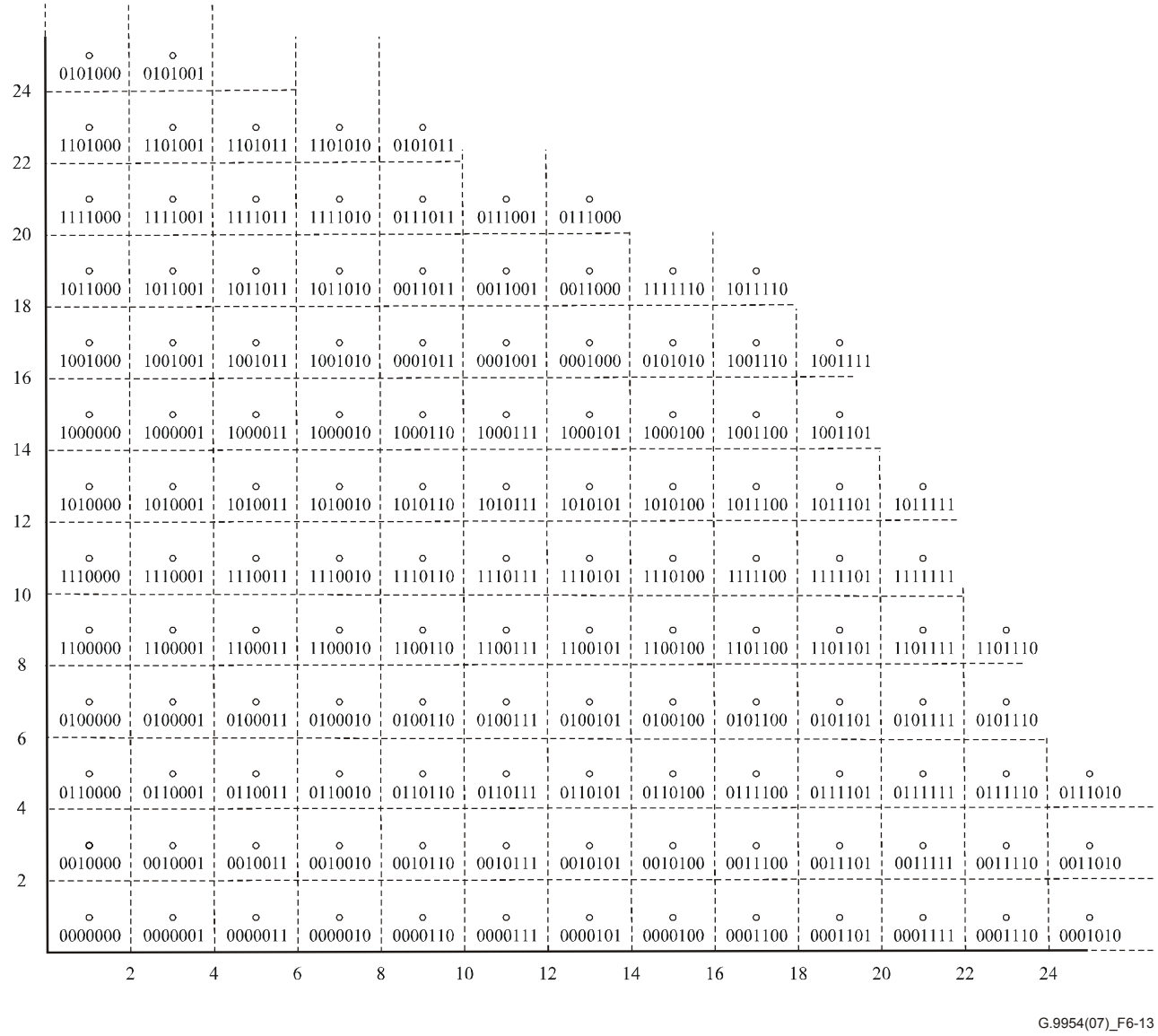

Figure 6-13 – 9 bits per symbol round constellation

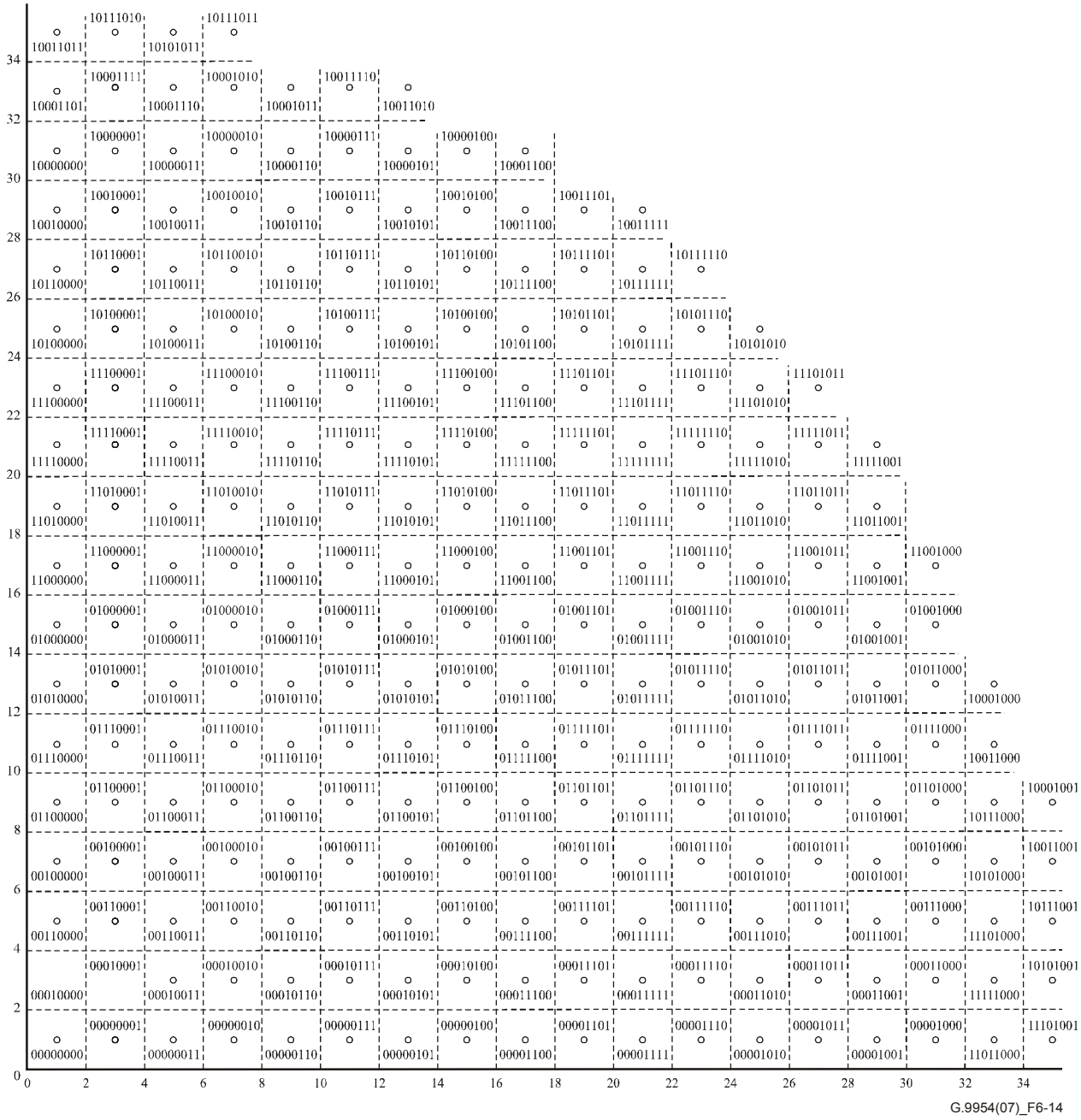

Figure 6-14 – 10 bits per symbol round constellation

6.5.3 Constellation scaling

The relative scaling of different constellations at a single baud is given by Tables 6-5 and 6-6, where the value of s(PE) in Table 6-5 is given in Table 6-6. The value of each constellation point must be accurate to within plus or minus 4 percent of the distance between nearest neighbours in that constellation.

NOTE – For example, at 2 MBaud, 2 bits per symbol, the tolerance on each point is ±0.08, while at 2 MBaud, 5 bits per symbol, the tolerance is ±0.02. Note that the tolerance is not implied by the number of significant digits in Table 6-6, i.e., the values in Table 6-6 should be considered exact.

Table 6-5 – Constellation reference points

Bits per symbol	Reference point(s)	Value
2	00	(1 + i)*s(PE)
3	000	(12 + 5i)*s(PE)
	001	(5 + 12i)*s(PE)
4	0000	(1 + i)*s(PE)
5	00000	(1 + i)*s(PE)
6	000000	(1 + i)*s(PE)
7	0000000	(1 + i)*s(PE)
8	00000000	(1 + i)*s(PE)
8-round	00000000	(1 + i)*s(PE)
9-round	000000000	(1 + i)*s(PE)
10-round	0000000000	(1 + i)*s(PE)

Table 6-6 – Constellation scale factors s(PE)

Symbol rate [MHz]	2 BPS	3 BPS	4 BPS	5 BPS	6 BPS	7 BPS	8 BPS	8 BPS round	9 BPS round	10 BPS round
2	1.0000	0.1111	0.3333	0.2500	0.1429	0.1111	0.0667	0.0800	0.0556	0.0400
4	0.7071	0.0786	0.2509	0.1812	0.1113	0.0835	0.0534	0.0617	0.0431	0.0306
8	0.5000	0.0556	0.1952	0.1396	0.0897	0.0664	0.0438	0.0470	0.0332	0.0235
16	0.3119	0.0335	0.1225	0.0860	0.0583	0.0418	0.0288	0.0296	0.0210	0.0148

The constellations are scaled based on a statistical measure of the peak-to-average ratio (PAR).

6.5.4 Symbol timing during baud transitions

On a transition from 2 MBaud to a higher baud, the first higher rate symbol shall occur 0.5 µs after the last 2-MBaud symbol.

On a transition from a higher baud to 2 MBaud, the first 2-MBaud symbol shall occur 0.5 µs after the last higher baud symbol.

For example, the transitions from 2 to 4 MBaud and from 4 to 2 MBaud are illustrated in Figure 6-15.

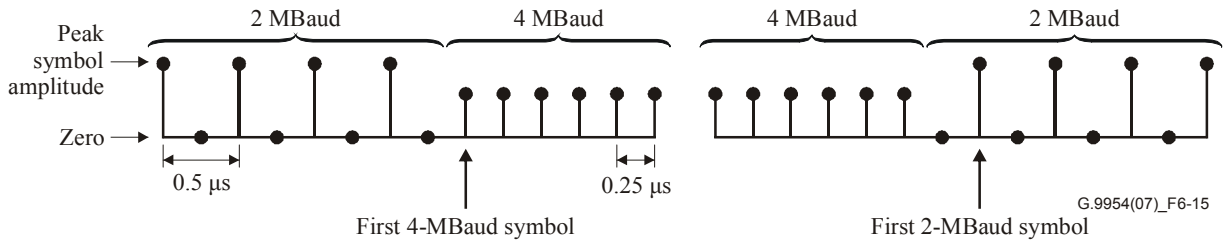

Figure 6-15 – Baud transitions

6.5.5 Encoding rate transitions

If the number of bits in a sequence is not an integer multiple of the number of bits per symbol, then enough zero bits shall be inserted at the end of the bit stream to complete the last symbol. The number of zero bits inserted shall be the minimum number such that the length of the appended bit stream is an integer multiple of the number of bits per symbol.

6.5.6 Modified header and trailer

The constellation encoder shall negate every other symbol of the header and trailer starting with the second symbol. That is, symbols 2,4,6...136 of the header and symbols 2 and 4 of the EOF shall be multiplied by –1.

6.6 QAM modulator

The modulator implements quadrature amplitude modulation (QAM). Figure 6-16 shows an example implementation. The carrier frequencies and transmit filters for each spectral mask do not depend on baud.

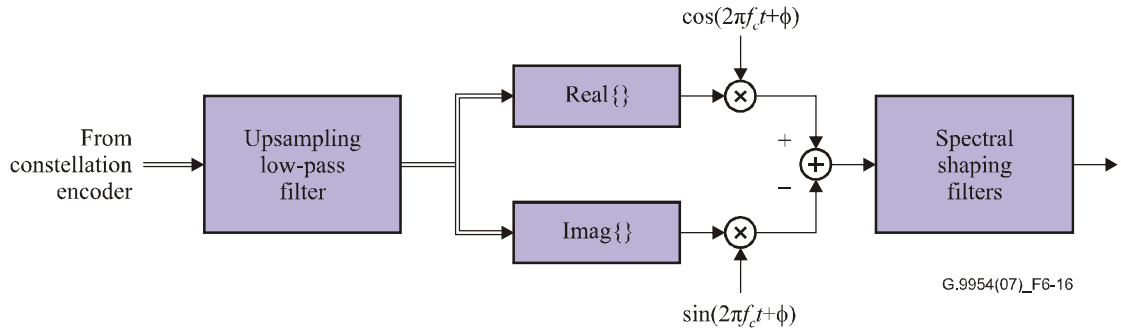

Figure 6-16 – QAM modulator

6.6.1 Carrier frequency and tolerance

Spectral modes A and B have the same carrier frequency: f_c = 12 MHz.

The carrier clock shall be locked to the symbol clock. So, the carrier frequency tolerance is derived from the clock tolerance defined in 6.9.2.

6.6.2 Transmit filters

The details of the transmit filters are implementation dependent. Clauses 6.8.3 and 6.8.4 constrain the transmit filter designs.

6.7 Minimum device requirements

A G.9954v2 station at a minimum shall be capable of transmitting and receiving over phoneline or over coax.

A station supporting phoneline at a minimum shall be capable of transmitting and receiving in one of the spectral modes, A or B, specified in clause 6.8.3.

Stations at a minimum shall be capable of transmitting and receiving 2-, 4-, 8- and 16-MBaud modulated frames.

Stations at a minimum shall be capable of transmitting all constellations from 2 bits per symbol to 8 bits per symbol and receiving all constellations from 2 bits per symbol to 6 bits per symbol.

6.8 Transmitter electrical specification

6.8.1 Transmit power

The transmit power during a 2-MBaud and 2-bits/symbol transmission shall be between –7 dBm and –9.5 dBm, measured across a 100-ohm load between tip and ring, integrated from 0 to 30 MHz.

6.8.2 Transmit voltage

The rms differential transmit voltage shall not exceed –15 dBV rms in any 2-µs window between 0 and 6 MHz, measured across a 135-ohm load between tip and ring for any payload encoding. The peak differential transmit voltage shall not exceed 580 mVpeak, measured across a 135-ohm load between tip and ring for any payload encoding.

Stations that are not transmitting shall emit less than –65 dBV rms measured across a 100-ohm load between tip and ring.

6.8.3 Spectral masks

Two spectral masks are defined for the two spectral modes. Stations shall use the spectral mask according to the spectral mode they transmit.

6.8.3.1 PSD upper bound

When transmitting in spectral mode A, the HNT metallic power spectral density (PSD) shall be constrained by the upper bound depicted in Figure 6-17 and in Table 6-7 with the measurement made across a 100-ohm load across tip and ring at the transmitter W1 interface. The upper bound shall apply to all symbol rates and constellations.

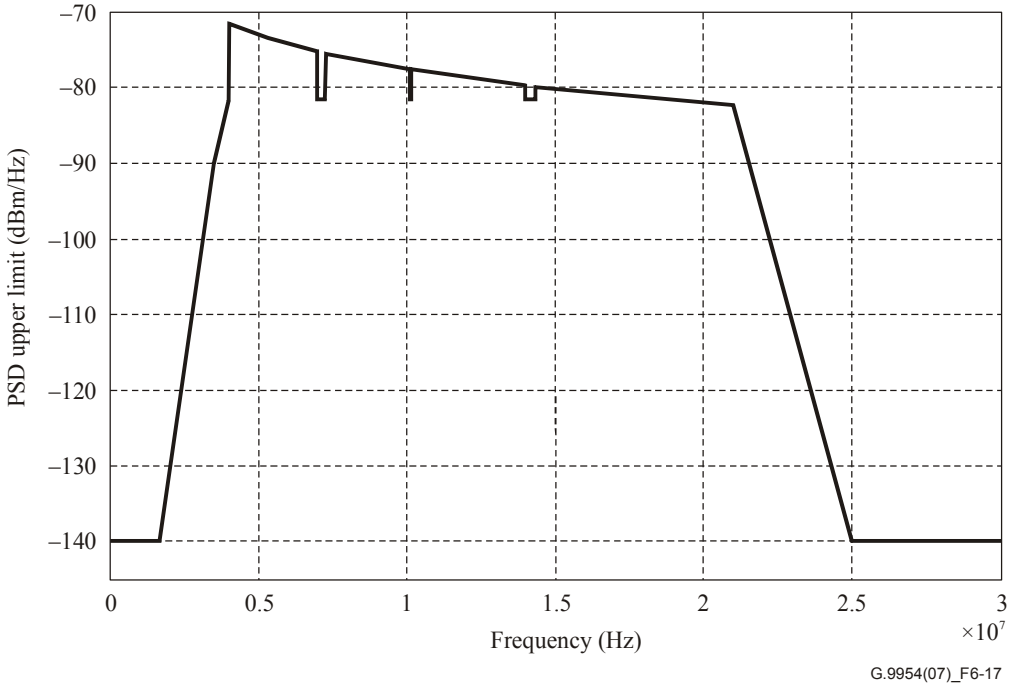

Figure 6-17 – Transmit PSD upper bound for spectral mode A

Table 6-7 – Transmit PSD upper bound for spectral mode A

Frequency [MHz]	PSD limit [dBm/Hz]
$0.015 < f \leq 1.7$	–140
$1.7 < f \leq 3.5$	$-140 + (f - 1.7) \times 50.0/1.8$
$3.5 < f \leq 4.0$	$-90 + (f - 3.5) \times 17.0$
$4.0 < f < 7.0$	$-71.5 - 15 \times \log_{10}(f/4)$
$7.0 \leq f \leq 7.3$	–81.5
$7.3 < f < 10.1$	$-71.5 - 15 \times \log_{10}(f/4)$
$10.1 \leq f \leq 10.15$	–81.5
$10.15 < f < 14.0$	$-71.5 - 15 \times \log_{10}(f/4)$
$14.0 \leq f \leq 14.35$	–81.5
$14.35 < f < 18.068$	$-71.5 - 15 \times \log_{10}(f/4)$
$18.068 \leq f \leq 18.168$	–81.5
$18.168 < f < 21.0$	$-71.5 - 15 \times \log_{10}(f/4)$
$21.0 \leq f < 25.0$	$-82.3 - (f - 21) \times 57.7/4.0$
$25.0 \leq f$	–140

When transmitting with spectral mode B, the HNT metallic power spectral density (PSD) shall be constrained by the upper bound depicted in Figure 6-18 and in Table 6-8 with the measurement made across a 100-ohm load across tip and ring at the transmitter W1 interface. The upper bound shall apply to all symbol rates and constellations.

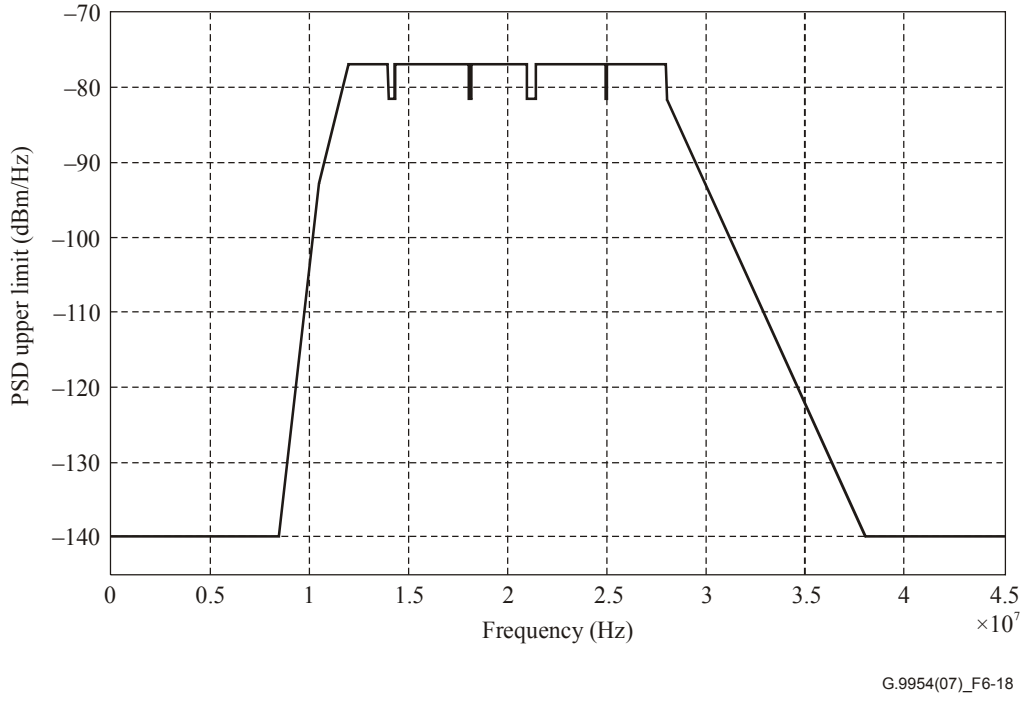

Figure 6-18 – Transmit PSD upper bound for spectral mode B

Table 6-8 – Transmit PSD upper bound for spectral mode B

Frequency [MHz]	PSD limit [dBm/Hz]
$0.015 < f \leq 8.5$	–140
$8.5 < f \leq 10.5$	$-140 + (f - 8.5) \times 47/2$
$10.5 < f \leq 12$	$-93 + (f - 10.5) \times 16/1.5$
$12 < f < 14$	–77
$14 \leq f \leq 14.35$	–81.5
$14.35 < f < 18.068$	–77
$18.068 \leq f \leq 18.168$	–81.5
$18.168 < f < 21$	–77
$21 \leq f \leq 21.45$	–81.5
$21.45 < f < 24.9$	–77
$24.9 \leq f \leq 25$	–81.5
$25 < f < 28$	–77
$28 \leq f < 38$	$-81.5 - (f - 28) \times 58.5/10$
$38 \leq f$	–140

When transmitting in spectral mode A, the resolution bandwidth used to make this measurement shall be 10 kHz for frequencies between 2.0 and 30.0 MHz, and 3 kHz for frequencies between 0.015 and 2.0 MHz. An averaging window of 213 seconds shall be used, and 1500-octet MTUs separated by an IFG duration of silence shall be assumed. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line under 2.0 MHz, with no sub-band greater than 20 dB above the limit line. A total of 100 kHz of possibly non-contiguous bands may exceed the limit line between 25.0 and 30.0 MHz, with no sub-band greater than 20 dB above the limit line.

When transmitting in spectral mode B, the resolution bandwidth used to make this measurement shall be 10 kHz for frequencies between 2.5 and 60.0 MHz, and 3 kHz for frequencies between 0.015 and 2.5 MHz. An averaging window of 213 seconds shall be used, and 1500-octet MTUs separated by an IFG duration of silence shall be assumed. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line under 8.5 MHz, with no sub-band greater than 20 dB above the limit line. A total of 100 kHz of possibly non-contiguous bands may exceed the limit line between 38.0 and 60.0 MHz, with no sub-band greater than 20 dB above the limit line. Note that the notches at 4.0, 7.0, 10.1, 14.0, 18.068, 21.0, and 24.9 MHz are designed to reduce RFI egress in the radio amateur bands.

NOTE – The masks should be tested at PE values of 2 MBaud and 2 bits/symbol, as these payload encodings result in the maximum transmitted power.

6.8.3.2 Passband ripple

For spectral mode A, the ripple relative to the PSD upper bound shall be less than 4 dB in the union of the following ranges: [5.00,6.25], [8.00,9.35], [10.90,13.50], [14.85,17.57], [18.67,20.25].

For spectral mode B, the ripple relative to the PSD upper bound shall be less than 4 dB in the union of the following ranges: [13.00,13.25], [15.1,17.25], [18.9,20.25], [22.1,24.15], [25.75,27.25].

The ripple relative to the PSD upper bound is the ripple of the gap between actual PSD and its corresponding upper bound.

6.8.4 Transmitter symbol response

The symbol response of the transmitter output shall be upper-bounded by the temporal mask shown in Figure 6-19. The response shall be measured across a 100-ohm load between tip and ring at the transmitter's W1 interface.

Output before $t = 0$ and after $t = 5.0$ µs shall be < 0.032% of the peak amplitude.

In Figure 6-19, the time $t = 0$ is arbitrary.

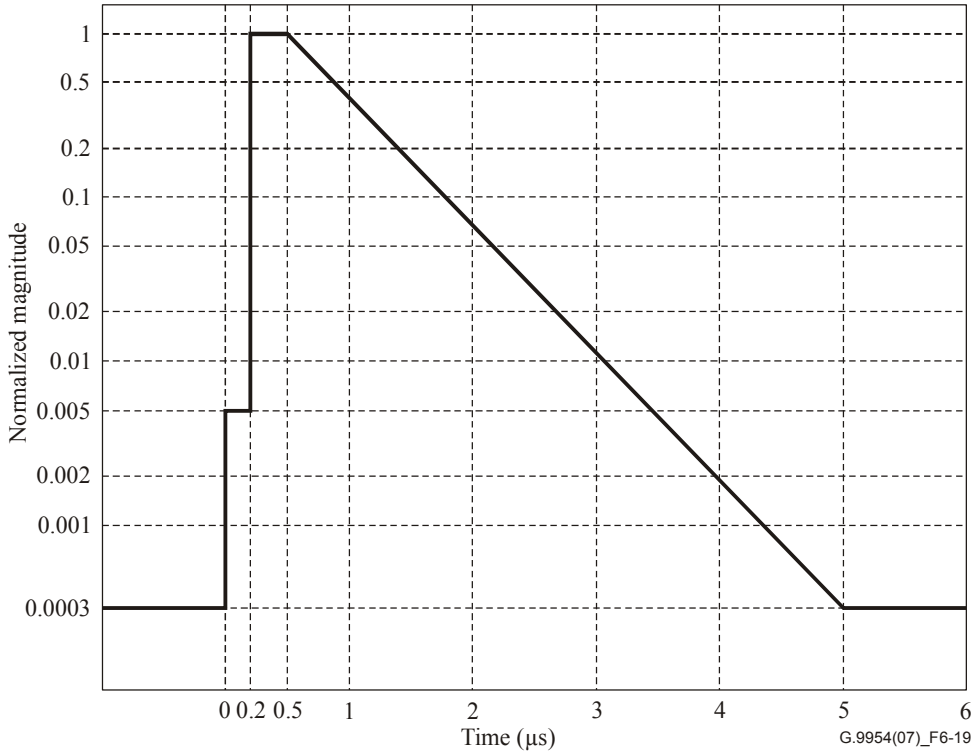

Figure 6-19 – Transmitter symbol response magnitude mask for spectral modes A and B

6.8.5 Spurious voice band output

The transmitter C-weighted output in the band extending from 200 Hz to 3000 Hz shall never exceed 10 dBrnC when terminated with a 600-ohm resistive load.

6.8.6 Common mode emissions

6.8.6.1 Common-mode output voltage

The transmitter shall emit no more than –55 dBV rms across a 50-ohm load between the centre tap of a balun with CMRR > 60 dB and the transceiver ground in the band extending from 0.1 MHz to 50 MHz.

6.8.7 Clock tolerance

In spectral mode A, the transmitter clock frequency shall be accurate to within ±100 ppm over all operating temperatures for the device. In spectral mode B, the transmitter clock frequency shall be accurate to within ±30 ppm over all operating temperatures for the device. The minimum operating temperature range for this requirement shall be 0 to 70°C.

In general, a ±50 ppm crystal will be required to meet this requirement for mode A and a ±20 ppm crystal will be required for mode B.

6.8.8 Clock jitter

The rms jitter of the transmitter clock shall be less than 70 ps, averaged over a sliding 10-µs window.

6.8.9 I/Q balance

There shall be no gain or phase imbalance in the transmitter, except as noted in 6.5.3.

6.9 Receiver electrical specification

6.9.1 Receiver sensitivity

6.9.1.1 Maximum signal

The receiver shall detect frames with peak voltage up to –6 dBV across tip and ring at a frame error rate of no greater than 10^{-3} with additive white Gaussian noise at a PSD of less than –140 dBm/Hz, measured at the receiver.

6.9.1.2 Minimum sensitivity

The receiver shall detect 1518-octet frames encoded as 2 bits/symbol and 2 Mbaud with rms voltage as low as 2.5 mV at no greater than 10^{-3} frame error rate. The rms voltage is computed only over time during which the transmitter is active.

The receiver shall detect no more than 1 in 10^4 1518-octet, 2 bits/symbol, 2 Msymbol/s frames with rms voltage less than 1.0 mV.

Both criteria assume additive white Gaussian noise at a PSD of less than –140 dBm/Hz, measured at the receiver, and assume a flat channel.

6.9.2 Clock tolerance

The receiver shall meet the requirements of clauses 6.9.4.1 and 6.9.4.2 on loop 1 when the transmitter clock frequency is within ±100 ppm and ±30 ppm of its nominal value for spectral mode A and spectral mode B, respectively.

6.9.3 Immunity to narrow-band interference

6.9.3.1 Differential input

The receiver shall demodulate frames with payload encoded at 4 Mbaud, 3 bits/symbol, and differential rms voltage as low as 20 mV (measured over the header) at a frame error rate less than 10^{-4} under the following conditions:

1) White Gaussian noise with PSD less than –130 dBm/Hz shall be added at the receiver.
2) A single tone interferer with any of the following frequency band and input voltage combinations, according to the spectral mode:

Table 6-9 – Interferer amplitudes

Frequency range [MHz]	Maximum peak-to-peak interferer level [volts] Spectral mode A	Maximum peak-to-peak interferer level [volts] Spectral mode B
0.01 to 0.1	6.0	6.0
0.1 to 0.6	3.3	3.3
0.6 to 1.7	1.0	1.0
1.7 to 4.0	0.1	1.0

Table 6-9 – Interferer amplitudes

Frequency range [MHz]	Maximum peak-to-peak interferer level [volts] Spectral mode A	Maximum peak-to-peak interferer level [volts] Spectral mode B
7.0 to 7.3	0.1	1.0
10.0 to 10.15	0.1	0.1
14.0 to 14.35	0.1	0.1
18.068 to 18.168	0.1	0.1
21.0 to 21.45	0.1	0.1
24.89 to 24.99	0.1	0.1
28.0 to 29.7	0.1	0.1

The applied voltage shall be measured across tip and ring at the input to the transceiver.

6.9.3.2 Common-mode input

The receiver shall demodulate frames with payload encoded at 4 MBaud, 3 bits/symbol, and differential rms voltage as low as 20 mV (measured over the header) at a frame error rate less than 10^{-4} under the following conditions:

1) White Gaussian noise with PSD less than –130 dBm/Hz shall be added at the receiver, differential mode.

2) A single-tone interferer, measured between the centre tap of a test transformer and ground at the input to the transceiver, with any of the frequency band and input voltage combinations in Table 6-10 according to the spectral mode:

Table 6-10 – Common-mode input requirements

Frequency range [MHz]	Maximum peak-to-peak interferer level [volts] Spectral mode A	Maximum peak-to-peak interferer level [volts] Spectral mode B
0.01 to 0.1	20.0	20.0
0.1 to 0.6	20.0	20.0
0.6 to 1.7	10.0	10.0
1.7 to 4.0	2.5	10.0
7.0 to 7.3	2.5	10.0
10.0 to 10.15	2.5	2.5
14.0 to 14.35	2.5	2.5
18.068 to 18.168	2.5	2.5
21.0 to 21.45	2.5	2.5
24.89 to 24.99	2.5	2.5
28.0 to 29.7	2.5	2.5

The common mode rejection of the test transformer used to insert the signal should exceed 60 dB to 100 MHz.

6.9.4 System margin requirements

Test loops, provided in clause B.2 shall be used to verify the minimum receiver requirements. The following impairments shall be applied in each loop test: additional (flat) attenuation, additive white Gaussian noise, narrow-band interferers, and 120-Hz impulse noise ("light dimmer noise").

The impairment level (defined in each subclause) must exceed the specified level at each specified payload encoding at the frame error rate (FER) point: 10^{-2}. A system margin requirement for a single time-varying channel is also defined.

Any entry of "–" in a table implies that there is no requirement under the specified conditions.

6.9.4.1 Attenuation requirements

The attenuator setting described in Table 6-11 is the additional attenuation applied in series with the specified wire loop.

Table 6-11 – Attenuation requirements

Payload encoding	FER	Required impairment attenuator setting [dB] Loop number					
		1	**4**	**5**	**6**	**8**	**9**
Mode A 4 Mbaud 3 BPS	10^{-2}	30	12	17	7	10	16
Mode A 16 Mbaud 3 BPS	10^{-2}	28	12	13	–	8	–
Mode B 4 Mbaud 3 BPS	10^{-2}	29	15	15	5	12	12
Mode B 16 Mbaud 3 BPS	10^{-2}	27	15	11	–	10	–

6.9.4.2 Additive white noise requirements

White noise power spectral density at 0 dB attenuator setting: –70 dBm/Hz. The output of the noise attenuator shall be added at the receiver. For loop 1, 20 dB of flat-channel attenuation shall be placed in series with the loop.

Table 6-12 – Additive white noise requirements

Payload encoding	FER	Required impairment attenuator setting [dB] Loop number					
		1	**4**	**5**	**6**	**8**	**9**
Mode A 4 Mbaud 3 BPS	10^{-2}	48	42	42	52	45	52
Mode A 16 Mbaud 3 BPS	10^{-2}	57	51	52	65	56	–
Mode B 4 Mbaud 3 BPS	10^{-2}	48	41	45	53	46	43
Mode B 16 Mbaud 3 BPS	10^{-2}	58	56	55	68	57	–

6.9.4.3 Narrow-band interference requirements

Narrow-band interference peak-to-peak amplitude at 0 dB attenuator setting: 2.0 volts at 7.0, 7.3, 10.1, 14.0, 14.35, 18.1, 21.0, 24.9 MHz. White Gaussian noise is simultaneously applied at a level of –135 dBm/Hz.

Table 6-13 – Narrow-band interference requirements

Payload encoding	FER	Required impairment attenuator setting [dB] Loop number					
		1	**4**	**5**	**6**	**8**	**9**
Mode A 4 Mbaud 3 BPS	10^{-2}	26	26	26	26	26	28
Mode A 16 Mbaud 3 BPS	10^{-2}	26	26	26	43	31	–
Mode B 4 Mbaud 3 BPS	10^{-2}	25	23	25	31	22	25
Mode B 16 Mbaud 3 BPS	10^{-2}	26	32	32	–	35	–

6.9.4.4 Impulse noise requirements

Impulse noise peak-to-peak amplitude at 0 dB attenuator setting: 3.0 volts. White Gaussian noise is simultaneously applied at a level of –135 dBm/Hz. The impulse shall be defined as two cycles of a 5.0-MHz square wave summed with four cycles of a 7.0-MHz square wave.

Table 6-14 – Impulse noise requirements

Payload encoding	FER	Required impairment attenuator setting [dB] Loop number	
		2	9
Mode A 4 Mbaud 3 BPS	10^{-2}	3	3
Mode A 16 Mbaud 3 BPS	10^{-2}	3	–
Mode B 4 Mbaud 3 BPS	10^{-2}	3	3
Mode B 16 Mbaud 3 BPS	10^{-2}	3	–

6.9.4.5 Dynamic channel system margin requirement

The receiver shall detect no more than five 1518-octet frames in error out of 3000 when sent at a rate of five frames per 10 ms over loop #2 under the following conditions:

1) During this test, the 330-pF capacitor terminating one of the stubs shall be switched in and out of the loop once per second, i.e., an open-circuit termination shall be used for a period of 1 second every 2 seconds.
2) White noise at a level of –140 dBm/Hz shall be added at the receiver.
3) The PE shall be 16 Mbaud, 3 bits/symbol in spectral modes A and B.

Switching a capacitor in and out of the loop simulates a switch-hook transition on a common telephone.

6.9.4.6 Telephony ringing signal performance

The HNT device shall be able to accommodate a telephony ringing signal event from a telephone central office. The signal shall consist of a 20-Hz sinusoid with a level of 90 Vrms superimposed on a DC bias level of –52 V (min.). The device shall be immune from a telephony ringing signal that is continuously repeated with an on-time of 2 seconds and an off-time of 4 seconds through the circuit defined in Figure 6-20.

The signal is injected into the circuit through two 500-ohm resistors as shown in Figure 6-20. Since most attenuators have low impedance at DC and could significantly reduce the ringing voltage, two 0.01-µF capacitors are required to provide DC isolation.

When subjected to the telephony ringing signal as defined above, the device frame error rate shall not exceed 0.1% when measured over 100'000 maximum-MTU UDP frames at 4 Mbaud, 3 bits/symbol in spectral modes A and B.

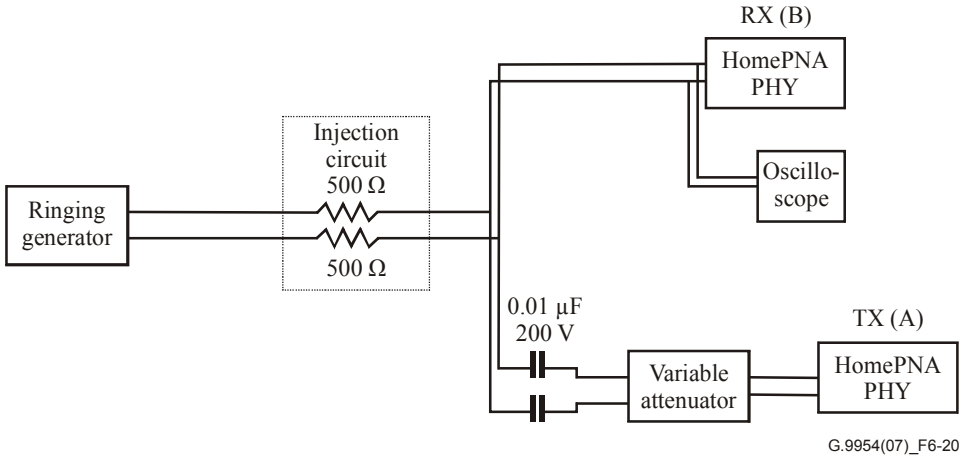

Figure 6-20 – Telephony ringing signal conditions

6.10 Input impedance

6.10.1 Passband return loss

Stations shall conform to the impedance mask corresponding to the spectral mode they transmit.

For stations capable of transmitting spectral mode A, the average return loss of the transceiver with respect to a 100-ohm resistive load shall exceed 12 dB between 4.75 and 20.25 MHz. This requirement applies to the transceiver powered on or in low-power mode (transmitter powered off). The average return loss with respect to a 100-ohm resistive load shall exceed 6 dB between 4.75 and 20.25 MHz with the transceiver removed from a source of power.

For stations capable of transmitting spectral mode B, the average return loss of the transceiver with respect to a 100-ohm resistive load shall exceed 12 dB between 12.75 and 27.25 MHz. This requirement applies to the transceiver powered on or in low-power mode (transmitter powered off). The average return loss with respect to a 100-ohm resistive load shall exceed 6 dB between 12.75 and 27.25 MHz with the transceiver removed from a source of power.

6.10.2 Stopband input impedance

Stations shall conform to the impedance mask corresponding to the spectral mode they transmit.

Stations transmitting in spectral mode A shall have input impedance magnitude greater than 10 ohms from 0-30 MHz and shall conform to the lower-bound mask in Table 6-15.

Table 6-15 – Input impedance lower-bound mask for spectral mode A

Frequency range [kHz]	Min. impedance [ohms]
$0 < f \leq 0.285$	1 M
$0.285 < f \leq 2.85$	100 k
$2.85 < f \leq 28.5$	10 k
$28.5 < f \leq 95$	4.0 k
$95 < f \leq 190$	2.0 k
$190 < f \leq 285$	1.4 k
$285 < f \leq 380$	1.0 k
$380 < f \leq 475$	850

Table 6-15 – Input impedance lower-bound mask for spectral mode A

Frequency range [kHz]	Min. impedance [ohms]
$475 < f \leq 570$	700
$570 < f \leq 665$	600
$665 < f \leq 760$	525
$760 < f \leq 855$	450
$855 < f \leq 950$	400
$950 < f \leq 1000$	350
$1000 < f \leq 1400$	175
$1400 < f \leq 2300$	100
$2300 < f \leq 2850$	50
$2850 < f \leq 3085$	25
$3085 < f \leq 4000$	10
$4000 < f \leq 4750$	30
$20250 < f \leq 21000$	30
$21000 < f \leq 25000$	25
$25000 < f \leq 30000$	50

Stations transmitting in spectral mode B shall have input impedance magnitude greater than 25 ohms from 0 to 30 MHz and shall conform to the lower-bound mask in Table 6-16.

Table 6-16 – Input impedance lower-bound mask for spectral mode B

Frequency range [kHz]	Min. impedance [ohms]
$0 < f \leq 1.5$	1'000'000
$1.5 < f \leq 3$	500'000
$3 < f \leq 9$	200'000
$9 < f \leq 17$	100'000
$17 < f \leq 30$	50'000
$30 < f \leq 70$	24'000
$70 < f \leq 100$	15'000
$100 < f \leq 200$	8'000
$200 < f \leq 400$	4'000
$400 < f \leq 600$	3'000
$600 < f \leq 800$	2'000
$800 < f \leq 1000$	1'600
$1000 < f \leq 1200$	1'400
$1200 < f \leq 1500$	1'150
$1500 < f \leq 1700$	1'000
$1700 < f \leq 2000$	850

Table 6-16 – Input impedance lower-bound mask for spectral mode B

Frequency range [kHz]	Min. impedance [ohms]
$2000 < f \leq 2500$	700
$2500 < f \leq 3000$	580
$3000 < f \leq 3500$	480
$3500 < f \leq 4000$	380
$4000 < f \leq 4500$	300
$4500 < f \leq 5500$	250
$5500 < f \leq 6500$	200

This requirement applies to the transceiver powered on, in low-power mode (transmitter powered off), or removed from a source of power.

NOTE – Implementers should be aware that the G.992.5 ("ADSL2plus") frequency band extends up to 2.2 MHz; therefore, to ensure compatibility, care should be taken to maintain an input impedance well above the specified minimum.

7 PHY layer specification over Coax

7.1 Overview

The G.9954v2 over coax PHY layer supports four spectral modes. Each spectral mode supports a different band range along with its corresponding payload baud rates set:

– Spectral mode A: 4-20 MHz; 2, 4, 8, 16 Mbaud;

– Spectral mode B: 12-28 MHz; 2, 4, 8, 16 Mbaud;

– Spectral mode C: 36-52 MHz; 2, 4, 8, 16 Mbaud;

– Spectral mode D: 4-36 MHz; 2, 4, 8, 16, 32 Mbaud.

The actual spectral mode to be used in the network is pre-configured according to considerations of co-existence with other services and line characteristics. The pre-configuration technique is implementation dependent and is out of the scope of this Recommendation. The network configuration is limited to either a homogenous spectral mode for all devices or a dual spectral mode D+A or D+B.

Constellation sizes range from 2 to 10 bits per baud, specifying PHY layer payload modulation rates that range from 4 Mbit/s to 320 Mbit/s.

Information is transmitted on the channel in bursts. Each burst or physical layer frame consists of PHY-layer payload information encapsulated with PHY preamble, header and postamble. The PHY-layer payload refers to the portion of the link level frame that is modulated at the payload rate, which is typically higher than the header rate. Hereafter, "payload" refers to the PHY-layer payload unless otherwise specified.

The following describes the physical layer formatting.

7.2 Transmitter reference model

The transmitter block diagram is shown in Figure 7-1. This consists of a frame processor, data scrambler, bit-to-symbol mapper, and QAM modulator, as defined in the following clauses.

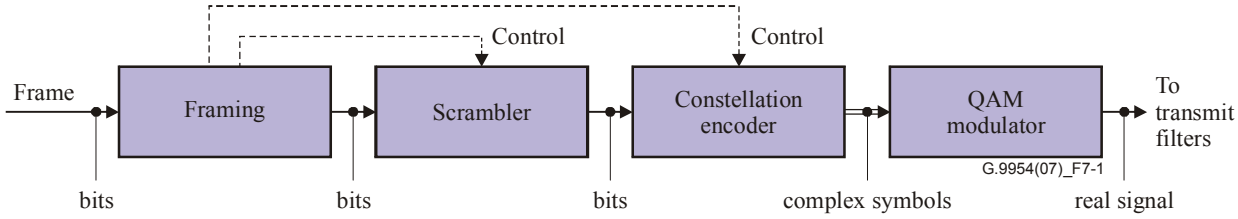

Figure 7-1 – Transmitter block diagram

7.3 Framing

The frame format is shown in Figure 7-2. This consists of a low-rate header section, a variable-rate payload section, and a low-rate trailer. Some parts of the frame are not scrambled, as described in clause 7.4.

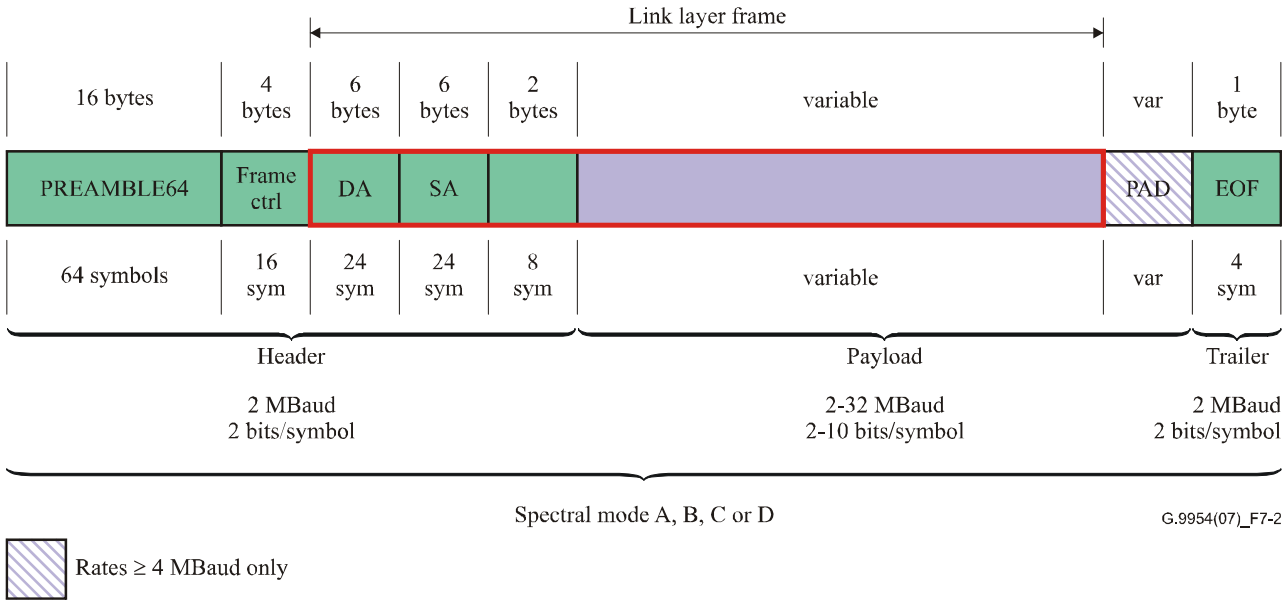

Figure 7-2 – PHY frame format

The interpretation of the two-byte field following the SA and the variable length field following it is given by the link layer frame format defined in clause 11.

7.3.1 Bit order

Except where otherwise stated, all fields are encoded most significant octet first, least significant bit first within each octet. Bit number 0 is the LSB within a field. Diagrams show MSB bits or octets to the left.

7.3.2 Preamble definition

The PREAMBLE64 is defined as a repetition of four 16-symbol sequences (TRN16) that result from encoding **0xfc483084** (in the order defined in clause 7.3.1) at 2 MBaud, 2 bits per symbol, with the scrambler disabled.

NOTE – The TRN16 is a white, constant amplitude QPSK sequence. The preamble was designed to facilitate:

- power estimation and gain control;
- baud offset estimation;
- equalizer training;
- carrier sense.

7.3.3 Frame control definition

The frame control field is a 32-bit field defined in Table 7-1.

Table 7-1 – Frame control fields

Field	Bit number	Bits	Description
FT	31:28	4	Frame type
EID	27:25	3	Extended identifier
RSVD	24:24	1	Reserved. This field shall be set to zero by the transmitter and the receiver shall discard frames with non-zero values.
ID	23:20	4	Identifier
SI	19:16	4	Scrambler initialization
PE	15:8	8	Payload encoding
HCS	7:0	8	Header check sequence

Hence, with the bit-ordering defined in clause 7.3.1, the frame control fields are transmitted in the order shown in Figure 7-3.

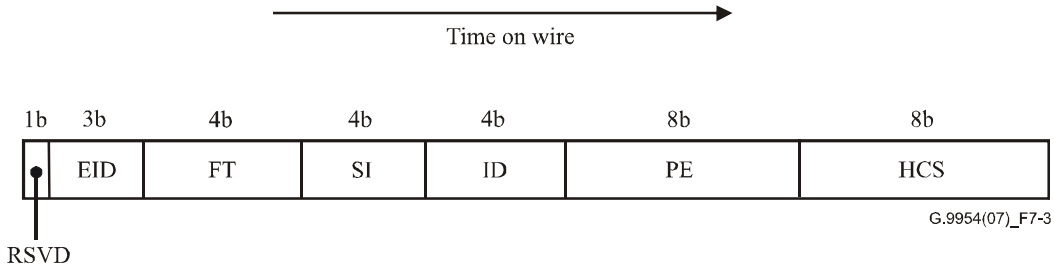

Figure 7-3 – Frame control field order

7.3.3.1 Frame Type

The Frame Type (FT) is a four-bit field. The details of this field are defined in clause 8.13.1.

7.3.3.2 Reserved bits (RSVD)

This field shall be set to zero by the transmitter, and the receiver shall discard any frame where this field is non-zero.

7.3.3.3 Extended Identifier (EID)

This Extended Identifier is a three-bit field. The details of this field are defined in clause 8.13.2.

7.3.3.4 Scrambler Initialization Bits

This 4-bit field shall be set to the value used to initialize the scrambler, as described in clause 7.4.

7.3.3.5 Identifier

The Identifier is a four-bit field. The details of this field are defined in clause 8.13.3.

7.3.3.6 Payload encoding

This field determines the spectral mask, baud and the constellation encoding of the payload bits. This field is defined by the sub-fields in Table 7-2.

Table 7-2 – Payload encoding fields

Field	Bit number	Bits	Description
EBPS	7	1	Extended bits per symbol
Baud	6:3	4	Symbol rate
BPS	2:0	3	Bits per symbol

7.3.3.6.1 Extended Bits per Symbol Bit

The EBPS is used to indicate an extended encoding of the BPS field. More specifically, it is used to extend the interpretation of the BPS field when EBPS = 1. This is described in detail in clause 7.3.3.6.3.

7.3.3.6.2 Symbol rate

This field indicates the symbol rate/baud rate of the payload bits in Table 7-3.

Table 7-3 – Symbol rates

Baud value	Interpretation
0-3	Reserved on transmit, discard frame on receive
4	Symbol rate = 2 MHz
5	Symbol rate = 4 MHz
6	Symbol rate = 8 MHz
7	Symbol rate = 16 MHz
8	Symbol rate = 32 MHz
9-15	Reserved on transmit, discard frame on receive

7.3.3.6.3 Bits per symbol

The values are defined in Table 7-4.

Table 7-4 – Bits per symbol encoding

EBPS value	BPS value	Interpretation
0	0	Reserved on transmit, discard frame on receive
0	1	2 bits per symbol
0	2	3 bits per symbol
0	3	4 bits per symbol
0	4	5 bits per symbol
0	5	6 bits per symbol
0	6	7 bits per symbol
0	7	8 bits per symbol
1	0	8-round constellation; 8 bits per symbol
1	1	9-round constellation; 9 bits per symbol
1	2	10-round constellation; 10 bits per symbol
1	3-7	Reserved on transmit, discard frame on receive

7.3.3.7 Header check sequence (HCS)

An 8-bit cyclic redundancy check (CRC) is computed as a function of the 128-bit sequence in transmission order starting with the FT bits and ending with the Ethernet source address (SA) bits, with zeros substituted for the as-of-yet uncomputed HCS field. The encoding is defined by the following generating polynomial.

$$G(x) = x^8 + x^7 + x^6 + x^4 + x^2 + 1$$

Mathematically, the CRC value corresponding to a given frame is defined by the following procedure.

The first 8 bits of the input bit sequence in transmission order are complemented.

The 128 bits of the sequence in transmission order are then considered to be the coefficients of a polynomial $M(x)$ of degree 127. (The first bit of the FT field corresponds to the x^{127} term and the last bit of the SA field corresponds to the x^0 term.)

$M(x)$ is multiplied by x^8 and divided by $G(x)$, producing a remainder $R(x)$ of degree ≤ 7.

$R(x)$ is multiplied by $H(x)$ to produce $N(x)$, where $H(x)$ is defined as $H(x) = x^7 + x^6 + x^5 + x^4 + x^2 + x + 1$.

$N(x)$ is divided by $G(x)$, producing a remainder $R'(x)$ of degree ≤ 7.

The coefficients of $R'(x)$ are considered to be an 8-bit sequence.

The bit sequence is complemented and the result is the CRC'.

The 8 bits of the CRC' are placed in the HCS field so that x^7 is the least-significant bit of the octet and x^0 term is the most-significant bit of the octet. (The bits of the CRC' are thus transmitted in the order $x^7, x^6, .. x^1, x^0$.)

Although the HCS is embedded within the protected bit stream, it is calculated in such a way that the resulting 128-bit stream provides error-detection capabilities identical to those of a 120-bit stream with an 8-bit CRC appended. The resulting 128-bit sequence, considered as the coefficients

of a polynomial of degree 127, when divided by G(x), will always produce a remainder equal to $x^7 + x^6 + x + 1$.

The input bits are unscrambled.

Because all fields covered by the HCS are transmitted at 2 MBaud and 2 bits per symbol (as described in clause 7.5.1), these fields should be received correctly in many cases where the payload is received in error. The HCS may be used in conjunction with soft-decision error statistics to determine with high probability whether the header was received correctly. This knowledge may be useful for optimizing the performance of ARQ and/or rate negotiation algorithms.

7.3.4 Link layer frame

The bit fields following the frame control field and preceding the pad field are defined in the G.9954v2 link-layer specification in clause 11. The first 6 octets are the Destination Address and the next 6 octets are the Source Address.

The presence of the DA and SA in the low-rate header enables reliable error-detection, which is useful for rate selection.

7.3.5 Pad

For payloads encoded at rates greater than or equal to 4 MBaud, a variable-length *pad* field consisting of an integer number of octets shall be inserted. The last octet of the pad field (PAD_LENGTH) shall be 255 (0xff) or the number of zero octets (0x00) preceding PAD_LENGTH, whichever is less. The number of zero octets shall ensure that the minimum length of the transmission, from the first symbol of the PREAMBLE64 through the last symbol of the end-of-frame delimiter, is at least 92.5 µs. For 2-MBaud payloads, there shall not be a pad field.

An example of a compliant formula for generating PAD_LENGTH is:

$$\min\left\{255, \left\lceil \frac{(92.5\,\mu s - 68\,\mu s - 2\,\mu s) \times \mathrm{B}\dfrac{Msymbol}{\text{second}} \times \mathrm{BPS}\dfrac{bit}{symbol}}{8\dfrac{bit}{octet}} \right\rceil - 1 - N\right\}$$

where the baud, B, is either 4, 8, 16, or 32, BPS is the bits per symbol, *N* is the number of octets in the part of the link layer frame transmitted in the payload-rate, 68 µs is the length of the header, and 2 µs is the length of the trailer. If the formula results in a negative quantity, it means that no pad is required.

7.3.6 End-of-Frame (EOF) delimiter

The End-of-Frame sequence consists of the first 4 symbols of the TRN sequence, or **0xfc** encoded as 2 bits-per-symbol at 2 MBaud.

This field is provided to facilitate accurate end-of-carrier sensing in low-SNR conditions. A station demodulating a frame can use this field to determine exactly where the last payload symbol occurred.

7.4 Scrambler

The scrambler is the frame-synchronized scrambler shown in Figure 7-4, which uses the following generating polynomial.

$$G(x) = x^{23} + x^{18} + 1$$

Bits 15 through 18 of the shift register shall be initialized with a 4-bit pseudo-random number. This value shall be placed in the SI field defined in 7.3.3.4 in the order such that register position 15 is the MSB (bit 19 of frame control) and bit 18 is the LSB (bit 16 of frame control).

The scrambler shall be bypassed during the preamble bit field and the first 16 bits of Frame Control. The scrambler shall be initialized and enabled starting with the 17th bit of the Frame Control field.

The scrambler shall be bypassed after the last bit of the link-layer frame, or the last bit of the PAD field, if present. The EOF sequence shall not be scrambled.

The use of a pseudo-random initial scrambler state results in a more uniform power spectral density (PSD) measured over multiple similar frames. This eliminates the problem of tones in the PSD from highly correlated successive packets.

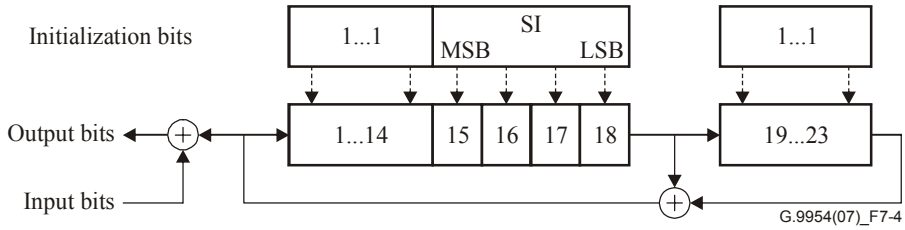

Figure 7-4 – Data scrambler

7.5 Constellation encoder

7.5.1 Constellation encoding control

All Header bits up to and including the first two bytes following the SA field shall be encoded at 2 MBaud, 2 bits per symbol. The output symbols shall be modified as described in clause 7.5.6.

Starting with the 1st bit following the two bytes following the SA field, the bits shall be encoded according to the PE field, (see Table 7-2), up to the last bit of the Link Layer Frame, or the last bit of PAD if it is present.

The EOF sequence shall be encoded at 2 MBaud, 2 bits per symbol. The output symbols shall be modified as described in clause 7.5.6.

7.5.2 Bit-to-symbol mapping

The incoming bits shall be grouped into N-bit symbols, where N is the number of bits per symbol specified in the PE field. The bit-to-symbol mapping is shown in Figures 7-5 through 4-14. The symbol values are shown with bits ordered such that the right most bit is the first bit received from the scrambler and the left most bit is the last bit received from the scrambler.

All constellations except for 3 bits per symbol lie on a uniform square grid, and all constellations are symmetric about the real and imaginary axes.

For the round constellations, only the 1st quadrant is shown and the 2 left most bits are omitted from the figures. For these cases, the 2 left most bits are specified in Figure 7-5.

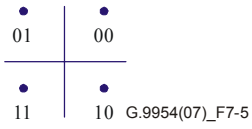

Figure 7-5 – 2 bits per symbol

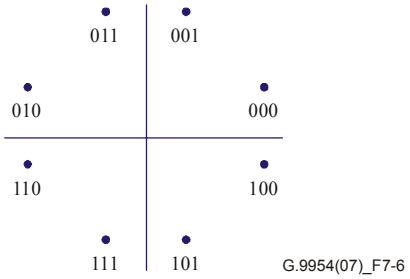

Figure 7-6 – 3 bits per symbol

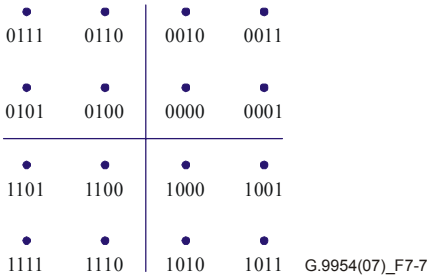

Figure 7-7 – 4 bits per symbol

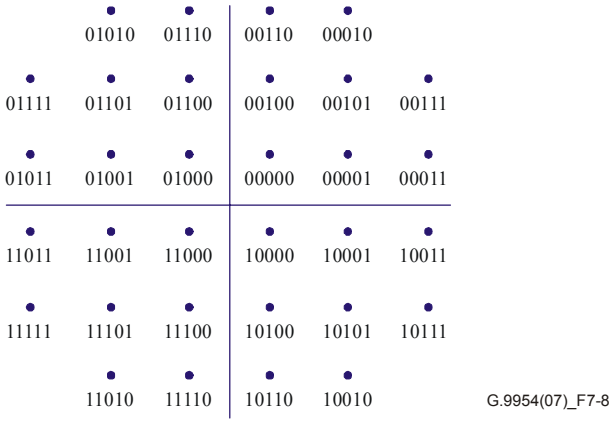

Figure 7-8 – 5 bits per symbol

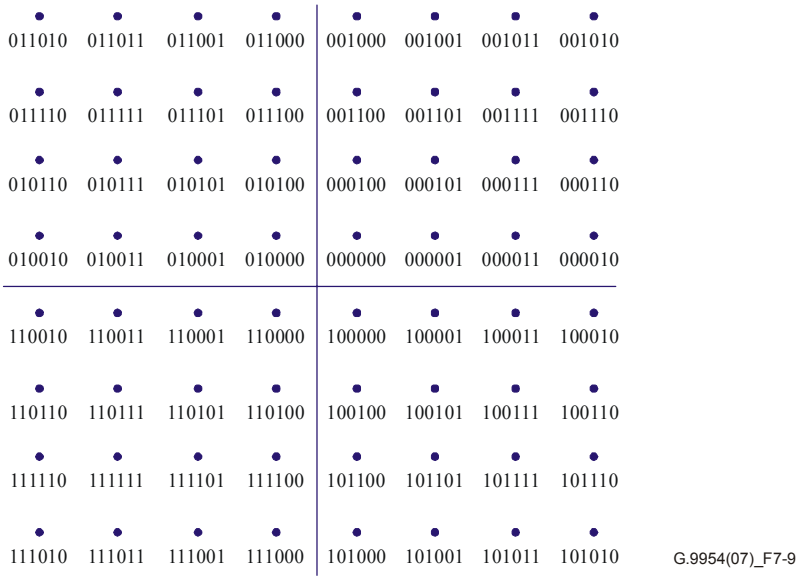

Figure 7-9 – 6 bits per symbol

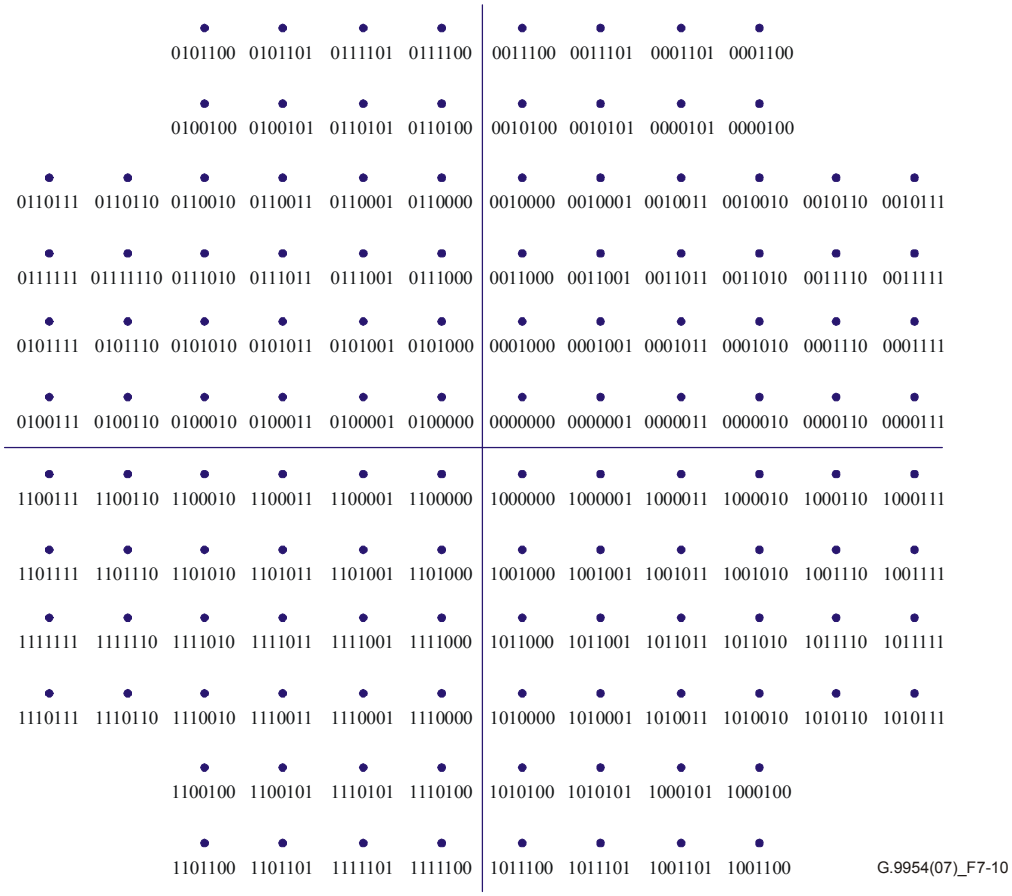

Figure 7-10 – 7 bits per symbol

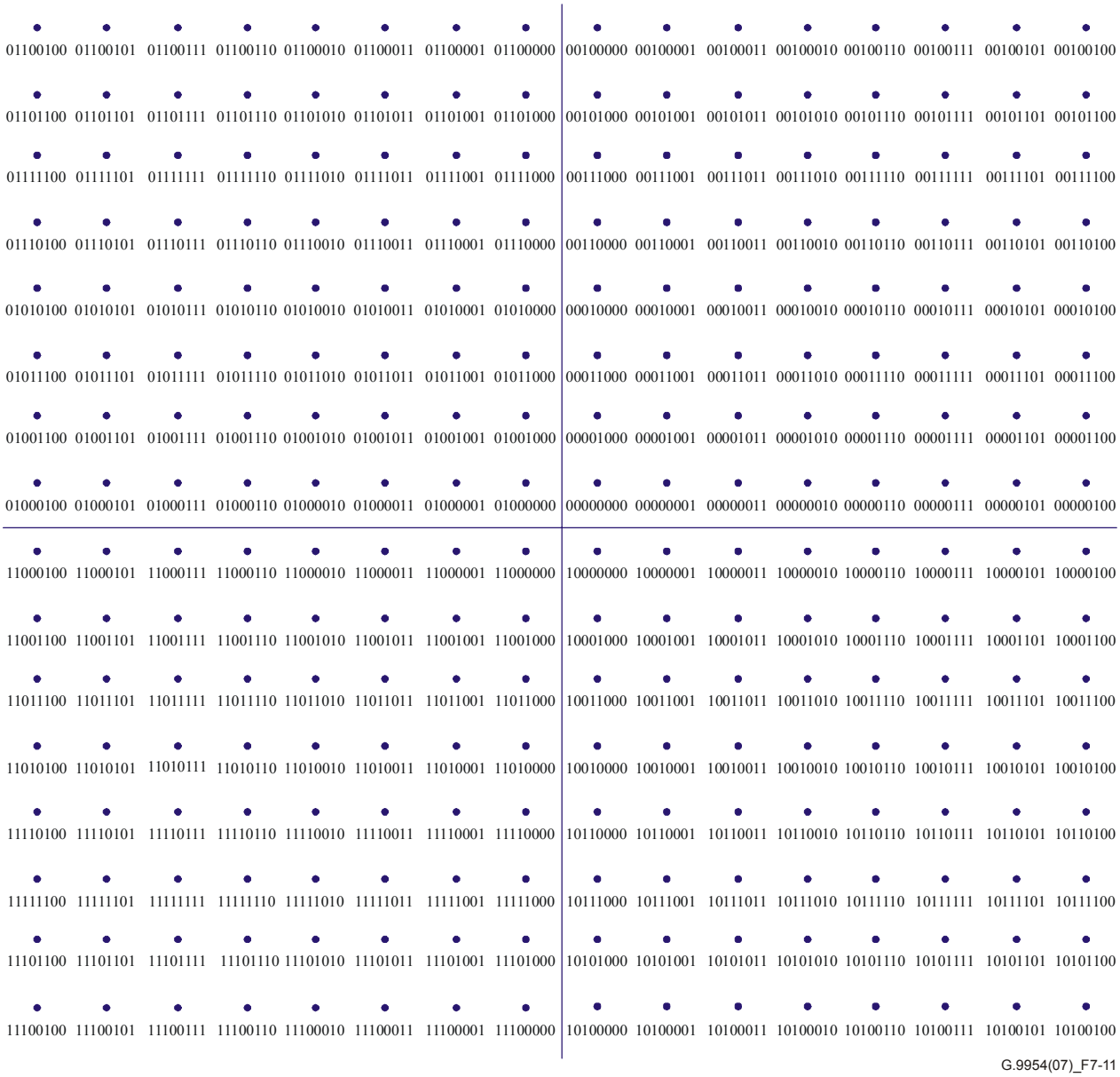

Figure 7-11 – 8 bits per symbol

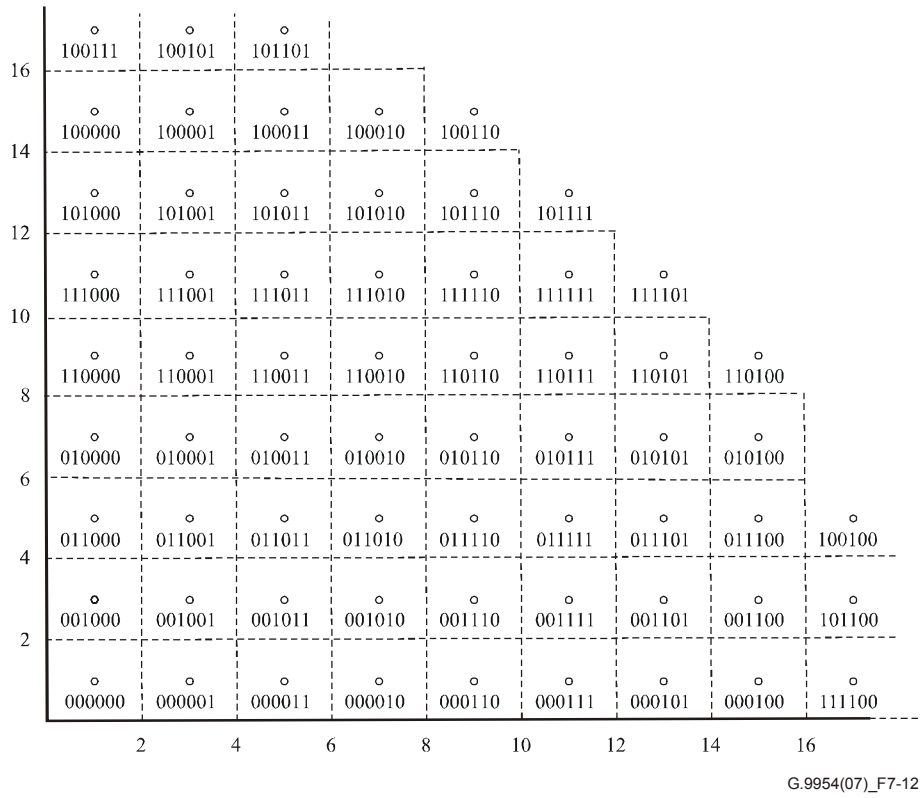

Figure 7-12 – 8 bits per symbol round constellation

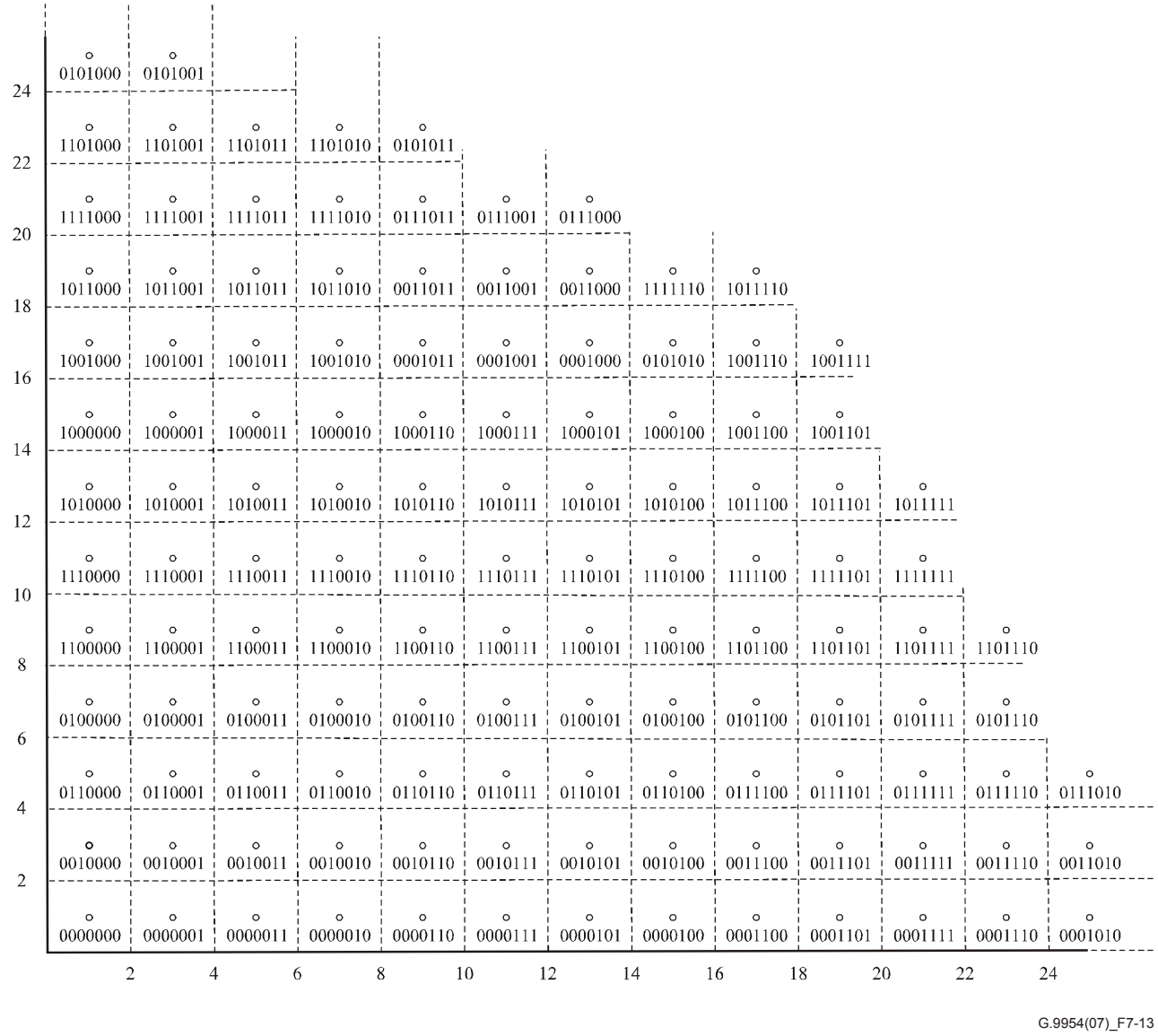

Figure 7-13 – 9 bits per symbol round constellation

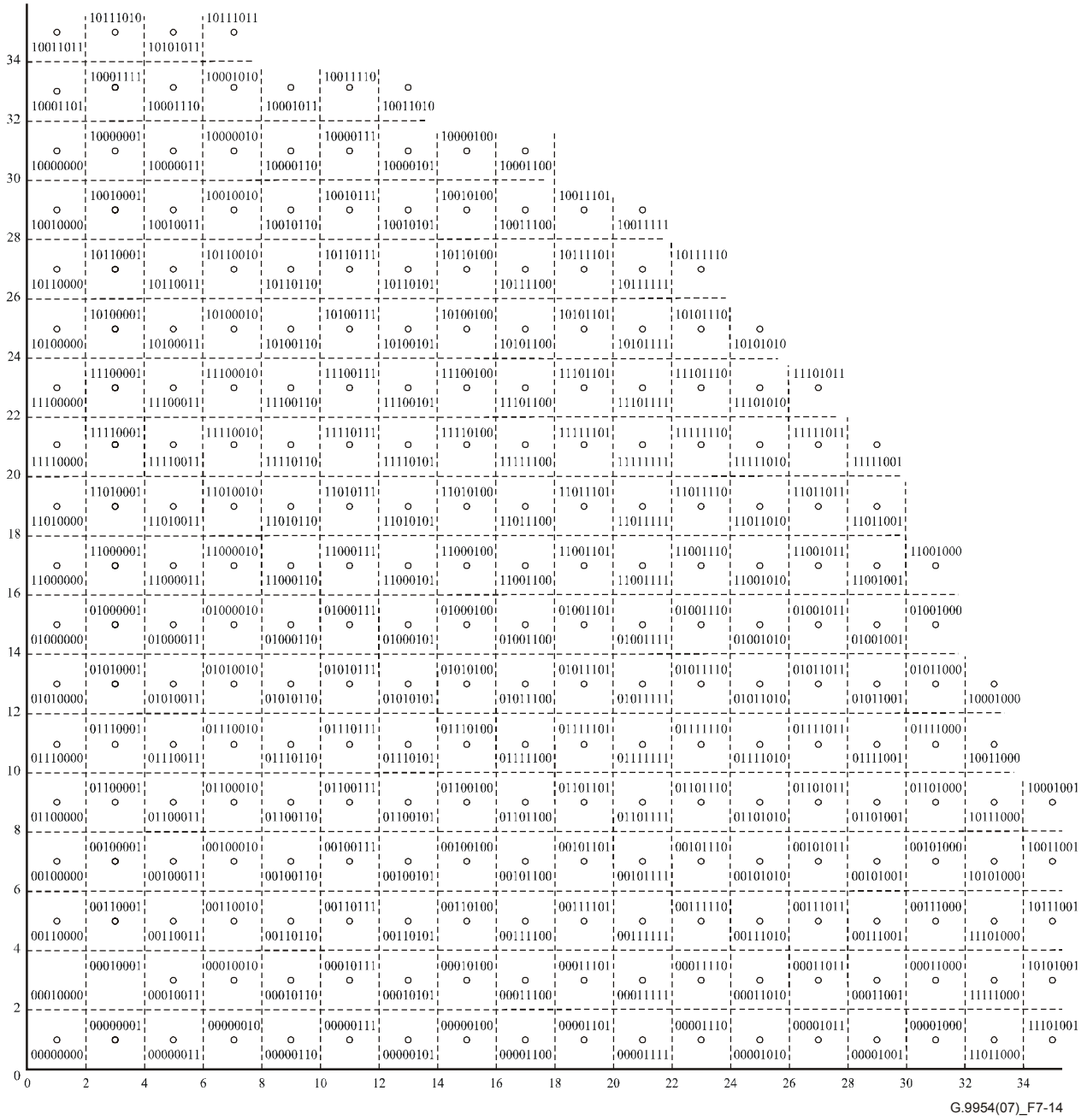

Figure 7-14 – 10 bits per symbol round constellation

7.5.3 Constellation scaling

The relative scaling of different constellations at a single baud is given by Tables 7-5 and 7-6, where the value of s(PE) in Table 7-5 is given in Table 7-6. The value of each constellation point must be accurate to within plus or minus 4 percent of the distance between nearest neighbours in that constellation.

NOTE – For example, at 2 MBaud, 2 bits per symbol, the tolerance on each point is ±0.08, while at 2 MBaud, 5 bits per symbol, the tolerance is ±0.02. Note that the tolerance is not implied by the number of significant digits in Table 7-6; i.e., the values in Table 7-6 should be considered exact.

Table 7-5 – Constellation reference points

Bits per symbol	Reference point(s)	Value
2	00	(1+i)*s(PE)
3	000	(12+5i)*s(PE)
	001	(5+12i)*s(PE)
4	0000	(1+i)*s(PE)
5	00000	(1+i)*s(PE)
6	000000	(1+i)*s(PE)
7	0000000	(1+i)*s(PE)
8	00000000	(1+i)*s(PE)
8-round	00000000	(1+i)*s(PE)
9-round	000000000	(1+i)*s(PE)
10-round	0000000000	(1+i)*s(PE)

Table 7-6 – Constellation scale factors s(PE)

Symbol rate [MHz]	2 BPS	3 BPS	4 BPS	5 BPS	6 BPS	7 BPS	8 BPS	8 BPS round	9 BPS round	10 BPS round
2	1.0000	0.1111	0.3333	0.2500	0.1429	0.1111	0.0667	0.0800	0.0556	0.0400
4	0.7071	0.0786	0.2509	0.1812	0.1113	0.0835	0.0534	0.0617	0.0431	0.0306
8	0.5000	0.0556	0.1952	0.1396	0.0897	0.0664	0.0438	0.0470	0.0332	0.0235
16	0.3119	0.0335	0.1225	0.0860	0.0583	0.0418	0.0288	0.0296	0.0210	0.0148
32	0.2500	0.0272	0.1118	0.0791	0.0546	0.0390	0.0271	0.0277	0.0196	0.0139

7.5.4 Symbol timing during baud transitions

On a transition from 2 MBaud to a higher baud, the first higher rate symbol shall occur 0.5 µs after the last 2-MBaud symbol.

On a transition from a higher baud to 2 MBaud, the first 2-MBaud symbol shall occur 0.5 µs after the last higher baud symbol.

For example, the transitions from 2 to 4 MBaud and from 4 to 2 MBaud are illustrated in Figure 7-15.

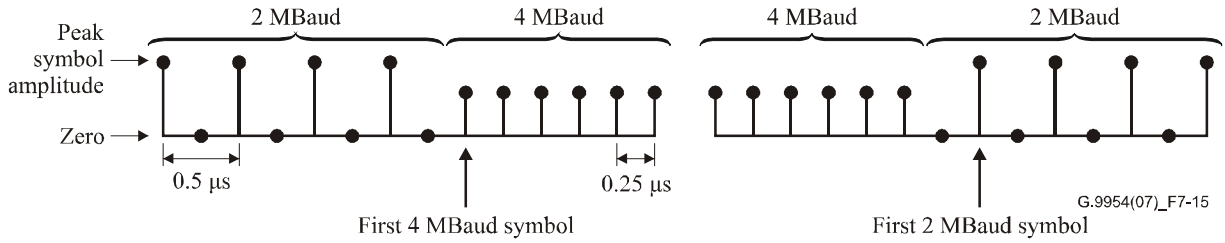

Figure 7-15 – Baud transitions

7.5.5 Encoding rate transitions

If the number of bits in a sequence is not an integer multiple of the number of bits per symbol, then enough zero bits shall be inserted at the end of the bit stream to complete the last symbol. The number of zero bits inserted shall be the minimum number such that the length of the appended bit stream is an integer multiple of the number of bits per symbol.

7.5.6 Modified header and trailer

The constellation encoder shall negate every other symbol of the header and trailer starting with the second symbol. That is, symbols 2,4,6...136 of the header and symbols 2 and 4 of the EOF shall be multiplied by –1.

7.6 QAM modulator

The modulator implements quadrature amplitude modulation (QAM). Figure 7-16 shows an example implementation. The carrier frequencies and transmit filters for each spectral mask do not depend on baud.

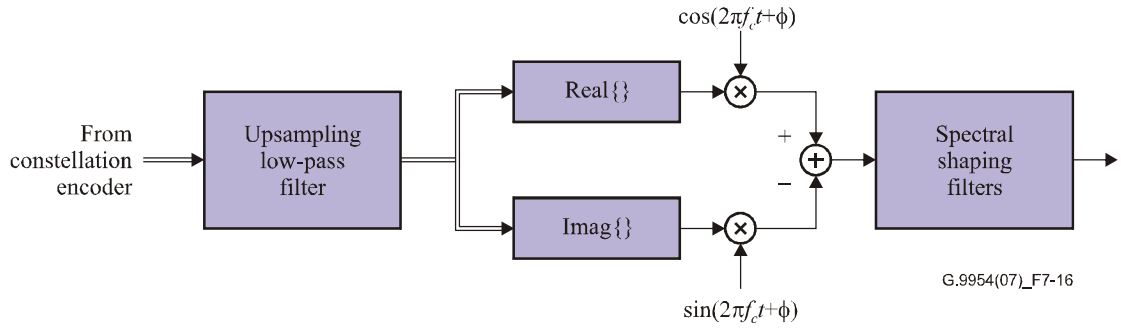

Figure 7-16 – QAM modulator

7.6.1 Carrier frequency and tolerance

Each spectral mode has its own carrier frequency f_c:

- Spectral mode A: f_c = 12 MHz;
- Spectral mode B: f_c = 12 MHz;
- Spectral mode C: f_c = –36 MHz;
- Spectral mode D: f_c = 12 MHz.

The carrier clock shall be locked to the symbol clock. So, the carrier frequency tolerance is derived from the clock tolerance defined in clause 7.8.5.

A common carrier frequency for modes A, B and D enables interoperability between spectral modes A and D in 16 MBaud and between spectral modes B and D in 16 MBaud. The negative carrier frequency for mode C results in transmitting the mirror image of the constellation encoder output spectrum.

7.6.2 Transmit filters

The details of the transmit filters are implementation dependent. Clauses 7.8.3 and 7.8.4 constrain the transmit filter designs.

7.7 Minimum device requirements

A G.9954v2 station at a minimum shall be capable of transmitting and receiving over phoneline or over coax.

A station supporting Coax at a minimum shall be capable of transmitting and receiving in one of the spectral modes: A, B, C or D specified in clause 7.8.3. Stations supporting transmission and reception of spectral mode D shall be capable of transmitting and receiving spectral modes A and B as well.

Stations at a minimum shall be capable of transmitting and receiving 2-, 4-, 8- and 16-MBaud modulated frames. Stations supporting spectral mode D shall be capable of transmitting and receiving 32 MBaud as well.

Stations at a minimum shall be capable of transmitting all constellations from 2 bits per baud to 8 bits per symbol and receiving all constellations from 2 bits per symbol to 8 bits per symbol.

7.8 Transmitter electrical specification

7.8.1 Transmit power

Stations shall transmit according to the transmit power limitations described in Table 7-7, corresponding to the spectral mode they transmit. Transmit power shall be measured during the header, across a 75-ohm load between centre and ground, integrated from 0 to 100 MHz.

Table 7-7 – Transmit power requirements

Spectral mode	Transmit power limit [dBm]
A	[–2 +1]
B	[–2 +1]
C	[–5 –2]
D	[–2 +1]

7.8.2 Transmit voltage

Stations that are not transmitting shall emit less than –85 dBVrms measured across a 75-ohm load between centre and ground.

7.8.3 Spectral masks

Four spectral masks are defined for the four spectral modes. Stations shall use the spectral mask according to the spectral mode they transmit.

7.8.3.1 PSD upper bound

When transmitting in spectral mode A, the HNT metallic power spectral density (PSD) shall be constrained by the upper bound depicted in Figure 7-17 and Table 7-8 with the measurement made across a 75-ohm load between centre and ground at the transmitter W1 interface. The upper bound shall apply to all symbol rates and constellations.

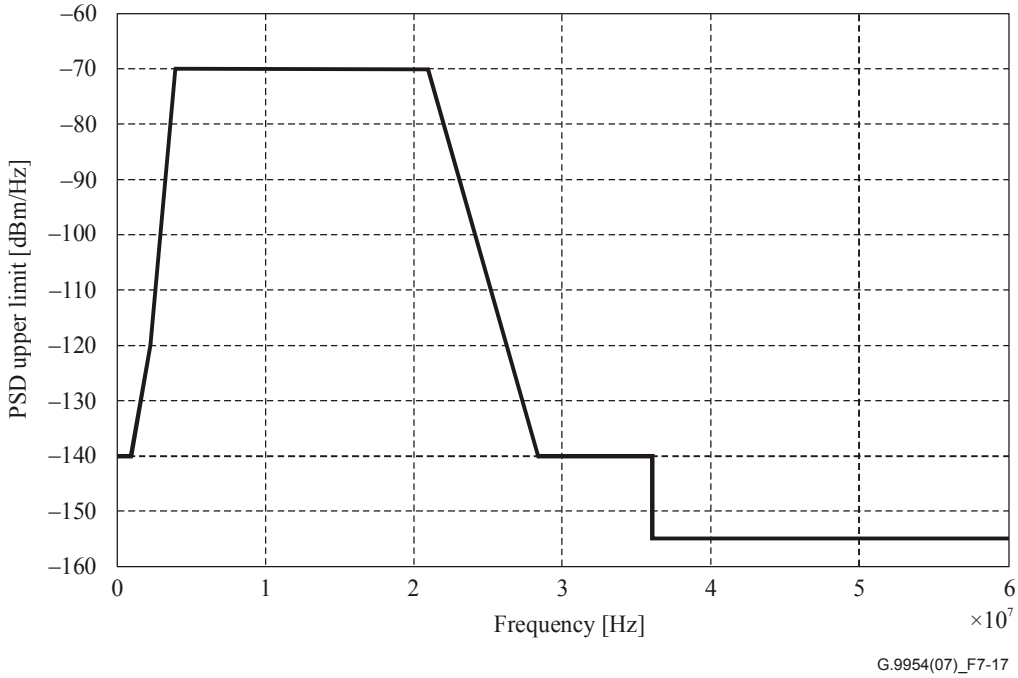

Figure 7-17 – Transmit PSD upper bound for spectral mode A

Table 7-8 – Transmit PSD upper bound for spectral mode A

Frequency [MHz]	PSD limit [dBm/Hz]
$0.015 < f \leq 1$	–140
$1 < f \leq 2.3$	$-140 + (f - 1) \times 20/1.3$
$2.3 < f \leq 4$	$-120 + (f - 2.3) \times 50/1.7$
$4 < f \leq 21$	–70
$21 < f \leq 28.4$	$-70 - (f - 21) \times 70/7.4$
$28.4 < f \leq 36$	–140
$36 \leq f$	–155

When transmitting in spectral mode B, the HNT metallic power spectral density (PSD) shall be constrained by the upper bound depicted in Figure 7-18 and Table 7-9 with the measurement made across a 75-ohm load between centre and ground at the transmitter W1 interface. The upper bound shall apply to all symbol rates and constellations.

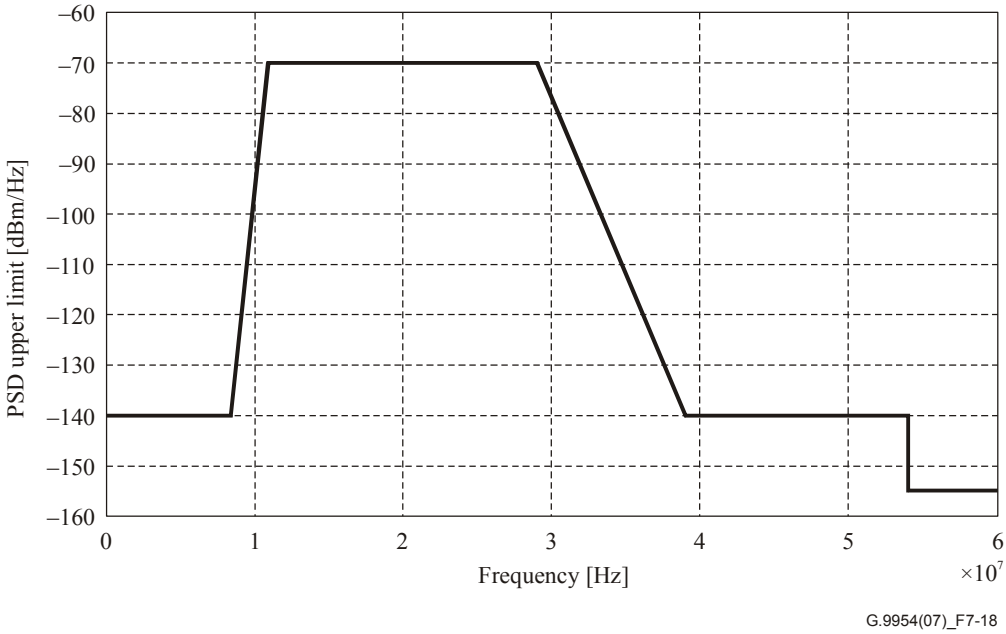

Figure 7-18 – Transmit PSD upper bound for spectral mode B

Table 7-9 – Transmit PSD upper bound for spectral mode B

Frequency [MHz]	PSD limit [dBm/Hz]
$0.015 < f \leq 8.5$	–140
$8.5 < f \leq 11$	$-140 + (f - 8.5) \times 70/2.5$
$11 < f \leq 29$	–70
$29 < f \leq 39$	$-70 - (f - 29) \times 70/10$
$39 < f \leq 54$	–140
$54 \leq f$	–155

When transmitting in spectral mode C, the HNT metallic power spectral density (PSD) shall be constrained by the upper bound depicted in Figure 7-19 and Table 7-10 with the measurement made across a 75-ohm load between centre and ground at the transmitter W1 interface. The upper bound shall apply to all symbol rates and constellations.

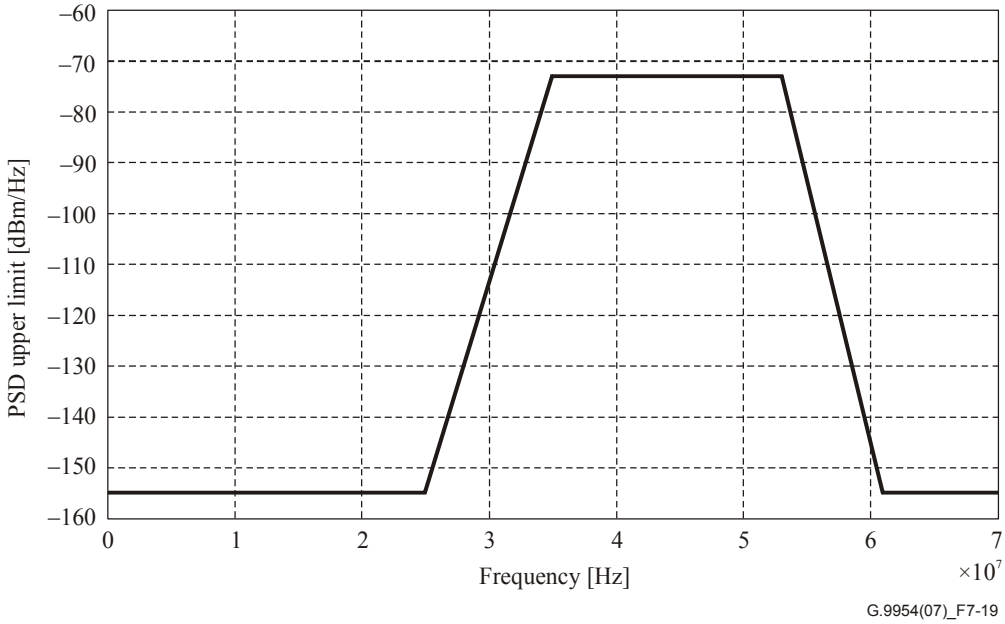

Figure 7-19 – Transmit PSD upper bound for spectral mode C

Table 7-10 – Transmit PSD upper bound for spectral mode C

Frequency [MHz]	PSD limit [dBm/Hz]
$0.015 < f \leq 25$	–155
$25 < f \leq 35$	$-155 + (f - 25) \times 82/10$
$35 < f \leq 53$	–73
$53 < f \leq 61$	$-73 - (f - 53) \times 82/8$
$61 \leq f$	–155

When transmitting in spectral mode D, the HNT metallic power spectral density (PSD) shall be constrained by the upper bound depicted in Figure 7-20 and Table 7-11 with the measurement made across a 75-ohm load between centre and ground at the transmitter W1 interface. The upper bound shall apply to all symbol rates and constellations.

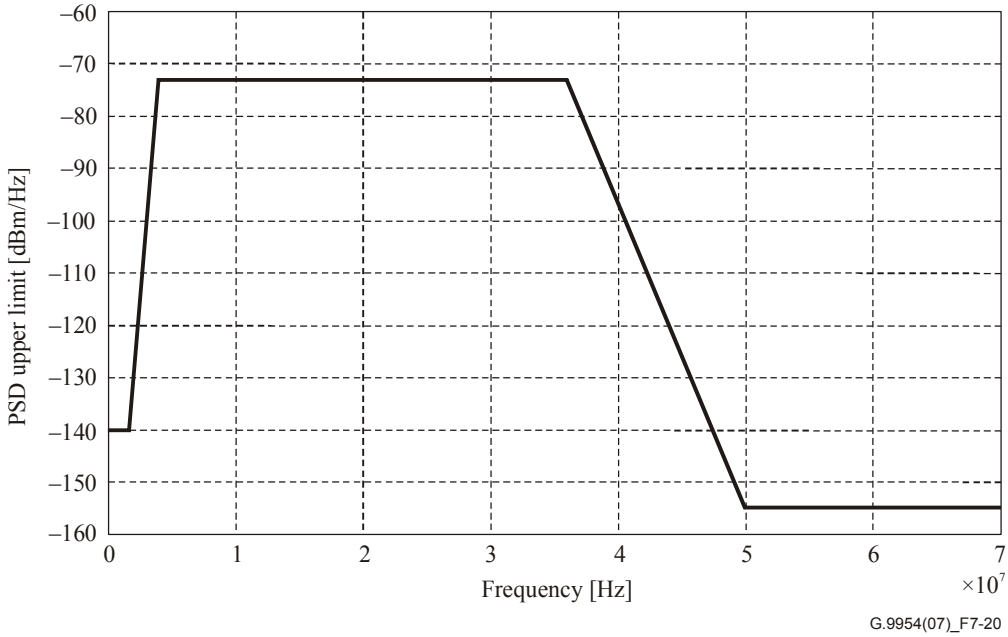

Figure 7-20 – Transmit PSD upper bound for spectral mode D

Table 7-11 – Transmit PSD upper bound for spectral mode D

Frequency [MHz]	PSD limit [dBm/Hz]
$0.015 < f \leq 1.7$	–140
$1.7 < f \leq 4$	$-140 + (f - 1.7) \times 67/2.3$
$4 < f \leq 36$	–73
$36 < f \leq 50$	$-73 - (f - 36) \times 82/14$
$50 \leq f$	–155

When transmitting in spectral mode A, the resolution bandwidth used to make this measurement shall be 10 kHz for frequencies between 2.5 and 60.0 MHz, and 3 kHz for frequencies between 0.015 and 2.5 MHz. An averaging window of 213 seconds shall be used, and 1500-octet MTUs separated by an IFG duration of silence shall be assumed. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line under 2.5 MHz, with no sub-band greater than 20 dB above the limit line. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line between 28.5 and 60.0 MHz, with no sub-band greater than 20 dB above the limit line.

When transmitting in spectral mode B, the resolution bandwidth used to make this measurement shall be 10 kHz for frequencies between 2.5 and 60.0 MHz, and 3 kHz for frequencies between 0.015 and 2.5 MHz. An averaging window of 213 seconds shall be used, and 1500-octet MTUs separated by an IFG duration of silence shall be assumed. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line under 8.5 MHz, with no sub-band greater than 20 dB above the limit line. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line between 39.0 and 60.0 MHz, with no sub-band greater than 20 dB above the limit line.

When transmitting in spectral mode C, the resolution bandwidth used to make this measurement shall be 10 kHz for frequencies between 2.5 and 80.0 MHz, and 3 kHz for frequencies between 0.015 and 2.5 MHz. An averaging window of 213 seconds shall be used, and 1500-octet MTUs separated by an IFG duration of silence shall be assumed. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line under 25.0 MHz, with no sub-band greater than 20 dB above the limit line. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line between 61.0 and 80.0 MHz, with no sub-band greater than 20 dB above the limit line.

When transmitting in spectral mode D, the resolution bandwidth used to make this measurement shall be 10 kHz for frequencies between 2.5 and 70.0 MHz and 3 kHz for frequencies between 0.015 and 2.5 MHz. An averaging window of 213 seconds shall be used, and 1500-octet MTUs separated by an IFG duration of silence shall be assumed. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line under 2.5 MHz, with no sub-band greater than 20 dB above the limit line. A total of 50 kHz of possibly non-contiguous bands may exceed the limit line between 50.0 and 70.0 MHz, with no sub-band greater than 20 dB above the limit line.

NOTE – The masks should be tested at a PE value of 2 MBaud and 2 bits/symbol, as this payload encoding results in the maximum transmitted power.

7.8.3.2 Passband ripple

Stations shall not exceed the maximum ripple described in Table 7-12, corresponding to the spectral mode they transmit. The requirements apply to a frequency range corresponding to the spectral mode.

Table 7-12 – Passband ripple requirements

Spectral mode	Frequency range [MHz]	Maximum ripple [dB]
A	[5.0 19.0]	3
B	[13.0 27.0]	3
C	[37.0 51.0]	3
D	[6.0 34.0]	3

7.8.4 Transmitter symbol response

The symbol response of the transmitter output shall be upper-bounded by the temporal mask shown in Figure 7-21. The response shall be measured across a 75-ohm load between centre and ground at the transmitter's W1 interface.

Output before $t = 0$ and after $t = 5.0$ μs shall be < 0.032% of the peak amplitude.

In Figure 7-21, the time $t = 0$ is arbitrary.

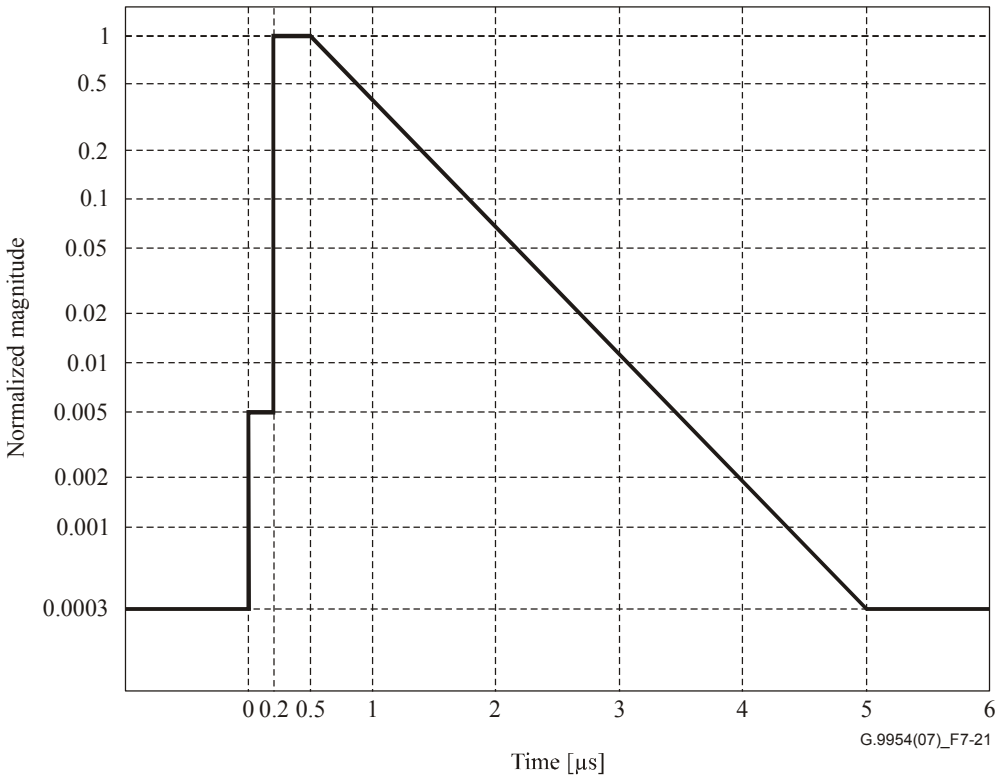

Figure 7-21 – Transmitter symbol response magnitude mask for all spectral modes

7.8.5 Clock tolerance

Transmitter clock frequency of the device shall be accurate to within the clock tolerance as described in Figure 7-22, corresponding to the spectral mode it transmits. Transmitter shall comply with the requirements over all operating temperatures for the device. The minimum operation temperature range for this requirement is 0 to 70°C.

Spectral mode	Clock tolerance [ppm]
A	[+100 –100]
B	[+30 –30]
C	[+30 –30]
D	[+30 –30]

Figure 7-22 – Clock tolerance requirements

In general, a ±50-ppm crystal will be required to meet the accuracy requirement of ±100 ppm and a ±20-ppm crystal will be required to meet the accuracy requirement of ±30 ppm.

7.8.6 Clock jitter

The rms jitter of the transmitter clock shall not exceed the maximum jitter requirements described in Table 7-13, corresponding to the spectral mode it transmits. The rms jitter is averaged over a 10-µs window.

Table 7-13 – Maximum jitter requirements

Spectral mode	Maximum rms jitter [ps]
A	70
B	70
C	50
D	50

7.8.7 I/Q balance

There shall be no gain or phase imbalance in the transmitter, except as noted in clause 7.5.3.

7.9 Receiver electrical specification

7.9.1 Receiver sensitivity

7.9.1.1 Maximum signal

The receiver shall detect frames with peak voltage up to 1.5 dBV between centre and ground for spectral modes A, B and D and up to –0.5 dBV for spectral mode C. The detection shall be at a frame error rate of no greater than 10^{-3} with additive white Gaussian noise at a PSD of less than –140 dBm/Hz, measured at the receiver.

7.9.1.2 Minimum sensitivity

For all spectral modes, the receiver shall detect 1518-octet frames encoded as 2 bits/symbol and 2 Mbaud with rms voltage as low as 500 μV at no greater than 10^{-3} frame error rate with a flat channel and additive white Gaussian noise at a PSD of less than –140 dBm/Hz, measured at the receiver. The rms voltage is computed only over time during which the transmitter is active.

7.9.2 Clock tolerance

The receiver shall meet the requirements of clause 7.9.3 for each spectral mode it supports, when the transmitter clock frequency is within any of its allowed range defined in clause 7.8.5.

7.9.3 System margin requirements

Variable flat attenuation shall be used to verify the minimum receiver requirements. For each spectral mode, three attenuation points are defined: Minimum attenuation, high-rate maximum attenuation and low-rate maximum attenuation. For every spectral mode supported by the receiver, all three attenuation points shall comply with a frame error rate (FER) of less than 10^{-4} using the required attenuation and its corresponding payload encoding (PE) described in Table 7-14. 1518-octet frames shall be used for this measurement.

Table 7-14 – Attenuation performance requirements

Spectral mode	Minimum attenuation		High-rate maximum attenuation		Low-rate maximum attenuation	
	PE	Attenuation [dB]	PE	Attenuation [dB]	PE	Attenuation [dB]
A	16 MBaud/7 BPS	0	16 MBaud/7 BPS	40	16 MBaud/3 BPS	55
B	16 MBaud/7 BPS	0	16 MBaud/7 BPS	40	16 MBaud/3 BPS	55
C	16 MBaud/7 BPS	0	16 MBaud/7 BPS	29	16 MBaud/2 BPS	48
D	32 MBaud/7 BPS	0	32 MBaud/7 BPS	40	32 MBaud/3 BPS	52

7.10 Input impedance

Stations shall comply with the input impedance requirements corresponding to the spectral mode they support. Stations supporting spectral mode D shall comply with the requirements defined for spectral mode D only.

7.10.1 Passband return loss

Stations shall exceed the minimum average return loss as described in Table 7-15 with respect to a 75-ohm resistive load, corresponding to the spectral mode they transmit. The averaging is performed over the frequency range corresponding to the spectral mode. One column in Table 7-15 applies to the transceiver powered on or in low-power mode (transmitter powered off). Another column applies to the transceiver removed from a source of power.

Table 7-15 – Passband return loss requirements

Spectral mode	Frequency range [MHz]	Minimum average return loss [dB] (transceiver powered on/low power mode)	Minimum average return loss [dB] (transceiver powered off)
A	[4.25 19.75]	10	6
B	[12.25 27.75]	10	6
C	[36.25 51.75]	10	6
D	[4.25 35.75]	10	6

7.10.2 Stopband input impedance

Stations shall exceed the minimum impedance magnitude as described in Table 7-16, corresponding to the spectral mode they transmit. The requirements apply to lower and upper frequency ranges corresponding to the spectral mode.

Table 7-16 – Stopband input impedance requirements

Spectral mode	Lower frequency range [MHz]	Upper frequency range [MHz]	Minimum impedance magnitude [ohm]
A	[0 4.25]	[19.75 1000]	10
B	[0 12.25]	[27.75 1000]	10
C	[0 36.25]	[51.75 1000]	10
D	[0 4.25]	[35.75 1000]	10

This requirement applies to the transceiver powered on, in low-power mode (transmitter powered off), or removed from a source of power.

8 Media access protocol specification

The G.9954v2 media access protocol is a synchronous protocol that uses carrier-sensing and collision avoidance methods (CSMA/CA) to coordinate access to a shared media amongst a set of G.9954v2 nodes. The G.9954v2 MAC protocol is suitable for shared media networks composed of phonelines, coax or hybrid phoneline/coax wiring.

The G.9954v2 MAC uses a resource reservation scheme to guarantee media resources to network devices and to prevent collisions between multiple network devices contending for access to the media.

A G.9954v2 network is composed of at least two network nodes. One of the network nodes takes the role of the network "master" and is referred to as the master while the other nodes are referred to as "endpoint"s. A G.9954v2 network node comprises of, amongst other items, a carrier sensor and a transceiver. The master also includes a scheduler. The master's scheduler sends to each device on the network a media access plan (MAP) at the beginning of each transmission cycle. The transceiver either transmits, or both transmits and receives data transmissions over the network.

Transmissions are performed within the context of a transmission cycle. The MAP that starts the transmission cycle describes the schedule of future transmission opportunities (TXOPs) that are available to specific network devices in the upcoming transmission cycle at specific and non-overlapping times. The start time and length of each scheduled TXOP in the upcoming transmission cycle, as well as the network devices and flows to which each TXOP is assigned, is determined by the scheduler and defined in the MAP. The transmission cycle is then initiated with the publication of the MAP by the scheduler to all the network devices on the network.

After the publication of the MAP, network device transmissions may begin. Each device recognizes, according to the MAP, a particular TXOP within which it is allowed to transmit and either utilizes (transmits within) the TXOP or passes on it.

A TXOP may be assigned to a single or multiple network devices and/or flows. When a TXOP is assigned to a single network device, the network device is considered to have exclusive access to the TXOP and media access is necessarily collision-free. When a TXOP is assigned to multiple network devices, the media access method based on carrier-sensing and collision-avoidance techniques is also used to allow collision-free burst-like media access within the shared TXOP.

To allow multiple network devices to perform collision-free media access within a shared TXOP, the G.9954v2 MAC defines a TXOP to be composed of a grid of smaller time-slots (sub-burst slots) which represents an opportunity for the initiation of a data transmission by a network device. The sub-burst slot is appreciably smaller in size than the containing TXOP and is smaller than a minimal-sized transmission burst. The advantage of the small sub-burst slot structure can be appreciated when a network device does not use its assigned sub-burst slot. In this case, only the sub-burst slot time is wasted before the opportunity to transmit is passed to the device assigned the next sub-burst slot. The assignment of devices and flows to sub-burst slots is defined by the scheduler and published in the MAP.

To support collision-free media access within shared TXOPs based on sub-burst slots, a G.9954v2 network device comprises of, amongst other things, a carrier sensor, a transceiver, a sub-burst slot grid aligner and a sub-burst slot scheduler. The sub-burst slot grid aligner is located in all G.9954v2 devices and is responsible for tracking the passage of sub-burst slots on the media in accordance with the MAP, identifying transmissions within sub-burst slots and re-aligning sub-burst slot grid timing after the occurrence of a transmission. The sub-burst slot scheduler is located only in the G.9954v2 master device and is responsible for creating and sending, to all devices on the network, a media access plan containing shared TXOPs with sub-burst slot assignments made to multiple network devices and/or flows.

In order for a G.9954v2 network device to act on an opportunity to transmit within an assigned sub-burst slot, it must initiate its transmission at the beginning of the sub-burst slot opportunity. The size (duration) of a sub-burst slot opportunity is small and configurable by the G.9954v2 master in the range of 8-64 μs. The window for acting upon the opportunity to initiate a transmission within a sub-burst slot shall be within 4 μs of the start of the sub-burst slot. The size of a sub-burst slot opportunity is defined by the G.9954v2 master and communicated to all devices on the network through the MAP.

In the event that a sub-burst slot opportunity S_n is not utilized by the network device to which the slot is assigned, the device assigned the next sub-burst slot in the sub-burst slot grid sequence, S_{n+1} is given the opportunity to transmit. When a sub-burst slot is utilized for a transmission, the sub-burst slot in which the transmission occurs can be thought of as expanding to allow the completion of the transmission burst started within it. Upon completion of the transmission burst, the sub-burst slot grid pattern continues in accordance with the MAP and with sub-burst slot grid timing adjusted relative to the end of the transmission.

The length of a transmission burst within a sub-burst slot, in general, is bounded only by the maximum transmission burst size. However, towards the end of the containing TXOP, the length of the transmission burst may also be bounded in time by the end of the containing TXOP. A transmission burst may extend beyond the end of the TXOP, extending the length of the TXOP if the transmission starts before the end of the TXOP and the attributes of the TXOP allow its length to be extended. By default, the length of a TXOP is strictly bounded.

At the beginning of each cycle, network devices receive a schedule of transmission opportunities (i.e., a MAP) detailing the timing of the TXOPs, an ordering of sub-burst slots within the TXOP and an assignment of sub-burst slots to devices and data flows. While the order of sub-burst slots in the grid is known to all devices on the network, the timing of the sub-burst slots in the grid changes, due to and in direct accordance with the transmissions that occur and the duration of the transmission. Consequently, following each transmission, the sub-burst slot grid aligner function in each network device on the network must recalculate the timing of the sub-burst slot grid so that each network device knows what time its next transmission opportunity is scheduled.

8.1 Modes of operation

The G.9954v2 MAC shall support a single mode of operation over all media types. This media access method assumes the presence of a single G.9954v2 device on the network acting in the role of "master" and coordinating media access timing between network nodes.

The presence of a G.9954v2 master on the network is detected by the reception of media access plan (MAP) messages. Upon reception and decoding of a MAP message, a G.9954v2 device shall start operating according to the G.9954v2 media access method and in accordance with timing described in the published MAP. In this state, a G.9954v2 node is said to be in "SYNC" and the network is said to be a "managed network".

MAP messages are expected periodically when operating in a managed network. The period between MAP messages may be variable and may vary between CYCLE_MIN and CYCLE_MAX. Failure to receive a MAP message for more than SYNC_TIMEOUT (see clause 8.14) interval from the last MAP message shall be detected by a G.9954v2 node as a loss of sync and the G.9954v2 node is said to be not in SYNC (NOSYNC) and the network is said to be "unmanaged".

The G.9954v2 MAC modes of operation and the transitions between modes are described in the following state-transition diagram (Figure 8-1).

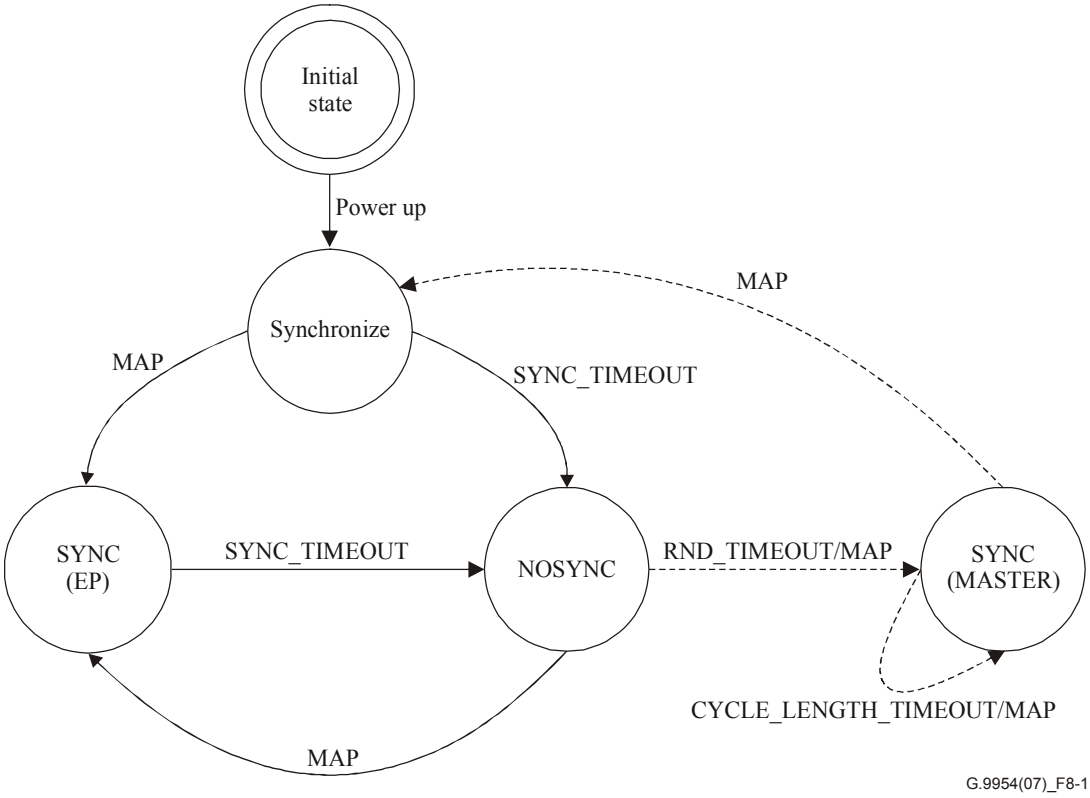

Figure 8-1 – G.9954v2 modes of operation – State diagram

When a G.9954v2 device powers up, it shall first attempt to synchronize with an existing MAC cycle by waiting for a MAP message for up to SYNC_TIMEOUT interval. If a MAP message arrives within SYNC_TIMEOUT interval, a managed network is assumed and the G.9954v2 device shall operate according to the media access rules for a managed network. The device is considered to be a G.9954v2 endpoint (EP) and in the SYNC (EP) state. If a MAP message is NOT received within SYNC_TIMEOUT interval, an unmanaged network is assumed and the G.9954v2 device shall operate according to the media access rules for an unmanaged network and is considered to be in the NOSYNC state. While in the NOSYNC state, a device shall enter into the SYNC-EP state upon receiving a MAP message. If a G.9954v2 device is capable of becoming a MASTER (optional), it may start transmitting MAP messages after a random timeout period (RND_TIMEOUT) and assume the role of the network master provided that it did not receive a MAP message during this period. In this case, the device is considered to be in the SYNC (MASTER) state and shall continue sending periodic MAP messages at the end of each MAC cycle period (MAC_CYCLE_TIMEOUT). If a G.9954v2 device that has currently assumed the role of master receives a MAP message from another device on the network, it should cease to send MAP messages and attempt to re-synchronize with the MAC cycle.

The current mode of operation of a G.9954v2 device is indicated by a flag in the G.9954v2 link-layer capability and status announcement (CSA) message. See the capability and status announcement flags in the G.9954v2 link-layer specification in clause 11.

8.1.1 Media access in a managed network

When operating in a managed network a G.9954v2 device shall ONLY perform media access within dedicated transmission opportunities (TXOPs and sub-burst slots) described in the MAP. A G.9954v2 device may transmit within a TXOP if the TXOP is allocated to the device or to a group that the device belongs to. All devices may transmit within spare (UNALLOCATED) TXOPs on a contention basis.

For more information on TXOPs, TXOP assignment and device groups, see clause 8.10 and its subclauses.

8.1.2 Media access in an unmanaged network

The media access method used by a G.9954v2 device in an unmanaged network is based on the same media access method for managed networks except that it uses carrier sensing only and is entirely contention-based. An unmanaged network can be thought of as being composed of a boundless UNALLOCATED TXOP (UTXOP). Network nodes may transmit on demand once the carrier sensor detects the media to be idle. Collisions are neither detected nor avoided and destructive collisions may occur. It is the responsibility of upper layers to effectively handle any packet loss through retransmissions and/or backoff methods. Note that this mode of operation is intended only as a transient mode to be used while selecting a new master.

8.1.3 Managed and unmanaged networks

A G.9954v2 device shall switch between managed and unmanaged methods in response to the appearance or disappearance of a G.9954v2 master-published MAP on the network as described in Figure 8-1 above.

8.2 Basic CSMA

The CSMA/CA media access method is the means by which two or more stations share a common transmission channel. To transmit, a station waits (defers) for a quiet period on the channel (that is, no other station is transmitting) and waits for arrival of a transmission opportunity assigned to it. It then sends the intended message modulated as per the PHY specification.

The carrier sensor detects the starting and ending times of a valid frame transmission on the wire. This is used to determine when frames are present on the channel.

Basic CSMA behaviour is specified for a transmitter and a receiver.

8.2.1 Transmitter behaviour

See Figure 8-2 for a description of a frame transmission that is valid with respect to the specified carrier sense (CS) function (valid CS frame).

NOTE – A transmitted valid CS frame will be affected by various signal impairments when seen by any receiver, and the performance limits of the CS function are implementation dependent.

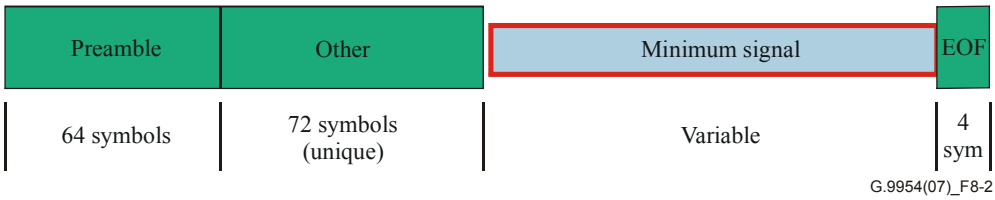

Figure 8-2 – Valid CS frame

A valid CS frame at the transmitter W1 interface has a length of TX_FRAME and consists of:

1) a sequence of symbols whose duration is equal to or greater than 92.5 µs (TX_FRAME minimum) duration, but less than the maximum specified in clause 8.13.4;

2) the first (64 + 16 + 24 + 24 + 8) symbols of which modulated at the base rate (2 MBaud QPSK, 2 bits per symbol), where the initial 64 symbols consist of the preamble sequence, where the next 64-symbol sequence is unique to the transmitting station, and where the next 8 symbols are the (likely non-unique) bits of the Ethertype field;

3) an arbitrary Minimum Signal, defined as a sequence of symbols whose rms value over any 8-µs window shall never be more than 9 dB less than 100 mV rms across 100 ohms for Phoneline modes A & B, 205 mV rms across 75 ohms for coax modes A, B, D and 145 mV rms across 75 ohms for coax mode C (NOMINAL_RMS_VOLTAGE);

4) 4 symbols of the EOF sequence;

5) a trailing transient, whose peak voltage does not exceed 0.1% of the absolute peak transmitted voltage across a 100-ohm load at the W1 interface at any point > 5 µs after the last transmitted symbol of the EOF;

6) a gap before the next transmission of this station of CS_IFG µs from the last symbol of the EOF to the first symbol of PREAMBLE of the next transmission, measured at the transmitter's W1 interface.

Receivers are only required to correctly detect valid CS frames.

The inter-frame gap shall be 29.0 µs (CS_IFG), where the gap is defined at the points at which the previous frame drops below 50% of its peak and the current frame rises above 50% of its peak.

8.2.2 Receiver behaviour

Timing of subsequent transmissions following a valid CS frame are based on a MAC timing reference, established by the receiver. Time following a transmission is divided into *slots*: an inter-frame gap (IFG); and a sequence of sub-burst slots. See Figures 8-3 and 8-4. During these time periods the MAC is *synchronized* and the slot timing is defined by the rules for valid transmissions in the previous clause.

When MAC timing is synchronized, stations shall commence any transmission no earlier than 0 and no later than 4 µs (TX_ON) after a slot origin, measured at the transmitter W1 interface.

The receiver carrier sense function, for phoneline modes A and B shall detect a maximum-amplitude valid CS frame over a range of 0 to at least 38 dB (CS_RANGE) flat-channel insertion loss and additive noise with a flat PSD of –140 dBm/Hz at the receiver with a missed frame rate of less than 10–4 and a premature end-of-frame declaration rate less than 10–4; see clause 6.9.1. With additive white Gaussian noise applied at the input with a PSD of –110 dBm/Hz, the false carrier detection rate shall be no greater than 1 per second.

For all coax modes, the receiver carrier sense function shall detect a maximum-amplitude valid CS frame over a range of 0 to at least 55 dB (CS_RANGE) flat-channel insertion loss.

8.3 Priority access

The G.9954v2 system is targeted for carrying media streams, such as audio and video. To reduce the latency variation in these streams and to prevent media resource starvation, a priority mechanism is implemented to allow higher layers to label outgoing frames with priority, and guarantee that those frames will have preferential access to the channel over lower priority frames. The access priority method implemented is to delay transmissions to a sub-burst slot that has an assigned priority-level that is less than or equal to the priority level of the frame waiting to be transmitted.

Sub-burst slots may be assigned priorities and organized into priority groups numbered in decreasing priority, starting at priority 7. All the sub-burst slots belonging to the same priority group shall have the same assigned priority level. Higher priority groups shall appear before lower priority groups and consequently higher priority transmissions commence transmission in earlier sub-burst slots and acquire the channel without contending with the lower priority traffic. A station's *Priority Group* is based on the priority number associated with the frame ready for transmission (PRI), as determined by the network stack and communicated to the MAC. The station uses any priority group to which it has sub-burst slots assigned with a number less than or equal to

PRI, normally the priority numbered exactly PRI. The station may only commit to transmit at the start of a priority sub-burst slot, i.e., if a station is ready to transmit a PRI = 7 frame only after the start of its assigned sub-burst slot within priority Group 7, it must wait until the start of its assigned sub-burst slot in Priority Group 6 to transmit. See Figure 8-3 for the relative timing of priority slots.

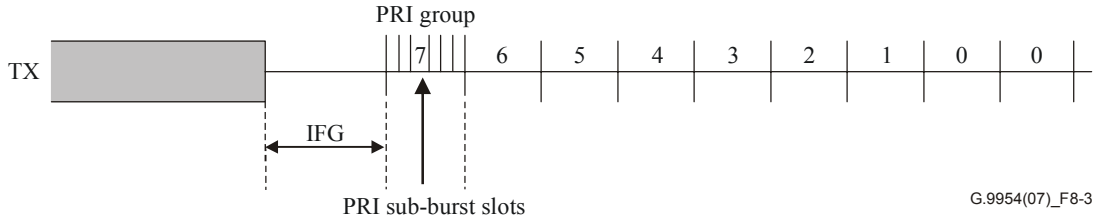

Figure 8-3 – Priority slots

No station shall transmit in a Priority Group numbered higher than the priority (PRI) assigned to the frame being transmitted and no station shall transmit in a Priority Group if it does not have sub-burst slots assigned to it in the Priority Group.

Stations not implementing priority shall assign the default link-layer priority value of 2 when transmitting.

Stations waiting for transmission shall monitor carrier sense (CS) and defer if CS was true prior to the start of the next priority sub-burst slot in which it can transmit. Any station ready to transmit at the start of the next priority sub-burst slot in which it can transmit shall transmit at the start of that priority sub-burst slot without deferring if CS was false prior to the start of that priority sub-burst-slot.

See Figure 8-4 for an example of video traffic at priority level 7 gaining access ahead of best effort traffic scheduled at level 0.

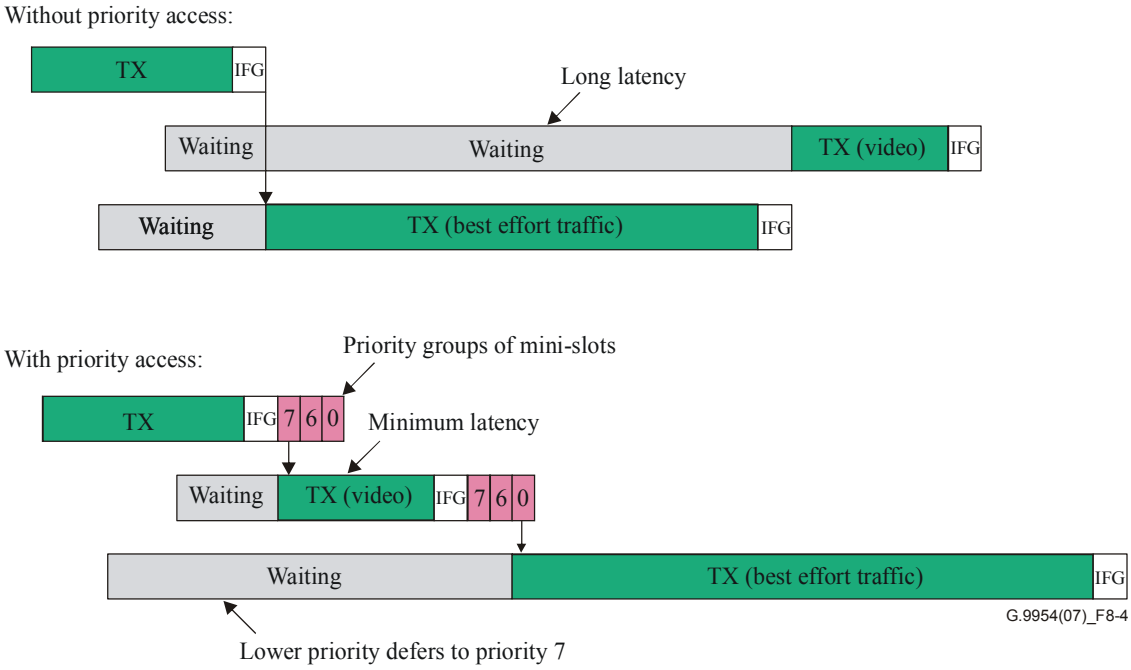

Figure 8-4 – Example of priority access

The Priority Group is restarted if there is some other transmission that acquires the channel while a station is waiting at a lower priority.

For further details on priority groups and sub-burst slots, see clause 10.2.

8.4 Priority mapping

The PRI value is the priority the MAC uses to schedule transmission and is the value present in the PRI field of the frame burst header. This value is determined by a higher layer in the network stack and the method of priority labelling is outside the scope of this Recommendation. The PRI field is used to transport the priority label from source to destination, to assist the destination in managing the receive queue. The 3-bit priority values referred to are "PHY priorities". PRI = 7 has the highest priority, PRI = 0 has the lowest.

There may be a mapping between PHY priorities and the link layer (LL) priority values as delivered to the link layer by the network layer. This mapping is described in the link layer protocol specification in clause 11.

In general, the IP network layer or application layer will determine what policy is used to map traffic onto LL priorities. For instance, IETF Integrated Services currently defines priority 0 as the default "best effort" priority, and priority 1 as the penalty "worse than best effort" priority – and most implementations will map best effort to PHY PRI = 2 and worse-than-best-effort to PHY PRI = 0.

The PHY priority mechanism is strict priority (as opposed to schemes which allocate lower priorities some minimum percentage of network capacity) – higher priority traffic always defers lower priority traffic. Higher priority traffic may be limited by admission control or other link layer policy mechanism to prevent over-subscription.

8.5 Network devices and device identifiers (Device_ID)

G.9954v2 devices are identified by their globally unique 48-bit universal MAC address.

G.9954v2 devices that operate in a managed network shall REGISTER with the master and identify themselves using their globally unique 48-bit universal MAC address. The MAC address is used by the master, during network admission, as a unique key for device identification.

A G.9954v2 network device that has been admitted by the master is assigned a *short* address, known as the Device_ID. The Device_ID is used to identify the assignment of TXOPs and sub-burst slots to devices. A G.9954v2 device is informed of its assigned Device_ID by the master during the network admission protocol; see clause 11.4.

The network Device_ID is a 6-bit structure with valid values in the range 0 to 63. Device_IDs are unique within the network.

The Device_IDs are defined in Table 8-1.

Table 8-1 – Device_ID definition

Device name	Device_ID	Description
Null device	0	The NULL (undefined) Device_ID
Master device	1	Identity of the selected G.9954v2 network master
Reserved	2-63	Device_IDs reserved for assignment by master to admitted G.9954v2 devices

8.6 Data flows and flow identifiers (Flow_ID)

A service or data flow (or just *flow* for short) is a simplex logical communication channel between a source and destination device. It is service-oriented and is defined by the type of information it transports. A device may support multiple service flows where each service flow is identified by a Flow_ID.

A Flow_ID is a 6-bit number defined in the range 0-63. A flow is uniquely identified in the network by the tuple (Source Address, Destination Address, Flow ID). The Flow_ID with a value of 0 represents the NULL (undefined) Flow_ID.

8.7 The MAC cycle

Media access in managed network is performed within the context of a periodic MAC cycle. Each period of the MAC cycle starts with the transmission of a media access plan (MAP) by the master and ends at the end of the planned media access period described in the MAP or upon the arrival of a new MAP. G.9954v2 network devices shall synchronize with the MAC cycle by detecting the presence of a MAP message and by performing media access according to the media access plan described in the MAP. The MAP describes the allocation of transmission opportunities or *TXOPs* to devices and/or service flows in the network. TXOPs are described by their start-time, duration and by the devices and/or services that may transmit within the TXOP. Timing references within the MAP are relative to the start of the MAC cycle. The beginning of the first symbol of the PREAMBLE of the received MAP transmission represents time zero.

The MAP shall describe the TXOPs in the MAC cycle immediately following the cycle in which the MAP is received. This means that the MAP message that starts MAC cycle N describes the TXOPs in MAC cycle N + 1. This is illustrated in Figure 8-5.

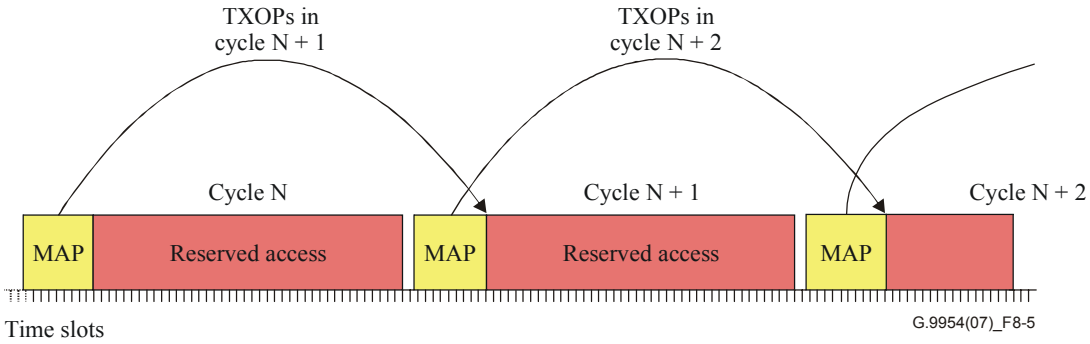

Figure 8-5 – MAC cycle and MAP reference

MAC cycles are separated by an inter-cycle gap (CS_ICG). An Inter-Cycle Gap is a guaranteed minimum period where the medium is idle based on the carrier sense function. The interval is measured from the last symbol of the EOF of the last frame in a MAC cycle to the first symbol of the PREAMBLE of the MAP transmission. Bursts within a MAC cycle are separated by an inter-frame gap (MAP_IFG) as defined in clause 8.14.

The G.9954v2 master shall allocate media time for the CS_ICG and MAP_IFG and encode it within the definition of the TXOPs described in the MAP. Each TXOP shall contain media time that includes the gap before the next transmission. See Figure 8-6.

Figure 8-6 – MAP_IFG and CS_ICG accounting

The actual length of a CS_ICG and MAP_IFG are defined in clause 8.14.

8.8 The MAC cycle length

MAC cycles are periodic and typically of constant length. The actual length of the MAC cycle may vary dynamically between cycles from CYCLE_MIN to CYCLE_MAX depending on scheduling constraints and decisions.

The length of the MAC cycle that a MAP describes is encoded implicitly in the MAP.

Since the MAP describes the media access plan for the next MAC cycle, it always takes two MAC cycles for a MAC cycle length modification to take effect. This is illustrated in Figure 8-7.

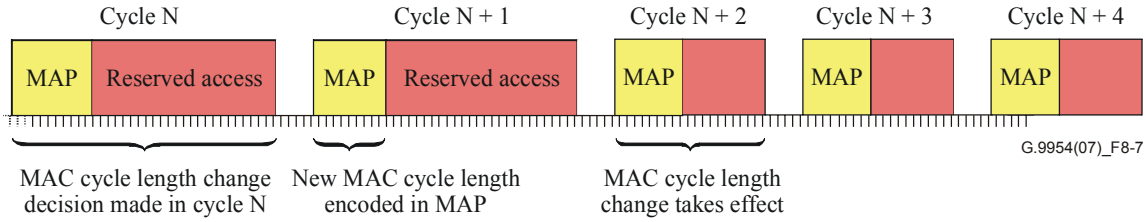

Figure 8-7 – Variable MAC cycle lengths

A MAC cycle may be prematurely terminated by the arrival of a new MAP message before the end of the scheduled cycle. Note that this mechanism enables the network to react quickly to changes in scheduling decisions.

8.9 Media access plan (MAP)

The MAP control frame signals the start of a MAC cycle (the "current" MAC cycle) and describes the TXOPs planned in the "next" MAC cycle. The "current" MAC cycle is identified by the sequence number contained in the MAP frame starting the cycle. The "next" MAC cycle is the MAC cycle that follows the "current" MAC cycle and contains a sequence number that is one more than the "current" MAC cycle accounting for modulo arithmetic.

The extent of the media access plan described by MAP frame is a single MAC cycle only. Since a MAP frame describes the media access plan for the next MAC cycle, a MAP becomes current at the beginning of the next MAC cycle and remains current until to the beginning of the following MAC cycle. The information in a MAP becomes out of date at the end of the MAC cycle that it describes.

A G.9954v2 network device shall NOT transmit within a MAC cycle for which it does not hold a valid and current (up-to-date) MAP.

A MAP frame shall be identified by a frame with frame type (FT = 0x9) (i.e., frame subtype (FS = 0x01)) and has the structure as in Figure 8-8.

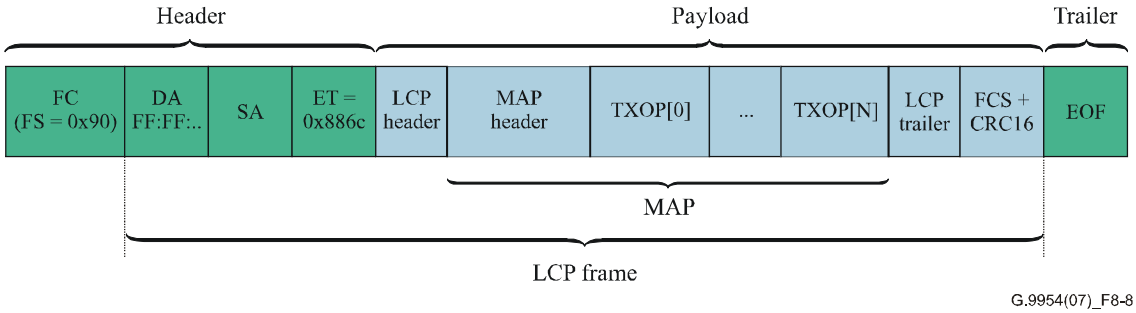

Figure 8-8 – MAP frame structure

The MAP frame shall be encoded as a link layer control protocol (LCP) Frame with the payload components composed of a fixed length MAP header followed by a variable length table of TXOP descriptors. The number of TXOP descriptors in the MAP as well as control and sequence information is encoded in the MAP header. The size of a MAP control frame shall not exceed the size of a standard Ethernet frame (i.e., 1500-byte payload).

For a standard Ethernet frame of payload size 1500 bytes, there are 1480 bytes available for the variable length TXOP table (after removing LCP and MAP headers). This means that the number of TXOP entries in the table will not exceed 370. Given a maximum MAC cycle size of 50 ms, and a minimum frame size of 92.5 µs + 29 µs (GAP), the theoretical maximum number of MAP entries is limited to 50000 / (92.5 + 29) = 411 TXOPs. In practice, the number of TXOPs in a MAP is expected to be significantly less than this theoretical limit, in the order of 10s of entries.

NOTE – The link control frame format is used for convenience in order to allow the MAP to be easily passed up to higher protocol layers (possibly residing in the driver stack behind an IEEE 802.3 standard interface).

The MAP frame shall always be sent to the "broadcast" destination address and the entire contents shall be transmitted using the most robust constellation encoding (2 Mbaud and 2 bits per symbol – PE = 33).

The set of TXOPs planned in the next MAC cycle and their association with devices/flows is described in the TXOP descriptor table. A TXOP is defined as a non-overlapping period of media access time that starts at a time T and extends for a duration of length L. The start time of the first TXOP in the MAP is, by definition, time zero ($T_0 = 0$) and the start time of an arbitrary TXOP (T_N) is calculated by adding the length of the previous TXOP to the start time of the same TXOP ($T_N = T_{N-1} + L_{N-1}$) where T_{N-1} and L_{N-1} represent the start time and length respectively of TXOP *N-1*.

A TXOP descriptor entry is defined in Table 8-2:

Table 8-2 – TXOP descriptor

Field	Bit number	Field Size [bits]	Description
Reserved	31	1	Reserved for future use
Length	30:16	15	Length allocated to associated TXOP in TIME_SLOT units where the size of a TIME_SLOT is determined by a base TICK size multiplied by a constant factor defined in the MAP. A length of zero indicates a sub-burst slot transmission.
Device_ID	15:10	6	Device associated with transmission opportunity Device_ID = 0 indicates a special MAP control directive. Device_ID > 0 identifies the associated device by Device_ID.
GroupType	9:7	3	Group Type identifier Identifies the collection of devices/flows associated with the same group as well as any implicit scheduling policy that applies to the group. For more information on groups, see clause 8.10.3.4.
FSelector	6:6	1	Field selector used to determine the interpretation of the following fields 0 – Priority interpretation 1 – Flow_ID interpretation
When FSelector = 0 (Priority)			
Reserved	5:3	3	Reserved for future use
Priority	2:0	3	Priority associated with transmission opportunity Defined in range 0-7 with priority 7 being the highest priority
When FSelector = 1 (Flow)			
Flow_ID	5:0	6	Identifies the flow associated with a transmission opportunity

Each TXOP descriptor entry in the MAP represents either an assignment of a device/flow to a transmission opportunity or special MAP control directives. Special MAP control directives are recognized by a table entry with a predefined Device_ID (=0). All other TXOP descriptors entries are recognized as assignments of device/flows.

The association of devices/flow entries to TXOPs in the MAP is by TXOP start time. Device/flow entries having the same TXOP start time all share the same TXOP. The order of appearance of entries in the MAP describes the relative scheduling order of devices/flows within the TXOP and scheduling policy may be implicitly defined through the Group_Type parameter or explicitly through a special control directive. A TXOP descriptor with zero length parameter presents the assignment of a device and flow to a sub-burst slot transmission opportunity. A TXOP descriptor with a non-zero length parameter indicates the last entry in the set of TXOP descriptors that describe the composition and attributes of a TXOP.

A device and data flow is uniquely identified in the TXOP descriptor by the (*Device_ID, Flow_ID*) or (*Device_ID,* Priority) pair where *Device_ID* identifies the device at the source of the flow and *Flow_ID* is a unique identifier of a flow within the context of *Device_ID*. In the (*Device_ID,* Priority) form of identification, *Device_ID* has the same semantics as above and *Priority* is the priority associated with the transmission opportunity and represents the lowest traffic priority that may be sent in the transmission opportunity. An entry with *Flow_ID = 0* represents a "wild-card" flow and stands for "any" flow belonging to the associated device i.e., the device/flow pair (N, 0) refers to "any" flow originating from device *N*. The Device/Flow pair value (0, 0) is a special identifier that is used to specify "any" flow from "any" device. A transmission opportunity that is associated with this "wild-card" device and flow may be used by any device including devices that have not yet been assigned a Device_ID (i.e., unregistered devices with DEVICE_ID=0) or by registered devices (Device_ID > 0) that have no other allocated transmission opportunity in the MAP.

Special MAP control directives are used to apply special, non-default, attributes to a TXOP and/or to specify a special scheduling policy to be applied within the TXOP after transmission or after a period of silence. The control directives are defined in Table 8-3.

Table 8-3 – Special MAP control directives

Device_ID	Flow_ID	Name	Semantics
0	0	Unallocated or registration TXOP	Identifies a "wild-card" transmission opportunity available to any flow from any device
	1	Next group	Defines a placeholder assignment of device/flow to TXOP that is used to bandwidth prevent starvation. For more details, see 8.10.3.6.1.
	2	Next MAP	Defines a placeholder sub-burst slot opportunity that can be used to transmit a MAP message before the end of the scheduled transmission cycle.
	3	*Explicit group separator*	Defines an explicit group separator used to apply explicit scheduling semantics to the group. For more information, see clause 8.10.3.5.
	4	*Explicit TXOP separator*	Defines an explicit TXOP separator. Used to apply special non-default attributes to a TXOP. For more information, see clause 8.10.2.
	5-63	–	Reserved for future use

For further information on the semantics of the special MAP control directives, see clause 8.10.

Table 8-4 illustrates an example MAP presented in tabular form. The columns of the table are defined by the parameters: Row_number, Device_ID, Group_type, Flow_ID, Length and TXOP_number where Row_number is a sequential serial number assigned to each row in the table; Device_ID and Flow_ID/Priority together identify the network device and flow associated with a transmission opportunity or a special MAP directive; Group_type identifies the collection of devices/flows associated with the same TXOP as well as any implicit scheduling policy that applies to the TXOP; Length specifies the duration allocated to a TXOP where a non-zero length entry indicates the end of the set of devices and flows sharing the same TXOP; and TXOP_number is a sequential serial number assigned to each sequential TXOP.

Table 8-4 – MAP tabular representation

Row_number	Device_ID	Group_type	Flow_ID/ Priority	Length	TXOP_number
0	1	4	0	L_0	[0]
1	1	5	0	0	[1]
2	2	5	0	0	
3	3	5	0	L_1	
4	1	6	0	0	[2]
5	2	6	0	0	
6	3	6	0	0	
7	4	6	0	L_2	
8	0	4	0	L_3	[3]

For example, the MAP in Table 8-4 contains four distinct TXOPs starting at times $T_0 = 0$, $T_1 = L_0$, $T_2 = T_1 + L_1$ and $T_3 = T_2 + L_2$ and having lengths L_0, L_1, L_2 and L_3 respectively. In addition, the first TXOP in the MAP, TXOP[0], defines a media access period allocated exclusively to the master for transmission of the MAP; TXOP[2] is a TXOP shared amongst devices with Device_ID = 1, 2 and 3; TXOP[3] is another TXOP shared amongst devices with Device_ID = 1, 2, 3 and 4; and finally TXOP[4] is a TXOP that is unassigned and can be used by any network device.

The first entry in the TXOP table shall be assigned to the master and shall be used for the transmission of the (next) MAP control frame itself.

For further details on the structure of a MAP control frame, see 11.13.1.

8.9.1 TXOP timing

The master shall plan the start-time of TXOP[N] equal to the start-time of TXOP[N – 1] plus the length of TXOP[N – 1].

The length of each TXOP shall include the media time required to transmit actual frame symbols as well as any required *inter-frame GAPs* needed to separate consecutive frame bursts. The length of the inter-frame GAP used by the master when calculating the length of TXOPs in the MAP shall be signalled to endpoint nodes in the MAP frame (*MAP_IFG).*

NOTE – Normally, a TXOP ends with a MAP_IFG. However, long TXOPs may contain intermediate MAP_IFGs to separate bursts within a TXOP.

If a TXOP has a fixed TXLimit (i.e., TXOP end-time is rigid and non-expandable), a G.9954v2 device shall not transmit within a TXOP later than $TXOP_{LatestTime} = TXOP_{StartTime} + TXOP_{Length} - MAP\_IFG$ (assuming that the next TXOP is assigned to a different device).

Two consecutive TXOPs with the same TXOP tuple assignment may logically be considered a single TXOP of extended length where the extended length equals the sum of the lengths of the two individual TXOPs. This supports TXOPs with a length greater than the limit imposed by the TXOP length field in the MAP. In this case, a MAP_IFG is not required between the two consecutive TXOPs and a transmission may extend across the boundary between them.

The transmission time line is divided into time slots of duration Time_slot where the duration of a Time_slot shall be 500 ns duration times a constant factor defined in the MAP. All TXOPs shall start on a Time_slot boundary. The master shall round up the length of TXOPs to integral numbers of Time_slots when calculating the MAP for a MAC cycle.

8.9.2 MAC timing synchronization

Network nodes shall synchronize to the master clock reference through the MAP control frame. All timing references specified by the master shall be made relative to the start of the first symbol of the preamble of the MAP frame. This reference point represents offset zero within the MAC cycle.

The current offset within the MAC cycle is reflected in a *synchronous clock counter*. The synchronous clock counter is reset by the arrival of the MAP and counts the progression of Time_slots relative to the start of the MAC cycle. The synchronization of transmission timing to the start of a TXOP shall be performed using to the synchronous clock counter.

8.9.3 Propagation delay compensation

Different devices on the network may receive the MAP at different times due to propagation delay. To account for the differences in the propagation delay between stations, the master shall plan for an inter-frame gap (MAP_IFG) in each TXOP that will guarantee, in the worst case, a CS_IFG µs gap between the end of one planned transmission and the start of the next planned transmission accounting for the largest deviation from scheduled TXOP time caused by *propagation delay*.

NOTE 1 – The actual IFG perceived at each station may vary depending on when it received the MAP relative to the master's clock and when the transmission was scheduled by the master. By planning for MAP_IFG and having every station guarantee at least CS_IFG, the effects of propagation delay are bounded and the cycle length will not drift.

The relationship between the minimum guaranteed IFG (CS_IFG), the actual IFG "perceived" by a device and the planned IFG (MAP_IFG) used in the MAP is defined as follows:

$$CS\_IFG \le IFG \le MAP\_IFG + 2 \times PD$$

where the parameters appearing in the inequality are described in Table 8-5.

Table 8-5 – IFG parameters

Parameter	Description
IFG	The actual inter-frame gap (IFG) "perceived" by a G.9954v2 device.
CS _ IFG	The inter-frame gap (IFG) required by the PHY to detect the end of one burst and the beginning of the next burst.
MAP _ IFG	The inter-frame gap used by the master in calculations of TXOP length in the MAP and defined by: $MAP\_IFG = CS\_IFG + 2 \times PD$
PD	Maximum propagation delay approximated by transmission at the speed of light (i.e., 300 m equals 1 µs delay).

The master shall plan for a MAP_IFG gap between bursts when calculating TXOP timing and length and shall advertise the value of MAP_IFG used in its calculations in the MAP. An endpoint device (any endpoint device including the master itself) shall guarantee to terminate its transmission at least MAP_IFG µs before the end of the TXOP.

Figure 8-9 illustrates the variation in perceived IFG from the perspective of different devices in the network in the presence of the effects of propagation delay.

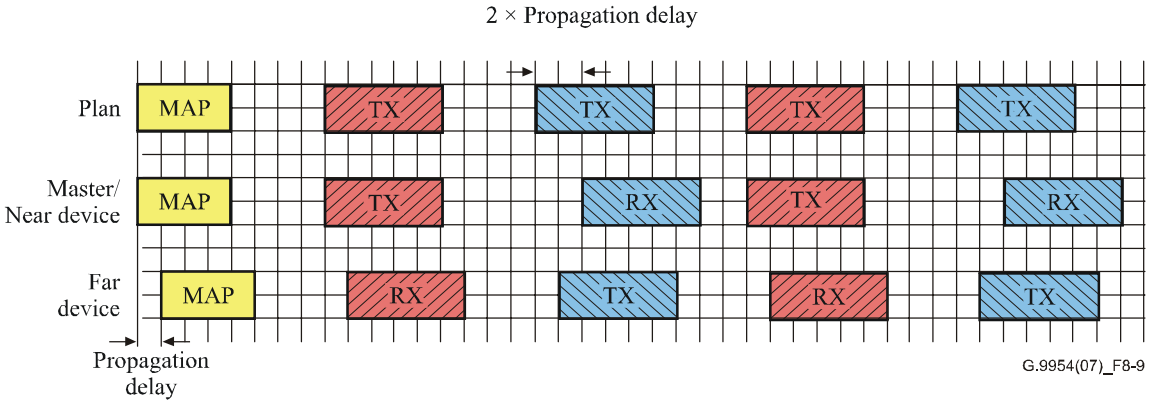

Figure 8-9 – Propagation delay

NOTE 2 – This mechanism is sufficient to eliminate the need to have non-master devices actually synchronize with the master clock reference. However, for some applications (e.g., voice) it may be important to synchronize with the master clock reference in order to synchronize sampling rates at upper layers. For this purpose, the master distributes its clock using a Link-Layer Timestamp Reporting message; see clause 11.17.

8.10 Transmission opportunities (TXOPs)

The internal structure of a MAC cycle is illustrated in Figure 8-10. It shows an example MAC cycle composed of transmission opportunities (TXOPs) of different types. The following types of TXOPs are defined.

- Contention-free TXOP (CFTXOP) – A TXOP allocated to a dedicated (single) network device.
- Contention TXOP (CTXOP) – A TXOP for which contention-based access is defined amongst a group of network devices.
- Unallocated TXOP (UTXOP) – An unallocated TXOP is a type of contention-based TXOP where any network device may transmit if the media is sensed as being idle.

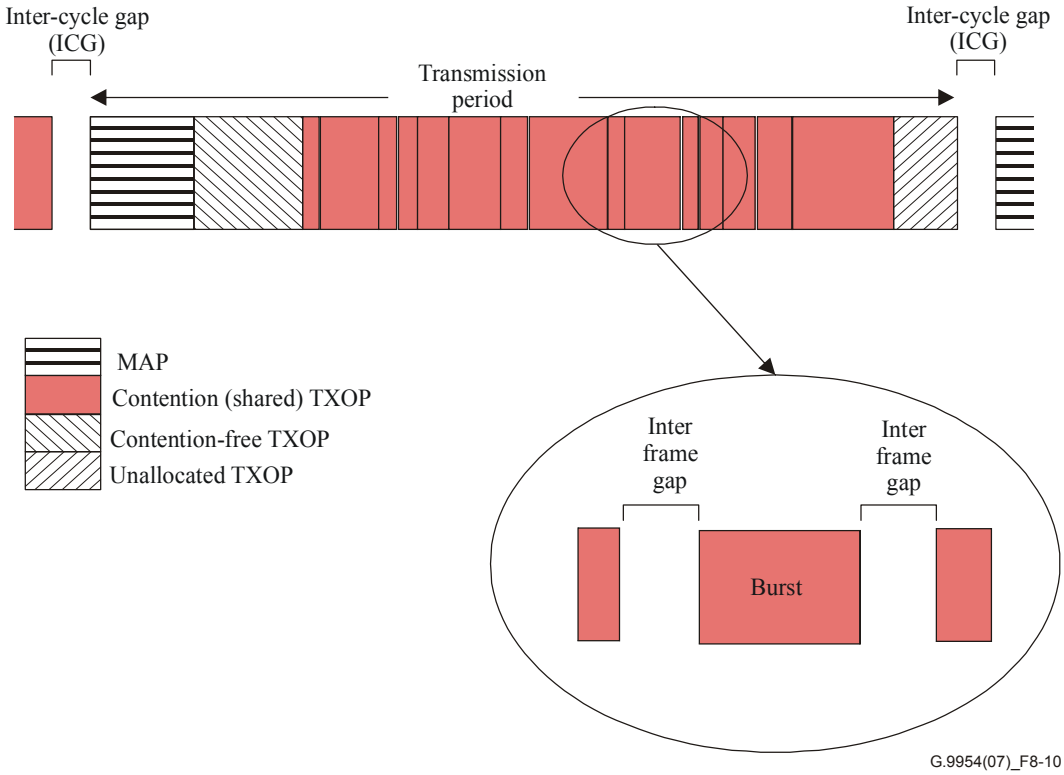

Figure 8-10 – MAC cycle structure

The *MAP* shall be sent at the beginning of each MAC cycle in the first TXOP of the cycle (as described in the previous MAP). The TXOP used for the transmission of the MAP is, by definition, a contention-free TXOP (CFTXOP) and allocated exclusively to the master.

The master shall plan media access during a MAC cycle by dividing the available media access time within the MAC cycle time into TXOPs. The master shall allocate contention-free TXOPs (CFTXOPs) and contention TXOPs (CTXOPs) in accordance with service requirements of network devices. The media access time remaining (if any) after all TXOPs have been allocated to specific devices, services or groups shall be assigned by the master as *UTXOPs.* These TXOPs may be used by any device, on a pure contention basis, for the transmission of non-scheduled traffic for which a TXOP has not been explicitly assigned.

NOTE – Bandwidth allocated to a network device for transmission may be spread out over a number of TXOPs within the MAC cycle. Although the resource management and scheduling algorithms in the master should attempt to concentrate allocated bandwidth together (in order to reduce the possible number of bursts) it may be necessary to spread the allocation throughout the cycle in order to meet QoS constraints. This may particularly be the case for CBR flows. Similarly, unallocated TXOPs may be scattered throughout the MAC cycle. The placement and length of TXOPs within a MAC cycle are all master scheduler decisions and as such beyond the scope of this Recommendation. Media access within CTXOPs shall be performed using the collision-free media access method based on sub-burst slots. Media access within UTXOPs shall be performed using contention-based media access methods based on carrier-sensing alone. Collisions which may occur within UTXOPs are neither detected nor resolved and it should be the responsibility of upper protocol layers to handle any packet loss.

8.10.1 Association of devices and flows to TXOPs

Devices/Flows are associated to a TXOP by the master. The association is performed based on master scheduler decisions. Such decisions may, for example, be initiated by LCP protocols that signal the setup of a new flow or the addition of a new class of service. The association, if it can be made, is reported back to the device at the source of the flow within the MAP.

All devices are implicitly associated with an unallocated TXOP (UTXOP) (if one exists). A device may transmit data of any type within the UTXOP.

A device shall only transmit within a TXOP to which it is associated.

8.10.2 TXOP separator

A TXOP is described in the MAP by a set of TXOP descriptor entries that terminate with an entry that has a non-zero length where the length specifies the duration of the TXOP. Typically, the entry containing the non-zero length field is a regular entry that defines an association between device/flow and TXOP. This entry represents an implicit TXOP separator, and default TXOP attributes shall be assumed. For default TXOP attributes, see Table 8-6.

The explicit TXOP separator is used to assign special non-default attributes to a TXOP. The length field in the explicit TXOP separator shall contain a non-zero length field but in addition it may contain an attribute specification that is different from the default value.

An explicit TXOP separator shall be identified by a TXOP descriptor entry in the MAP with a (Device_ID, Flow_ID) tuple value of (0, 4). Table 8-6 describes the structure and attributes of the explicit TXOP separator.

Table 8-6 – Explicit TXOP separator

Field	Bit number	Field size [bits]	Description
Reserved	31:31	1	Reserved for future use. Set to zero by transmitter and ignored by receiver
Length	30:16	15	Length of the TXOP in TIME_SLOT units where length of a TIME_SLOT is determined by the base TICK size multiplied by a constant factor defined in the MAP
DeviceID	15:10	6	= 0
TXLimit	9:9	1	Transmission limitations that apply within the TXOP 0 = All transmissions shall end before the end of the TXOP (default). 1 = All transmissions shall start before the end of the TXOP but may end after the scheduled end of the TXOP.
TXOPType	8:6	3	The type of the TXOP 0 – CFTXOP 1 – CTXOP with 8-µs-wide sub-burst slots 2 – CTXOP with 16-µs-wide sub-burst slots (default) 3 – CTXOP with 32-µs-wide sub-burst slots 4 – CTXOP with 64-µs-wide sub-burst slots 5-7 – Reserved for future use
FlowID	5:0	6	= 4

The TXLimit field is used to specify a TXOP that has an elastic end limit that may expand beyond the scheduled limit of the TXOP. The amount that the TXOP may expand is bounded by the size of the longest transmission within the TXOP since all transmissions within a TXOP shall start before the end of the TXOP.

It should be noted that a TXOP with an elastic end-limit will cause the delay of the next TXOP immediately following. The delay caused by the expansion of an elastic TXOP shall not normally be additive. This means that if a TXOP expands beyond its end limit by a time duration ΔT, due to a

transmission, the start of the succeeding TXOP shall be delayed by the same duration ΔT and the length of the TXOP shall be shortened by the same amount. The only exception to the above is in the case of a transmission that exceeds beyond the end of the MAC cycle. In this case, the TXOP used to transmit the MAP shall retain its original length.

By default, a TXOP shall have a TXLimit = 0.

Figure 8-11 illustrates the concepts described above.

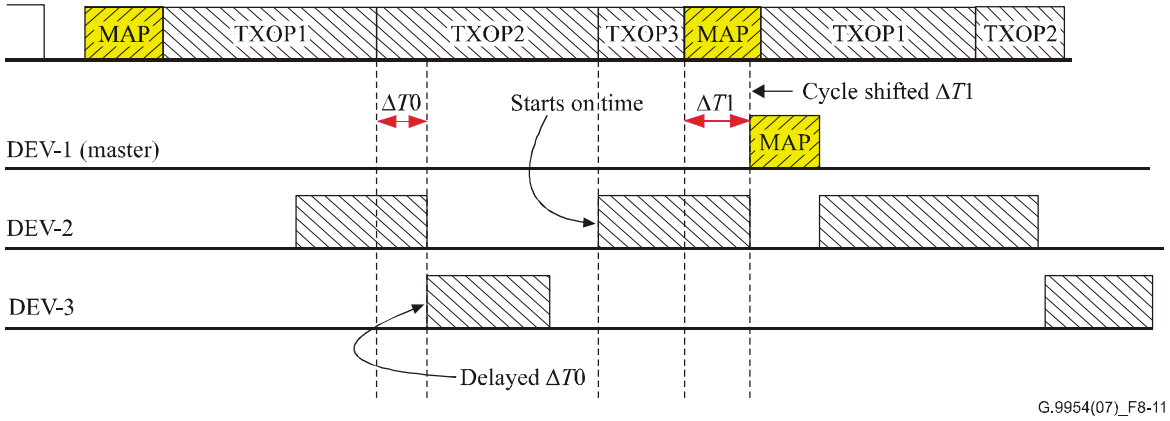

Figure 8-11 – TXOPs with elastic end-limits

8.10.3 TXOPs and sub-burst slots

To provide collision-free media access to shared TXOPs, a TXOP is divided into sub-burst slots where each sub-burst slot represents an opportunity for a device to start transmitting. Sub-burst slots are short in time (SBS_SLOT µs), and transmissions must start at the beginning of a sub-burst slot or else wait until the next occurrence of the same sub-burst slot.

Sub-burst slots are organized into groups with each sub-burst slot identified by its ordinal position within the group sequence. Each sub-burst slot is also associated with a network device or flow, identified by Device_ID and Flow_ID, which identify the device and flow which has the permission to transmit within that sub-burst slot. A TXOP is composed of one or more sub-burst slot groups which, together with scheduling attributes assigned to the groups, specify an ordered sequence of sub-burst slots and sub-burst slot assignments which may repeat throughout the TXOP. The order of appearance of sub-burst slots and the manner in which the sequence repeats vary based on different scheduling policies. For example, one simple scheduling policy supported is the "round-robin" policy whereby the sub-burst slot sequence repeats regularly until the end of the TXOP (see Figure 8-12).

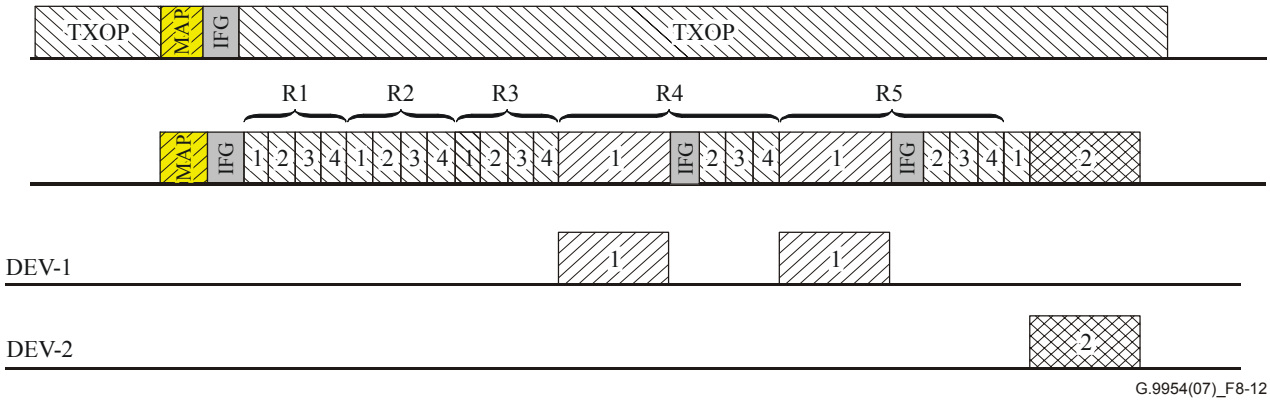

Figure 8-12 – Round-robin sub-burst slot scheduling policy

Sub-burst slots divide a TXOP into a grid of transmission opportunities. The grid starts at the beginning of a TXOP and the grid is rebuilt after each transmission.

Devices that share a TXOP shall track the passage of sub-burst slots on the line and guarantee to transmit only within their assigned sub-burst slots. Collisions are avoided by assuring that each device is assigned a distinct set of sub-burst slots and by synchronizing network devices in time and in scheduling policy. It is the responsibility of the master to perform this synchronization through the distribution of the MAP.

The media access method based on sub-burst slots used in shared TXOPs is illustrated in Figure 8-12 and shows a timing diagram for an example transmission cycle. The transmission cycle is initiated with the publication of the MAP and contains a TXOP shared by multiple network devices. The shared TXOP is composed of multiple scheduled sub-burst slots with each sub-burst slot assigned to a unique device, identified by Device_ID, and representing an opportunity for the assigned device to initiate a transmission. Collectively, the sub-burst slots form a grid of transmission opportunity start times. Initially, the grid is calculated relative to the beginning of the shared TXOP. The grid pattern repeats regularly throughout the TXOP providing multiple opportunities for devices to transmit. A network device associated with a sub-burst slot may choose to act on the opportunity to transmit or it may choose to pass on as illustrated in Figure 8-13.

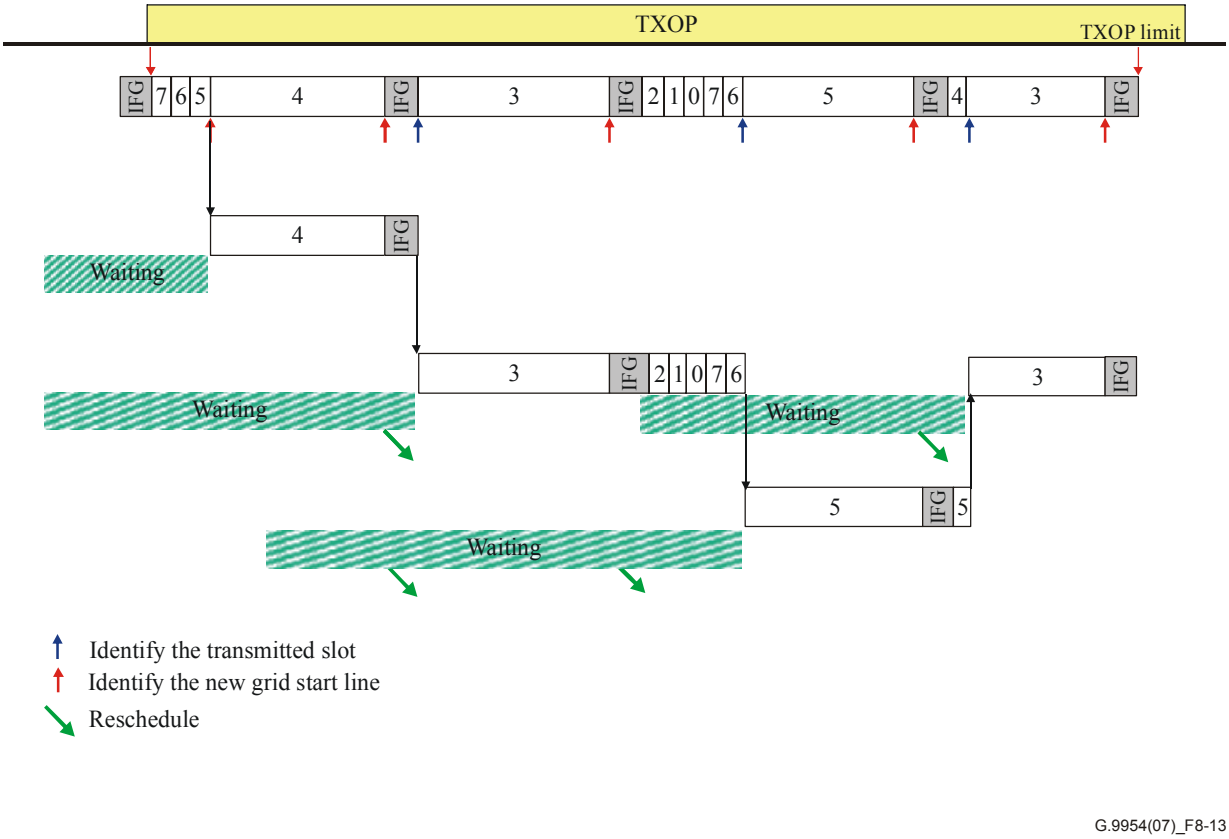

Figure 8-13 – CSMA media access method

8.10.3.1 Sub-burst slot size and timing

Sub-burst slot start times are calculated relative to a single time base T_{base}. The time base T_{base} shall initially set to the start of the TXOP and is adjusted at the end of each transmission on the media. The adjusted time base T'_{base} is set to the time $T_{end} + IFG$ where T_{end} is the timestamp at the end of the last transmission and IFG is the size of the inter-frame gap.

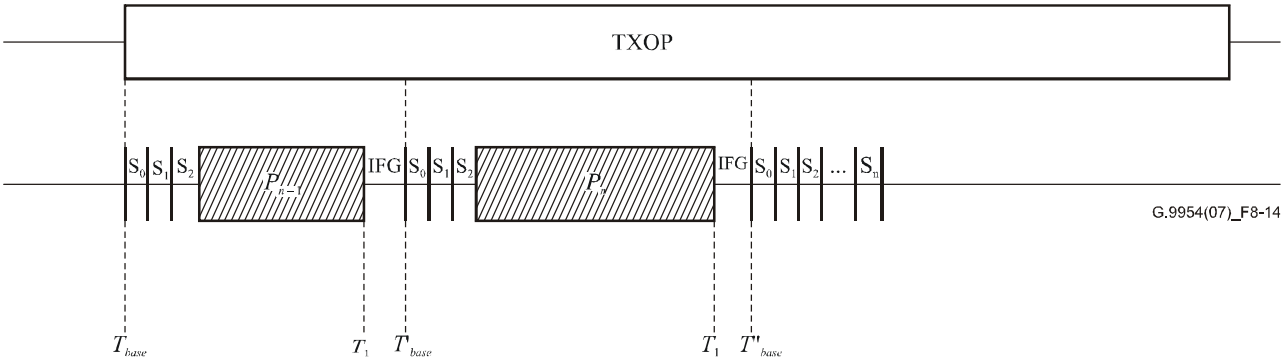

Figure 8-14 – Sub-burst slot timing

The width of a sub-burst slot is SBS_slot µs which accounts for worst-case variances related to clock accuracy, propagation delays and hardware implementation delays.

Valid values for SBS_slot widths are 8, 16, 32 or 64 µs. The actual sub-burst slot width used in a TXOP shall be published by the master in the MAP. All G.9954v2 devices shall support at least 16-µs-wide SBS_slots and shall adapt their SBS_slot timing parameters in accordance with the value published in MAP. Support for 8-µs SBS_slot timing is optional. A device shall publish the SBS_slot timing parameters that it supports during registration and the master shall switch the SBS_slot timing used in a TXOP to the lowest common denominator amongst network devices.

8.10.3.2 Sub-burst slot transmissions

Sub-burst slots scheduled in the MAP within a TXOP form a grid of transmission opportunity start times that start at the beginning of the TXOP. Each sub-burst slot S*n* in the grid serves as a placeholder, reserving the opportunity for its associated network device or data flow to transmit at the time which the sub-burst slot occupies in the sequence of sub-burst slots in the grid.

For example, as shown in Figure 8-15, the first opportunity to transmit is reserved for the network device or data flow associated with sub-burst slot S1. The network device or data flow associated with S1 may act on this opportunity to transmit, or pass on it.

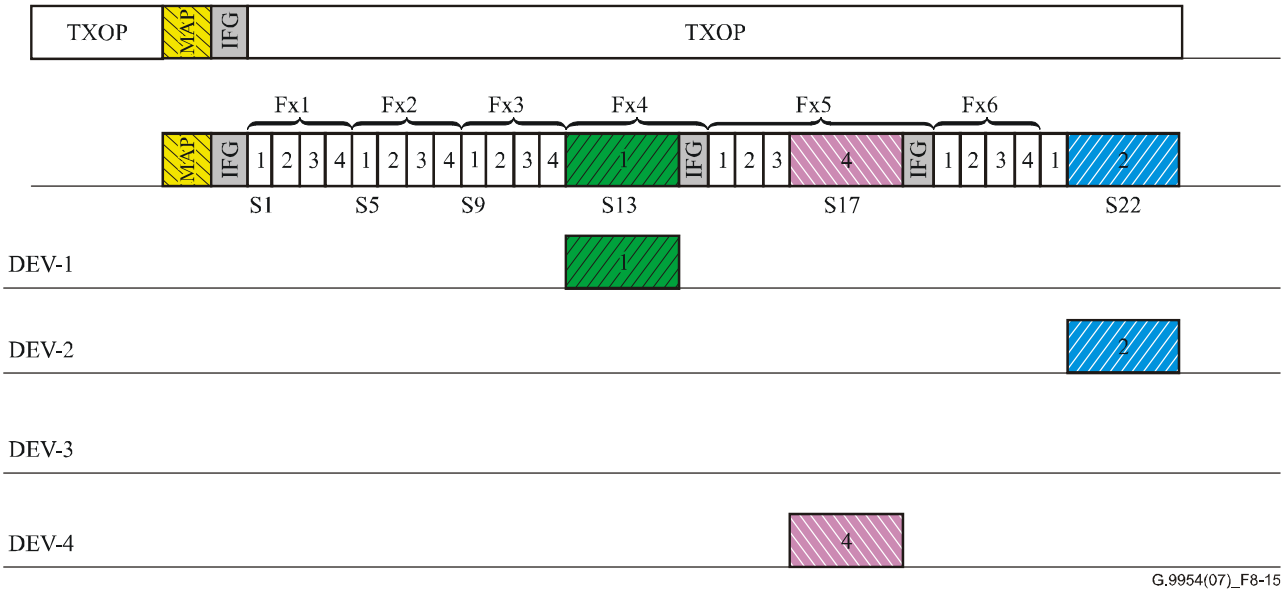

Figure 8-15 – Sub-burst slot transmissions

In order for a network device to act on an opportunity to transmit, it shall initiate its transmission at the beginning of its associated sub-burst slot. The window for acting upon the opportunity to initiate a transmission may be the first 4 μs (TX_ON) of the sub-burst slot.

In the event that the transmission opportunity provided by a sub-burst slot S*n* is not utilized by the network device or data flow associated with it, the network device or data flow associated with the next sub-burst slot S(n + 1) in the grid sequence shall then be given the opportunity to transmit.

For example, from Figure 8-15, it can be seen that no transmission occurs during the TXOP until the fourth transmission opportunity allotted to network device DEV-1. As shown in the transmission activity diagram above for network device DEV-1, network device DEV-1 passes on the transmission opportunities presented by the sub-burst slots S1, S5 and S9 numbered "1", and transmits for the first time during sub-burst slot S13. During S13, network device DEV-1 begins transmitting, and occupies the transmission medium until its transmission is complete.

When utilized for a transmission, the sub-burst slot assigned to a device or flow shall expand to allow the completion of the transmission begun by the network device during the sub-burst slot. The length to which the sub-burst slot may expand may be unlimited, or it may be limited depending on the attribute assigned to the containing TXOP. The end of a shared TXOP may constitute a limit to the expansion of sub-burst slot during a transmission. Alternatively, a transmission may extend the length of a shared TXOP.

When transmissions occur, the sub-burst slot grid aligner function in each G.9954v2 network device on network shall recalculate the timing of the sub-burst slot grid so that each network device knows at what time its next transmission opportunity is scheduled. While the order of the sub-burst slots in the grid is known to each network device through the advertised MAP, the timing of the grid is changed due to, and in direct accordance with, the duration of the transmissions which occur.

8.10.3.3 Sub-burst slot groups

Sub-burst slots may be organized in modular groups, and these modular group units may be assembled into group sequences of one or more groups within a TXOP and scheduled in the MAP. A group may consist of a particular sequence of sub-burst slots, each of which is associated with a particular network device or data flow. Each group may also be associated with a particular group type. Group types dictate scheduling policy and influence the sub-burst slot grid pattern that applies after a transmission within the group type. There exists a set of predefined scheduling schemes for the repetition, rotation or omission of a group or sequence of groups in a TXOP. Unless explicitly specified, the scheduling scheme associated with a group are dictated by group type.

Group affiliation, i.e., which sub-burst slots comprise a group of which type, is encoded in the MAP via a particular value assignment of the group type parameter. A sequence of sub-burst slots entries in the MAP with the same group type belong to the same group. A new group of sub-burst slots begins with a change in the value of the group type. Alternatively, a group may be explicitly defined using an explicit group separator directive, as described in clause 8.10.3.5.

8.10.3.4 Group types

The group type parameter is used to organize sub-burst slots into groups within a TXOP and to define scheduling semantics to be applied within a group and between groups following periods of silence or transmissions.

The set of predefined group types are defined in Table 8-7.

Table 8-7 – Group types

Group type	TX policy	Group scheme	Description
0	STATE	LEGACY	Reserved for legacy G.9954 TXOP scheme.
1			Reserved
2			Reserved
3			Reserved
4	EDGE	FIXED	Sub-burst slot media access method with fixed scheme
5	EDGE	ROTATED	Sub-burst slot media access method with round-robin scheme
6	EDGE	REPEATED	Sub-burst slot media access method with repeating round-robin scheme.
7			Reserved

The TX policy field specifies timing requirements on transmissions of data within a TXOP. The "state" TX policy indicates that transmission of data within such a TXOP may occur at any time during the TXOP while the "edge" TX policy associated with a TXOP indicates that transmission of data within this type of TXOP must occur at the beginning of a sub-burst slot boundary within TX_ON μs. All TXOPs supporting sub-burst slotted media access use an "edge" TX policy.

Scheduling semantics defines the grid pattern of sub-burst slots defined after the occurrence of one of the following two events:

1) ***Proceed*** – Event generated following the passage of ALL sub-burst slots in a group without the occurrence of a transmission within the group.

2) ***Abort*** – Event generated following the transmission of a burst within a group.

Scheduling semantics for the *Proceed* event specify the *next* group that follows the *current* group if none of the sub-burst slots in the *current* group are used for transmissions. Scheduling semantics for the *Abort* event specify, in addition to which group, control is passed following a transmission, also the order of appearance of sub-burst slots in the current group in which the transmission occurred after the transmission.

In general, following an *Abort* event, control returns to the first group in the group sequence. Following a proceed event, control passes to the next group in the sequence. If the current group is the last group in the sequence, control returns to the first group in the sequence if the last group is of group type Rotated or Fixed and remains in the current group if the last group is of group type Repeated.

Note that the default behaviour defined above for the transition of control between groups in response to the *Proceed* and *Abort* events may be explicitly overridden and using the Explicit Group Separator control directive. In this case, control can be passed explicitly to any group in the sequence. For further information on the Explicit Group Separator control directive see clause 8.10.3.5.

The remainder of this clause specifies, in further detail, the scheduling semantics for the Fixed, Rotated and Repeated group types in terms of their response to the Proceed and Abort events.

8.10.3.4.1 Fixed Group types

In a Fixed Group scheme, a transmission within the fixed group (Abort event) constitutes an interruption of the continuity of the current group sequence. After the transmission, group control returns to the first group in the sequence and the scheduled order of sub-burst slots in the Fixed group is reset to the first sub-burst slot in the group.

When the sequence of sub-burst slots in a Fixed group passes without the occurrence of a transmission (Proceed event), control is passed onto the next group in the group sequence or control returns to the first group if the current group is the last group in the sequence.

For example, assume the existence of a TXOP containing a single group Fx, a fixed type group comprising the sequence of sub-burst slots numbered "1,2,3,4", where each sub-burst slot is associated with a network device with the same device number (DEVICE_ID). While no transmissions occur (i.e., on the Proceed event), the Fx pattern repeats itself regularly using the same sub-burst slot number ordering. This implies the following sub-burst slot pattern:

[1,2,3,4] $P\rightarrow$ [1,2,3,4] $P\rightarrow$ [1,2,3,4] $P\rightarrow$ [1,2,3,4] $P\rightarrow$…

where the bracketed numbers represent repetition of the sub-burst slots in a group and $P\rightarrow$ and $A\rightarrow$ represent the Proceed and Abort event transitions respectively.

Assuming a transmission by device number 1 in the second repetition of group Fx, Fx2 and in the fourth repetition of Fx, Fx4, the sub-burst slot grid pattern is defined as follows:

[1,2,3,4] $P\rightarrow$ [**1**] $A\rightarrow$ [1,2,3,4] $P\rightarrow$ [**1**] $A\rightarrow$ [1,2,3,4] $P\rightarrow$…

where bold numbers represent actual transmissions by the device with the specified device number.

8.10.3.4.2 Rotated Group types

In a Rotated Group scheme, following a transmission within the rotated group (Abort event) the group's sub-burst slot sequence remains uninterrupted and control returns to the first group in the sequence.

In case of the Proceed event, group control behaviour shall be the same as for Fixed Groups.

For example, assume a TXOP containing a single rotated group R composed of sub-burst slots "1,2,3,4" and assigned to devices with the same device numbers respectively. In this scheme, while no transmissions occur, the sub-burst slot pattern is as defined for Fixed groups as follows:

[1, 2, 3, 4] $P\rightarrow$ [1, 2, 3, 4] $P\rightarrow$ [1, 2, 3, 4] $P\rightarrow$ [1, 2, 3, 4] $P\rightarrow$…

Assuming a transmission by device number 1 in the second repetition of the group R, R2 and fourth repetition of R, R4, the sub-burst slot grid pattern is defined as:

[1, 2, 3, 4] $P\rightarrow$ [**1**] $A\rightarrow$[2, 3, 4, 1] $P\rightarrow$ [2, 3, 4, 1] $P\rightarrow$ [**1**] $A\rightarrow$ [2, 3, 4, 1] $P\rightarrow$…

8.10.3.4.3 Repeated Group Types

In a Repeated Group scheme, following a transmission within the repeated group (Abort event) the group's sub-burst slot sequence remains uninterrupted and control returns to the first group in the sequence.

In the case of a Proceed, group control remains in the current group if and only if the Repeated group is the last group in a group sequence. If the Repeated group is NOT the last group in the sequence, a Repeated Group behaves the same as a Rotated group.

The Repeated Group type may be used together with a Rotated Group type in order to simply mark the division of sub-burst slots into separate groups. The group boundary is marked by the point of transition from one Group Type to the other.

For example, assume a TXOP containing two groups Rotated (Ro) and Repeated (Rp) groups in that order and composed of sub-burst slots "1,2,3,4" and "5,6" in each group and assigned to devices with the same device numbers respectively. In this scheme, while no transmissions occur, the sub-burst slot pattern is defined as follows:

[1, 2, 3, 4] $P\rightarrow$ [5, 6] $P\rightarrow$ [5, 6] $P\rightarrow$ [5, 6]…

Assuming a transmission by device number 5 in the first and second repetition of the group Rp, the sub-burst slot grid pattern is defined as:

[1, 2, 3, 4] *P*→ [**5**] *A*→ [1, 2, 3, 4] *P*→ [6, **5**] *A*→ [1, 2, 3, 4] *P*→ [6, 5] *P*→ [6, 5] *P*→…

Note that if the order of the Rotated (Ro) and Repeated (Rp) groups in the TXOP is reversed, the sub-burst slot pattern would be the same as for two consecutive Rotated groups. Using the same example as above, while no transmissions occur the sub-burst slot pattern would be:

[5, 6] *P*→ [1, 2, 3, 4] *P*→ [5, 6] *P*→ [1, 2, 3, 4] *P*→…

Similarly, in case of a transmission by device 1 in the first and second repetition of Ro, the sub-burst slot pattern would be:

[5, 6] *P*→ [**1**] *A*→ [5, 6] *P*→ [2, 3, 4, **1**] *A*→ [5, 6] *P*→ [2, 3, 4, 1] *P*→ …

8.10.3.5 Group separator

The explicit group separator is a control directive entry in the MAP that is used to explicitly mark the end of a set of member sub-burst slots comprising a group and to assign special attributes and controls to the group. An explicit group separator is identified by a MAP entry with a (*Device_ID, Flow_ID*) tuple value of (0, 3).

The explicit group separator is composed of the data items shown in Table 8-8.

Table 8-8 – Explicit group separator

Field	Bit number	Field size [bits]	Description
Reserved	31:31	1	Reserved for future use. Shall be set to zero by transmitter and ignored by receiver
NumSBslots	30:24	7	Number of sub-burst slots in group. May be greater than or less than the actual number group members defined in the MAP. A value of zero indicates that the number of sub-burst slots in the group is equal to the number of slots defined in the MAP.
ProceedGroup	23:20	4	Group number of Next group to which control is passed upon occurrence of the Proceed event. A value of zero indicates default semantics for the Group Type.
AbortGroup	19:16	4	Group number of Next group to which control is passed upon occurrence of the Abort event. A value of zero indicates default semantics for the Group Type.
DeviceID	15:10	6	= 0
Reserved	9:6	4	Reserved for future use. Shall be set to zero by transmitter and ignored by receiver
FlowID	5:0	6	= 3

When the explicit group separator is used it shall be included in the MAP table after a row pertaining to the final (last) group member of a group where each group member represents a specification of device and data flow that is associated with the group.

8.10.3.5.1 NumSBslots

Using the NumSBslots field, it is possible to define a group whereby the number of sub-burst slots in the group is less than or greater than the number of members in the group. When the number of sub-burst slots is less than the number of group members, only a portion of the group members can be serviced during each repetition of the group and it will take more than one repetition of the group to service each group member. For example, assume a group containing five-group members identified by the IDs 1, 2, 3, 4, and 5 and a group composed of only three sub-burst slots. In this case, the pattern of sub-burst slots is defined:

$$[1, 2, 3]\ P\rightarrow [4, 5, 1]\ P\rightarrow [2, 3, 4]\ P\rightarrow [5, 1, 2]\ldots$$

Alternatively, if the number of sub-burst slots available is greater than the number of group members, then a group member may be assigned more than one sub-burst slot each repetition of the group. For example, assume a group containing three group members identified as 1, 2, and 3 and a group composed of five sub-burst slots. In this case, the sub-burst slot pattern formed would be as follows:

$$[1, 2, 3, 1, 2]\ P\rightarrow [3, 1, 2, 3, 1]\ P\rightarrow [2, 3, 1, 2, 3]\ P\rightarrow [1, 2, 3, 1, 2]\ldots$$

8.10.3.5.2 Proceed and Abort Group number

Using the ProceedGroup and AbortGroup numbers, it shall be possible to explicitly specify to which group control is passed in response to each of the events. The group to which control is passed is identified by group number. Group numbers are assigned in the range 1..15 where group number 1 represents the first group encountered in the TXOP and group number *N + 1* represents the next group encountered in the TXOP following group number *N*.

For example, it is possible to define the scheduling scheme illustrated in Figure 8-16 using the MAP described in Table 8-9.

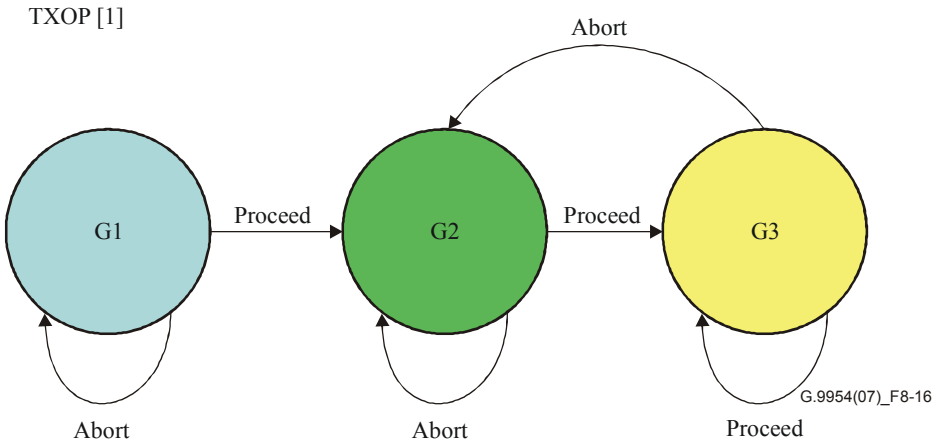

Figure 8-16 – Explicit group transitions

Table 8-9 – Explicit Proceed/Abort controls

#	Device ID	Group Type	Flow Id	Length			TXOP
2	1	5	0	0			[1]
3	2	5	0	0			
4	3	5	0	0			
5	0	–	3	**AbortGroup**	**ProceedGroup**	**NumSBslots**	
				1	2	0	
6	4	6	0	0			
7	5	6	0	0			
8	0	-	3	**AbortGroup**	**ProceedGroup**	**NumSBslots**	
				2	3	0	
9	6	5	0	0			
10	7	5	0	L1			
11	0		3	**AbortGroup**	**ProceedGroup**	**NumSBslots**	
				2	3	0	

8.10.3.6 Special sub-burst slot types

Usually, a sub-burst slot is assigned to a uniquely identifiable device (and flow) using the (Device_ID, Flow_ID) or (Device_ID, Priority) tuple. However, sometimes it may be required to assign a sub-burst slot not to a single device but rather to a set of devices that may use the sub-burst slot in a well-defined order or based on well-defined semantics.

The following clause describes these special types of sub-burst slots.

8.10.3.6.1 "Next Group" sub-burst slot type

The Next Group sub-burst slot is a sub-burst slot type, identified by the (*Device_ID, Flow_ID*) tuple value (0, 1), and used as a mechanism for starvation prevention.

Network nodes may suffer from starvation of available transmission opportunities if, for example, they belong to a group having a low priority for the allotment of transmission opportunities, and other groups having a higher priority for the allotment of transmission opportunities are consuming all the available sub-burst slots.

To prevent starvation, the *Next Group* sub-burst slot type is used to provide a priority upgrade scheme that allows sub-burst slots from lower-priority groups to be temporarily upgraded into a higher priority group. The *Next Group* sub-burst slot is a "wild-card" transmission opportunity that represents a place-holder for sub-burst slots from the next group. Sub-burst slots from the next group are assigned to the *Next Group* sub-burst slot based on their current scheduling order.

For example, assume a TXOP composed of two groups G1 and G2 with both groups defined a round-robin (rotated) scheduling scheme. Assume that G1 contains sub-burst slots assigned to devices "1, 2, 3, N" where N represents the *Next Group* opportunity and G2 contains sub-burst slots assigned to devices "4, 5". The sub-burst slot grid pattern when no devices are transmitting would be as follows:

[1, 2, 3, 4] *P*→ [5, 4] *P*→ [1, 2, 3, 5] *P*→ [4, 5] *P*→ [1, 2, 3, 4] ….

Now assume that devices 3 and 4 are transmitting continuously in every available sub-burst slot assigned to them. The sub-burst slot grid pattern would appear as follows:

[1, 2, **3**] *A*→ **[4]** *A*→ **[**1, 2, **3**] *A*→ **[**5, 1, 2, **3**] *A*→ **[4]** *A*→ **[**1, 2, **3**] *A*→….

Note that without the *Next Group* opportunity, the grid pattern would be as follows and devices "4" and "5" would suffer from starvation:

[1, 2, **3**] *A*→ **[**1, 2, **3**] *A*→ **[**1, 2, **3**] *A*→ **[**1, 2, **3**] *A*→**[**1, 2, **3**] *A*→….

8.10.3.6.2 "Next MAP" sub-burst slot type

While a long transmission cycle provides certain advantages with respect to efficient bandwidth usage, such as minimizing the usage of bandwidth for MAP advertisements over the network, the disadvantage of a long, uninterrupted transmission cycle is that the network is committed to a particular media access plan for the entire cycle. This imposes a certain limitation on the network and making it slow to react to changes in the media access plan.

The *Next Map* sub-burst slot opportunity solves this limitation by providing explicit transmission opportunities to the master to schedule the transmission of a new MAP. The transmission of a new MAP during an existing transmission cycle has the effect of interrupting the current TXOP and transmission cycle and starting a new one based on a new media access plan. Since the media access plan published in the MAP only takes effect in the following cycle, it is possible to send MAPs in two consecutive *Next MAP* opportunities.

The *Next Map* sub-burst slot shall be identified in the MAP by the (*Device_ID, Flow_ID)* tuple value (0,2). More than one *Next Map* sub-burst slot may be scheduled per-cycle.

8.10.3.6.3 Registration opportunity

To join a G.9954v2 network, devices shall register with the master by sending a *registration request* message. The process of registration results in the assignment of a Device_ID to the new device and the subsequent assignment of transmission opportunities to the device.

Since a new device joining the network does not yet have a transmission opportunity assigned to it, it shall send the *registration request* message in a transmission opportunity assigned for contention-based access and identified by the (Device_ID, Flow_ID) tuple value (0,0). This transmission opportunity is also known as the *registration opportunity*.

Since a new device may attempt to join a network at any time, there should be sufficient *registration opportunities* to provide reasonable "network join" response time of at least one opportunity per REG_SLOT_PERIOD.

For bandwidth efficiency, the registration opportunity should have minimum overhead so that a minimum amount of media time is wasted in case the registration opportunity is not used. One method of achieving this is to use a registration TXOP with an elastic TXOP limit and width of a single sub-burst slot (SBS_slot). Such a transmission opportunity will waste only SBS_slot microseconds if no *registration request* is sent and will expand to the required length of a *registration request* if one is sent during this time. An elastic registration opportunity should not be placed at the end of transmission cycle to avoid adding jitter to the cycle.

This is illustrated in Figure 8-17:

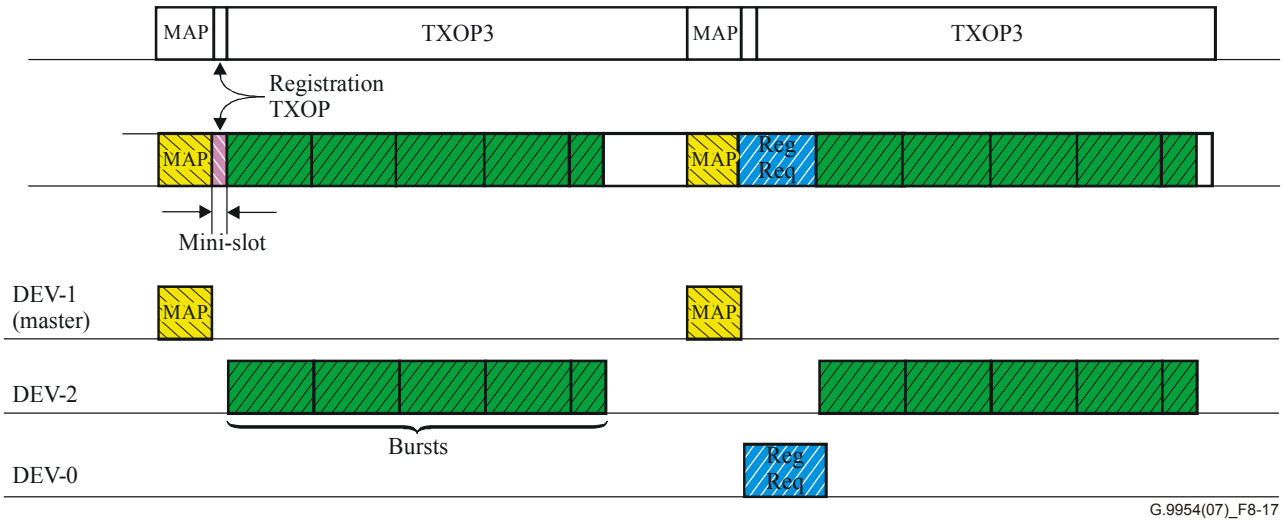

Figure 8-17 – Registration TXOP

8.11 The G.9954v2 master node functional capabilities

A G.9954v2 master-capable node is a G.9954v2 node that, in addition to supporting all of the required capabilities of a G.9954v2 "endpoint" node, is also able to assume the role of master in the absence of an active master on the network.

In addition to the G.9954v2 node requirements listed above, a node desiring to be a G.9954v2 master node shall support all of the following master-related MAC and Link Layer functions:

1) **Network admission** – Manage the admission of G.9954v2 nodes to the network (see clause 8.11.1).

2) **Dynamic master selection** – Detect the presence or absence of an operational master on the network and assume the role of network master, if required; see clauses 8.1 and 11.15.

3) **Flow and bandwidth management** – Manage the setup, modification and teardown of service flows and the allocation of associated media bandwidth resources in accordance with the services QoS constraints (see clause 8.11.2).

4) **Scheduling** – Plan the media cycle and schedule transmissions such that QoS bandwidth, latency and jitter constraints are met.

5) **MAP generation and distribution** – Generate a media access plan (MAP) that represents the output of the bandwidth management and scheduling functions and distribute the MAP each MAC cycle (see clause 8.11.4).

The G.9954v2 master shall perform media access using the same media access rules as for endpoint (non-master) devices and using the same media access plan distributed to the endpoint devices.

A G.9954v2 master-capable device shall be able to coexist on the same home network with other (active) master-capable devices. Only a single master device shall exist on the network at any one time. The G.9954v2 device selected to become master in an environment of multiple master-capable devices shall be performed automatically using the automatic master selection methods described in clauses 8.1 and in 11.15.

NOTE 1 – It is NOT a requirement that every G.9954v2 device be capable of becoming a master.

NOTE 2 – Much of the distinguishing behaviour between different G.9954v2 master implementations is defined by the bandwidth management and scheduling policies that it implements. Since these aspects of the

master are beyond the scope of this Recommendation, related material appearing in this Recommendation should be considered as informative only.

8.11.1 Network admission

All G.9954v2 devices shall first "register" with the G.9954v2 master using the Network Admission protocol; see clause 11.14.

A G.9954v2 master shall respond to a Network Admission protocol REGISTRATION request message and should check the requesting devices authorization to join the network. The master should use the MAC address, sent by the requesting device (in the REGISTRATION request) as the device identifier or key for device authentication. If the requesting device is able to be admitted to the network, the master shall assign a DEVICE_ID and return the assigned DEVICE_ID and network configuration parameters to the requesting device in the REGISTRATION response message. If the requesting device is not admitted to the network, the master shall return a status indicating the reason in the REGISTRATION response.

8.11.2 Flow and bandwidth management

The G.9954v2 master should maintain state information concerning the allocation of media resources in the network and should control the admission of new services and the allocation of media resources.

Admission control should guarantee that minimum data-rate as well as maximum latency, jitter and BER characteristics for existing services are not violated.

The master shall service requests to add/delete network service flows and requests to change service flow characteristics using the G.9954v2 link layer flow signalling protocol.

If a request is made to add a new service flow and the requested level of service cannot be met, the master shall deny the admission of the new service and a denial of service status shall be returned to the requestor.

Note that denial of service means that no QoS contracts can be given to a particular service. In this case, media access may still be performed on a priority basis within contention-based TXOPs (CTXOPs).

Similarly, if changes in line conditions over a logical channel results in a reduction in available network capacity and violation of QoS constraints for an admitted service on that channel, the master shall notify the source device of the violation of QoS service parameters, by returning a denial of service status using the flow signalling protocol.

Changes in line conditions are actually detected through rate negotiation between devices at the endpoints of a channel. If the line conditions change and the transmitter is forced to use a different payload encoding (PE) for an admitted service, the master shall be notified by the transmitting device using the flow modification signalling protocol. The master should then recalculate media bandwidth reservations to account for the change in PE.

For further information on quality of service and details of the protocols used to add new services and modify and remove existing services in the network, see clauses 10.7 and 11.1.2.

8.11.3 Scheduling

This clause contains general scheduling requirements for a G.9954v2 scheduler. These requirements are intended only as a guideline for implementations.

The G.9954v2 master should be capable of allocating media transmission opportunities to services such that a G.9954v2 device transmitting within the assigned transmission opportunity should meet QoS bandwidth, latency and jitter constraints for the admitted service flows.

The scheduler should be responsible for managing available bandwidth and bandwidth demands and

for balancing the demands for media bandwidth defined by the traffic specifications of the various admitted services with the total amount of available bandwidth.

The output of the scheduling process shall be a media access plan (MAP) that defines the allocated transmission opportunities for the various service flows.

For each admitted service flow, the master scheduler shall calculate the TXOPs required by the service, the start time of the TXOP and the TXOP length.

NOTE – The scheduling algorithm, although beyond the scope of the G.9954v2 specification, should aim to deliver deterministic guarantees for CBR (isochronous) services, statistical guarantees to variable bit-rate (VBR) services and no hard guarantees for best-effort services.

Interoperability between master and endpoints from different vendors is guaranteed through the MAP mechanism although QoS performance may vary between solutions.

8.11.4 MAP generation and distribution

The G.9954v2 master shall generate and distribute a media access plan (MAP) each MAC cycle.

A new MAP shall be generated at least each cycle although the table of TXOPs in the MAP should change only after a change in scheduling decisions as a result of the addition, removal or modification of service flows, or if network conditions change.

The G.9954v2 master shall distribute the MAP by broadcasting the MAP control frame to all nodes in the network. The MAP control frame shall be broadcast using the most robust payload encoding (PE = 33, 2 Mbaud, 2 bits per symbol).

For further details on the MAP and the structure and timing of the MAC cycle, see clause 8.7. For further information on the MAP control frame, see clause 11.13.1.

8.12 G.9954v2 endpoint node requirements

A G.9954v2 endpoint node shall detect the existence of a G.9954v2 master node on the network and operate according to the media access rules for a managed network.

As a minimum requirement, a G.9954v2 endpoint node shall support the following MAC functions:

1) **MAC cycle synchronization** – A G.9954v2 endpoint shall synchronize with the master-generated MAC cycle in a managed network.
2) **Synchronized transmissions** – A G.9954v2 endpoint node shall comply with the transmission directives in the current MAP and guarantee that it shall only transmit within a TXOP that is allocated exclusively to it (CFTXOP) or to a group to which it belongs (CTXOP) or within an unallocated TXOP (UTXOP).

A G.9954v2 device shall respect bandwidth allocations described in the MAP to other devices.

If a G.9954v2 device is not allocated a transmission opportunity in the media cycle, it shall bound its transmissions strictly within the bounds of the UTXOPs.

NOTE – The above minimum requirement represents the core functionality upon which the higher-level protocol functions (e.g., registration, flow set up etc.) can be bootstrapped.

In order to support QoS contracts of bandwidth reservation for flows, a G.9954v2 endpoint device shall support the following G.9954v2 MAC and link-layer functions:

1) **Registration** – Once an endpoint node has synchronized with the master, the endpoint shall REGISTER with the master. REGISTRATION is the process whereby an endpoint requests entry to the network and, if authorized, is supplied a network Device_ID and network configuration data.
2) **Flow signalling** – In order to manage QoS flows, an endpoint shall support the flow signalling protocol. The flow signalling protocol is used to set up, modify or tear down flows.

8.12.1 Synchronization

A G.9954v2 endpoint node shall synchronize with the master-generated MAC cycle by detecting the existence of a MAC media access plan (MAP) transmission. Upon detection of a MAP control frame, a G.9954v2 endpoint node shall reset its synchronous clock counter to zero at the time corresponding to the arrival time of the first symbol of the preamble of the MAP transmission at the wire-interface in the receiver. A G.9954v2 endpoint device shall schedule its synchronous transmissions within the MAC cycle according to the synchronous clock counter.

If a G.9954v2 node fails to receive an MAP transmission for SYNC_TIMEOUT, the G.9954v2 endpoint node shall switch to the media access mode defined for unmanaged networks.

When operating in an unmanaged network, upon detection of a MAP transmission, a G.9954v2 endpoint node shall switch to a mode of operation appropriate for a managed network. The mode switch shall be initiated immediately and within the same transmission cycle.

8.12.2 Synchronized transmissions

In a managed network, a G.9954v2 endpoint device shall perform media access according to the current active media access plan advertised by the master. It shall transmit only within a TXOP assigned exclusively to it or to a group to which it belongs.

A G.9954v2 endpoint node shall accurately schedule its synchronous transmissions using the synchronous clock counter and comply with synchronous timing constraints specified in clause 8.9.2.

8.12.3 Registration

A G.9954v2 endpoint node shall register with the master, using the LLC REGISTRATION protocol.

A G.9954v2 endpoint node shall transmit REGISTRATION protocol messages within a UTXOP (see registration TXOP in clause 8.10.3.6.3)

NOTE – Endpoint devices may initially contend for access to the REGISTRATION opportunity. Collisions may be handled by the G.9954v2 collision resolution methods and/or by retrying after a random number of admission opportunities.

A G.9954v2 endpoint node shall notify the master of its assigned MAC address in the REGISTRATION protocol message.

Authentication is part of the REGISTRATION process and may be performed by checking that the device, identified by the endpoints MAC address, is authorized to join the network. The authorization procedure is implementation dependent.

The G.9954v2 endpoint device shall use the Device_ID assigned to it by the G.9954v2 master in subsequent flow signalling protocol sequences.

For further details on the REGISTRATION protocol, see clause 11.14.

8.12.4 Flow signalling

A G.9954v2 endpoint node shall support the flow signalling protocol if it supports flows with varying QoS parameters.

In a managed network, the G.9954v2 endpoint at the source of a QoS contract flow shall inform the master of flow set-up, modification and teardown requests. It shall notify the master of negotiated rate changes between a source and destination device (i.e., changes in the flows' current PE) over a logical channel using the flow modification protocol.

For further information on flow signalling and rate negotiation, see clauses 11.16 and 11.4, respectively.

8.12.5 Endpoint processing and scheduling

G.9954v2 endpoint nodes require little local intelligence in order to schedule transmissions in a master-controlled G.9954v2 network. Scheduling can be performed solely based on the directions provided in the received MAP. QoS scheduling intelligence is concentrated in the master and expressed in the MAP.

Endpoint nodes may exercise local scheduling intelligence by reassigning the association of services to its allocated transmission opportunities at its own discretion provided that it does not conflict with the semantics defined for a transmission opportunity (e.g., priority association). In other words, services that are allocated specific transmission opportunities by the master may be reassigned, by the endpoint device, to other services, if it so desires.

If local scheduling is performed, the resulting QoS achieved for services originating from the endpoint shall be no worse than that which would be achieved using the master schedule alone.

NOTE – "worse" here is measured in terms of QoS throughput, latency, jitter and BER.

8.13 MAC layer framing

This clause provides details of the sub-fields of the G.9954v2 frame control field that are used by the G.9954v2 MAC layer.

8.13.1 Frame type

The frame type (FT) is an eight-bit field that is used to define different frame formats.

G.9954v2 devices may transmit frames with FT = 0x8 or 0x9. All other values are reserved and frames received with such frame types shall be discarded.

The FT is intended to provide a mechanism for forward compatibility, allowing extensions to use frame formats differing from G.9954v2.

The FT field is composed of sub-fields as shown in Table 8-10:

Table 8-10 – Frame type field

Field	Bit number in frame control	Bits	Description
G.9954	31:31	1	0 = Reserved for legacy use 1 = G.9954v1 and G.9954v2 MAC frame
FS	30:28	3	Frame subtype 0 = Ethernet frame 1 = MAP 2-7 = Reserved for future use

The bits of the frame type field are transmitted in the order shown in Figure 8-18.

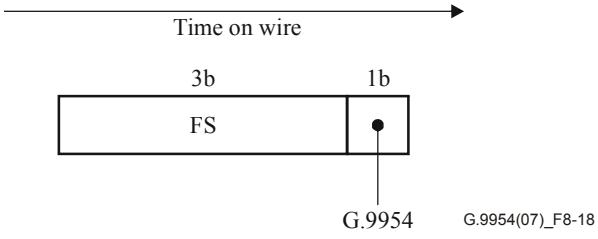

Figure 8-18 – Frame type field order

8.13.1.1 Frame Subtype (FS)

This field is used to define the Frame Subtype. The following subtypes are defined:

- 0 = Ethernet frame
- 1 = MAP frame used in synchronous MAC protocol
- 2-7 = Reserved for future use.

Reserved frame subtype values are intended for use in future.

8.13.1.2 G.9954

This bit-field is used to indicate a G.9954v1 MAC frame.

8.13.1.3 Reserved bits (RSVD)

This field shall be set to zero by the transmitter, and the receiver shall discard any frame where this field is non-zero.

8.13.2 Extended ID (EID)

The interpretation of the extended ID (EID) field and identifier (ID) fields is dependent on the value of the most significant bit (EID[2:2]) of the EID. When EID[2:2] is zero, the remainder of the EID field is undefined and the interpretation of the ID field is as for legacy G.9954v1 systems. When EID[2:2] is set, the EID and ID fields are taken together and interpreted as the Device_ID of the HNT device transmitting the frame where EID[1:0] are the high-order bits and ID[3:0] are the low-order bits of the Device_ID respectively.

Field	Bit number in frame control	Bits	Description
EID[2:0]	27:25	3	Extended identifier
EID[2:2]	27:27	1	0 = G.9954v1 legacy mode 1 = Extended identifier field defined
RSVD	26:25	2	When EID[2:2] = 0: Field is reserved. Should be set to zero by transmitter and ignored by receiver
Device_ID[5:4]	26:25	2	When EID[2:2] = 1: High-order 2 bits of the Device_ID of the transmitting HNT device

8.13.3 ID

The interpretation of this field is dependent on the value of the most significant bit (MSB) in the EID field (i.e., EID[2:2]). When EID[2:2] = 0, the ID field assumes the legacy G.9954v1 interpretation of the FLOW_ID associated with the frame. When EID[2:2] = 1, the ID field represents the low order 4 bits of the Device_ID of the transmitting HNT device.

Field	Bit number in frame control	Bits	Description
ID	23:20	4	Identifier field
FLOW_ID	23:20	4	When EID[2:2] = 0: Flow_ID associated with frame
Device_ID[3:0]	23:20	4	When EID[2:2] = 1: Low-order 4-bits of the Device_ID of the transmitting HNT device

8.13.4 Minimum and maximum link-level frame sizes

The link-level frame consists of the DA through FCS fields, prior to the PHY-level frame encapsulation. All G.9954v2 stations shall transmit link-level frames with a minimum of 64 octets. The payload field of link-level frames smaller than minFrameSize shall be padded with any value octets appended after the supplied payload to make the frame minFrameSize long.

The maximum standard Ethernet frame is 1518 octets, but some G.9954v2 link-layer encapsulations may add additional octets.

All G.9954v2 stations shall be able to transmit and receive link-level frames with up to 1526 octets. No G.9954v2 station shall transmit link level-frames with more than 512 * bits per symbol * baud [octets]. The number of octets specified counts DA through FCS, and does not count preamble, header, CRC-16, PAD or EOF. This will result in a maximum frame duration (maximum TX_FRAME value) of 4166 µs. A G.9954v2 station shall default the maximum length frame it will send to a given DA to 1526 octets until it can determine that the receiver can support larger transmission units (e.g., by use of the CSA announcement of CSA_MTU; see "Link Protocols for G.9954v2").

These maxima establish an upper bound on the duration of a given transmission and an upper bound on the maximum frame size that receivers must accommodate.

8.13.5 Frame bursting (Packet aggregation)

G.9954v2 devices shall support the aggregation of link layer frames (packets) into a single physical layer frame (burst). The purpose of packet aggregation is to reduce overheads associated with the physical layer frames by concatenating packets from the same source and to the same destination into a single burst. Packets aggregated into a burst shall either all belong to the same flow or shall all have a priority greater or equal to the priority of the first packet in the aggregated frame.

Aggregation reduces the per-packet overhead by removing the IFG between aggregated packets. In addition, the low-baud and low-constellation burst header is shared amongst all aggregated packets.

The aggregation frame format shall use the G.9954v2 link layer frame burst control frame to encapsulate the aggregated packet data. This link layer control frame format is described in full detail in clause 11.12.

Aggregation shall conform to the following basic rules:

- The maximum length of the aggregated frame shall not exceed the maximum allowed time on the wire.
- The maximum number of aggregated frames in a burst shall be negotiated between source and destination, either using the CSA protocol or flow signalling protocol.
- All aggregated frames in a burst shall have the same source and destination address. The destination address may be a BROADCAST or MULTICAST address.
- The priorities of all aggregated frames in a burst shall be all greater than or equal to the priority of the first sub-frame of the burst.
- A burst termination trailer shall be used to indicate the end of a burst.

Aggregation may be performed up to the limits of the size of the TXOP in which the frame will be transmitted or up to the maximum link level frame size, whichever is smaller.

Figure 8-19 shows a breakdown of the frame aggregation format.

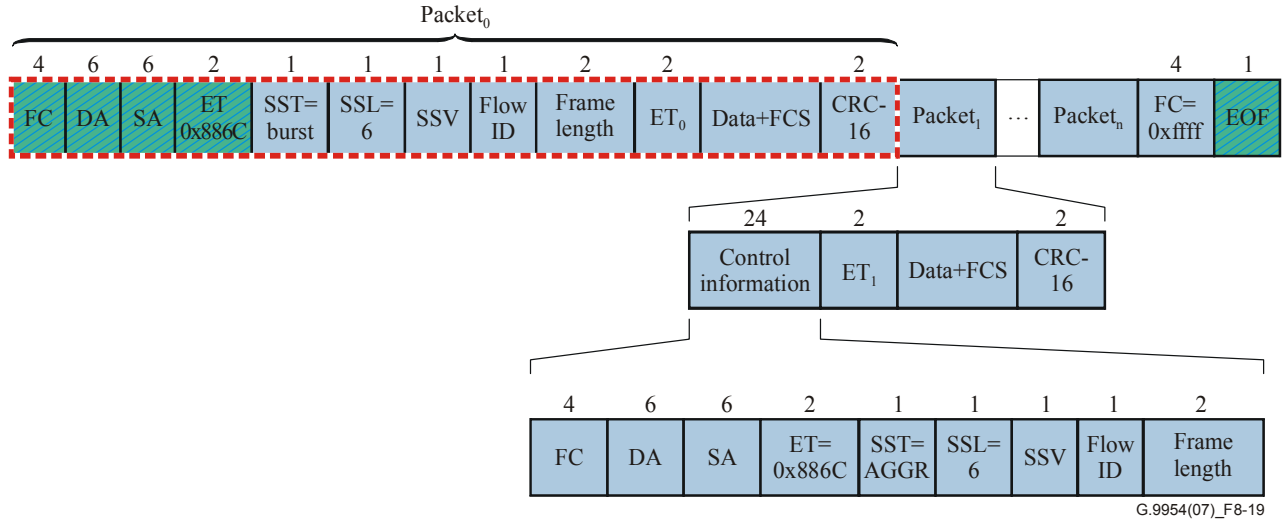

Figure 8-19 – Aggregation frame format

Each aggregated frame shall contain a full packet header.

The FCS field shall have the same meaning as described in IEEE Std 802.3 frame and is calculated over the DA, SA, EtherType and Data fields of the frame. The CRC-16 shall be calculated over the same respective fields.

8.14 MAC parameters

This clause is determinative of MAC parameters, to supersede any other value of these parameters in other parts of this Recommendation. Where a tolerance is indicated, Δ = 63 nanoseconds.

Table 8-11 – MAC parameters

Clause	Parameter	Min	Max	Units
8.1 Modes of operation	SYNC_TIMEOUT	–	150	milliseconds
8.2 Basic CSMA	NOMINAL_RMS_VOLTAGE – Phoneline modes A & B	100	–	mV rms
	NOMINAL_RMS_VOLTAGE – Coax Modes A, B, D	205		mV rms
	NOMINAL_RMS_VOLTAGE – Coax modes C	145		mV rms
	CS_RANGE – Phoneline Modes A & B	38	–	dB
	CS_RANGE – Coax Modes (ALL)	55		dB
	CS_IFG	29.0 – Δ	29.0 + Δ	microseconds
	minFrameSize	64	–	octets
	maxFrameSize	1526	See 8.13.4	octets
	TX_FRAME	92.5	See 8.13.4	microseconds
	Round-trip time (RTT) for 1000 ft		3.0	microseconds
	TX_ON	0	4.0	microseconds

Table 8-11 – MAC parameters

Clause	Parameter	Min	Max	Units
8.7 The MAC cycle	CS_ICG	CS_IFG		microseconds
	MAP_IFG	CS_IFG	63	microseconds
	CYCLE_MAX		50	milliseconds
	CYCLE_MIN	5		milliseconds
	TICK	500	500	nanoseconds
	TIME_SLOT	1	8	TICKs
8.10.3.1 Sub-burst slot size and timing	SBS_SLOT	8	64	8, 16, 32 or 64 μs
8.10.3.6.3 Registration opportunity	REG_SLOT_PERIOD		1	seconds

9 Compatibility specification

9.1 Spectral compatibility with other services on the same wire

The PSD mask specified is such that compliant transmitters should be able to meet FCC Part 68 Section 308-e-1-ii.

The mask A over phoneline specifies a limit of –140 dBm/Hz below 1.7 MHz, which ensures compatibility with G.992.1, G.992.2 and ISDN.

The mask B over phoneline specifies a limit of –140 dBm/Hz below 8.5 MHz. The mask includes notches covering the Radio Amateur bands (e.g., between 7.0 and 7.3 MHz), which reduces the maximum PSD to –81.5 dBm/Hz. This is lower than the VDSL recommendations for PSD in the amateur bands. Since the VDSL spectral compatibility has been developed over the last several years in several standards bodies, including the ITU-T, this spectral mask should be compatible with RFI emission requirements in countries outside North America, such as UK, Japan, Germany and France.

9.2 Coexistence and interoperability with G.9951/G.9952

G.9954v2 uses the same PHY header, frame format as G.9951/G.9952. This means that under certain conditions a G.9954v2 device may be able to receive a G.9951/G.9952 frame and to decode it. In addition, the G.9951/G.9952 LLC protocols represent a subset of the G.9954v2 protocols and are therefore compatible with G.9954v2 LLC protocols.

However, since G.9954v2 is, by definition, a synchronous protocol, based on CSMA/CA methods (with no collision-detection capabilities) while G.9951/G.9952 uses CSMA/CD methods, these methods will not coexist together on the same wire without destructive collisions causing packet error.

9.3 Coexistence and interoperability with G.9954

G.9954v2 node shall coexist and interoperate with G.9954 nodes on the same network.

A G.9954v2 master shall detect the existence of a legacy G.9954v1 node on the network and shall guarantee that the legacy G.9954v1 nodes will not share a TXOP with other G.9954v1 nodes except for the unallocated TXOP (UTXOP). The G.9954v2 master may allocate a unique contention-free TXOP per G.9954v1 legacy node or it may enforce all legacy G.9954v1 nodes to operate only

within a UTXOP period. The particular approach taken depends on system requirements. A G.9954v1 master may support a configurable parameter to control the particular method preferred.

Detection of G.9954v1 legacy nodes shall be performed on the basis of information contained in the LLC registration and CSA protocol messages.

A G.9954v2 node shall be able to detect a G.9954v1 master and shall issue a bandwidth reservation request in order to create a unique contention-free TXOP for its transmissions.

10 G.9954v2 quality of service

10.1 General description

The G.9954v2 MAC supports both priority-based and parameter-based QoS methods.

G.9954v2 support for eight priority levels provides a basic Quality of Service (QoS) mechanism for differentiating between different kinds of services. This mechanism is compatible with several existing mechanisms for differentiating between classes of service such as the IEEE 802.1D recommendations for the VLAN priority tag (IEEE 802.1P) and the PRECEDENCE bits defined in the original interpretation of the type Of service (TOS) field found in an IP packet header using the differentiated services (Diffserv) protocol. The priority-based scheme defines a relative ordering amongst packets with high-priority packets guaranteed access to the media before lower-priority packets.

Although priority classification of services provides some level of QoS support, it cannot provide QoS guarantees with strict latency and jitter budgets. In order to provide strict QoS contracts, the G.9954v2 MAC also provides a parameter-based QoS method based on the concept of *flows* that is compatible with a resource reservation-like protocols and supports the specification of QoS in terms of explicit traffic and rate parameters and not just a relative ordering of packets. Traffic shaping, scheduling and policing mechanisms, based on these well-defined QoS parameters, are subsequently used to provide strict control over network throughput, latency and jitter performance.

This clause specifies the G.9954v2 QoS solution.

10.2 Priority-based QoS

Support for priority-based QoS shall be provided using the priority access scheme described in clause 8.3. This scheme shall be implemented as follows:

The priority access scheme specifies a set of up to eight priority groups composed of sub-burst slots assigned to particular devices in the network. A device may transmit within a sub-burst slot within a group if and only if the packet it is transmitting has a marked priority that is greater than (higher) or equal to the priority associated with the group. Furthermore, the priority access scheme specifies that following a transmission, the state machine tracking the current priority group should be restarted.

The priority access scheme shall be implemented in a G.9954v2 device using sub-burst slot groups in the following manner. Each priority is represented by a sub-burst slot group. Within a group, the group type should be either "Rotated" or "Repeated" in order to achieve fairness of bandwidth allocation within a group. Note that it is also possible to use a group type of "Fixed" in order to achieve a further level of prioritization within a group. The Group Type of the lowest priority group should be "Repeated" in order to allow low priority packets to gain access to the media without delay in the absence of any higher priority traffic.

The priority access scheme can be described by the state diagram in Figure 10-1, where each state represents a priority group PG-*n* where *n* is the priority assigned to the sub-burst slots in the group. From this diagram, it can be seen that while there are no transmissions of higher-priority packets, the state machine progress to the lower-priority group and remains in that group until a transmission

occurs. Following a transmission, the state machine is restarted to the highest-priority group allowing immediate access to high-priority packets. This bounds the maximum media access latency of a high-priority packet.

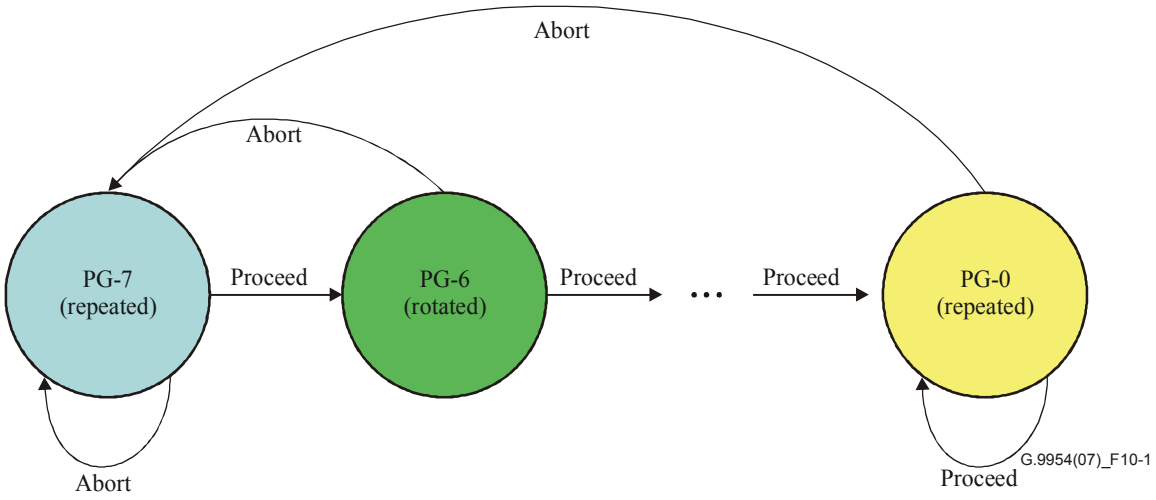

Figure 10-1 – Priority access scheme implementation

The MAP structure described in Table 10-1 shall be used to describe the state-machine for the priority access scheme. The priority-based QoS scheduler that generates the priority-based MAP shall guarantee that a sub-burst slot is assigned to a device in each priority group that has a lower priority than the first priority group in which a device appears. This guarantees that a device can transmit a high-priority packet within any lower-priority group. The priority-based QoS scheduler shall also guarantee that the last priority group in a sequence (i.e., the lowest priority) has the assigned priority value of zero.

Table 10-1 – Priority scheme MAP

#	Device Id	Group type	Flow_ID/ priority	Length	TXOP
0	1	4	0	L_0	[0]
1	1	6	7	0	[1]
2	2	6	7	0	
3	3	6	7	0	
4	1	5	6	0	
5	2	5	6	0	
6	3	5	6	0	
…	…	….	….	….	
1	1	6	0	0	
2	2	6	0	0	
3	3	6	0	L_1	

Note that in order that the last priority group be of Group type "Repeated", it is necessary to start the first group with either "Rotated" or "Repeated" depending on whether an even or odd number of priority groups are encoded into the MAP respectively.

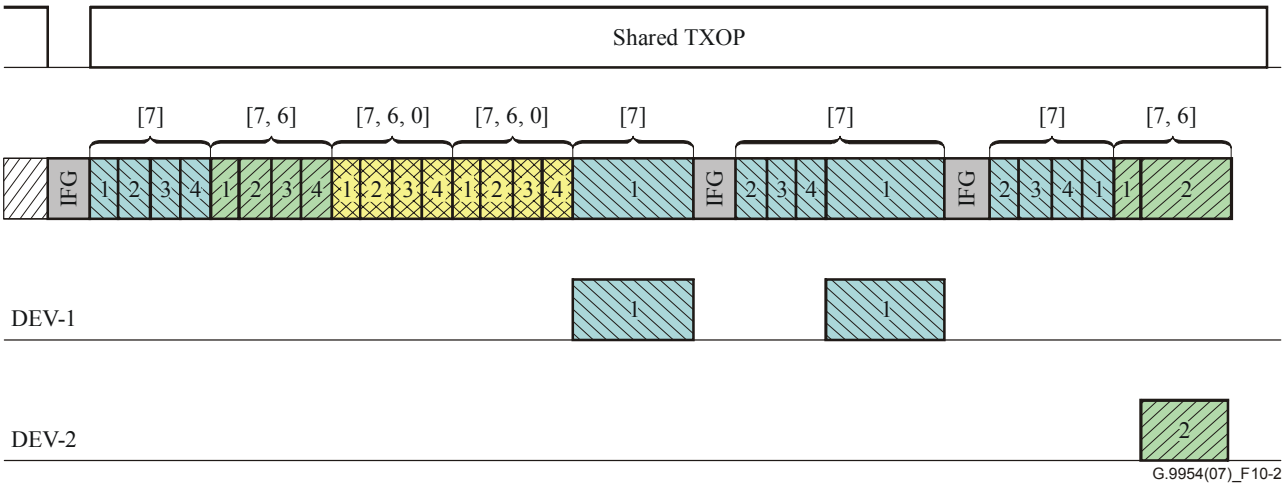

Figure 10-2 – Priority access scheme

It shall be the responsibility of each device to determine the priority associated with a packet that it is transmitting. A device may use priority information provided by upper layers (e.g., priority tag within a VLAN tagged packet, or IP TOS DSCP codepoints), or the device may assign a priority based on priority information associated with a flow specification. A device may also map or remap link layer priority information associated with a packet to a media access priority as defined within the MAP.

A priority-based QoS scheduler should support at least three priority levels (i.e., priority groups).

10.3 Parameter-based QoS

Parameter-based QoS is based on the use of traffic specifications to describe traffic in terms of rate, latency, jitter, PER and other parameters. This provides the basis for fine-grain QoS controls over bandwidth allocation, latency and jitter. Whereas the priority-based method guarantees a relative ordering between packets, the parameter-based QoS method guarantees resource allocation and a media access plan that matches the characteristics of the traffic flows on the network providing the necessary conditions that allow a traffic flow to meet its QoS constraints.

The G.9954v2 QoS mechanism is based on the concept of a *data flow* (or *flow* for short). A flow represents a unidirectional flow of data between network nodes based on well-defined QoS traffic and rate parameters, that allow strict control over network throughput, latency, jitter and BER parameters.

Flows are set up and torn down on a service-by-service basis. The G.9954v2 master is responsible for allocating bandwidth for flows, upon request, and for advertising the bandwidth allocation decision in the media access plan (MAP). Network nodes are responsible for scheduling their transmissions according to the constraints of the advertised MAP.

The bandwidth allocation algorithm shall enforce and guarantee QoS parameters. Consequently, bandwidth reservation requests associated with a flow set-up are subject to admission control policing and shaping by the master. Flow set-up requests that cannot be met according to the requested parameters are rejected and QoS parameters may be subsequently re-negotiated.

Bandwidth requirements for a flow may be modified throughout its lifetime in order to more effectively support changing bandwidth requirements that are characteristic of "bursty" and variable bit-rate (VBR) data streams and changing line conditions.

Flows are set up by convergence layers, either implicitly – upon identification of a new service, or explicitly – in response to higher-level protocol messages or upon network admission according to a

predefined specification/configuration. Flows may similarly be torn down implicitly, upon detection of inactivity or explicitly upon termination of a service, in order to free network resources associated with the flow.

It is the responsibility of the convergence sublayer to map incoming data streams onto the appropriate flow that meet their individual QoS requirements.

In summary, the main QoS features supported by G.9954v2 parameter-based QoS method are as follows:

- Statistical and deterministic QoS guarantees for bandwidth, jitter, latency and BER.
- *Traffic classes* and *service flows* described by well-defined traffic and rate parameters.
- Constant and variable bit-rate flows.
- Flow management including flow admission control, resource reservation, QoS negotiation/re-negotiation, flow set-up and tear-down.
- Frame classification based on traffic filter specification e.g., IP TOS, VLAN priority tag, protocol type, source/destination address, etc.
- QoS flow policing, shaping and scheduling.

10.4 Service flows and QoS parameters

A flow describes a simplex communication channel, with well-defined QoS characteristics, between source and destination device. The QoS characteristics of a flow are described by a set of traffic and rate parameters which are communicated between G.9954v2 devices using the *Flow Signalling* protocol (see clause 10.7 for more details).

A flow is uniquely identified by the tuple (*Source Address, Destination Address, Flow_Id*). The QoS characteristics of a flow are defined by the parameters summarized in Table 10-2 and defined in subclauses below.

Table 10-2 – Flow properties

Field name	Description
Source address	The MAC address of the device at source of the flow
Destination address	The MAC address of the device that is the destination of the flow (may be the broadcast address)
Flow ID	Unique identifier of the flow between the source and destination addresses. The flow ID is assigned by the G.9954v2 device at the source of the flow.
Priority	Link layer priority assigned to the flow
Service type	Defines the type of service that the flow supports: 0 – CBR 1 – rt-VBR 2 – nrt-VBR 3 – BE
Max. latency	Maximum tolerable transmission and queuing delay according to Table 11-67.
Max. jitter	Maximum delay variation according to table defined in defined in Table 11-68.
ACK policy	0 – None 1 – LARQ

Table 10-2 – Flow properties

Field name	Description
FEC policy	0 – None 1 – RS 2 – 3 Reserved
Aggregation policy	0 – No aggregation 1 – MAC-level aggregation
CRC error handling policy	0 – Do not discard packets with CRC errors. 1 – Discard packets with CRC error.
Nominal packet size	The nominal packet size in octets for packets associated with the service. A value of 0 indicates an unspecified or unknown value.
Max. data rate	Peak burst rate in 4-kbit/s-per-second units. Takes into account the net (payload) data rate
Average data rate	Average bit rate required by the service in units of 4 kbit/s
Min. data rate	Minimum required bit rate in 4-kbit/s units for the service to operate. This number is expected to be different from zero only for real-time traffic requiring a minimum transmission delay.
BER	Service-level BER. Used in rate negotiation to select the desired PE that achieves the highest raw bit rate and also meets BER requirements.
Payload encoding	Payload encoding used on logical channel This parameter shall only be set when communicating flow parameters to the master. Between flow endpoints, the payload encoding is negotiated using rate negotiation.
Packet timeout	The amount of time in milliseconds a packet will remain queued before being deleted from the flow queue. A value of 0 indicates that packets never time out and remain queued until transmitted on the line.
TX timeslot	Timeslot of first TXOP defined for the flow. This field can be set by upper layers during flow set-up in order to synchronize allocated TXOPs with an external source. This is intended for isochronous services. Time is measured in units of 2-13 ms with reference to the master's time reference as advertised in the timestamp report indication; see clause 11.17.
Flow inactivity timeout	Amount of time (in milliseconds) a flow will remain "alive", in the absence of any traffic, before the flow is automatically torn down and resources released. A value of 0 indicates that the flow is not automatically torn down. For further information on flow tear-down, see clause 11.16.

10.4.1 Source and destination address

The source and destination address of a flow is identified by the respective source and destination device addresses. The source address is a unicast 48-bit MAC address identifying the device at the source of the flow. The destination address identifies the destination of the flow and may be a unicast, multicast or broadcast 48-bit MAC address.

10.4.2 Flow ID

The Flow ID is a unique *flow identifier* between source and destination addresses. The Flow ID shall be assigned locally by the device at the source of the flow.

For more information on Flow IDs, see clause 8.6.

10.4.3 Priority classification

The priority classification represents the link layer priority assigned to the flow. The priority value shall be used to define the media access priority for transmissions when priority-based QoS is used. It may also be used by the scheduler in the master to rank flows in scheduling decisions.

Priority assignment should follow IEEE 802.1D and 802.1P recommendations for the mapping of user priorities to traffic classes. For further information, see clause 11.6.

10.4.4 Service type

The *service type* of a flow defines the type of QoS commitment guarantees required by the service. The *service types* are defined in Table 10-3.

Table 10-3 – Service types

Service type	Description
Unsolicited grant (CBR)	Supports real-time low-latency, fixed-size, periodic (CBR) data. The resource scheduler guarantees allocation of a fixed amount of bandwidth periodically without explicit bandwidth requests. Used for "deterministic" QoS guarantees.
Real-time (rt-VBR)	Supports variable bit rate (VBR) data by supporting periodic variable size data grants. Suitable for MPEG video streams.
Non-real-time (nrt-VBR)	Similar to a real-time service except that the scheduler services non-real-time flows at a lower rate than real-time flows.
Best effort (BE)	Similar to non-real-time service except that the scheduler services Best-Effort traffic at a lower rate than non-real-time traffic.

The *service type* parameter may be used by the master scheduler in scheduling decisions and by the source node in resource management decisions.

10.4.5 Maximum latency

This parameter defines the maximum tolerable transmission and queuing delay for a service. The parameter is defined by an enumerated value from a set of defined latencies expressed in milliseconds.

The amount of latency a service can tolerate affects the amount of memory (buffer space) required. For devices that have less buffer space available than the amount implied by the latency parameter, an alternative (lesser) latency value may be specified by the destination device in the Flow Set-up/Modify Response message used in the flow signalling protocol.

The *Maximum Latency* parameter shall be used by the master scheduler in scheduling decisions concerning the interval between transmission opportunities and the number of transmission opportunities assigned to the service within the MAC cycle. This parameter may also be used to control the length of a burst of aggregated packets belonging to the same service.

For further details of the supported latency values and the flow signalling protocol, see clause 11.16.

10.4.6 Maximum jitter

The maximum jitter parameter defines the maximum delay variation in latency values for a service above and below the mean latency value. Maximum jitter is expressed as (±Max) ms.

The *maximum jitter* parameter should be used by the master scheduler in scheduling decisions concerning the position of TXOPs within the MAC cycle.

Jitter values are expressed as an enumerated value within a set of defined jitter values. For further details of the supported jitter values, see the description in clause 11.16.

10.4.7 ACK policy

This flag indicates whether the flow requires link-layer acknowledgements using the LARQ mechanism in order to reduce the packet error rate (PER). ARQ is specified per ARQ channel where an ARQ channel is defined by a flow, i.e., by the tuple (*Source Address*, *Destination Address*, *Flow ID*) or by the (*Source Address*, *Destination Address*, *Priority*).

NOTE – TCP-based protocols are natural candidates for applying a link-layer ACK policy as TCP performance may degrade significantly with an increase in packet errors.

10.4.8 Forward error control (FEC) policy

This flag indicates whether Reed-Solomon coding should be applied on the communication channel defined by the flow. This indication shall be used by a receiver to determine whether Reed-Solomon redundancy information should be sent to the transmitter, at the flow source, during rate negotiation.

Since support for Reed-Solomon coding is optional, support for this parameter is also optional.

10.4.9 Aggregation policy

Latency characteristics of a flow are used by the scheduler to determine how much flow data can be aggregated into a single transmission burst (frame). Scheduling decisions that account for a flow's latency requirements are taken by the master when calculating the size of a TXOP in the MAP. Similarly an endpoint device performing local traffic scheduling may use latency characteristics to determine the amount of aggregation and transmission burst size.

Aggregation can be disabled completely for a flow, irrespective of the latency parameter, by specifying an aggregation policy of "*No Aggregation*".

NOTE – A "No Aggregation" policy may be useful when aggregation is performed at upper protocol layers and no further aggregation is desired.

10.4.10 CRC error-handling policy

This clause specifies the policy to be used by the MAC when handling packets with CRC errors. Erroneous packets may be discarded by the MAC/link layers or passed up to higher protocol layers with erroneous bits in contained within.

The particular CRC error-handling policy effects the semantics of the BER parameter as described in clause 10.4.13.

NOTE – Some services are tolerant of a small number of erroneous bits in the data stream. If the CRC error-handling policy specifies that erroneous packets should be discarded, then this implies a BER = 0 since no bit errors will be passed up to higher protocol layers. However, discarding complete packets introduces packet errors and the PER parameter becomes the dominant measure. A PER = 0 can be achieved using LARQ (ACK policy) at the expense of latency.

10.4.11 Nominal packet size

This is the nominal packet size, in octets, for packets associated with the service. A value of 0 indicates an unspecified or unknown value.

10.4.12 Maximum, average and minimum data rates

The peak, average and minimum bit rates required for a service to operate effectively. Data rates are expressed in units of 4 kbit/s.

For CBR flows, the minimum, maximum and average data rates are typically all equal.

Given a service's nominal packet size and data rates, it is possible to police and shape traffic into a form that conforms to the service specifications. This may be required in some implementations to ensure that a flow does not consume more resources than defined by its traffic specification. The

allocation of transmission opportunities in the MAP inherently imposes traffic shaping on the endpoints.

10.4.13 Bit error ratio (BER)

Each service has an associated BER requirement that specifies the ratio of bit errors to "non-error" bits that a service is able to tolerate before QoS is affected.

The BER parameter is used to describe either the per-bit error probability, if packets with CRC errors delivered to upper layers, or the packet error rate (PER) divided by the mean number of bits per packet, if packets with CRC errors are discarded. The policy for handling packets with CRC errors is specified by the CRC error-handling policy flag (see clause 10.4.10).

For example, consider a service using 1500-byte packets and requiring a $PER = 10^{-2}$, then $BER = 10^{-2} / (1500 \times 8) \approx 10^{-6}$.

NOTE – The service-level BER is used during rate negotiation to determine the best payload encoding that can be used in order to provide the highest throughput communication channel that is able to meet the BER requirements for a service. For further information on rate negotiation, see clause 11.4.

10.4.14 Payload encoding

This parameter defines the payload encoding (PE) to be used on the channel. The PE chosen is determined through rate negotiation and represents the PE providing the highest raw bit rate that meets the BER parameter for the service.

10.4.15 TX timeslot

In order to support the synchronization of a flow's TXOPs with an external source (e.g., upstream timeslots in a broadband access network), the initiator of a flow set-up sequence can indicate the timing of TXOPs desired on the home network. Timing is specified by an absolute time measured with respect to the master's time reference.

NOTE 1 – This feature requires an endpoint node to synchronize its clock with the master's clock reference using master timestamp reference protocol. The time specified is an absolute time (remember we are synchronized with the master clock). The master knows the requested time and the max latency and so it can calculate where it should allocate the TXOPs in time. This parameter is only intended to help masters make scheduling decisions.

When allocating bandwidth for the specified flow, the master scheduler may use this information to influence the location of the associated TXOPs within the MAC cycle. When no timing information is provided, the master scheduler is free to allocate TXOPs as it sees fit. There is no requirement that the master meets the requested timing specification.

NOTE 2 – Timing information about the location of TXOPs is returned to upper convergence layers through the MAP mechanism. This allows upper layers to similarly synchronize to actual timing on the home network if so required.

For further details on clock synchronization, see the description of the timestamp report indication protocol in clause 11.17.

Since the timestamp report indication protocol is optional, support for this parameter is also optional.

10.4.16 Flow inactivity timeout

This parameter specifies the amount of time a flow may remain inactive before the flow is automatically "torn down" and its resources released. A flow is inactive in the absence of any traffic on the flow. A flow may be torn down by either device found at the endpoints of a flow.

A flow inactivity timeout with a value of zero disables flow inactivity ageing.

NOTE – It is strongly suggested that flows be defined with inactivity ageing enabled in order to guarantee the release of media (and other) resources in case of service termination.

10.5 Bandwidth allocation models

Parameter-based QoS requires the QoS scheduler to allocate bandwidth in such a manner so as to guarantee traffic specifications. Although QoS scheduling algorithms are beyond the scope of this Recommendation, in general, it can be said that the G.9954v2 scheduler supports two distinct bandwidth allocation models:

1) fixed bandwidth;

2) dynamic bandwidth.

In the fixed bandwidth model, the QoS scheduler allocates fixed-size chunks of media time (TXOPs) for the exclusive use of a device and flow assuming contention-free media access within that TXOP. The size and location of the TXOPs should be set appropriately by the QoS scheduler in accordance with the flow's associated traffic rate and latency/jitter characteristics. This model is best suited for constant bit-rate (CBR) traffic.

In the fixed bandwidth model a fixed quantity of media bandwidth is reserved exclusively for the use of the device at the source of the CBR traffic flow. The reserved bandwidth is guaranteed to be available each cycle while the service is still up and the bandwidth allocated cannot be compromised by other services sharing the media.

In the dynamic bandwidth allocation model, the QoS scheduler allocates bandwidth within a TXOP in such a way as to guarantee statistically the QoS traffic specifications. In general, the TXOP should be shared amongst devices and flows also requiring only statistical guarantees. This model is best suited for handling "bursty" variable bit-rate traffic.

The dynamic bandwidth model uses as much bandwidth as is required allowing other devices sharing the same TXOP to use the unused bandwidth if required. This provides for efficient usage of the available bandwidth amongst devices transmitting bursty traffic.

Figure 10-3 illustrates fixed and dynamic bandwidth allocation models. In this diagram DEV-2 is allocated a fixed block of media bandwidth (TXOP1) while devices DEV-1 and DEV-3 share a CTXOP (TXOP2) with bursty traffic using the dynamic bandwidth allocation model.

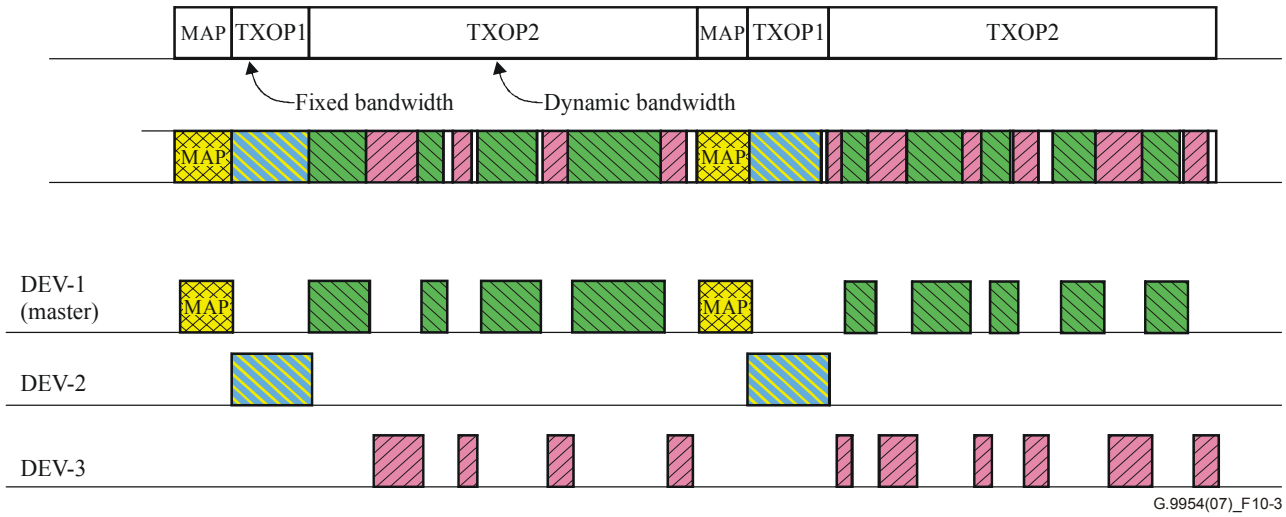

Figure 10-3 – Fixed and dynamic bandwidth allocation

It is the responsibility of the QoS scheduler to determine which of the bandwidth allocation models is most appropriate for the type of traffic. In general, it is recommended that the QoS scheduler uses the fixed bandwidth model for flows with a service type of CBR and a dynamic bandwidth model for flows of other service types.

10.6 Convergence layer traffic classification

Packets from upper protocol layers are mapped to an underlying G.9954v2 flow by the protocol convergence layer. The result of the mapping is a reference to the *flow descriptor* that describes the properties of the flow to which a packet belongs. The default mapping of a packet is to the *default flow (Flow_ID = 0)* using a priority-based media access method.

Packets are mapped to flows using *traffic classifiers*. A *traffic classifier* defines a protocol specific set of selection criteria that are applied to incoming packets in order to test their association with a specific flow. Multiple classifiers may be active at the same time in a convergence layer. Traffic classifiers are processed in an order that is defined by their relative priority.

Traffic classifiers may be installed in the convergence layer at the flow source by upper-layer management operations, during network admission or through flow set-up/modification signalling operations.

For further details on the establishment of convergence layer traffic classification filters, see the description of the network admission and flow signalling protocols in clause 11.

10.7 Flow signalling protocol

To establish a flow with well-defined QoS parameters, as defined in clause 10.4, a *flow* shall be "set up" between source and destination devices. *Flow set-up* may be initiated by either the flow source or destination devices.

If a flow requires QoS guarantees, bandwidth for the flow shall be allocated in the MAP. To allocate bandwidth for a flow in the MAP, the master shall be informed of the set-up of the flow.

Flow set-up shall be performed using the *flow signalling protocol and* will involve a message exchange sequence, between the initiating node and the target node, whereby the initiator shall specify the properties of the flow (as defined in Table 10-2) to be set up.

To set up a flow with QoS Contracts, the device at the source of the flow shall notify the master using the same *flow signalling protocol*. The master shall perform admission control on the flow set-up request in order to determine whether sufficient media resources exist. If admitted, the master scheduler shall allocate transmission opportunities in the MAP that meet the QoS requirements of the requested flow. If the flow is not admitted, the master shall signal the error to the source of the flow set-up request. It is implementation-dependent as to the behaviour of a device upon failure to set up a flow.

NOTE – An implementation may tear down a flow if bandwidth cannot be reserved for it by the master. Alternatively, an implementation may continue to transmit data over the flow channel although fixed bandwidth cannot be reserved and other QoS parameters cannot be guaranteed.

During the lifetime of a flow, its specification may need to be modified in order to accommodate for changing (variable) bit-rate requirements, resource constraints (e.g., latency/jitter buffers) and achievable payload rates. Modifications to a flow specification are signalled between devices at the flow endpoints. In addition, if modifications to flow properties are such that they may affect media resource allocation in the MAP, the master shall be signalled by the device at the source of the flow. Signalling shall be performed using the *flow modification* protocol. The master shall perform admission control on the requested flow modification.

During a flow modification, the QoS parameters that affect media resource allocation in the MAP are defined as follows:

– Maximum, Average, Minimum Data Rates (clause 10.4.12) – Change as a consequence of traffic statistics collected at flow source.

– Payload Encoding (clause 10.4.14) – Caused by changing line conditions and detected by rate negotiation.

– Maximum latency or jitter (clauses 10.4.5 and 10.4.6) – Caused by changes in memory resource constraints at source or destination of flow.

– Nominal packet size (clause 10.4.11) – Caused by variableness in nature of packets in traffic stream.

Other flow properties are static and do not change during the lifetime of a flow.

When a flow is no longer needed or in use, it shall be torn down. *Flow teardown* shall be performed by the convergence layer either explicitly, in response to a "teardown" request from upper layers or implicitly through the aging out of inactive flows. If a flow has media resources allocated to it (i.e., transmission opportunities in the MAP), the master shall be signalled of the flow teardown by the device at the source of the flow. The master shall be informed using the *flow teardown* protocol. When a flow is torn down, the resources it binds shall be freed.

For a full description of the flow signalling protocol, see clause 11.16.3.

10.8 Admission control

Admission control shall be performed by the master when a request is received to add a new flow or to change the properties of an existing flow to more stringent QoS parameters.

Admission control involves the following functions:

1) bandwidth testing;

2) latency/jitter bound testing.

Upon receiving a *Flow Set-up* or *Modify* request, the master shall check for the availability of sufficient media resources (i.e., unallocated media time) in order to meet the flows' throughput demands given the flows' *minimum, maximum* and *average data rate* requirements and given the *payload encoding* required on the channel. Furthermore, the master shall verify that the location of available transmission opportunities is such that allocation of transmission opportunities to the flow will allow the flow to meet its requirements for latency and jitter bounds.

If admission control testing results in the failure of either or both of the admission control tests, the master shall return an ERROR in the *flow signalling "Response"* frame.

A flow's latency/jitter specification represents a maximum allowable bound, and consequently, the master may allocate media resources in a manner such that it exceeds the original latency and jitter specification for the flow.

In order to meet QoS constraints of a flow specification, the master may need to reorganize the location and size of allocated transmission opportunities for other flows. This may be needed in order to "make space" for the addition of the new flow. The master should attempt to accommodate flows within the existing available media time before considering reorganization of other flows in order to localize the effect of the change in the media access plan and in order NOT to introduce unnecessary (albeit transient) latency and jitter in other flows.

If admission control testing succeeds and the requested flow can be set up or modified according to the specified parameters, the master shall reserve media resources for the flow and advertise the reservation in the MAP.

For further information on the flow signalling protocol, see clause 11.16.

10.9 QoS support levels

A G.9954v2 device shall provide one of the following levels of QoS support:

1) G.9954v1 QoS level;
2) Best-effort QoS level;
3) Priority-based QoS level;
4) Full QoS level.

QoS support levels are organized into an ordered hierarchy, from lowest to highest support levels, where each level includes all the functionality of the level preceding it. For example, best-effort QoS support includes G.9954v1 QoS support and priority-based QoS support includes both best-effort and G.9954v1 QoS support.

The semantics of each QoS support level are described below in the following subclauses.

10.9.1 G.9954v1 QoS level

G.9954v1 QoS support level defines G.9954v1 legacy QoS mode. In this mode, QoS support is provided by the parameter-based QoS model using fixed bandwidth allocation. Media access is performed only within contention-free TXOPs of fixed length. Media access based on sub-burst slots is not supported. A G.9954v2 master shall not use the explicit TXOP or group separator in a MAP that includes a G.9954v1 legacy device.

10.9.2 Best-Effort QoS level

Best-effort QoS support includes G.9954v1 QoS and also supports media access based on sub-burst slots. A G.9954v2 device providing best-effort QoS support (also known as a G.9954v2 best-effort QoS device) shall perform media access with CTXOPs using sub-burst slots and shall support at least one priority group containing low priority sub-burst slots.

A G.9954v2 master shall assign a G.9954v2 best-effort QoS device a single sub-burst-slot transmission opportunity within the lowest priority group within a CTXOP. The G.9954v2 master shall not use explicit TXOP or group separators within a CTXOP that includes a G.9954v2 best-effort QoS device and shall not use next group sub-burst slots.

10.9.3 Priority-based QoS level

Priority-based QoS support is a superset of best-effort QoS support and defines the support for multiple priority groups of sub-burst slots according to the model described in clause 10.3.

A G.9954v2 device supporting priority-based QoS shall support at least three priority groups. A G.9954v2 master shall assign a sub-burst slot to a G.9954v2 priority-based QoS device in each of the priority groups that the device participates. The G.9954v2 master shall not use the explicit TXOP or group separator in a MAP that includes a G.9954v2 priority-based or best-effort QoS device and shall not use next group sub-burst slots.

10.9.4 Full QoS level

Full QoS is a superset of priority-based QoS and includes support for multiple groups of sub-burst slots; the assignment of multiple sub-burst slots to a device within a group; explicit TXOP and Group separators including explicit proceed and abort event transitions; next group and next MAP sub-burst slots. This represents the highest level of QoS support.

11 Link-layer protocol specification

11.1 Overview

This Recommendation specifies the link layer format to be used for G.9954v2 stations. In addition, for link layer frames that are identified by IEEE assigned Ethertype value (0x886c) in the Type/Length field of the frame, these frames carry link control functionality, and definitions for this functionality are provided in this Recommendation.

The LLC sublayer is responsible for performing link control functions. In particular, it is responsible for managing information concerning network connections, for enforcing Class of Service (CoS) and Quality of Service (QoS) constraints defined for the various service flows and for ensuring robust data transmission using Rate Negotiation, optional Reed-Solomon coding techniques and ARQ (Automatic Repeat ReQuest) techniques.

The following link control functions are defined in G.9954v2 link layer:

- Rate negotiation;
- Link integrity;
- Capability announcement;
- Limited automatic repeat request (LARQ);
- Frame bursting capability;
- MAC cycle synchronization;
- Registration;
- Flow signalling;
- Master selection;
- Certification protocol;
- Reed-Solomon encapsulation;
- Timestamp reporting.

These link functions use control frames to carry protocol messages between stations. G.9954v2 includes a standardized mechanism for link layer network control and encapsulation. Individual sub-types further distinguish control frames. The link control entities may be implemented in hardware or driver software. Link control frames are not seen by layer 3 (IP) of the network stack, and shall not be bridged between network segments.

11.1.1 Minimal link protocol support profile for G.9954v2 link protocols

The minimal link protocol support profile for G.9954v2 link protocols allows less complex implementations of the G.9954v2 specification. While all control protocols serve an important function in the operation of the network, it is possible to implement a minimal subset of Link Layer Protocols that are compatible with fully functional implementations and does not detract from the overall performance of other stations. The shorter name, "minimal profile", will be used in the rest of this Recommendation.

Full support of all the link protocols, called the Full Link Protocol Support Profile, is assumed throughout the rest of this Recommendation unless Minimal Profile is explicitly mentioned.

A G.9954v2 device supporting a Minimal Profile shall support the following G.9954v2 link-layer protocols:

- Rate negotiation;
- Link integrity;
- Capability announcement;
- MAC cycle synchronization;
- Frame bursting;
- Certification protocol;
- Minimal LARQ;
- Network admission.

Such a device is able to synchronize with the master-generated synchronous MAC cycle, to register with the Master in the network and to contain its transmissions within TXOPs assigned to it and defined in the master-generated media access plan (MAP). Media access is performed according to the media access rules specified in clause 8. Frame bursting is used to more efficiently utilize media time. Rate negotiation is performed over logical channels between source and destination devices.

11.1.2 G.9954v2 device supporting QoS contracts

In addition to the link-layer protocols in the minimal profile (above), a G.9954v2 device supporting QoS contracts shall also support the following G.9954v2 link-layer protocol:

- Flow signalling (endpoint device).

Such a device shall be able to perform all the functions of a minimal profile G.9954v2 device and shall also be able to manage *flows* with QoS contracts, request bandwidth reservations for *flows* and perform rate negotiation and LARQ at the level (granularity) of a flow.

11.1.3 G.9954v2 master-capable device

A G.9954v2 device that is capable of becoming a network master, called a master-capable device for short, shall, in addition to the link-layer protocols described above also support the following G.9954v2 link-layer protocols:

- Dynamic master selection (according to state diagram in Figure 8-1);
- MAC cycle generation;
- Flow signalling (master device);
- Timestamp reporting (Master clock reference).

A master-capable device shall be able to assume the role of master in a master-less network and to generate periodic MAC cycles for synchronous operation. It shall be able to engage in flow signalling and to convert flow signalling requests to scheduler input. It shall also implement dynamic master selection according to the state diagram described in Figure 8-1. A master-capable device shall be also able to act as a master clock reference, by periodically advertising its internal clock allowing endpoint devices to synchronize their local clocks to the master's internal clock.

11.1.4 G.9954v2 optional link-layer protocols

The following link-layer protocols are optional for all G.9954v2 devices:

- Full dynamic master selection (see clause 11.15);
- Timestamp reporting (endpoint slave);
- Reed-Solomon encapsulation.

11.2 Basic link layer frame format

The basic link layer frame format is described in Table 11-1.

Table 11-1 – Basic link-layer format

Field	Length	Explanation
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	Ethernet ethertype. Arbitrary value. If equal to 0x886c (HNT Link Protocol Frame assigned by IEEE), then frame is for link protocol control frame.
Data	Variable	Payload data
Pad	Variable	Padding (if required to meet minimum length frame)
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence, described in clause 11.2.1

The G.9954v2 basic link-layer frame format is based on the IEEE Std 802.3 Ethernet frame format (not including the IEEE Std 802.3 preamble and SFD fields) with an additional CRC-16 frame check sequence. The HNT frame bit fields starting with the destination address (DA) field and ending with the FCS field are identical to the corresponding fields described in IEEE Std 802.3 (see Figure 11-1), and are referred to as the link-level Ethernet frame. The bits of a PHY-level Ethernet frame have an Ethernet preamble and start-frame-delimiter (SFD) bits prepended to the link-level frame; these bits are not present in G.9954v2 frames.

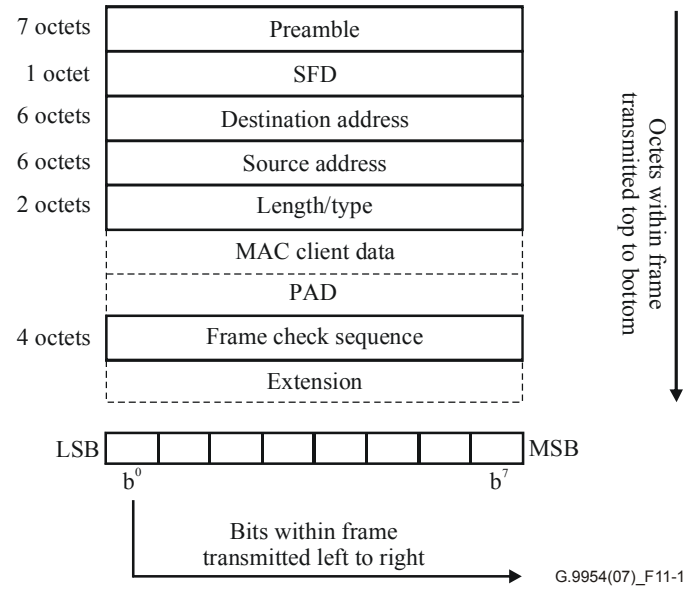

Figure 11-1 – Ethernet PHY-level frame format

It is intended that IEEE assigned Ethernet MAC addresses shall be used for destination address (DA) and source address (SA).

The link-level Ethernet frame consists of an integer number of octets.

An additional CRC-16 shall be appended after the frame check sequence, as described in clause 11.2.1

In the frame formats defined above, before transmission, the link control frame shall be converted into a G.9954v2 physical layer frame by adding preamble, frame control, PAD and EOF as shown in Figure 6-2.

11.2.1 CRC-16

A 16-bit cyclic redundancy check (CRC) shall be computed as a function of the contents of the (unscrambled) Ethernet link-level frame in transmission order, starting with the first bit of the DA field and ending with the last bit of the FCS field. The encoding is defined by the following generating polynomial.

$$G(x) = x^{16} + x^{12} + x^5 + 1$$

Mathematically, the CRC value corresponding to a given frame is defined by the following procedure.

The first 16 bits of the frame in transmission order are complemented.

The n bits of the frame in transmission order are then considered to be the coefficients of a polynomial $M(x)$ of degree $n - 1$. (The first bit of the destination address field corresponds to the $x^{(n-1)}$ term and the last bit of the FCS field corresponds to the x^0 term.)

$M(x)$ is multiplied by x^{16} and divided by $G(x)$, producing a remainder $R(x)$ of degree ≤ 15.

The coefficients of $R(x)$ are considered to be a 16-bit sequence.

The bit sequence is complemented and the result is the CRC.

The 16 bits of the CRC shall be placed in the CRC-16 field so that x^{15} is the least significant bit of the first octet, and the x^0 term is the most significant bit of the last octet. (The bits of the CRC are thus transmitted in the order x^{15}, x^{14}, .. x^1, x^0.)

NOTE – The HNT CRC-16, in conjunction with Ethernet's FCS, provides more protection from undetected errors than the FCS alone. This is motivated by environmental factors that will often result in a frame error rate (FER) several orders of magnitude higher than that of Ethernet, making the FCS insufficient by itself.

11.3 Link-layer control frames

Link-layer frames with ethertypes equal to 0x886c are link-layer control frames. These frames are not based on the IEEE Std 802.3 Ethernet frame format. There are two basic formats for a link control frame: a long subtype and a short subtype. The long subtype format is provided for future specified control frames where the amount of control information exceeds 256 octets. The control and encapsulation frames described in this Recommendation use the short subtype format.

In the frame formats defined in Table 11-2, before transmission the link control frame shall be converted into a physical layer frame by adding preamble, frame control, PAD and EOF as shown in Figure 6-2.

11.3.1 Short format

The short-format link control frame is defined in Table 11-2. The SSVersion field should be used by all protocols using the Short Format Link Control Frame header. This field specifies which format version of the control information is used. This allows future extension of each SSType.

Table 11-2 – Short-format link control frame

Field	Length	Explanation
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link protocol frame assigned by IEEE)

Table 11-2 – Short-format link control frame

Field	Length	Explanation
SSType	1 octet	0 – 127 assigned by HNT 0 – Reserved 1 – Rate request control frame 2 – Link integrity short frame 3 – Capabilities announcement 4 – LARQ 5 – Vendor-specific short format type 6 – Frame bursting 7 – Dynamic master selection 8 – Timestamp report indication 9 –127 Reserved Values 128-255 correspond to the long subtype
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field (or the first octet following SSLength if it is not defined as SSVersion) and ending with the second (last) octet of the Next Ethertype field. Min is 2 and max is 255.
SSVersion	1 octet	Version number of the control information
Control Data	0-252 octets	Control information
Next Ethertype	2 octets	Ethertype/length of next-layer protocol; 0 if none
Payload data	Variable	If not encapsulating frame, then this field is 0 octet long.
PAD	41-0 octets	Padding required to meet minimum if data < 41 octets
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

SSLength must be checked to ensure that enough control information is present. New, backwards-compatible frame formats may contain additional fixed data fields, but shall always contain the fixed fields specified in earlier formats. Protocol implementations must interpret all supported SSType frames using the latest supported SSVersion that is less than or equal to the SSVersion indicated in the received frame. Unknown fields shall be ignored. Encapsulated data from unsupported (newer) SSVersions of supported encapsulating SSType frames shall be passed to the layer above. Protocol extensibility is addressed in clause 11.10.

The Next Ethertype field is required for all short-format link control frame headers. Among other things, it supports backward compatibility by enabling receivers to always strip short format link layer headers. If the Next Ethertype field is zero, then the frame is a basic control frame and should be dropped after processing the control information it contains. The Next Ethertype field shall be the last two octets of the control header. The position of the Next Ethertype field in the frame shall be determined using the SSLength field in order to ensure forward compatibility.

If the Next Ethertype field is non-zero, then the frame is an *encapsulating* control frame. An encapsulated data frame is an encapsulating control frame with any Next Ethertype field not matching x0000 or 0x886c. G.9954v2 receivers shall be capable of removing at least one encapsulating short-format link control frame header from any received encapsulated data frame. When Next Ethertype is restricted by the specification to the value x0000 for a specific link-layer control frame SSType or LSType, then encapsulation of data frames is not allowed when using that

link-layer control frame type. The only link-layer frame type that supports encapsulation of data frames is the LARQ frame.

If the SSType is not understood by the receiver (a fact possibly announced via future CSA options), then the frame shall be dropped. All nodes are required to understand the LARQ SSType (although they are not required to implement LARQ). Protocol extensibility is addressed in clause 11.10.

The header and trailer for standard Ethernet frames are shaded in gray, in order to highlight the formats of the control information frames.

11.3.2 Long format

The long-format link control frame is defined in Table 11-3. An LSVersion, similar to SSVersion, should be used by all long-format subtypes. A Next_Ethertype field is required for all long-format subtypes. If long-format subtypes (LSType values) are not understood by the receiver (a fact possibly announced via future CSA options), then they shall be dropped. Processing requirements with respect to forwards compatibility, dropping of unknown frame types with Next_Ethertype = 0, and removal of long-format headers with Next_Ethertype != 0, are identical to those for short-format control frame headers.

Table 11-3 – Long-format link protocol frame

Field	Length	Explanation
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link protocol frame assigned by IEEE)
LSType	2 octets	32768 Reserved 32769 Vendor-specific long-format 32770 Certification protocol 32771 Reed-Solomon encapsulating header 32772 MAP Synchronization Protocol 32773 Network Admission Protocol 32774 Flow Signalling Protocol 32775 to 65534 Reserved, assigned by HNT 65535 Reserved
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field (or the first octet following SSLength if it is not defined as SSVersion) and ending with the second (last) octet of the Next Ethertype field. Min is 2 and max is 65535.
LSVersion	1 octet	Version number of the following protocol information
Data	LSLength – 3 octets	LSType protocol-dependent data
Next Ethertype	2 octets	Ethertype/length of next layer protocol; 0 if none.
Payload Data	Variable	If not encapsulating frame, then this field is 0 octet long.
PAD	42-0 octets	PAD to minimum size if needed
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

11.3.3 Order of transmission

Network transmission order of frame fields is from the top to the bottom of each table.

Within a field, the MSByte of the field shall be the first octet of the field to be transmitted, with the LSBit of each octet transmitted first. Subsequent bytes within a field are transmitted in decreasing order of significance.

When subfields are indicated in any table, the ordering shown is decreasing significance from the top to the bottom of the table.

11.4 Rate negotiation control function

The PHY payload modulation can use 2- to 8-bits per symbol constellations and one of several defined *bands* which are associated with symbol rates. For some bands 8-, 9- and 10-bits per symbol constellations optionally exist; see clause 6.3.3.6.

The payload encoding (PE) that can be achieved is a function of the channel quality between source and destination, and the channel quality generally differs between each pair of stations depending on the wiring topology and specific channel impairments. Therefore the rate negotiation function in a destination station uses rate request control frames (RRCF) to provide information to a source station as to the payload encoding that the source station should use to encode future frames sent to this destination, and to generate test frames to assist a receiver in selecting the most appropriate band to use.

The policy that the destination station uses to select the desired payload encoding and the policy it uses to decide when to transmit rate request control frames are implementation dependent. Stations should avoid transmission policies that can result in excessive RRCF traffic.

Rate negotiation in G.9954v2 is defined over a logical channel (see clause 11.4.3, Terms and definitions, and clause 11.4.3.1) where a logical channel is defined by the tuples { Source Address, Destination Address, Priority } and/or { Source Address, Destination Address, Flow ID }. This allows a fine degree of control over the selected rate for a logical channel by allowing different rates to be negotiated per logical channel, even when the different channels are over the same source-destination pair. Since each logical channel represents a different service or flow, possibly with distinct BER/PER requirements, rate negotiation is adaptive per service.

The goal of rate negotiation is to select the payload encoding that achieves the highest raw bit rate while still meeting the BER/PER requirements for the logical channel.

11.4.1 Rate request control frame format

The RRCF specifies a maximum constellation (bits per symbol) that the receiver (ReqDA) wishes to be used in a given band, or indicates that a given band is not supported. See Table 11-4.

Table 11-4 – Rate request control frame definition

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	=SUBTYPE_RATE (1)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. The minimum value of SSLength is 18 for SSVersion 0.
SSVersion	1 octet	= 0

Table 11-4 – Rate request control frame definition

Field	Length	Meaning
OpCode	1 octet	Operation code for this control message. See Table 11-6 for definitions.
NumBands	1 octet	Number of bands specified in this control. Each band has a two-octet descriptor. The bands refer to the modulation type: *Band* *Reference* 1 Reserved for legacy systems 2 Reserved for legacy systems 3 2-Mbaud modulation 4 4-Mbaud modulation 5 8-Mbaud modulation 6 16-Mbaud modulation 7 32-Mbaud modulation NumBands shall be 6 or 7 on transmission for G.9954v2 stations, and stations shall ignore band entries beyond Band7 on receive if NumBands is larger than 7. The value 0 is not allowed. Values greater than 6 can be ignored if the G.9954v2 station does not support 32-Mbaud modulation.
NumAddr	1 octet	Number of addresses specified in the payload of this control message. NumAddr may be zero. The SA in the Ethernet header is always used, and is referred to hereafter as RefAddr0.
Band1_PE	1 octet	The PE value that should be used to send data when band 1 is selected
Band1_rank	1 octet	The rank order of the ReqDAs' preference for this band 1 is highest preference, and the other bands within the spectral mask are assigned successively larger rank values.
. . .		Additional instances of band information
BandN_PE	1 octet	The PE value that should be used to send data when band N is selected
BandN_rank	1 octet	The rank order of the ReqDAs' preference for this band 1 is highest preference, and the other bands within the spectral mask are assigned successively larger rank values.
RefAddr1	6 octets	Optional. Present if NumAddr ≥ 1. The second MAC address for which the rates are being specified; only broadcast and multicast address types are allowed.
RefAddr2	6 octets	Optional. Present if NumAddr ≥ 2. The third MAC address for which the rates are being specified; only broadcast and multicast address types are allowed.
. . .		[additional instances of RefAddr, until the number of RefAddr fields equals NumAddr]
[Additional TLV extensions]		Flow ID/Priority extension information. See clause 11.4.2.
Next Ethertype	2 octets	= 0
Pad		To reach minFrameSize if required
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

Additional bands may exist in future versions of this Recommendation, and can be described with band descriptors {PE, rank} added after band 7. If additional bands are present, their descriptors will appear between BandN_Rank and RefAddr1, and G.9954v2 stations take their presence into account when determining the location of the RefAddr list.

G.9954v2 stations shall ignore band specification beyond NumBands = 7. If a receiver does not specify a band in an RRCF, or specifies a PE of 0 for a band, then transmitters shall not use that band. In order to allow unambiguous determination of which bands are present as future bands are added, intervening unsupported bands must use PE = 0 to indicate non-use. Bands may only be unspecified if no other band information follows.

The NumBands and NumAddr fields are placed next to each other so that all the fixed fields can be referenced at known offsets in the frame.

Table 11-5 describes the assigned values that may appear in the band description entries in the rate request control frame.

Table 11-5 – PE values for rate request control frames

PE	Data rate	Meaning
0	N/A	Means this band is not supported
1-32	N/A	Reserved for legacy systems
33	4 Mbit/s	2 Mbaud, 2 bits per symbol
34	6 Mbit/s	2 Mbaud, 3 bits per symbol
35	8 Mbit/s	2 Mbaud, 4 bits per symbol
36	10 Mbit/s	2 Mbaud, 5 bits per symbol
37	12 Mbit/s	2 Mbaud, 6 bits per symbol
38	14 Mbit/s	2 Mbaud, 7 bits per symbol
39	16 Mbit/s	2 Mbaud, 8 bits per symbol
40	N/A	Reserved
41	8 Mbit/s	4 Mbaud, 2 bits per symbol
42	12 Mbit/s	4 Mbaud, 3 bits per symbol
43	16 Mbit/s	4 Mbaud, 4 bits per symbol
44	20 Mbit/s	4 Mbaud, 5 bits per symbol
45	24 Mbit/s	4 Mbaud, 6 bits per symbol
46	28 Mbit/s	4 Mbaud, 7 bits per symbol
47	32 Mbit/s	4 Mbaud, 8 bits per symbol
48	N/A	Reserved
49	16 Mbit/s	8 Mbaud, 2 bits per symbol
50	24 Mbit/s	8 Mbaud, 3 bits per symbol
51	32 Mbit/s	8 Mbaud, 4 bits per symbol
52	40 Mbit/s	8 Mbaud, 5 bits per symbol
53	48 Mbit/s	8 Mbaud, 6 bits per symbol
54	56 Mbit/s	8 Mbaud, 7 bits per symbol
55	64 Mbit/s	8 Mbaud, 8 bits per symbol
56	N/A	Reserved
57	32 Mbit/s	16 Mbaud, 2 bits per symbol

Table 11-5 – PE values for rate request control frames

PE	Data rate	Meaning
58	48 Mbit/s	16 Mbaud, 3 bits per symbol
59	64 Mbit/s	16 Mbaud, 4 bits per symbol
60	80 Mbit/s	16 Mbaud, 5 bits per symbol
61	96 Mbit/s	16 Mbaud, 6 bits per symbol
62	112 Mbit/s	16 Mbaud, 7 bits per symbol
63	128 Mbit/s	16 Mbaud, 8 bits per symbol
64		Reserved
65	64 Mbit/s	32 Mbaud, 2 bits per symbol
66	96 Mbit/s	32 Mbaud, 3 bits per symbol
67	128 Mbit/s	32 Mbaud, 4 bits per symbol
68	160 Mbit/s	32 Mbaud, 5 bits per symbol
69	192 Mbit/s	32 Mbaud, 6 bits per symbol
70	224 Mbit/s	32 Mbaud, 7 bits per symbol
71	256 Mbit/s	32 Mbaud, 8 bits per symbol
72-159	N/A	Reserved
160	16 Mbit/s	2 Mbaud, 8-round constellation; 8 bits per symbol
161	18 Mbit/s	2 Mbaud, 9-round constellation; 9 bits per symbol
162	20 Mbit/s	2 Mbaud, 10-round constellation; 10 bits per symbol
163-167	N/A	Reserved
168	32 Mbit/s	4 Mbaud, 8-round constellation; 8 bits per symbol
169	36 Mbit/s	4 Mbaud, 9-round constellation; 9 bits per symbol
170	40 Mbit/s	4 Mbaud, 10-round constellation; 10 bits per symbol
171-175	N/A	Reserved
176	64 Mbit/s	8 Mbaud, 8-round constellation; 8 bits per symbol
177	72 Mbit/s	8 Mbaud, 9-round constellation; 9 bits per symbol
178	80 Mbit/s	8 Mbaud, 10-round constellation; 10 bits per symbol
179-183	N/A	Reserved
184	128 Mbit/s	16 Mbaud, 8-round constellation; 8 bits per symbol
185	144 Mbit/s	16 Mbaud, 9-round constellation; 9 bits per symbol
186	160 Mbit/s	16 Mbaud, 10-round constellation; 10 bits per symbol
187-191	N/A	Reserved
192	256 Mbit/s	32 Mbaud, 8-round constellation; 8 bits per symbol
193	288 Mbit/s	32 Mbaud, 9-round constellation; 9 bits per symbol
194	320 Mbit/s	32 Mbaud, 10-round constellation; 10 bits per symbol
195-255	N/A	Reserved

Table 11-6 describes the values that may appear in the OpCode entry in the Rate Request Control Frame.

Table 11-6 – OpCode values for rate request control frames

OpCode	Meaning
0	Rate change request
1	Rate test request
2	Rate test reply
3-255	Reserved

11.4.2 Receiver indication of logical channel TLV extension to the LCP SUBTYPE_RATE subtype

In order to support rate negotiation over a logical channel defined by { Source Address, Destination Address, Priority } or { Source Address, Destination Address, Flow ID } a TLV extension to the Rate Request Control Frame (RRCF) is defined.

Two additional parameters are included for each RefAddr defined in the RRCF. These parameters indicate the Priority or Flow identifier for the logical channel whose source address is the DA in the Ethernet header of the RRCF frame and whose Destination Address = RefAddr<n>.

There are three types of logical channels defined for rate negotiation. These are as follows:

1) Simple Channel – Defined by the { Source Address, Destination Address } pair. No additional channel identifier is required. For a simple channel a PER = 1e-4 shall be used as the minimum PER parameter for rate selection.
2) LARQ Priority Channel – Defined by the tuple { Source Address, Destination Address, Priority }. For a LARQ priority channel a PER = 1e-2 shall be used as the minimum PER parameter for rate selection.
3) Flow Channel – Defines the logical channel identified by { Source Address, Destination Address, Flow ID }. The BER/PER used as input for rate selection is defined in the flow parameters negotiated and signalled between source and destination during flow signalling. For further information on flow parameters and the flow signalling protocol, see clause 11.16.

The logical channel identifier TLV extension is optional. If, however, the extension is found in the RRCF frame, there shall be a pair of parameters (RefChanType<n>, RefChanId<n>) for each RefAddr in the frame (i.e., NumAddr + 1 entries). The first entry shall correspond to RefAddr0 and the last entry to $RefAddr_{NumAddr}$.

Table 11-7 – Flow ID/priority extension information

Field	Length	Meaning
SETag	1 octet	= RRCF_CID_TAG(3), Optional logical channel identifier
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. Must be (NumAddr+1) * 2. Minimum is 4.
RefChanType0	1 octet	The logical channel type defined by (DA, RefAddr0, RefId0). The channel type defines the semantics of RefId as follows: 0 – Simple channel; RefId is undefined. 1 – LARQ priority channel; RefId is interpreted as Priority. 2 – Flow channel; RefId is interpreted as FlowId.

Table 11-7 – Flow ID/priority extension information

Field	Length	Meaning
RefId0	1 octet	The RefId according to the semantics defined by RefChanType
RefChanType1	1 octet	The logical channel type defined by (DA, RefAddr1, RefId1). The channel type defines the semantics of RefId as follows: 0 – Simple channel; RefId is undefined. 1 – LARQ priority channel; RefId is interpreted as Priority. 2 – Flow channel; RefId is interpreted as FlowId.
RefId1	1 octet	The RefId according to the semantics defined by RefChanType1
…		[additional instances of Channel Identification information, until the number of channels equals NumAddr + 1. The channel identification table is optional as indicated by TLV extension mechanism. If the TLV extension does not exist, all logical channels are assumed simple channels. Otherwise, there must be an explicit channel identification entry for each defined RefAddr from RefAddr0..$RefAddr_{NumAddr}$]

11.4.3 Terms and definitions

Table 11-8 – Terms and definitions

Term	Definition
band specification	A payload encoding (PE) and rank associated with a given band. A band is equivalent to a symbol rate. Seven bands are defined in G.9954v2.
logical channel, channel	A flow of frames from a sender to one or more receivers on a single network segment, consisting of all the frames with a single combination of: 1) DA and SA, or 2) DA, SA and Priority, or 3) DA, SA and Flow ID. Each combination represents a different channel type referred to as a Simple, LARQ Priority, and Flow Channels respectively.
receiver	A station that receives frames sent on a particular channel. If the destination is a unicast address, there is at most one receiver. If the destination is a group address (including broadcast), there may be many receivers.
receiver PE	The preferred PE to be used on this channel, as determined by the receiver.
RRCF	Rate Request Control Frame. Sent from the receiver to the sender to effect a change in PE.
RefAddr0	The SA in the Ethernet header of the RRCF frame. This is the DA of the receiver (for the channel), and is always used by the channel sender as the first RefAddr processed.

Table 11-8 – Terms and definitions

Term	Definition
RefAddr1..RefAddr<n>	Other addresses including Broadcast and Multicast addresses for which the receiver is indicating rate information to the sender. The channel receiver's station address (RefAddr0) should not be put in the list of additional RefAddr's. NOTE – At least one RefAddr field is necessary to support rate negotiation for Broadcast and Multicast addresses since these cannot be used as the source address in the Ethernet header.
sender	The sending station for a channel, usually the station owning the source MAC address.
sender PE	The preferred PE associated with a channel, as noted by the sender.

11.4.3.1 Channels

Rate negotiation is defined over simplex logical channels. A separate channel is defined for each combination of Ethernet 1) DA, SA or 2) DA, SA and Priority or 3) DA, SA and Flow ID. The different combinations represent different channel types and are referred to as Simple, LARQ Priority and Flow Channels respectively. There is no explicit channel set-up procedure for Simple and LARQ priority channels. A new channel is implicitly defined when a packet is received from a new SA or sent to a new DA. Flow channels are set up by flow signalling between channel source and destination devices. For further information on the flow signalling protocol, see clause 11.16.

Each channel has a single sender but can have multiple receivers. Receivers operate independently.

11.4.3.2 Sending RRCFs

Rate control frames (all OpCodes) should be sent with a priority corresponding to link-layer priority 7. RRCFs shall never be sent with a link-layer priority of 6. RRCFs may be sent with a lower link-layer priority, from the set [5,4,3,0]. However, the link-layer priority of an RRCF shall never be lower than the highest link-layer priority received in the last two seconds from the station to which the RRCF is being sent. Rate change requests (OpCode = 0) shall always be sent with an encoding of 2 Mbaud at 2 bits per symbol (PE = 33) when the channel source is a G.9954v2 device. Selection of the encoding for rate test request frames and rate test reply frames is described below.

11.4.3.3 Interval timer

Each station should maintain a timer with a period of 128 seconds. There should be no attempt to synchronize this timer between stations. Receipt or transmission of any frames should not modify the timer. The timer interval is used when determining which nodes have been actively sending to multicast and broadcast addresses (see clause 11.4.4.2) and when sending reminder RRCFs in reference to multicast and broadcast addresses (see clause 11.4.5.1).

11.4.4 Sender operation

11.4.4.1 Sender – Transmit data frame

Access the logical channel state information to determine the sender PE to use for transmission. Create the channel if necessary, and default the sender PE to PE = 33 (2 Mbaud 2 bits per symbol) if the destination node is G.9954v1 or G.9954v2. Logical channel state information includes the node type, the sender PE and the receiver PE for each band for which this information has been specified.

11.4.4.2 Sender – Receive rate change request (RRCF OpCode 0)

For each of the RefAddrs in the RRCF (starting with RefAddr0, the SA of the RRCF frame), access the logical channel state information, if any exists, corresponding to the RefAddr and optionally RefId (further referenced by the tuple (RefAddr, [RefId]), where the square brackets indicate an *optional* element), and update the sender PE according to the band specification in the RRCF. If no logical channel state information exists for (RefAddr0, [RefId0]), the station should create a new logical channel state entry and initialize the sender PE according to the band specification in the RRCF. If no logical channel state information exists for additional (RefAddrs, [RefIds]), the station may either ignore those addresses or create new logical channel state entries and initialize the sender PE according to the band specification in the RRCF.

For multicast addresses and the broadcast address, senders should use a payload encoding (PE) that is receivable by all nodes actively listening to that address. Sender stations may enforce a minimum PE which they will use to transmit to a given multicast channel, based on application-level information about QoS. It is desirable to send at the highest rate supported by the channel. Hence, if a RefAddr is a multicast address or the broadcast address, the sender should use the PE value which yields the highest raw bit rate, but which is not greater than any of the band specifications provided by the nodes actively listening to that address. Active multicast listeners shall be defined as any stations which have, in either of the last two 128-second intervals, either:

1) sent any frame to the multicast address; or

2) sent a RRCF to this station with the multicast address listed in the RefAddr list.

Active broadcast listeners shall be defined as any stations which have, in either of the last two 128-second intervals, either:

1) sent any frame to the broadcast address; or

2) sent a RRCF to this station with the broadcast address listed in the RefAddr list.

In a master-controlled network, the sender (i.e., the station at the source of the logical channel) shall update the master of the change in the negotiated PE on a flow channel. The master shall be notified of the change in the flow's PE parameter by sending a Flow Modify Request with the new PE for each flow identified in the RRCF message.

This protocol is illustrated in Figure 11-2.

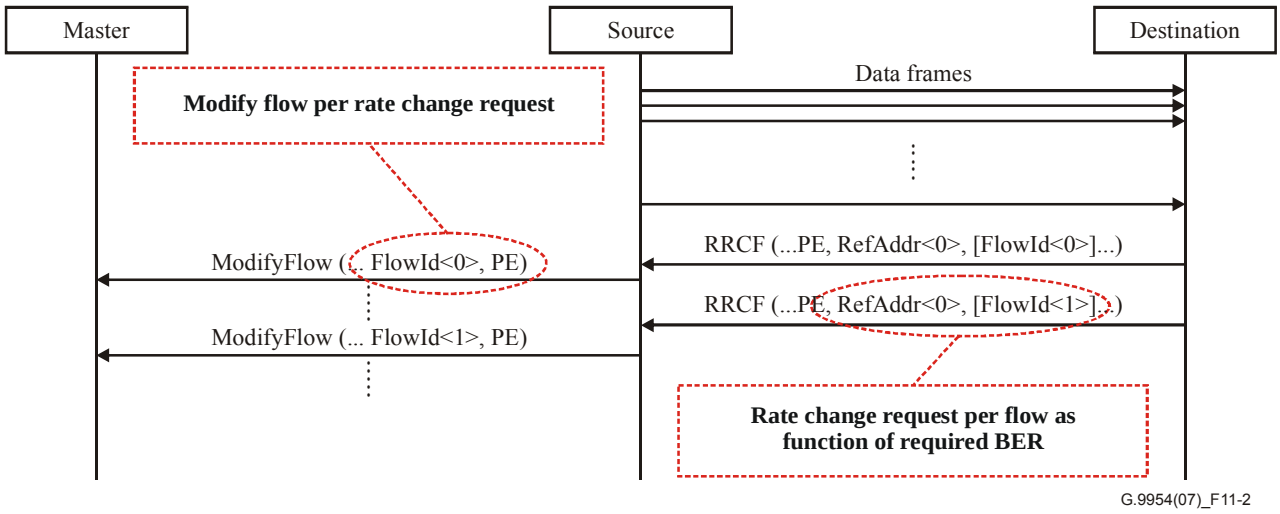

Figure 11-2 – Rate negotiation protocol

11.4.4.3 Sender – Receive rate test request frame (RRCF OpCode 1)

For each supported band encoding, generate a rate test reply frame (RRCF OpCode 2) to the requestor encoded using the specified payload encoding. The contents of the RRCF shall be the current logical channel state info.

Support for rate test request frames is only required in all G.9954v2 stations.

11.4.4.4 Sender – Active HNT nodes

An active HNT node is any station from which a frame has been received in either of the last two 128-second intervals.

11.4.5 Receiver operation

11.4.5.1 Receiver – Receive a frame

The following baseline algorithm for limiting the number of RRCFs should be employed. Alternative implementations shall not generate more RRCFs than the suggested implementation. Nodes that are interested in receiving frames of a specific multicast address or of the broadcast address shall provide a mechanism to ensure that all sources of frames sent to that multicast address (or the broadcast address, as appropriate) are reminded of this node's desire to receive frames directed to that address at least once every 128 seconds (see clause 11.4.4.2).

For each channel, maintain a rate control backoff limit (RCBL) that ranges in value from 1 to 1024, and a rate control backoff frame count (RCBFC) and a receiver_PE for each supported band. (Only receiver_PE is arrayed by the number of supported bands. RCBL and RCBFC are per channel). RCBL is initialized to 1, and RCBFC is initialized to 0. Receiver_PE shall be initialized to 0 for band2. No other restrictions on receiver PE initialization are necessary. If a link integrity frame is received with PE = 1, no RRCF shall be transmitted (see clause 11.5).

For each received frame, compute the new desired PE for the channel (new_pe) for each band. See clause 11.4.5.1.1 for a sample algorithm for selection of desired PE for a band. If the new desired PE is different from the previous value of the desired PE for any supported band, then reset RCBL to 1, and reset RCBFC to 0. Save the new value for desired PE (new_pe) per band, as receiver_PE. If the PE of the received frame is different from the new desired PE, then increment RCBFC by 1. If RCBFC is now greater than or equal to RCBL, then send an RRCF to the source of the frame, with band1_PE set to receiver_PE, for band1 and band2_PE set to receiver_PE for band2, reset RCBFC to 0, and double RCBL up to a maximum of 1024. If a multicast or broadcast channel is active (based on receiving frames other than RRCFs within the last two 128-second intervals), and 128 seconds have passed since the receiver has sent a frame to this multicast or broadcast address, transmit an RRCF with the current receiver PE to any nodes that have sent frames to that multicast or broadcast address, with a RefAddr set to the multicast or broadcast address in question. Multiple multicast addresses may be aggregated into a single RRCF being sent to a node that has been active on multiple multicast addresses. However, only addresses for which the intended recipient of the RRCF has been active should be included.

In RRCF messages, requesting stations should attempt to specify the maximum payload encoding that they believe will have an acceptable error rate, in order to maximize the aggregate throughput of the network.

At a minimum, the 2-MBaud band shall always be specified in an RRCF.

11.4.5.1.1 Sample payload encoding selection algorithm

This clause describes an example algorithm suitable for use by devices implementing a single band on networks with additive white noise and impulse noise. Other algorithms are possible which may better optimize the selected payload encoding based on the measured channel conditions.

For each implementation, compile a table of average slicer mean squared error (ASMSE) required

for each payload encoding (except PE = 8) to achieve a packet error rate (PER) of 1e-3. Define this table as DOWN_LARQ. Compile a second table with a target PER of 1e-6. Define this table as DOWN_NOLARQ. Define UP_LARQ as DOWN_LARQ with all ASMSE values decreased by 2 dB and UP_NOLARQ as DOWN_NOLARQ with all ASMSE values decreased by 2 dB.

The following steps describe how to select the new payload encoding desired for a particular channel, (new_pe), given the current payload encoding desired on that channel, (curr_pe), and a new frame is received on that channel:

1) Keep a history window of 16 HNT frames per channel. For each channel, compute the ASMSE over all frames in the history window that did not have a CRC error.
2) If all the frames in the history window were received with a CRC error, set new_pe = 1 and exit. Else:
3) If LARQ is in use on a channel, find the greatest payload encoding in the UP_LARQ table with an ASMSE greater than or equal to the ASMSE computed in step 1. If LARQ is not in use, use the UP_NOLARQ table. Define this payload encoding as new_up_pe.
4) If LARQ is in use on a channel, find the greatest payload encoding in the DOWN_LARQ table with an ASMSE greater than or equal to the ASMSE computed in step 1. If LARQ is not in use, use the DOWN_NOLARQ table. Define this payload encoding as new_down_pe.
5) If new_up_pe > curr_pe, set new_pe = new_up_pe and exit. Else:
6) If new_down_pe < curr_pe, set new_pe = new_down_pe and exit. Else:
7) If neither 5 nor 6 is satisfied, set new_pe = curr_pe

NOTE – The offset between the up and down rate selection tables provides the algorithm with hysteresis to provide stability in selection of a payload encoding in the presence of minor variations in ASME. Due to this offset, conditions 5 and 6 cannot both be satisfied simultaneously.

The combination of the 16-frame history window with the selection hysteresis prevents the rate selection algorithm from generating an excessive number of rate changes while remaining responsive to significant changes in the channel conditions.

The selection algorithm for the value PE = 8 should also include hysteresis to avoid generating an excessive number of rate changes while remaining responsive to significant changes in the channel conditions.

11.4.5.2 Receiver – Send rate test request frame (RRCF OpCode 1)

Periodically, but at a rate not to exceed once every 128 seconds (except as described below), a receiver may send a rate test request frame to a sender to test if the channel can support a different band. The band encodings represent the encodings for which the receiver would like the sender to generate test frames. NumAddr shall be set to 0 in rate test request frames.

Rate test request frames should be sent encoded at the current negotiated rate for the channel from the receiver to the sender.

Support for rate test request frames is required in all stations.

11.4.5.3 Receiver – Receive rate test reply frame (RRCF OpCode 2)

Upon receipt of a rate test reply frame, the receiver should use the demodulation statistics for this frame, and any previously received rate test reply frames using this encoding, to make a decision as to the channel's capability to support the tested band encoding. If the decision is that the channel is not capable of supporting the tested band encoding, the receiver shall not generate another rate test request frame for at least 128 seconds. If the decision is that the channel is capable of supporting the tested band encoding, the receiver may repeat the test to collect more data, at a maximum rate of one rate test request frame every second, with a maximum of 16 additional tests. At this point, the

receiver should generate a Rate Change Request to the sender specifying the new band encoding.

Support for rate test reply frames is only required in stations that implement additional bands beyond band1. Stations that only implement band1 my silently discard received rate test reply frames.

11.5 Link integrity function

The purpose of the link integrity function is to provide a means for hardware and/or software to determine whether or not this station is able to receive frames from at least one other station on the network. In the absence of other traffic, a station periodically transmits a link integrity control frame (LICF) to the broadcast MAC address, with the interval between such transmissions governed by the method described below.

All stations shall implement the following function to ensure that, with high probability, within any 1-second interval there is either:

1) at least one LICF sent to the broadcast MAC address from this station; or

2) at least one packet addressed to the broadcast MAC address received from each of at least two other stations. Additionally, all stations shall send at least one LICF every 64 seconds.

The method is described below:

- Stations SHOULD support generation of the existing LI frame even in inactive or sleep mode. While in sleep or inactive mode, HNT stations that do not want to be or cannot be awakened SHOULD not send LI frames.
- A link packet may be any broadcast frame received with a valid header FCS. Only LICF frames should be treated as link packets.
- Each station maintains a free-running timer with a period of 1 second. There should be no attempt to synchronize this timer between stations. The timer should not be modified by any link state transitions or by the reception of any frames. This timer is the source of the timeout event used in the link integrity state table in Table 11-9.
- Each station maintains a 6-bit FORCE_SEND counter that is initialized to a random value between 30 and 63. This initialization value may be selected once at node startup and used for each re-initialization of the FORCE_SEND counter, or a new random value may be selected for each re-initialization of the FORCE_SEND counter.
- Each station has a register (SA1) that can be set from the SA of a received link packet.
- An LICF should be sent with a priority corresponding to link layer priority 7.
- The PE for an LICF shall be determined by accessing the RRCF logical channel information for the broadcast channel. An exception to this criterion is if LI frames are not sent with the currently-negotiated broadcast PE value, then they SHALL be sent with PE = 1. This allows, for example, terminals in sleep or inactive mode to maintain active status on the network. Receipt of a LI frame with PE = 1 SHALL not cause a transmission of an RRCF by any HNT terminal.
- While in sleep or inactive mode, the terminal SHOULD perform link integrity and wake-up processing on all receive packets. No further processing of receive packets is necessary. The relevant power-management processing shall be done on LARQ and non-LARQ data frames and the understanding is that non-WoLAN frames should be discarded.
- Each station shall send a link integrity control frame (LICF) with the format shown in Table 11-10, according to the state table in Table 11-9.

Figure 11-3 state diagram gives a pictorial view of the state transitions, with some minor loss of detail, including omission of events that do not cause state transitions (and have no associated

actions), and the collapsing of multiple events into a single transition with a more complex description of the action.

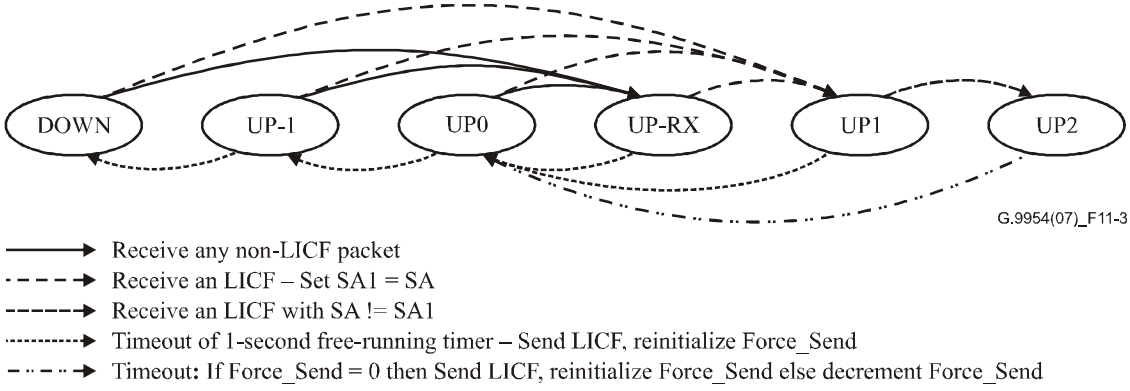

Figure 11-3 – Link integrity state diagram

Table 11-9 is a complete state table, with associated actions. The timeout event is the periodic expiration of a one-second free-running timer.

Initial state: DOWN, Force_Send initialized: 30 ≤ Force_Send ≤ 63.

Table 11-9 – Link integrity finite state machine (FSM)

	DOWN	**UP-1**	**UP0**	**UP-RX**	**UP1**	**UP2**
Receive any non-LICF	UP-RX	UP-RX	UP-RX	UP-RX	UP1	UP2
	(none)	(none)	(none)	(none)	(none)	(none)
Receive LICF with SA == SA1	UP1	UP1	UP1	UP1	UP1	UP2
	Set SA1<-SA	Set SA1<-SA	Set SA1<-SA	Set SA1<-SA	(none)	(none)
Receive LICF with SA != SA1	UP1	UP1	UP1	UP1	Native: UP2 Compat: UP1	UP2
	Set SA1<-SA	Set SA1<-SA	Set SA1<-SA	Set SA1<-SA	(none)	(none)
Timeout and Force_Send == 0	DOWN	DOWN	UP-1	UP0	UP0	UP0
	Send LICF[a)], reinit Force_Send	Send LICF [a)], reinit Force_Send	Send LICF [a)], reinit Force_Send	Send LICF, reinit Force_Send	Send LICF, reinit Force_Send	Send LICF, reinit Force_Send
Timeout and Force_Send > 0	DOWN	DOWN	UP-1	UP0	UP0	UP0
	Send LICF [a)], reinit Force_Send	Send LICF [a)], reinit Force_Send	Send LICF [a)], reinit Force_Send	Send LICF, reinit Force_Send	Send LICF, reinit Force_Send	decrement Force_Send

[a)] Devices which can transmit using more than one MAC source address (e.g., a bridge) should send a CSA request frame to the broadcast address instead of sending an LICF for the cases indicated in the table.

Link integrity status shall be indicated when in any state but DOWN. All stations should include a visible link status indicator (LSI) (e.g., an LED) for indicating link integrity status.

Table 11-10 – Link integrity short frame

Field	Length	Meaning
DA	6 octets	Destination address = FF:FF:FF:FF:FF:FF
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_LINK (2)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum is 4 for SSVersion 0.
SSVersion	1 octet	= 0
LI_pad	1 octet	Ignored on reception.
Next Ethertype	2 octets	= 0
Pad	40 octets	Any value octet
FCS	4 octets	
CRC-16	2 octets	HNT frame check sequence

11.6 Capability and status announcement

A mechanism is defined for network-wide negotiation, capability discovery and status announcement. It is based on periodic broadcast announcements, called capabilities and status announcements (CSA) sent in CSA control frames (CSACFs). The defined status flags allow determination of a station's HNT version, optional feature support, and link-layer priority usage, as well as communication of network configuration commands.

The purpose of the protocol is to distribute to all stations the complete set of status flags in use on the network, so that stations can make operational decisions based on those flags with no further interaction.

Stations shall use the CSA control frame as described in Table 11-11 and the CSA flag definitions shown in Table 11-12.

Stations shall send a CSA control frame once per minute or when a change in the station's current status requires the announcement of new (or deleted) flags.

A station sending a CSA control frame announcing a status change shall send a second copy of the most recent CSACF a short interval after the first, since it is always possible to lose a frame due to temporary changes in the channel, impulse noise, etc. The interval should be randomly selected (not simply fixed), and chosen from the range 1 to 1000 milliseconds, inclusive.

CSA control frames are sent with a priority corresponding to link layer priority 7.

CSA control frames are always sent to the broadcast address (0xFFFFFFFFFFFF).

The PE for a CSA control frame shall be determined by accessing the RRCF logical channel information for the broadcast channel.

A Request OPCode is defined to allow a station to quickly gather complete information about all stations. Upon receiving a CSA control frame with the Request OpCode, a station shall transmit a current CSA message after a delay of a short interval, using the same mechanism (and parameters) that delays the second copy of CSA announcements, described above.

11.6.1 CSA control frame

Table 11-11 defines the format of a capabilities and status announcement control frame. The first three fields beyond the Ethernet header comprise the standard header for short format control frames.

Table 11-11 – Capability and status announcement frame

Field	Length	Meaning
DA	6 octets	Destination address = FF:FF:FF:FF:FF:FF
SA	6 octets	Source address, not necessarily corresponding to the MAC address to which the frame contents are applicable (see CSA_SA)
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_CSA (3)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum is 32 for SSVersion 0.
SSVersion	1 octet	= 0
CSA_ID_Space	1 octet	Identifies the registration space of CSA_MFR_ID 0 – Unspecified 1 – JEDEC 2 – PCI
CSA_MFR_ID	2 octets	HW manufacturer ID – Identifies the manufacturer of the PHY controller chip. The purpose of this field plus the part number and revision is to identify specific implementations of the PHY specification. This is not a board or assembly-level identifier.
CSA_Part_No	2 octets	HW manufacturer part number – The part number of the PHY controller chip
CSA_Rev	1 octet	HW revision
CSA_Opcode	1 octet	0 – Announce 1 – Request
CSA_MTU	2 octets	Maximum size link-level PDU this receiver accepts in octets; the default value is 1526 octets. 1526 is the minimum value that shall be advertised by an HNT station.
CSA_SA	6 octets	MAC address of the station to which the capabilities and status are applicable
CSA_device_id	1 octet	Device ID assigned (by the master) during registration; it is reported to the HNT device with the MAC address identified in the SA field. A value of NULL_ID indicates that the device is not registered with the master. NOTE – More than one station (identified by the CSA_SA) field may have the same CSA_device_id.
CSA_pad	1 octet	Reserved for version 0. Shall be sent as 0, ignored on reception. Creates field alignment to 32-bit WORD boundaries.
CSA_CurrentTxSet	4 octets	Configuration flags, plus all current in-use status for this station. Flag definitions are specified in Table 11-12.

Table 11-11 – Capability and status announcement frame

Field	Length	Meaning
CSA_OldestTxSet	4 octets	A copy of the "oldest" TX flags for this stations, from the period ending at least one period (minute) earlier. Flag definitions are specified in Table 11-12.
CSA_CurrentRxSet	4 octets	The union of recent flags received from other stations. Flag definitions are specified in Table 11-12.
Next Ethertype	2 octets	= 0
Pad		Pad to reach minFrameSize if necessary
FCS	4 octets	
CRC-16	2 octets	HNT frame check sequence

11.6.2 Status, configuration, option and priority flags

The flags as shown in Table 11-12 shall be used for CSA_CurrentTxSet, CSA_OldestTxSet, and CSA_CurrentRxSet in capabilities and status announcement control frames.

Table 11-12 – CSA flag set

Octet	Field	Length [Bits]	Description
Flags0	TxPriority7	1	Station is (was) transmitting frames with LL priority 7. (always set)
	TxPriority6	1	Station is (was) transmitting frames with LL priority 6.
	TxPriority5	1	Station is (was) transmitting frames with LL priority 5.
	TxPriority4	1	Station is (was) transmitting frames with LL priority 4.
	TxPriority3	1	Station is (was) transmitting frames with LL priority 3.
	TxPriority2	1	Station is (was) transmitting frames with LL priority 2.
	TxPriority1	1	Station is (was) transmitting frames with LL priority 1.
	TxPriority0	1	Station is (was) transmitting frames with LL priority 0. (always set)
Flags1	QoS support level	2	Level of QoS support provided by G.9954v2 device as described in clause 10.9. 0 – G.9954v1 QoS 1 – Best-effort 2 – Priority-based 3 – Full QoS

Table 11-12 – CSA flag set

Octet	Field	Length [Bits]	Description
Flags1	Highest mask # or spectral mode supported	2	Highest mask # supported by G.9954v1 transmitter (i.e., when highest version is G.9954v1). Support for Mask N assumes full support for all symbol rates in Mask N – 1. 0 – Mask #1 1 – Mask #2 2 – Mask #3 The spectral mode supported by the G.9954v2 device (i.e., when highest version is G.9954v2). 0 – Spectral Mode A 1 – Spectral Mode B 2 – Spectral Mode C 3 – Spectral Mode D
	Supports frame bursting	1	This station supports frame bursting.
	Smallest sub-burst slot supported	2	Size (duration) of smallest sub-burst slot supported. 0 – 8 microseconds 1 – 16 microseconds 2 – 32 microseconds 3 – 64 microseconds This field should be ignored for G.9954v1 devices.
	Supports high constellation encoding	1	Supports high round constellation encodings of 8, 9 and 10 bits per symbol.
Flags2	Frame burst packet limit	3	0 – No limit (actually limited by maximum link-level frame size in the highest PE). 1 – This station supports bursts of up to 16 frames. 2 – This station supports bursts of up to 32 frames. 3 – This station supports bursts of up to 64 frames. 4 – This station supports bursts of up to 128 frames. 5 – This station supports bursts of up to 256 frames.

Table 11-12 – CSA flag set

Octet	Field	Length [Bits]	Description
Flags2	Frame burst size limit	3	0 – No limit (actually limited by maximum link-level frame size in the highest PE). 1 – This station supports bursts of up to 8 kbytes. 2 – This station supports bursts of up to 16 kbytes. 3 – This station supports bursts of up to 32 kbytes. 4 – This station supports bursts of up to 64 kbytes. 5 – This station supports bursts of up to 80 kbytes. For the purpose of burst size limitations, a burst consists of all the link layer frames (i.e., all the frame excluding the physical layer preamble, frame-control, pad and EOF). For further information on frame bursting and aggregation, see clause 11.12.
	Synch mode	1	This station is operating in synchronous MAC mode and is currently synchronized with the master MAC cycle. 0 – The station is NOT operating in synchronous MAC mode. 1 – The station is operating in synchronous MAC mode.
	Reserved	1	Shall be sent as 0 and ignored by stations when received.
Flags3	Reserved	5	Shall be sent as 0 and ignored by stations when received.
	Highest version	3	This station's highest supported HNT version: 0x000 – Reserved 0x001 – Reserved for legacy usage 0x010 – Reserved for legacy usage 0x011 – G.9954v1 0x100 – G.9954v2 0x101-0x111 – Reserved for future use

Thirty-two bit-flags shall be supported for announcing status and configuration information. The flags are divided into three basic groups: mode selection flags including HNT version information, supported options, and in-use TX link layer priority announcements. These flags shall be added to the global state as soon as announced, and removed when no longer announced by any station, either through explicit deletion or by timing them out. An in-use TX link layer priority shall be announced for a period of one to two minutes after the last frame actually sent with the priority, until the aging mechanism causes it to be deleted from CurrentTxSet.

The default set of status flags, used to initialize the NewTxSet (defined below), is defined to be the priorities 0 and 7, the station's HNT version, and any supported options.

11.6.3 Terms and parameters

11.6.3.1 Capabilities and status period (CS period)

The basic time interval used to age out non-persistent status information shall be one minute. Each station has a repeating timer set to this interval. The timers in different stations are not synchronized, and synchronization should, in general, be avoided. The description below refers to the time between one expiration of this timer and the next as a "period". The "current" period refers to the time since the most recent expiration of the timer.

A CSA frame shall be sent at the end of each interval.

11.6.3.2 Variables, etc.

- DeleteSet: A computed value used to detect newly removed status information.
- NewRxFlags, ReallyNewRxFlags: Computed values used to detect new status flags.

11.6.3.3 Timers

- CSP_Timer: A free-running timer with a period of 60 seconds.
- RetransmitTimer: A one-shot timer, set to a random interval in the range 1 ms to 1000 ms, inclusive, after sending a CSA in which CSA_CurrentTxSet and CSA_OldestTxSet are different, or when a CSA is received with the CSA_Opcode set to 1 (Request). This timer is cancelled if a second CSA is sent as a result of the CSP_Timer expiring.

11.6.4 Status and priority set state variables

Each station maintains five basic sets of status and priority information. In addition, three more composite sets are defined as the union of two or more of the basic sets. See Table 11-13.

Table 11-13 – Set state variables

NewTxSet	The set of flags announced during the current CS period, updated immediately when a new link layer priority is used or new volatile status is set. When the CSP_Timer expires, CurrentTxSet is given the value of NewTxSet, and NewTxSet is reset to the default set.
PreviousTxSet	The set of flags that were announced during the previous CS period (the ending value of NewTxSet from the previous CS period).
OldestTxSet	The set of flags rolled over from PreviousTxSet at the end of the previous CS period (the value of PreviousTxSet from the previous CS period). Flags that are present in OldestTxSet and missing from PreviousTxSet were not actively used or detected (by the sender) for an entire CS period, and will be deleted. This set is sent in CSA frames as CSA_OldestTxSet.
NewRxSet	The union of all CSA_CurrentTxSet flags received in CSAs from other stations during the current CS period. This is rolled over into PreviousRxSet at the expiration of the CSP_Timer, then reset to the empty set (0). A volatile status flag (one of the priority flags) in this set may subsequently be deleted if the only station previously announcing that flag stops using it. The deletion from that station's CurrentTxSet is noted by the difference from its OldestTxSet. The fact that it was the only sender is noted by the absence of the flag in that station's CurrentRxSet, indicating that it has received the flag from no other stations. If deleted from NewRxSet, a flag shall also be deleted from PreviousRxSet.
PreviousRxSet	The set of announced flags received during the previous CS period (the ending value of NewRxSet from the previous CS period). A flag may be deleted from this set, as described under NewRxSet above.
CurrentTxSet	The set of flags that were announced during the previous CS period plus any new status and priority flags (or changed configuration/options flags) used during the current CS period, i.e., the union of PreviousTxSet and NewTxSet. This set is sent in CSA frames as CSA_CurrentTxSet.
CurrentRxSet	The union of NewRxSet, PreviousRxSet. This set is sent in CSA frames as CSA_CurrentRxSet.
CurrentInUseSet	The union of CurrentTxSet and CurrentRxSet. This set is used to determine the operational mode of the station and to modify the mapping between the LL priority of the frame and the actual PHY priority usage.

11.6.5 Capabilities and status announcement protocol operation

11.6.5.1 New transmit frame – Priority detection

The CSA protocol does not directly process transmit frames. When the LARQ protocol is in use, CSA looks at the LL priority of the frame as it would normally be sent to the driver.

1) If the LL priority is not already in NewTxSet, add it to NewTxSet.
2) If the LL priority was not already in NewTxSet and it is not in PreviousTxSet, then send a new CSA control frame with the CSA_Opcode set to 0 (Announce), and start the RetransmitTimer. If the timer was already running, then cancel and restart it. Update the current Phy priority mapping function for the driver.

11.6.5.2 Receive CSA control frame

The receiver may want to save a copy of some or all of the most recent CSA from each other station as a simple way of tracking other station's capabilities and status.

1) Record (optionally) the status and options flags from the CSA_CurrentTxSet in a table indexed by the CSA_SA address. The options flags are used to select use of optional functions between pairs of stations that implement the same options.
2) If the CSA_Opcode in the frame is 1 (Request), then start the RetransmitTimer. If the timer is already running, it should be left running although this is not required and cancellation followed by restart is allowed.
3) If CSA_CurrentTxSet has a flag not already in NewRxSet, then add the flag to NewRxSet, and check to determine if this flag is not present in the PreviousRxSet. The corresponding boolean expressions are as follows:
   - NewRxFlags = (CSA_CurrentTxSet & ~NewRxSet)
   - NewRxSet |= NewRxFlags
   - ReallyNewFlags = NewRxFlags & ~(PreviousRxSet | CurrentRxSet)
4) Compare CSA_OldestTxSet with CSA_CurrentTxSet. If a flag has been deleted, and if that flag is also missing from CSA_CurrentRxSet, then delete the flag from NewRxSet, and PreviousRxSet. The corresponding boolean expressions are as follows:
   - DeleteSet = (CSA_OldestTxSet & ~CSA_CurrentTxSet) & ~CSA_CurrentRxSet
   - NewRxSet = NewRxSet & ~DeleteSet
   - PreviousRxSet = PreviousRxSet & ~DeleteSet
   - CurrentRxSet = NewRxSet | PreviousRxSet
5) If either ReallyNewFlags or DeleteSet are non-zero, then update the network mode and priority mapping, as necessary.

11.6.5.3 CSP_Timer timeout

When a CSP_Timer timeout occurs, a new CS period has begun. Roll over the various status sets, re-compute the composite sets, and send a CSA. Set the RetransmitTimer, if needed.

1) OldInUseSet = CurrentInUseSet
2) Move NewRxSet to PreviousRxSet.
3) Set NewRxSet to 0 (empty set).
4) Move PreviousTxSet to OldestTxSet.
5) Move NewTxSet to PreviousTxSet.

6) Set NewTxSet to the default set, consisting of this station's highest supported version, current configuration flags if any (normally none), currently supported options, and the default priority set {0,7}.

7) Update CurrentTxSet, CurrentRxSet, and CurrentInUseSet (at least logically, an implementation need not keep separate copies of these values).

CurrentRxSet = NewRxSet | PreviousRxSet

CurrentTxSet = NewTxSet | PreviousTxSet

CurrentInUseSet = CurrentRxSet | CurrentTxSet

8) Send a CSA frame with the CSA_Opcode set to 0 (Announce), including the updated flags.

9) If CSA_CurrentTxSet and CSA_OldestTxSet in the CSA frame just sent were different, start the RetransmitTimer. If the timer was previously running, then cancel it and restart it.

10) If one or more status flags have been deleted, then recompute the network operating mode and/or priority mapping function due to changed status flags. The mode/mapping recomputation should be performed if CurrentInUseSet is not equal to OldInUseSet.

11.6.5.4 Retransmit timeout

If the RetransmitTimer expires, send a current CSA frame for this station with the CSA_Opcode set to 0 (Announce). The timer shall not be restarted.

11.6.6 Priorities

There is a cost of slightly lower maximum attainable bandwidth associated with lower Phy priorities in the HNT MAC protocol if a default mapping scheme of link layer to Phy layer priorities is employed. This cost becomes especially burdensome when only lower-priority traffic is being carried on the network. Therefore, the CSA protocol includes procedures for remapping lower LL priorities to higher Phy layer priorities when no station on the network is sending traffic marked for those higher priorities.

The choice of physical layer (Phy) priority for a given frame is based on its assigned link-layer (LL) priority. The default mapping from LL priority to Phy priority is specified in clause 11.6.6.3. The LL priority of a frame at the sender must be conveyed to the receiving station in order to allow proper recovery of link layer protocol at the receiver. This requires either a fixed, one-to-one, mapping of LL-to-Phy priorities, or some mechanism for carrying the LL priority within each frame. The LARQ protocol, defined in clause 11.7, carries the assigned LL priority from a sending station to a receiving station, providing the required mechanism, and thereby creating the opportunity to apply non-default LL-to-Phy priority mappings, which in turn, allows for higher maximum attainable bandwidth. A station may optionally use an 802.1q header to convey the LL priority. However, since support for 802.1q headers is optional, a station employing this method should attempt to determine that all receivers of the frame support the use of 802.1q headers. Stations that do not support 802.1q headers are unlikely to properly receive frames that include an 802.1q header.

11.6.6.1 Transmit frames – Choice of physical priority

When the assignment of a physical layer priority to the frame occurs, any changes to the Phy priority remapping function due to the use of a new priority should already have been made. The driver should use the remapped Phy priority to transmit the frame (including placing this value in the frame control header) unless the frame has no LARQ header, in which case the default LL-to-Phy mapping shall be used.

11.6.6.2 Received frame priorities

The LL priority of received frames indicated up the protocol stack by the driver (before any reassignment due to a LARQ or 802.1q header) shall be determined using the default Phy-to-LL priority map. The mechanism that guarantees correct LL priority for received frames is the restoration of LL priority from the LARQ (or optionally, 802.1q) header or from the flow specification. LARQ header processing shall be performed after the default LL priority has been assigned in the receive path. If a received frame can be mapped to a flow channel, the priority information in the associated flow specification shall be used to recover the LL priority.

11.6.6.3 Default link layer-to-physical layer map

The IEEE 802.1p specification places the default (unassigned/best-effort) priority above both priorities 1 and 2, when an 8-level priority system is in use. Therefore, Link Layer priority 0 shall be mapped above both LL 1 and LL 2 for default physical layer priority assignment. IEEE 802.1p designates priority level 7 for Network Control and priority level 6 for traffic requiring latency of <10 ms (typically characterized as voice-like traffic). However, on HNT networks, Phy priority level 7 shall be reserved for traffic requiring latency of <10 ms, and network control traffic shall be redirected to HNT Phy priority level 6. Link layer priority 5 shall be reserved for traffic requiring latency of <100 ms. So the default mapping for LL-to-Phy priorities includes the swapping of priorities 6 and 7.

For transmitted frames, the set of LL priorities [0,1,2,3,4,5,6,7] shall be mapped, by default, in order to the following set of Phy priorities [2,0,1,3,4,5,7,6].

For received frames, Phy priorities [0,1,2,3,4,5,6,7] shall be mapped, by default, to LL priorities [1,2,0,3,4,5,7,6].

11.6.7 Priority mapping and LARQ

The Phy priority remapping shall be performed below LARQ in the protocol stack, and shall not be applied to the priority field in the LARQ (or optionally, 802.1q) header. Phy priority remapping shall not be performed on data frames (those that are not link control frames) unless a LARQ (or optionally, 802.1q) header has been added with the original LL priority. Phy priority remapping shall be performed on link control frames.

11.6.8 Priority remapping based on CurrentInUseSet

Without priority mapping, a station would pass the original LL priority into the driver, where that value would be used to select the associated Phy priority from the default map. With priority remapping, the default-assigned Phy priorities are increased to make use of higher Phy priorities that would otherwise be unused. The remapping function is simple. For each Phy priority P that corresponds to an in-use LL priority, the new priority P' to use shall be that priority increased by the number of higher unused priorities. For example, if [1,3,4,7] are in use, then priority 4 will be increased by 2 to 6, since there are two higher unused priorities (5,6). Figure 11-5 contains a few more examples that should make this clear (including the default LL-to-Phy translation). The columns in Figures 11-4 and 11-5 represent LL priorities before mapping. The left hand section shows some sets of in-use priorities, with the right-hand section showing the new Phy priority that the driver should use in each case.

									TX LL priority							
									0	1	2	3	4	5	6	7
CurrentInuse priorities (any)									Default TX Phy priorities							
a	n	y	t	x	s	e	t		2	0	1	3	4	5	7	6

Figure 11-4 – Default LL-to-Phy TX priority mapping

									TX LL priority							
									0	1	2	3	4	5	6	7
CurrentInuse Priorities (LL)									Remapped TX Phy Priorities							
0							7		6	5	5	6	6	6	7	7
0						6	7		5	4	4	5	5	5	7	6
0	1			4			7		5	4	4	5	6	6	7	7
0			3		5	6	7		3	2	2	4	4	5	7	6

Figure 11-5 – Direct LL-to-Phy TX priority remapping

The shaded entries show mappings that no sender should be using. However, if there is any possibility of an implementation sending with an out-of-date mapping, or sending a priority that has not been included in the mapping, then it should always use the priority of the next lower valid mapping.

Here is one example in detail. If the CurrentInuse are [0,1,4,7], then the corresponding set of in-use Phy priorities is [2,0,4,6]. Then increase each by the number of missing higher priorities 2 → 5, 0 → 4, 4 → 6 and 6 → 7. Just to be safe, the any unused Phy priorities are also remapped to the new value of the next lower in-use priority, giving: 2 → 4, 3 → 5, 5 → 6, 6 → 7.

So the in-use LL priorities [0,1,4,7] result in transmitting Phy priorities [5,4,6,7]. A complete map for all the LL priorities adds the remaining remapped values for the default priorities corresponding to the unused LL priorities: LL[0,1,2,3,4,5,6,7] gives Phy[5,4,4,5,6,6,7,7].

11.7 LARQ: Limited automatic repeat request protocol

Limited automatic repeat request (LARQ) is a protocol that reduces the effective error rate when frame errors occur. Its primary distinction from similar, sequence number-based protocols is that it does not guarantee reliable delivery of every frame, but instead conceals errors in the physical layer through fast retransmission of frames. The goal is to significantly enhance the usability of networks that may, at least occasionally, have frame error rates (FER) of 1 in 10^{-2} or worse. Protocols such as TCP are known to perform poorly when FER gets high enough, and other applications, such as multimedia over streaming transport layers, are also susceptible to poor performance due to high FER conditions.

The protocol provides a negative acknowledgment (NACK) mechanism for receivers to request the retransmission of frames that were missed or received with errors. There is no positive acknowledgment mechanism. There is no explicit connection set-up or tear-down mechanism. A reminder mechanism gives receivers a second chance to detect missing frames when relatively long gaps (in time) occur between frames.

LARQ functions as an adaptation layer between the Ethernet link layer (layer 2) and the IP network layer (layer 3). It is commonly implemented in the device driver.

Stations implement LARQ per "LARQ channel", where a LARQ channel is identified by either the tuple {source address, destination address, priority}, referred to as a LARQ-Priority Channel or by the tuple {source address, destination address, flow id}, referred to as a LARQ-Flow Channel.

LARQ-priority channel is defined (and set-up) in an implementation-dependent way. A LARQ-flow channel is defined when the ACK-policy for the associated flow (in the flow specification) is set to "LARQ" and set up in conjunction with the set-up of the flow.

Stations may enable or disable LARQ processing on a channel dynamically, based on information about network frame error rates. However, LARQ should be left enabled at all times, since the per-packet processing overhead is quite low, and the complexity associated with enabling and disabling the protocol (including determination of appropriate parameters) probably outweighs any likely

performance gains.

Stations should implement LARQ, and if they do so, they shall use the specified control frame formats and should use the procedures defined below.

For a simple channel (i.e., a logical channel defined by SA, DA without an associated flow specification) stations not adding LARQ (or optionally, 802.1q) headers shall not remap Phy priorities, and shall treat all received traffic as "best effort", that is, all traffic shall be assigned to link layer priority 0. For a flow channel (i.e., logical channel defined by SA, DA, flow id) Phy priority remapping and LL priority recovery is performed using the priority information in the flow specification.

Stations may choose to add LARQ headers on transmitted frames with the LARQ_NoRtx flag set to 1. This flag indicates that the station does not retransmit frames for this channel, but adding the LARQ header allows the station to use Phy priority remapping since the LL priority of successfully received frames will be restored from the LARQ header.

All stations SHALL be capable of removing LARQ headers from received frames (de-encapsulating the original payloads). Furthermore, if the implementation supports multiple LL priorities in its receive protocol processing, then it shall restore the LL priority from the LARQ header, if one is present. If a station does not implement LARQ, then it shall drop LARQ control frames and it shall discard frames marked as retransmissions in the LARQ header.

11.7.1 Frame formats – Encapsulating headers

The text below uses the terms "insert" and "remove" when discussing LARQ headers. The formal definition of the LARQ frame format provides a Next Ethertype field that contains the original frame's Ethertype value. In practice, it will generally be the case that LARQ frames will be created by inserting the 8 octets starting with the Ethertype 0x886c into the original frame between the Ethernet header's source address and the original frame's Ethertype. The original frame's Ethertype becomes relabelled as the Next Ethertype field of the final frame.

The LARQ header carries LLC priority across the network. The use of 802.1q headers is not required for this function, and HNT drivers are not required to support the use of 802.1q headers for conveying priority. See Tables 11-14 to 11-17.

Table 11-14 – LARQ reminder control frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_LARQ (4)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength is 6 for SSVersion 0.
SSVersion	1 octet	= 0
LARQ_hdr data	3 octets	LARQ control header data with LARQ_ctl bit = 1, LARQ_NACK = 0.
Next Ethertype	2 octets	= 0
Pad	38 octets	
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

Table 11-15 – LARQ NACK control frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_LARQ (4)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength is 12 for NACK frames with SSVersion 0.
SSVersion	1 octet	= 0
LARQ_hdr data	3 octets	LARQ control header data with LARQ_ctl bit = 1, LARQ_NACK = 1..7.
NACK_DA	6 octets	Original destination Address
Next Ethertype	2 octets	= 0
Pad	32 octets	
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

Table 11-16 – LARQ encapsulation frame

Field	Length	Meaning
DA	6 octets	Destination address (from original Ethernet PDU)
SA	6 octets	Source address (from original Ethernet PDU)
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_LARQ (4)
SSLength	1 octet	= 6 Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength is 6 for SSVersion 0
SSVersion	1 octet	= 0
LARQ_hdr data	3 octets	LARQ encapsulation header data (with LARQ_ctl bit = 0)
Next Ethertype	2 octets	From original Ethernet PDU
Payload	Min. 46 octets	From original Ethernet PDU payload
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

Table 11-17 – LARQ encapsulation header data

Octet	Field	Length	Meaning
Flags0	LARQ_Mult	1 bit	Multiple retransmission flag. 0 in the original transmission of a data frame. For retransmitted frames (LARQ_Rtx = 1), set to the value of LARQ_Mult in the NACK frame that caused the retransmission. This flag can be used by receivers to measure the round-trip times associated with the miss/NACK/receive-rtx process.
	LARQ_Rtx	1 bit	0 for first transmission of a frame, 1 if frame is retransmitted. Stations not implementing LARQ shall drop any data frame if this bit is 1.
	LARQ_NewSeq	1 bit	1 if the sequence number space for the channel has been reset, and older sequence numbers should not be NACKed, 0 otherwise.
	LARQ_NoRtx	1 bit	0 if implementation supports retransmission, 1 if only priority is meaningful. May be used on a per-channel basis.
	LARQ_Ctl	1 bit	"0" when in encapsulation format
	Priority/FlowID	3 bits	Link layer priority/Flow ID of this frame
Flags1_Seq0	FlowID	1 bit	High order bit of FlowID if FSelector = 1
	FSelector	1 bit	Select interpretation of priority/flow ID field. 0 – Priority interpretation 1 – Flow ID interpretation
	Reserved	2 bits	Reserved for future use
	LARQ_seq_high	4 bits	High 4 bits of sequence number
Seq1	LARQ_seq_low	8 bits	Low 8 bits of sequence number

The exact application of the LARQ_Rtx, LARQ_NewSeq and LARQ_NoRtx bits requires further explanation as found in Table 11-18.

Table 11-18 – LARQ_Rtx, LARQ_NewSeq and LARQ_NoRtx bits interpretation

LARQ_ Rtx	LARQ_ NewSeq	LARQ_ NoRtx	Interpretation
0	0	0	Normal transmission on an active channel This combination is used for the first transmission of a frame on an active LARQ channel. The receiver of this frame should send NACKs for earlier sequence numbers that are determined to be missing when this frame is received, or for this frame, if this frame has a CRC error but the LARQ header appears to be in sequence for the channel.

Table 11-18 – LARQ_Rtx, LARQ_NewSeq and LARQ_NoRtx bits interpretation

LARQ_ Rtx	LARQ_ NewSeq	LARQ_ NoRtx	Interpretation
0	0	1	Used for the first transmission of a frame which will not be retransmitted in response to a NACK The sender should use this combination when it does not save the frame for retransmission in response to receiving a NACK. If a receiver is keeping state, then it should send this frame up when it has either received frames for all previous sequence numbers, or given up attempts to receive frames for all previous sequence numbers.
0	1	0	Used for the first transmission of a frame with a new sequence number space The sender uses this combination when there are no saved frames for the channel, excepting this frame. The receiver should send all frames for this channel up to the next layer, since there is no longer the possibility of receiving any frames with previous sequence numbers. The receiver of this frame should send a NACK for this frame, if this frame has a CRC error but the LARQ header appears to be in sequence for the channel.
0	1	1	Used for the first transmission of a frame with a new sequence number space which will not be retransmitted in response to a NACK The sender uses this combination when there are no saved frames for the channel. The receiver should send all frames for this channel up to the next layer, since there is no longer the possibility of receiving any frames with previous sequence numbers.
1	0	0	Retransmission of a frame for this channel Sender uses this combination to send a frame which has been transmitted before, and for which a NACK will cause an additional retransmission. The receiver must accept this frame if it is not a duplicate. If the receiver is not maintaining state for the channel, then this frame must be discarded because it would be impossible to determine the duplicate status for the frame. The receiver of this frame should send a NACK for this frame, if this frame has a CRC error but the LARQ header appears to be in sequence for the channel.
1	0	1	Retransmission of a frame for this channel Sender uses this combination to send a frame which has been transmitted before, but has not been saved for retransmission in response to receiving a NACK. The receiver must accept this frame if it is not a duplicate. If the receiver is not maintaining state for the channel, then this frame must be discarded because it would be impossible to determine the duplicate status for the frame.

Table 11-18 – LARQ_Rtx, LARQ_NewSeq and LARQ_NoRtx bits interpretation

LARQ_ Rtx	LARQ_ NewSeq	LARQ_ NoRtx	Interpretation
1	1	0	Retransmission of a frame for this channel The sender uses this combination when there are no older saved frames for the channel, excepting this frame. The receiver must accept this frame if it is not a duplicate. If the receiver is not maintaining state for the channel, then this frame must be discarded because it would be impossible to determine the duplicate status for the frame. The receiver should send this frame and all older frames for this channel up to the next layer, since there is no longer the possibility of receiving any frames with previous sequence numbers. The receiver of this frame should send a NACK for this frame, if this frame has a CRC error but the LARQ header appears to be in sequence for the channel.
1	1	1	Retransmission of a frame for this channel The sender uses this combination when there are no older saved frames for the channel. The receiver must accept this frame if it is not a duplicate. If the receiver is not maintaining state for the channel, then this frame must be discarded because it would be impossible to determine the duplicate status for the frame. The receiver should send this frame and all older frames for this channel up to the next layer, since there is no longer the possibility of receiving any frames with previous sequence numbers.

Table 11-19 – LARQ control header data

Octet	Field	Length	Meaning
Flags0	LARQ_Mult	1 bit	Multiple retransmission flag. 0 in the first NACK sent for a given sequence number, 1 in all retransmitted NACKs.
	LARQ_NACK	3 bits	NACK count If 0 in a LARQ control frame, then this is a reminder.
	LARQ_Ctl	1 bit	Set to 1 for LARQ control header data format
	Priority/FlowID	3 bits	Link layer priority/flow ID of this frame
Flags1_Seq0	FlowID	1 bit	High order bit of FlowID if FSelector = 1
	FSelector	1 bit	Select interpretation of priority/flow ID field. 0 – Priority interpretation 1 – Flow ID interpretation
	Reserved	2 bits	Reserved for future use
	LARQ_seq_high	4 bits	High 4 bits of sequence number
Seq1	LARQ_seq_low	8 bits	Low 8 bits of sequence number

11.7.2 Terms and definitions

- **control frame**: A frame generated by a LARQ protocol module that contains only a LARQ protocol header as its payload.
- **current sequence number**: The most recently received new sequence number for a channel.
- **data frame**: Any standard Ethernet frame from higher (than LARQ) protocol layers. A LARQ-enabled station encapsulates the original payload of an Ethernet frame by inserting a LARQ header (short form control header with LARQ_hdr data) between the source address and the remainder of the frame before the frame is passed down to the driver for transmission on the network.
- **forget timer**: An implementation-dependent mechanism to allow a receiver to reset the sequence number space of a channel when a received sequence number is not the next expected (Current sequence number + 1). One second is a suggested default value.
- **hold timer, lost timer**: An implementation-dependent timing mechanism that limits the time a receiver will hold onto a received frame while waiting for a missing frame to be retransmitted. Conceptually, there is one such timer per missing sequence number. The timer interval is maximum hold interval.
- **logical channel, channel**: A flow of frames from a sender to one or more receivers on a single network segment consisting of all the frames with a single combination of destination address, source address, and link layer priority or flow ID.
- **NACK, Nack, nack**: An indication from a receiver to a sender requesting retransmission of one or more frames. Also, the action of providing such an indication. E.g., "to nack a sequence number" meaning to send a NACK indication.
- **NACK timer**: An implementation-dependent timing mechanism used by a receiver to retransmit NACKs for missing sequence numbers. Conceptually, there is one such timer per missing sequence number per logical channel. The timer is reset each time a NACK is sent for a sequence number. The timer interval is NACK retransmission interval.
- **new**: A new sequence number is one whose difference from the current sequence number for the channel, modulo the size of the sequence number space and considered as a signed integer, is greater than 0. In particular, the numbers (current + 1) through (current + 2047).
- **old**: An old sequence number is one whose difference from the current sequence number for the channel, modulo the size of the sequence number space and considered as a signed integer, is less than or equal to 0. In particular, the numbers (current – 2048) through (current) are old. However, most of the old sequence numbers are also out of sequence.

- **out of sequence**: Any sequence number that falls outside a reasonable range, old or new, of the current sequence number for a logical channel is considered out of sequence. Plus or minus twice the value of MaximumSaveLimit (defined below) should be used as the "reasonable range" when checking for out of sequence.
- **receiver**: A station that receives frames sent on a particular channel. If the destination address is a unicast address, there is at most one receiver. If the destination address is a group address (including broadcast), then there may be many receivers.
- **reminder**: A control frame sent by the channel sender with the most recently used sequence number for a channel which has been inactive for reminder interval after its most recent data frame.
- **reminder timer**: An implementation-dependent timing mechanism used by a sender to generate a reminder frame after a period of inactivity for a channel. The timer is reset each time a new data frame is transmitted. Conceptually, there is one such timer per channel. The timer interval is reminder interval.
- **save timer**: An implementation-dependent timing mechanism that limits the time a sender will save a frame waiting for retransmission requests. The timer interval is maximum save interval.
- **sender**: The sending station for a channel, usually the station owning the source MAC address.
- **sequence numbers**: Sequence numbers are maintained separately for each logical channel by the sender.

11.7.2.1 Channels

LARQ is defined for operation on simplex logical channels. A separate logical channel is defined for each combination of Ethernet destination address, Ethernet source address and link-layer priority or Ethernet destination address, Ethernet source address and Flow ID. There is no explicit channel set-up procedure. A new channel is implicitly defined when a station chooses to send LARQ encapsulated frames for a new combination of DA, SA and link layer priority or Flow ID. For a flow channel, an associated LARQ channel may be implicitly set up when the flow is set up if the ACK policy defined for the flow is LARQ.

The station that sends such LARQ encapsulated frames (usually the owner of the SA, except in the case of a bridge masquerading as SA) is the sender for the channel. Each channel has a single sender. Any station that receives the frames and processes the LARQ headers is a receiver. There may be any number of receivers. Receivers operate independently.

11.7.3 Sender operation

11.7.3.1 Variables and parameters

- **Send Sequence Number**: The sequence number of the most recently transmitted data frame.
- **Reminder Timer Interval**: A fixed interval. The default is 50 ms. Lower values will increase the overhead of reminders on network load, while *higher* values increase the latency for end-of-sequence frames requiring retransmission. Implementations should not use values outside of the range 25-75 ms, based on 150-ms maximum save and hold times.

- **Minimum Retransmission Interval**: An interval used to prevent too-frequent retransmissions of a single frame. Most important for multicast channels. The default is 10 ms.
- **Maximum Save Limit:** The maximum number of frames that will be saved for a single logical channel. This is implementation dependent, and varies with the maximum frame rate the sender is expected to support. Values of 100 or more can be useful for high-speed applications such as video.
- **Maximum Save Interval**: The maximum time that the sender will normally save a frame for possible retransmission. The default is 150 ms.

11.7.3.2 Sender – New channel

Select implementation-dependent parameters, if necessary.

Select an initial value for Send Sequence Number.

11.7.3.3 Sender – Transmit new data frame

The link layer priority for the frame is determined in an implementation-dependent manner, for instance, by examining the 802.1p priority passed along with packets in newer NDIS implementations.

Access the logical channel state information for the DA, SA and link-layer priority/flow ID of the frame.

Increment Send Sequence Number, modulo 4096 (the size of the sequence number space).

Build the LARQ header with the new value of Send Sequence Number, and the multiple retransmission flag set to 0. The Priority field in the LARQ header shall be set to the link-layer priority value specified for the frame. If no priority is specified, then the priority shall be set to 0. The method of specifying priority and the choice of value are implementation dependent and outside the scope of this Recommendation for LARQ-priority channels. For LARQ-flow channels, the LL priority shall be set using the priority specified in the flow specification.

Insert a LARQ header (short form control frame format with LARQ_hdr data) between the SA and the Ethertype/Length field of the original frame. The new frame is eight bytes longer than the original.

Save a copy of the frame.

Send the frame.

Restart the reminder timer for the channel.

Start a save timer for the sequence number. When no other resource limitations apply, a sending station should normally save a frame for Maximum Save Interval, which corresponds to Maximum Hold Interval used by LARQ receivers.

11.7.3.4 Sender – Process a NACK control frame

The priority/flow ID and original destination address (NACK_DA) are read from the LARQ NACK header.

Access the logical channel state information for the sender channel, where the channel DA is the NACK_DA and the channel SA is the Ethernet DA from the Nack control frame.

The NACK count in the LARQ header indicates the number of sequence numbers requested for retransmission. The first indicated sequence number is the value sequence number in the NACK header, followed by the next (NACK Count – 1) sequence numbers. For each indicated sequence number starting with the first:

– If a copy of the original frame is no longer available, go to the next sequence number.
– If the most recent retransmission of the frame is within Minimum Retransmission Interval of the current time, go to the next sequence number.
– Prepare a copy of the original frame with its original LARQ header for retransmission.
– Copy the value of the multiple retransmission flag from the NACK header into the LARQ header of the frame to be retransmitted.
– Set the LARQ_Rtx flag to 1.
– Send the retransmitted frame.

Do not send a retransmission if a received Nack control frame has an error.

11.7.3.5 Sender – Reminder timer expiration

If the reminder timer expires, create a reminder control frame, with the sequence number set to the current value of send sequence number for the channel. The priority for the reminder control frame shall be the same as the priority for the channel.

Send the frame.

Do not restart the reminder timer for the channel.

11.7.3.6 Sender – Save timer expiration

The save timer is implementation dependent. Its purpose is to set an upper bound on how long frames will be saved by a sender for possible retransmission. If set too long, host resources may be wasted saving frames that will never be retransmitted.

This timer is conceptually implemented per sequence number. Release any resources associated with the saved frame.

11.7.3.7 Sender – Resource management

A LARQ implementation requires careful attention to resource management. The resources include the buffers used for saving copies of data for retransmission, the buffers and other resources used to manage the re-ordering of frames to incorporate retransmissions, and the various timers used to govern proper behaviour and efficient protocol operation. Resource management is implementation dependent. However, the following guidelines should be followed.

Saved copies of frames should be kept for Maximum Save Interval (default is 150 ms), other considerations notwithstanding.

Maximum Save Limit, the maximum number of saved frames for any channel, should be a function of the maximum rate that new frames may be generated. Very slow devices might usefully save only a couple of frames for retransmission. A high-speed device serving video streams might save 100 or more frames for a single channel.

Senders that save relatively few frames are more likely to receive NACK control frames for sequence numbers that can no longer be retransmitted. Such behaviour is inefficient, but causes no other problems.

11.7.4 Receiver operation

11.7.4.1 Channel variables and parameters

The description below of correct protocol operation uses the following variables. The actual implementation may vary so long as the behaviour remains unchanged.

- **Current Sequence Number**: The most recent sequence number received in a LARQ header for the channel, whether in a data frame or a reminder control frame.
- **Oldest missing sequence number**: The oldest sequence number for a frame not yet received which has not been declared lost.
- **Maximum Hold Interval**: The longest interval that a frame will be held awaiting an earlier missing frame. The default is to use the same value as Maximum Save Interval, which has a default of 150 ms.
- **Maximum Receive Limit**: The maximum number of frames that a receiver will buffer while awaiting an earlier missing frame. The default should normally be the same as the Maximum Save Limit.
- **NACK Retransmission Interval**: The interval after which a receiver will retransmit a Nack control frame for a missing sequence number, with the expectation that earlier Nack control frames or data frame retransmissions were lost. The default for fixed implementations is 20 ms.

11.7.4.2 Receiver – New channel

When a data frame with a LARQ header or a LARQ reminder control frame is received, the receiver shall determine the identity of the LARQ channel (i.e., either {DA, SA, priority} or {DA, SA, flow id}) using information in the LARQ frame (i.e., frame-control and LARQ encapsulation header) and determine whether it is a new channel. If the LARQ channel is new, the receiver shall initialize state information for a new channel. For a flow channel, the associated LARQ channel may be set up during Flow Set-up if the set-up flow has an ACK Policy = LARQ.

The primary piece of state information is the Current Sequence Number for the channel. Current Sequence Number shall be initialized to the sequence number immediately preceding that found in the LARQ header of the received frame. This assignment shall take place prior to processing the received frame and will result in the frame either appearing to be the next expected data frame, or the reminder for the next expected data frame.

11.7.4.3 Receiver – LARQ data or reminder frame

Look up the channel state information based on the Ethernet DA and SA in the received frame plus the link-layer priority/flow ID from the LARQ header. (Set up a new channel if necessary.)

If the received sequence number of the received frame is out of sequence, the channel state may be reset. If the sequence number (before resetting) is old, and the forget timer has expired, then the sequence space may be reset to the value of the received frame's sequence number.

If the received sequence number is newer than the Current Sequence Number (after any reset of the sequence number space), then perform new sequence number processing steps below; otherwise perform the old sequence number processing steps.

11.7.4.4 Receiver – LARQ frames with CRC or other errors

For best performance, implementations should allow the LARQ protocol module to process errored frames, such as those with payload CRC errors. This will, in many cases, allow Nack indications to be sent more quickly since the receiver will not have to wait for the next frame to detect the loss. At the same time, it provides a second opportunity for detecting lost frames at the end of a sequence, when a later reminder would be the only protection.

If errored frames are used, they shall be used only to detect a very small set of missing sequence numbers for an existing channel (preferably one missed frame). In particular, if the errored frame appears to have a valid LARQ header, and the frame's source MAC address, destination MAC address, and LARQ header priority/flow ID match an existing logical channel, and if the sequence number is (Current Sequence Number + 1), then treat this frame as a reminder control frame for the purposes of processing. Reminder control frames are always dropped after processing.

In all other cases, drop the errored frame with no further processing. Do not set up a new channel if the frame has an error. Do not send a retransmission if a Nack control frame has an error. Do not reset a channel (for sequence numbering purposes) for an errored frame.

11.7.4.5 Receiver – New sequence number

If the frame has an error indicated by a lower layer driver, such as a CRC error, and the sequence number of the frame is anything other than (Current Sequence Number + 1), then drop the frame with no further processing. Otherwise, process the frame as a Reminder control frame.

If the difference between the new sequence number of the received frame and the oldest missing sequence number is greater than (Maximum Receive Limit – 1), then repeat the following steps until the acceptable limit is reached.

– Cancel the Nack retransmission timer and the lost frame timer for the oldest missing sequence number.

– If there is a saved frame for the next sequence number, then deliver in-sequence frames to the next layer above until the next sequence number with a missing frame is reached (which may be the next expected sequence number for the channel, (Current Sequence Number + 1)). The value from the Priority/Flow ID field from the LARQ header for each frame is delivered to the next layer along with each associated frame. The method of specifying priority/flow ID to the next layer is implementation dependent and outside the scope of this Recommendation.

If the sequence number is the next expected sequence number (Current Sequence Number + 1) and the frame is a good data frame and there are no older missing sequence numbers, then send the frame up to the next layer.

If the sequence number is newer than (Current Sequence Number + 1), or is a reminder for (Current Sequence Number + 1), then send one or more Nack control frames requesting retransmission of the missing frame(s).

The destination address for the Nack will be the source address of the received frame. The source address will be this station's MAC address. The destination address of the received frame shall be placed in the original destination address field (NACK_DA) in the LARQ Nack control frame header. The Multiple Retransmission flag shall be set to 0. The [first] missing sequence number shall be placed in the sequence number field. The priority for the NACK control frame shall be the same as the priority for the channel.

If multiple Nack control frames shall be sent, the earliest sequence number shall be sent first.

For each missing sequence number, a Nack retransmission timer shall be started, set to expire at the current time plus Nack Retransmission Interval.

For each missing sequence number, a lost frame timer shall be started, set to expire at the current time plus Maximum Hold Interval.

If the frame is a good data frame and was not delivered to the next layer, then save it.

If the frame is a reminder frame (or an errored data frame), then drop it.

Advance the Current Sequence Number to the sequence number in the received frame.

11.7.4.6 Receiver – Old sequence number

If the sequence number is the same or older than Current Sequence Number, then it shall not generate control frames, although it may itself be dropped, held, or sent up to the next higher layer, possibly causing other held frames to be sent up as well. It may cause the cancellation of a Nack retransmission timer or lost frame timer associated with that sequence number.

If the frame is not a good (e.g., bad CRC) data frame, or its sequence number is older than the oldest missing frame, or it has already been received (this is a duplicate retransmission), or it is a reminder frame, then drop the frame and skip further processing for this frame.

Cancel the Nack retransmission timer and the lost frame timer for the sequence number.

If the sequence is not the oldest missing sequence number, then save the frame.

If the sequence number is the oldest missing sequence number, then deliver the frame up to the next higher layer. If there is a saved frame for the next sequence number, then deliver in-sequence frames to the layer above until the next sequence number with a missing frame is reached (which may be the next expected sequence number for the channel). The value from the priority/flow ID field from the LARQ header for each frame shall be delivered to the next layer along with each associated frame. The method of specifying priority/flow ID to the next layer is implementation dependent and outside the scope of this Recommendation.

11.7.4.7 Receiver – Nack retransmission timer expires

If a Nack retransmission timer expires, then send another Nack control frame for the associated sequence number. The priority for the NACK control frame shall be the same as the priority for the channel. Multiple sequence numbers may be Nacked at the same time, if their timers expire at similar times.

The multiple retransmission flag shall be set to 1 for Nack control frames sent as a result of retransmission timer expiration.

While there is no explicit limit on the number of Nack control frames sent for a particular sequence number, the Nack timer shall be cancelled if the frame will be received or if the sequence number will be declared lost.

11.7.4.8 Receiver – Lost frame timer expires

The lost frame timer is implementation dependent. Its purpose is to set an upper bound on how long frames will be held before they are sent up when a frame is really lost. If set too long, network resources may be wasted on NACK control frames sent for frames that the sender on the channel would never retransmit. Further, higher layer transport timers may also become involved. The default value of 150 ms is strongly suggested as an upper bound.

Upon expiration, the sequence number shall be declared lost, resulting in the cancellation of the Nack retransmission timer and the lost frame timer for the sequence number. If there is a saved frame for the next sequence number, then send up in-sequence frames until the next sequence number with a missing frame is reached (which may be the next expected sequence number for the channel).

If the lost frame timers for multiple sequence numbers expire at the same time, then the timers are processed in sequence from oldest to newest.

11.7.4.9 Receiver – Forget timer

The forget timer is an implementation-dependent mechanism to allow a receiver to reset the sequence number space of a channel when a received sequence number is not the next expected (Current Sequence Number + 1) and a relatively long interval has expired since the last frame received on the channel. Once expired, a receiver should accept any unusual sequence number as the next expected sequence number, allowing for undetected resets of other stations, disconnection from the network, etc. The definition of "unusual sequence number" is implementation dependent, but generally means any old sequence number or any new sequence number that is not close to the current sequence number, where "close" is 1 or some other small integer. A one-second default is suggested.

11.7.4.10 Receiver – Resource management

In general, the receiver should set upper bounds on the number of held frames per channel and the number of held frames across channels. The bounds may vary based on the priority/flow ID of the channel.

Timer intervals may vary based on factors such as the priority/flow ID of the channel, or measured intervals for successful retransmissions.

The description above suggests per-sequence number timers. This is for descriptive purposes only, and does not imply any implementation mechanism.

11.8 Vendor-specific formats

The following two types (see Tables 11-20 and 11-21) allow vendor-specific extensions which may be reasonably handled by implementations that do not otherwise support them. The short format vendor-specific format allows short control messages and encapsulation headers, while the long format subtype allows other extensions that require longer messages.

Table 11-20 – Vendor-specific short frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_VENDOR_SHORT (5)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength shall be ≥ 6 for SSVersion 0.
SSVersion	1 octet	= 0
Vendor OUI	3 octets	An IEEE-assigned Organizationally Unique Identifier
Control data	0-249 octets	Vendor-specific control data
Next Ethertype	2 octets	= next Ethertype if an encapsulation format, or 0 if no encapsulated frame
Pad	0-38 octets	Any value octet
FCS	4 octets	
CRC-16	2 octets	HNT frame check sequence

Table 11-21 – Vendor-specific long frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_VENDOR_LONG (32769)
LSLength	2 octets	Number of additional octets starting with the LSVersion field and ending with the second (last) octet of the Next Ethertype field. LSLength shall be > 6 for LSVersion 0.
LSVersion	1 octet	= 0
Vendor OUI	3 octets	An IEEE-assigned Organizationally Unique Identifier
Control data	1-65531 octets	Vendor-specific data
Next Ethertype	2 octets	= next Ethertype if an encapsulation format, or 0 if no encapsulated frame
Pad	40-0 octets	If needed to make minimum size frame. Should be zero
FCS	4 octets	
CRC-16	2 octets	HNT frame check sequence

11.9 HNT certification and diagnostics protocol

11.9.1 Scope

This protocol is required for G.9954v2-compliant nodes being submitted for certification testing. Use of this protocol by G.9954v2 nodes is required.

Devices submitted for HNT certification testing only need to implement the server portion of the protocol. The same driver implementation should be used for both certification testing and production devices. However, for devices that have stringent resource constraints, the certification and diagnostics protocol may be implemented in a special driver used only for the certification tests.

11.9.2 Overview

The HNT certification and diagnostics protocol is designed to provide the required framework for testing of systems providing HNT interfaces. Specifically, it aims to provide a common set of functionality required (equivalent to cert_tool.exe and the UDP functionality of epi_ttcp) for certification testing while minimizing the impact on system design. This protocol is a component of a solution which should provide a control and test interface that enables execution and reporting of a complete certification test case suite, regardless of DUT implementation.

This Recommendation specifies the protocol itself, and does not address details of using the protocol for a specific test or diagnostic function. Such details are dependent on the specific test(s) being performed (e.g., HNT certification testing vs. network diagnostics) and as such are outside the scope of this Recommendation.

The protocol is designed to be operating-system and platform-independent, and is intended to support certification testing, with possible extensions to support network-wide diagnostics, system development, and manufacturing and QA testing.

For brevity, we will use the term "cert" to refer to the G.9954v2 certification and diagnostics protocol.

All cert activity (control and data frames) is restricted to the physical segment under test. There is no support for doing cert through another interface. All control frames received on an interface are

only relevant to that interface.

11.9.3 Control

One node on the network is the protocol controller, which will be referred to as the "client". The client initiates and coordinates all certification and diagnostics activity. The client portion of the protocol should be enabled on only one node in a network at any time.

All other nodes on the network are "servers". They service requests from the client by adjusting their configuration as directed by the client, or by sourcing and sinking cert data frames as requested by the client. Client nodes should also provide all the functionality of a server. Generally, the server will be implemented within the device driver for HNT nodes, but it may be implemented at a higher layer above any network device assuming that HNT link control frame (LCF) frames can be passed by the server to and from the device driver. In order to minimize the impact on system resources, the server functionality of the cert protocol is intended to be as minimal and straightforward as possible. Cert frames are grouped into two categories: control and data frames. Control frames are used to configure nodes and collect information from nodes. Data frames are used to test transmit and receive capabilities of nodes. Control request frames are only generated by the client. Servers generate replies to the control requests, and generate data frames as directed by the client.

Servers shall reply to control requests within five seconds. Servers shall complete any configuration changes (e.g., HNT mode changes) initiated by the control request within five seconds after receipt of the control request.

Devices submitted for HNT certification testing using this protocol shall implement the server portion of the protocol. Implementation of the client portion of the protocol is not required. Cert frames shall not be bridged by any node.

Control frames should be sent at link layer (LL) priority 7. Data frames shall be sent at the LL priority/flow ID specified by the client when initiating the data transmission. If any encapsulating protocols (e.g., LARQ) are enabled on a node, the data frames shall be sent with the enabled encapsulation(s) to facilitate testing of the protocol implementation(s). Control frames may be encapsulated. Cert clients and servers shall be able to de-encapsulate cert control frames to the same extent that they are required to de-encapsulate data frames. HNT nodes shall be capable of removing one encapsulating Short Format Link Control Frame header from cert control frames.

11.9.4 Frame format

Cert frames use the basic HNT link-layer control frame (LCF) format defined in "Interface Specification for HNT Technology Link-Layer Protocols". A single long-subtype frame format is defined with a common header structure used with all cert frames and a variable number of command or data segments as shown in Table 11-22.

Table 11-22 – Certification and diagnostics frame format

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_CERT (32770)
LSLength	2 octets	Number of additional octets in the control header, starting with the LSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum is 6 for LSVersion 0.
LSVersion	1 octet	= 0

Table 11-22 – Certification and diagnostics frame format

Field	Length	Meaning
OpCode	1 octet	Command segment set used in this frame
Reserved	4 octets	
Cert_Seq	2 octets	Frame sequence number
CommandData	0-1486 octets	Command data, may be empty, or contain one or more Command Segments, or one Data Segment
Next Ethertype	2 octets	= 0
Pad	40-0 octets	Should be zero
FCS	4 octets	
CRC-16	2 octets	HNT frame check sequence

The command segments use the format shown in Table 11-23.

Table 11-23/G.9954 – Command segment format

Field	Length	Meaning
CSType	2 octets	The type of the command segment.
CSLength	2 octets	Number of octets in the CSPayload field. Valid values are nominally 0 to 1482. However, for some CSType values the CSLength field is fixed. The high 3 bits are reserved in version 0; they shall be sent as 0 and ignored on reception.
CSPayload	0-1482 octets	Command-specific information. May be empty.
CSPad	0-3 octets	If present, shall be sent as 0, ignored on reception. Aligns subsequent command segments on 32-bit boundaries. Shall be present if CSLength is not a multiple of 4.

Data segments use the format shown in Table 11-24.

Table 11-24 – Data segment format

Field	Length	Meaning
DSType	2 octets	The type of the data segment.
DSLength	2 octets	Number of octets in the DSPayload field. Valid values are nominally 1 to 1482. The high 3 bits are reserved in version 0; they shall be sent as 0 and ignored on reception.
DSPayload	1-1482 octets	Data

Replies from a server can span multiple frames, but individual command segments shall not extend across frame boundaries.

When multiple command segments are present in a frame, they shall be sent in order by ascending tag value.

All command segments shall be aligned on 4-byte boundaries. All command segments shall be padded to a multiple of 4 bytes. Data segments shall not be padded, and shall not be combined with command segments.

11.9.5 Opcodes

Server nodes generate the opcodes shown in Table 11-25.

Table 11-25 – Server node opcodes

Mnemonic	Opcode
OK	0x00
ERROR	0x01
TESTDATA	0x02
SAMPLEDATA	0x03

Client nodes generate the opcodes shown in Table 11-26.

Table 11-26 – Client node opcodes

Mnemonic	Opcode
ENABLECERT	0x08
DISABLECERT	0x09
CONFIGNODE	0x10
CONFIGSEND	0x11
STARTSEND	0x12
STOPSEND	0x13
ECHOREQUEST	0x14
CONFIGRECV	0x15
STOPRECV	0x16
REPORTSTATS	0x17
REPORTCONFIG	0x18
RESETSTATS	0x19
REPORTNODE	0x20
STARTSAMPLE	0x30
VENDOR	0x40

11.9.6 Command segments

Command segments are listed in groups, with the Opcode(s) that use them preceding each group. See Table 11-27.

Table 11-27 – Command segment groups

Mnemonic	CSType	CSLen	CSPayload values	Description
Opcode: ERROR				
ERRORCODE	0x0001	1	1-8	An index indicating the error from the list: 1 – UNK 2 – UNSUP_OP 3 – INVALID_PARAM 4 – UNSUP_CMDSEG 5 – UNSUP_DGEN 6 – INVALID_SEQ 7 – INVALID_FRAME 8 – INVALID_OP
Opcodes: OK(REPORTCONFIG) OK(REPORTSTATS) OK(REPORTNODE)				
INFOREPLY	0x0002	2	Two 8-bit values	Number of reply frames – 1, plus index of current frame (starting with 0).
Opcodes: STARTSEND STOPSEND STOPRECV				
REFSEQ	0x0005	2	Any	REFSEQ value contains the Cert_Seq value from a previous command.
Opcode: VENDOR				
OUI	0x0023	3	IEEE OUI	Vendor commands are sent with this command segment first.
Opcodes: CONFIG_NODEOK (in response to REPORTCONFIG)				
TXPE	0x0010	1	1-7 9-15 (optional) 255 (default)	Fixed PE, rate negotiation disabled. Fixed PE, rate negotiation disabled. Rate negotiation enabled.
TXPRI	0x0011	1	0-7 255 (default)	Fixed transmit PHY priority. Use LL priority, negotiate priority map via CSA.
LINKINT	0x0012	1	0 1 (default)	Link integrity disabled Link integrity enabled
TXMODE	0x0013	1	0 1 (default) 2	Disable all transmissions. Enable all transmissions. Enable only HNT link control frame transmissions.
HPNAMODE	0x0016	1	0-7 (default)	Reserved for legacy usage
LARQ (optional)	0x0020	1	0 1	LARQ disabled (but headers stripped). LARQ enabled

Table 11-27 – Command segment groups

Mnemonic	CSType	CSLen	CSPayload values	Description
CSA (optional)	0x0021	1	0 1	CSA disabled. CSA enabled.
CSAHPNAMODE (optional)	0x0022	1	0 (default) 1 2 3	Do not set any mode config flags in CSA messages. Reserved for legacy usage Reserved for legacy usage. Reserved for legacy usage.
Opcodes: STARTSAMPLE				
SAMPLE	0x0030	14	MAC address	Octet 0-5: SA of channel
			MAC address	Octet 6-11: DA of channel
			0 = None 1 = GAP 2 = PREAMBLE	Octet 12: Test type
			0	Octet 13: Reserved – Shall be set to zero by the transmitter and ignored by the receiver
Opcodes: CONFIGSEND, CONFIGRECV				
DGEN_TYPE	0x0084	1	1,2	Data generator to use for the data segment of the frames. See clause 11.9.17.
DGEN_DATA	0x0085	4	Any	Initialization value for data generator. See clause 11.9.17.
LENGTH	0x0086	2	1-1482	Length of the data segment of the frames to be sent.
SA	0x0081	6	Unicast MAC address	MAC address of the node that will be the source of the data frames (generally the MAC address of the recipient of the CONFIGSEND request).
DA	0x0083	6	Any MAC address	MAC address of the node(s) that will be the recipient(s) of the data frames. A total of ten DA segments may be present, and must be supported.
Opcode: CONFIGSEND				
NPKTS	0x0087	4	Any (default = 0)	Total number of packets to send. 0 means send frames continuously until a STOPSEND request is received.
BURST_INT	0x0088	2	Any (default = 0)	Interval between start of bursts in milliseconds. 0 means send frames without any pacing.
BURST_NPKTS	0x0089	2	! = 0 (default = 1)	Number of packets to send per burst

Table 11-27 – Command segment groups

Mnemonic	CSType	CSLen	CSPayload values	Description
NUMACKS	0x008a	1	! = 0 (default = 1)	Number of ACK and EOT frames to send. (See clause 11.9.10.2.)
TXPE_TEST	0x008b	1	All valid values as defined in Table 11-5 255 (default)	Fixed PE, rate negotiation disabled. Rate negotiation enabled Applies only to test frames being generated by the server
TXPRI_TEST	0x008c	1	0-7 255 (default)	Fixed transmit PHY priority Use LL priority, negotiate priority map via CSA Applies only to test frames being generated by the server
Opcode: OK (in response to REPORT_STATS)				
RECV_NPKTS	0x0105	4	Any	Total number of data frames received without errors, not including EOT frames
RECV_NBYTES	0x0106	4	Any	Total number of data bytes received without errors
RECV_SEQ_MISS	0x0107	4	Any	Number of missing data frames detected via gaps in sequence numbers
RECV_SEQ_ERR	0x0108	4	Any	Number of data frames received with unexpected sequence numbers
RECV_DATA_ERR	0x0109	4	Any	Number of data frames received with detected data corruption
RECV_FCS_ERR	0x010c	4	Any	Number of frames received with FCS errors
RECV_HDR_ERR		4	Any	Number of frames received with detected header errors
RECV_ERR	0x010a	4	Any	Number of frames with other recv errors
RECV_ELAPSED_ TIME	0x010b	4	Any	Receive test elapsed time in milliseconds
XMT_NPKTS	0x0101	4	Any	Total number of data frames sent without errors reported by lower layers (e.g., excessive collisions), not including EOT frames
XMT_NBYTES	0x0102	4	Any	Total number of data bytes sent without errors
XMT_NERRS	0x0103	4	Any	Number of transmit errors reported by lower layers that resulted in lost frames (e.g., excessive collisions).

Table 11-27 – Command segment groups

Mnemonic	CSType	CSLen	CSPayload values	Description
XMT_ELAPSED_TIME	0x0104	4	Any	Transmit elapsed time in milliseconds
Opcode: OK (in response to REPORTNODE)				
PRIMARY_ID	0x8301	4	Any	Primary vendor/device ID
SUBSYSTEM_ID	0x8302	4	Any	Subsystem vendor/device ID
MAC_ADDRESS	0x8303	6	Any	IEEE 48-bit MAC address
SERIAL_NUM	0x8304	≤ 16	ASCII	
DEVICE_TYPE	0x8305	1	0-24	An index indicating the device type: 0 – Other 1 – CI NIC (includes miniPCI, Cardbus) 2 – USB NIC 3 – Cable modem bridge 4 – DSL modem bridge 5 – Broadband wireless bridge 6 – V90 bridge 7 – Stand-alone bridge 8 – Cable modem router 9 – DSL modem router 10 – Broadband wireless router 11 – V90 router 12 – Stand-alone router 13 – Audio device 14 – Video device 15 – Disk device 16 – CD/DVD device 17 – Backup device 18 – Digital cable set-top 19 – Digital satellite set-top 20 – Printer 21 – Print server 22 – Scanner 23 – FAX 24 – Phone
VEND_NAME	0x8306	≤ 32	ASCII	
VEND_DRIVER	0x8307	≤ 16	ASCII	
VEND_DATE	0x8308	4	1 to $(2^{32} – 1)$	
MANUF_DATE	0x8309	4	1 to $(2^{32} – 1)$	
TIMER_GRAN	0x830a	2	1-1000	Timer resolution in milliseconds

11.9.7 Data segments

Data segments are listed in groups (see Table 11-28), with the Opcode(s) that use them preceding each group.

Table 11-28 – Data segment groups

Mnemonic	CSType	CSLen	CSPayload values	Description
Opcodes: TESTDATA ECHOREQUEST OK (in response to ECHOREQUEST)				
DATA	0x8108	1-1482	Any	Data
Opcode: TESTDATA				
EOT	0x8109	0	N/A	End of transmission: Marks the end of the server's data transmission
Opcode: SAMPLEDATA				
SAMPLES	0x8133	1-1482	MAC address	Octet 0-5: Source address of channel
			0-65535	Octet 6-7: Total number of samples in test
			0-65535	Octet 8-9: Index of first sample in this segment
			0 = None 1 = GAP 2 = PREAMBLE	Octet 10: The test type (from CSPayload of command segment)
			0	Octet 11: Reserved for future use. Shall be set to zero by the transmitter and ignored by the receiver.
			Samples…	Octet 12 to (DSLength-13): Signed 16-bit samples

11.9.8 Server Opcode usage

11.9.8.1 OK

Opcode OK messages are generated in response to control requests that are successfully completed. Opcode OK messages contain variable number of command segments, depending on the control request. Opcode OK messages with zero command segments are referred to as "empty OK" messages.

The Cert_Seq field in the OK message shall be set to the value of the Cert_Seq field from the control request.

If multiple OK messages are being generated in response to a single command request, the INFOREPLY command segment shall be the first segment in each reply frame. An INFOREPLY command segment may be included as the first command segment when a single OK message is being generated.

11.9.8.2 ERROR

Opcode ERROR messages are generated in response to control requests which are malformed, not understood, or could not be completed successfully. The Cert_Seq field in the ERROR message shall be set to the value of the Cert_Seq field from the control request. Opcode ERROR messages shall contain one or two command segments. The first command segment shall have CSType = ERRORCODE. The second command segment, if present, shall be an ERRORPOINTER command

segment with CSPayload containing the first four octets (CSType and CSLength) from the first command segment that caused the problem, if it can be identified.

11.9.8.3 TESTDATA

Opcode TESTDATA frames are used to measure performance (e.g., frame error rate) or implementation (e.g., in-order delivery of LARQ encapsulated frames) characteristics of the nodes being tested, and are typically sent between two servers. The Cert_Seq field in the TESTDATA messages typically starts at 0 for each test, and increases by one for each subsequent TESTDATA frame sent as part of that test.

Opcode TESTDATA messages shall contain a single data segment of with DSType = DATA or command segment with CSType = EOT.

11.9.8.4 SAMPLEDATA

Opcode SAMPLEDATA frames are used to support a spectral analysis of a HNT channel from server A to server B as seen by server B. Upon receiving STARTSAMPLE command, the source of the tested channel shall send a Link-Layer Link Integrity message to the destination of the channel. The destination of the channel shall send SAMPLES data segment(s) to the server containing 32 symbols worth of samples using its native sample rate. If the samples span more than one data segment, then the segments should be sent in an ascending order of sample index.

When the test type is PREAMBLE, the samples shall represent symbols 25 to 56 of the preamble for the frame received from the source of the channel.

When the test type is GAP, the samples shall represent a period in the inter-frame-gap that starts 8 microseconds after the reception of the frame.

11.9.9 Client opcode usage

11.9.9.1 ENABLECERT

At startup, or after receipt of a DISABLECERT request, servers shall be in "cert disabled" mode. While in "cert disabled" mode, the node shall silently ignore all received cert frames except DISABLECERT and ENABLECERT requests until an error-free ENABLECERT request has been received. After receipt of an ENABLECERT request, the node shall check the format of the received frame. If no errors are detected, the node shall switch to (or remain in) "cert enabled" mode and reply with an empty OK message. If an error in the frame format is detected, the node shall reply with an ERROR message and shall not switch modes.

11.9.9.2 DISABLECERT

After receipt of a DISABLECERT request, servers shall check the format of the received frame. If no errors are detected, the node shall reply with an empty OK message, switch to (or remain in) "cert disabled" mode, and then silently ignore all subsequent received cert frames except DISABLECERT and ENABLECERT requests. If an error in the frame format is detected, the node shall reply with an ERROR message and shall not switch modes.

11.9.9.3 CONFIGNODE

Opcode CONFIGNODE messages may contain exactly one of the following command segments:

- TXPE;
- TXPRI;
- LINKINT;
- TXMODE;
- HPNAMODE;
- LARQ;

- CSA;
- CSAHPNAMODE.

All servers shall support the TXPRI, LINKINT, and TXMODE command segments. All servers shall support TXPE settings as defined in Table 11-5 and the value of 255 as defined in Table 11-27. Servers shall support the LARQ command segment if and only if they implement the LARQ protocol. Servers shall support the CSA and CSAHPNAMODE command segments if and only if they implement the CSA protocol.

If a server receives a CONFIGNODE request with an unsupported or invalid command segment, it shall reply with an ERROR message. Otherwise, it shall reply with an empty OK message.

11.9.9.4 CONFIGSEND

These command segments shall be provided in a CONFIGSEND request, in the order listed:

- DGEN_TYPE;
- DGEN_DATA;
- LENGTH;
- SA;
- DA.

DA is the only CSType in a CONFIGSEND request that may be repeated, and if repeated, all DA segments shall be contiguous. Implementations shall support at least ten DA command segments in a CONFIGSEND request. CONFIGSEND command segments shall only be sent to unicast addresses.

The traffic generator is responsible for generating the data in the frames, the size of the frames, and the distribution of frames in the case of multiple DAs. The most commonly used generator is fixed data, fixed-length frames, round-robin distribution to all DAs.

The following command segments are optional in a CONFIGSEND request, but if present, all shall be sent in the order listed:

- NPKTS;
- BURST_INT;
- BURST_NPKTS;
- NUMACKS;
- TXPE_TEST;
- TXPRI_TEST.

If the server cannot provide the resolution implied by BURST_INT, then the value shall be rounded up to the closest value which the server can provide.

If BURST_INT is not specified or is 0, then the data sending node shall generate frames as fast as possible without dropping frames on the transmit side.

If NPKTS is not specified or is 0, then the data-sending node shall generate data frames until a STOPSEND request is received.

The receiving node shall reply with an ERROR message if any unsupported parameters (or unsupported values for supported parameters) are included in the CONFIGSEND request, if the receiving node is already in the process of sending cert data frames from a previous CONFIGSEND/STARTSEND set of requests, if more than one CONFIGSEND is received before a STARTSEND request is received, or if the SA in the CONFIGSEND request is not the receiving node's MAC address. Otherwise, the receiving node shall reset the transmit counters listed in clause 11.9.11, set any optional parameters not included in the CONFIGSEND request to their

default values, and reply with an empty OK message.

11.9.9.5 STARTSEND

STARTSEND requests contain one or more command segments of CSType = REFSEQ. Each REFSEQ value matches the Cert_Seq value of a CONFIGSEND request that was previously issued. Receiving nodes shall follow the protocol defined in clause 11.9.10.2.

11.9.9.6 STOPSEND

STOPSEND requests include one or more command segments of CSType = REFSEQ. Each REFSEQ value matches the Cert_Seq value from a CONFIGSEND request that created a data stream. When a server receives a STOPSEND request, it compares the Cert_Seq value(s) in the request to the Cert_Seq value from the last CONFIGSEND request it received. If there is a match, the server shall reply with a single OK message, containing one command segment of CSType = REFSEQ with the Cert_Seq value that matched. If a STOPSEND request is received while data frames are being sent, the transmitting node shall stop sending data frames. If there is no match, or if the node has not received any CONFIGSEND requests, then it shall silently ignore the request.

11.9.9.7 ECHOREQUEST

ECHOREQUEST frames contain a single data segment of DSType = DATA. The client fills the DSPayload field with the data it wishes to get echoed back (from 1 to 1482 bytes), and sets the DSLength field appropriately. The receiver shall reply with an OK message containing a copy of the data segment from the ECHOREQUEST command.

11.9.9.8 CONFIGRECV

These command segments shall be provided in a CONFIGRECV request, in the order listed:

- DGEN_TYPE;
- DGEN_DATA;
- LENGTH;
- SA;
- DA.

DA is the only CSType in a CONFIGRECV request that may be repeated, and if repeated, all DA segments shall be contiguous. CONFIGRECV command segments shall only be sent to unicast addresses.

The receiving node shall reply with a single ERROR message if any unsupported parameters (or unsupported values for supported parameters) are included in the CONFIGRECV request, or if its MAC address does not appear in any of the DA command segments. Otherwise, the receiving node shall reset the receive counters listed in clause 11.9.11, set any optional parameters not included in the CONFIGRECV request to their default values, and reply with an empty OK message.

11.9.9.9 STOPRECV

STOPRECV requests include one or more command segments of CSType = REFSEQ. Each REFSEQ value matches the Cert_Seq value from a CONFIGRECV request that created a data stream. When a server receives a STOPRECV request, it compares the Cert_Seq value(s) in the request to the Cert_Seq value from the last CONFIGRECV request it received. If there is a match, the server shall immediately compute the elapsed time from the start of the test or, if no data frames have been received, set the elapsed time to zero, and reply with a single OK message, containing one command segment of CSType = REFSEQ with the Cert_Seq value that matched. Any subsequent data frames received shall be ignored. If there is no match, or if the node has not received any CONFIGRECV requests, then it shall silently ignore the request.

11.9.9.10 REPORTSTATS

The receiver shall reply with an OK message containing the counters listed in clause 11.9.11, in the order listed in that clause. The counters shall not be reset after being reported, in case the reply is lost and the client needs to repeat the REPORTSTATS command. The reply message shall start with an INFOREPLY command segment, followed by command segments for each of the required counters.

11.9.9.11 REPORTCONFIG

The receiver shall reply with an OK message containing the current settings for the configuration parameters listed in clause 11.9.9.3. The reply message shall start with an INFOREPLY command segment, followed by command segments for each of the required parameters. The command segments shall be sent in the order listed in clause 11.9.9.3. The first five configuration parameters shall be reported, while the last three, LARQ, CSA, and CSAHPNAMODE, shall be reported only if supported.

11.9.9.12 RESETSTATS

The receiver shall reset all the counters listed in clause 11.9.11 and reply with an empty OK message.

11.9.9.13 REPORTNODE

The receiver shall reply with an OK message containing fixed information pertaining to the node, such as identifiers, software/hardware versions, etc. The reply frames shall each begin with an INFOREPLY command segment, followed by command segments from the following list, sent in the order listed:

- PRIMARY_ID;
- SUBSYSTEM_ID;
- MAC_ADDRESS;
- SERIAL_NUM;
- DEVICE_TYPE;
- VEND_NAME;
- VEND_DRIVER;
- VEND_DATE;
- MANUF_DATE;
- TIMER_GRAN.

11.9.9.14 STARTSAMPLE

The client shall initiate sampling of the channel by sending a SAMPLE command segment. The DA of the Certification and Diagnostics frame shall be BROADCAST. The client shall then wait for all the "SAMPLES" data segments to arrive. The application should use a proper time-out in case the server(s) do not reply.

11.9.9.15 VENDOR

This opcode allows vendors to implement a private set of functions. The first command segment shall be CSType = OUI with CSPayload set to the vendor's OUI. A node receiving a vendor-specific command request with an OUI that does not match an OUI it understands shall return an INVALID_PARAM error message. The behaviour of nodes that receive a vendor-specific command request with a matching OUI is at the discretion of the vendor, and is outside the scope of this Recommendation.

11.9.10 Control request protocol

11.9.10.1 General control requests

All control requests other than STARTSEND and VENDOR follow a trivial protocol: The client sends a single-frame request, and the server replies with one or more frames – all control frames sent by the client are explicitly "acked" with either OK or ERROR or SAMPLEDATA in case of STARTSAMPLE. For most cases, a single frame is generated. Each control frame generated by the client shall be sent with a monotonically increasing (ignoring rollover) value for Cert_Seq. The Cert_Seq field in the acknowledgement frames from the server nodes use the Cert_Seq value from the control request to ensure that the client can properly identify which request is being acked. The client shall be responsible for dealing with unacknowledged requests, e.g., by resending the request after some timeout, with a possible delay between attempts. Failure to receive an ack can mean that either the original request frame was lost, or the ack was lost. For all currently defined requests except STARTSEND, there is no negative impact to resending a request. The timeout value used by the client is dependent on the specific request being issued. For config commands, a timeout of 50 ms should be used. The client behaviour if repeated failures are encountered is dependent on the goals of the client (certification testing vs. network diagnostics) and is not specified here.

In the case of a REPORTSTATS or REPORTCONFIG request, some number (≥ 1) of reply frames are generated by the server. The first command segment of all reply frames sent in response to REPORTSTATS, REPORTCONFIG, and REPORTNODE requests shall be an INFOREPLY command segment indicating the total number of frames to be sent and the relative number of the current frame.

Subsequent command segments contain the data being returned by the server.

All reply frames are sent with Cert_Seq value set to the Cert_Seq from the client's request. The client shall be responsible for ensuring that all frames have been received and reissuing the request if any frames will be lost.

11.9.10.2 Protocol for STARTSEND control requests

In order to provide an uninterrupted flow of data frames during a test, a somewhat different protocol shall be used for STARTSEND requests. After issuing the appropriate CONFIGRECV and CONFIGSEND requests to configure all nodes, the client issues a STARTSEND request with a list of control segments of type REFSEQ each containing the Cert_Seq for a previous CONFIGSEND request. Any node expecting a STARTSEND request (i.e., one which has received a CONFIGSEND but has not yet received a STARTSEND request) which receives the STARTSEND request looks through the list of REFSEQ control segments in the STARTSEND request for a Cert_Seq value which matches the sequence number from the CONFIGSEND request. If no match is found, the server silently ignores the STARTSEND request. If a match is found, the node sends NUMACKS OK control replies to the client with a REFSEQ control segment containing the Cert_Seq value of the CONFIGSEND request. The server then sends the requested data frames to the destination address(es). The Cert_Seq field in the data frames starts at zero, and increases by one (modulo 2^16) for each data frame sent. After all data frames have been sent, the server then sends NUMACKS data frames with command segment type EOT, with CSValue set to the sequence number of the CONFIGSEND request, to each of the destination addresses. Upon receipt of the EOT frame, the destination node(s) measure the elapsed time of the data sent and discard any data frames received after the EOT. The EOT frame shall not be counted in the receive statistics. A timeline of a typical data test is shown in Figure 11-6.

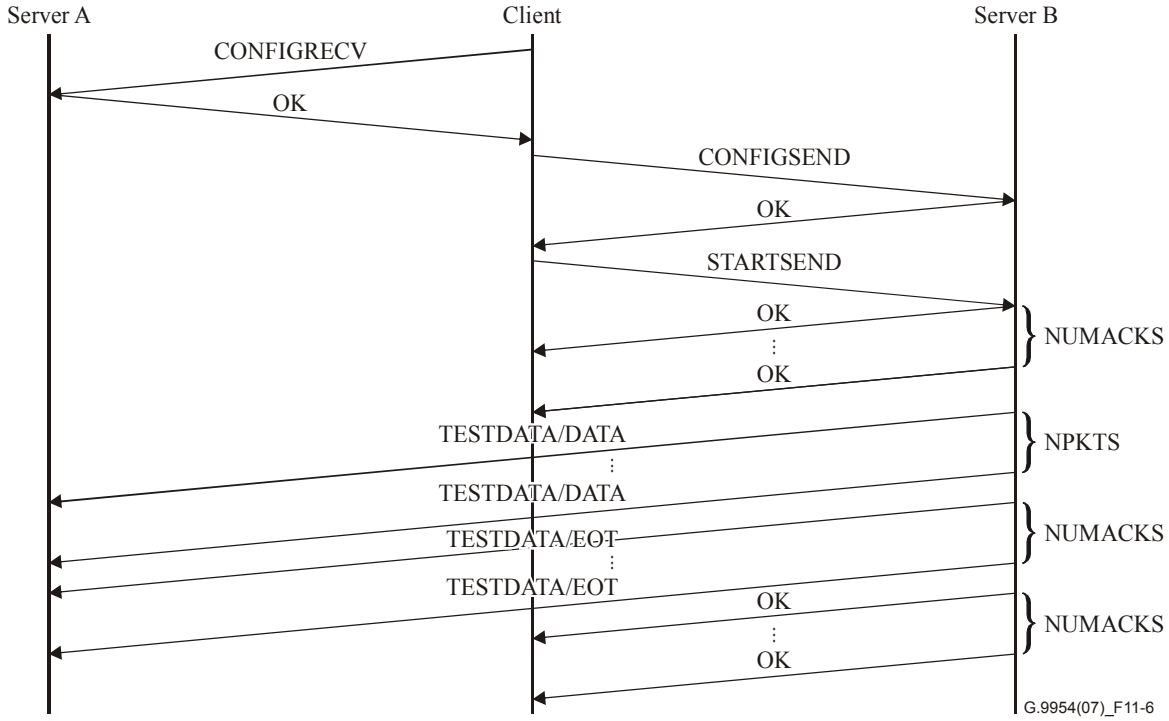

Figure 11-6 – Protocol timeline for data test

The server then sends NUMACKS OK control reply frames with command segment type EOT, with CSValue set to the sequence number of the CONFIGSEND request, to the client.

If no transmit priority has been configured, then the data frames are sent at the default LL priority 0, and the control reply frames to the client are sent at LL priority 7. The server should ensure that all data frames (including the EOT frames) have been sent on the wire before the control reply frames are sent to the client.

If a server receives a duplicate STARTSEND request for a given CONFIGSEND request (indicating that the client did not receive any of the initial NUMACKS OK control replies), the server shall return an ERROR frame to the client. The client shall be responsible for issuing any necessary STOPSEND requests, reconfiguring the nodes as necessary and restarting the test.

For STARTSEND requests, the timeout which the client should use while looking for the initial acks is 50 ms. The timeout for the final acks (those sent back to the client after the data frames containing EOT segments have been sent) needs to be calculated based on the amount of data being transmitted and the worst-case throughput for the test.

11.9.10.3 Protocol for VENDOR control requests

The protocol for VENDOR control requests is at the discretion of the vendor, and is outside the scope of this Recommendation.

11.9.11 Stats

11.9.11.1 Receive counters

The following counters shall be maintained by a server receiving data frames, and reported in response to a REPORTSTATS command:

- RECV_NPKTS;
- RECV_NBYTES;
- RECV_SEQ_MISS;
- RECV_SEQ_ERR;

- RECV_DATA_ERR;
- RECV_FCS_ERR;
- RECV_HDR_ERR;
- RECV_ERR;
- RECV_ELAPSED_TIME.

11.9.11.2 Transmit counters

The following counters shall be maintained by a server sending data frames, and reported in response to a REPORTSTATS command:

- XMT_NPKTS;
- XMT_NBYTES;
- XMT_NERRS;
- XMT_ELAPSED_TIME.

All counters shall be maintained and reported as 32 bits.

Elapsed time shall be measured from transmit or receive of the first data frame until transmit or receive of the first EOT frame.

11.9.12 Receiver processing of control frames

Frames with HCS, FCS or CRC-16 errors are not used by cert. Since some implementations may exist as a separate layer above the device driver, there is no guarantee across implementations that frames with these errors will reach the cert layer. Thus, for consistency, all cert implementations shall ignore any frames received with any of these errors.

11.9.13 Receiver processing of data frames

For each received data frame:

- If DGEN_TYPE and DGEN_DATA were specified in the CONFIGRECV request, the receiver generates a local copy of the packet using the data generator and compares to the received packet data. If the data fails to match, the receiver increments recv_data_err. If no errors are detected, the receiver increments recv_npkts.
- The receiver tracks the sequence number of the received frames and increments RECV_SEQ_MISS for any frames that have been missed (as evidenced by gaps in the sequence numbers) and increments RECV_SEQ_ERR for any frames received out of sequence.

The following logic shall be used to increment recv_seq_miss and recv_seq_err:

```
if (((received_seq – expected seq) & 2^15) != 0) recv_seq_err++;
else {
     recv_npkts++;
     if (received_seq == expected_seq) expected_seq = (expected_seq + 1) %
     2^16;
     else {
          if (received_seq > expected_seq) recv_seq_miss += (received_seq –
          expected_seq);
          else recv_seq_miss += (2^16 + received_seq – expected_seq);
          expected_seq = (received_seq + 1) % 2^16;
     }
}
```

Duplicate frames will also increment recv_seq_err.

11.9.14 General requirements

Server nodes should be capable of sourcing and sinking data frames simultaneously, but are not required to do so. Servers shall be able to handle receipt and processing of control frames while sending data frames. This version of the protocol does not specify support for simultaneously generating multiple data streams or simultaneously receiving and validating multiple data streams.

11.9.15 Timing

The resolution on all timing (timestamps and sending intervals) should be 10 ms, and it shall not be more than 50 ms. Jitter requirement shall be ±10% of the provided resolution.

11.9.16 Error codes

The error codes have been defined as shown in Table 11-29.

Table 11-29 – Error codes

Mnemonic	Value
UNK	1
UNSUP_OP	2
INVALID_PARAM	3
UNSUP_CMDSEG	4
UNSUP_DGEN	5
INVALID_SEQ	6
INVALID_FRAME	7
INVALID_OP	8

11.9.17 Data generators

11.9.17.1 DGEN_TYPE = 1

The 4 bytes of DGEN_DATA specified in the CONFIGSEND request are replicated, as a group, to fill the length of the payload. If the payload length is not a multiple of 4, the remaining bytes are filled with the portion of DGEN_DATA that fits. For example, if DGEN_DATA = 0x01020304 and the payload length is 11, then the payload shall be filled with 0x01020304010203040102030401020304010203.

If the number of destination addresses is greater than one, then the generated frames are multiplexed to the destination nodes in the order they were listed in the CONFIGSEND request.

11.9.17.2 DGEN_TYPE = 2

The least significant byte of DGEN_DATA shall be used to initialize an 8-bit counter. Payload bytes shall be sequentially filled with the value of the counter, and the counter shall be incremented by one per payload byte. For example, if DGEN_DATA = 0xf9 and the payload length is 11, then the payload shall be filled with 0xf9fafbfcfdfeff00010203. If the number of destination addresses is greater than one, then the generated frames are multiplexed to the destination nodes in the order they were listed in the CONFIGSEND request. The three most significant bytes shall be sent as zero and ignored on receipt.

11.10 Link-layer framing extensions

This clause of the link-layer specification describes how extensions to frame formats are accomplished.

In addition, two extensions for CSA control frames to support the use of optional and/or extended features between compatible stations are defined. The first extension is a list of optional LCP frame

subtypes supported by the implementation (beyond the four basic version HNT types). New frame types, such as one for a Reed-Solomon encoded frame, would be announced by stations that implement them, allowing for simple pairwise "negotiation" of support for optional types. The second extension is a standard format for announcing parameters associated with an extended feature.

Finally, this clause adds some additional rules governing the design and usage of new/revised LCP protocols, including some more concrete guidelines on LCP header lengths and alignment restrictions.

11.10.1 Definitions

- **embed** – Place data, typically an Ethernet/802.3 frame payload, within the structure defined for an LCP subtype header, possibly encoded, in a manner that requires understanding of the structure to extract the original payload. (i.e., the original payload becomes part of the LCP header).
- **embedded payload** – The data encoded within an embedding header, typically the payload of an Ethernet/802.3 frame starting with the Type/Length field.
- **embedding header** – A header that contains an embedded payload, for which the header's function must be understood to make use of the enclosed data.
- **encapsulating header** – A header that can be removed without further processing (e.g., a LARQ header), leaving something useful, typically an Ethernet/802.3 frame payload. An encapsulating header has a non-zero Next Ethertype field.
- **encapsulate** – To insert an LCP header into a frame, prior to the original Type/Length field, without modifying the rest of the frame. Removal of the header restores the frame to its original state (i.e., the original payload follows the LCP header).
- **tag length value** – A type of structure consisting of an assigned identifier, the Tag, followed by a Length field specifying the size of the data to follow, followed by the Value(data) itself.
- **TLV** – Tag length value.

11.10.2 Extension mechanism

Extensions to existing frame formats shall be added using tag-length-value (TLV) encoding, with tags assigned by HNT. The TLV format has short and long versions. The short format has an 8-bit tag and an 8-bit length field, while the long-format has a 16-bit tag and a 16-bit length. The short format uses tag values 1-127, and the long format uses tag values 32768-65535, with most significant bit of the most significant octet of the Tag field distinguishing the two formats.

Tags values are assigned independently for each LCP SSType or LSType from the full range of values (i.e., the ranges are overlapping). The tag value 0x00 is explicitly reserved as a pad value, the use of which is described below.

When TLV blocks are added, they shall precede the Next Ethertype field and follow all other non-TLV-encoded fields. The definition of new TLV extensions for a particular subtype does not automatically force the assignment of a new version for SSVersion (or LSVersion). All implementations shall ignore unknown TLV blocks. Once the first TLV extension has been defined for a particular subtype, all extensions to that subtype in the future shall require TLV encoding, including any permanent additions to future versions.

The SSVersion or LSVersion field shall be incremented when a permanent extension is defined for all future versions of an LCP subtype, or when a formerly reserved field in the permanent portion of a subtype is defined to have a use within the protocol. The version field should not be incremented for optional extensions.

11.10.3 Header size restrictions, and LCP padding

All encapsulating G.9954v2 LCP headers, short or long format, shall have lengths that are multiples of 4 octets (32 bits). The reserved tag value 0x00 shall be used as padding within the TLV portion of an LCP header to ensure the required alignment of fields (see next paragraph) and to ensure that the total length of the LCP header is a multiple of 4 octets. This requirement minimizes the cost of frame handling by higher layer protocols when headers are removed.

It is further required that all senders shall ensure natural alignment of 16-bit and 32-bit values as measured from the start of the SSType or LSType field. One, two or three octets of padding (value is 0) shall be used each time padding is required to align a following field.

11.10.4 Required support

11.10.4.1 Support for optional LCP extensions

Stations supporting G.9954v2 shall use the supported subtypes CSA extension to announce support for optional LCP subtypes, including embedding header subtypes, new control header subtypes and encapsulating subtypes other than LARQ.

For all received subtypes, stations shall ignore unknown extensions, when present, and shall process all known extensions normally as if any unknown extensions were not present.

11.10.4.2 Use of encapsulating headers

Stations shall be capable of removing an unknown encapsulating header and processing the remaining frame as if the unknown header were not present. However, stations shall not add any encapsulating header except for the standard 8-octet LARQ header unless all recipients of the frame are known to support frame lengths long enough to accommodate the extended message size that results when the encapsulating header is present. In addition, encapsulating headers other than the LARQ header should only be sent if all active listeners of the DA of the frame are known to support that type. Active Listener is defined in clause 11.4.4.2 (referred to as "active multicast/broadcast listeners").

A station is considered to support G.9954v2 or higher only if a CSA message indicating that status has been received from the station within the last two minutes. Stations that go to sleep typically do not generate CSA messages, and will therefore drop from the ranks of "known to be G.9954v2 or higher", requiring further traffic to be sent with G.9954v2 default capability limitations assumed for the receiving nodes (for example, the default MTU size) in order to ensure reasonable wake-up behaviour.

This means that encapsulating headers other than 8-octet LARQ headers shall not be used for broadcast or multicast traffic unless CSA messages from every active listener for the broadcast or multicast group indicates support for MTU lengths sufficiently long to accommodate the extended message size that results when the encapsulating header is present. A "station on the wire" is a HNT station that sends link integrity frames. The source MAC address used in the link integrity frames identifies the station. If no CSA message has been recently received (during the last two minutes) with the same source MAC address, then the station is asleep, and must be treated as G.9954v2 (see clause 11.6.5).

Stations shall not add an LCP encapsulating header subtype other than an 8-octet LARQ header if any active listener of the frame's support for MTU lengths is not sufficiently long to accommodate the extended message size that results when the encapsulating header is present. When all active listeners advertise sufficient MTU sizes, a station should not add an LCP encapsulating header subtype other than LARQ unless at least one active listener is known to support the subtype via the Supported Subtypes CSA extension.

Stations should not send an LCP control frame to stations not known to support the subtype. If the MAC destination address is a multicast/broadcast group address, then at least one active listener should be known to understand the subtype.

11.10.4.3 Use of embedding headers

Stations should not send an LCP frame with an embedded payload unless all active listeners are known to understand the subtype (via receipt from all receivers of the new supported subtypes extension to CSA control frames with the embedding subtype).

11.10.5 TLV extension formats

See Tables 11-30 to 11-33.

Table 11-30 – Short format TLV extension

Field	Length	Value/Meaning
SETag	1 octet	1-127. Tag value assigned for extension
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. Minimum is 0; maximum is 255.
SEData	0-255[a)] octets	Additional data for extension.
[a)] Limited by available space in physical or link-layer frame format.		

SELength shall not be used as an indicator of the version of information present in the SEData portion of the TLV.

Table 11-31– Long format TLV extension

Field	Length	Value/Meaning
LETag	2 octets	32768-65535. Tag value assigned for extension
LELength	2 octets	Total length of TLV extension excluding the tag and length octets. Minimum is 0; maximum is 65526.
LEData	0-65526 [a)] octets	Additional data for extension.
[a)] Limited by available space in physical or link-layer frame format.		

LELength shall not be used as an indicator of the version of information present in the LEData portion of the TLV.

Table 11-32 – Pad, may be used with all TLV extensions

Field	Length	Value/Meaning
LCP_Ext_Pad	1 octet	= 0 (LCP_EXT_PAD). May be repeated up to three times in succession.

Table 11-33 – Example: Short format frame with TLV extension

Field	Length	Value/Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c
SSType	1 octet	= x
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. The total length from SSType through Next Ethertype must be a multiple of 2 (natural alignement of Next Ethertype) with SSLength being an even integer, or a multiple of 4 (encapsulating header), with SSLength mod 4 equal to 2.
SSVersion	1 octet	= x
Fixed/known data for SSVersion		
SETag	1 octet	Tag value assigned for extension
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. Minimum is 0; maximum is 255.
SEData	0-255[a)] octets	Additional data for extension
[Additional TLV extensions]		
[padding if needed]	0-3 octets	Must be zero
Next Ethertype	2 octets	
[a)] Limited by available space in physical or link-layer frame format.		

11.10.6 CSA extensions

11.10.6.1 CSA extension for supported optional subtypes

The following extension is defined for CSA frames to allow implementations to advertise support for each optional subtype. Optional subtypes are defined as those subtypes that are defined but not required, by some version of the HNT specification. Initially, this will include any new G.9954v2 subtypes for which support is not required in G.9954v2 devices. Rather than attempt to conserve a little space, all frame types are treated as 16-bit integers, sent most significant octet first.

Table 11-34 – Supported subtypes TLV extension for CSA

Field	Length	Value/Meaning
SETag	1 octet	CSA_SUBTYPES_TAG
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. 2 × number of advertised subtypes.
Subtype1	2 octets	First supported optional subtype as a 16-bit integer (may be short or long sub-type).
[Subtype2,…,n]	2 × (n – 1) octets	Additional optional subtypes supported by the implementation.

11.10.6.2 CSA extension for subtype parameters

The following extension is defined for CSA frames to allow implementations to advertise implementation-specific parameters for individual LCP subtypes. Not all LCP frame types will require additional parameters. The definition of parameters is subtype-dependent, and outside the scope of this Recommendation.

Table 11-35 – Subtype parameters TLV extension for CSA

Field	Length	Value/Meaning
SETag	1 octet	CSA_PARAMS_TAG
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. Minimum is 3; maximum is 255.
Subtype	2 octets	The subtype for which additional parameters are being specified.
Parameter Data	1+ octets	Implementation-specific data.

11.10.6.3 Vendor-specific extension, short format

The following extension is defined for all extensible subtypes.

Table 11-36 – Vendor-specific short format TLV extension

Field	Length	Value/Meaning
SETag	1 octet	VENDOR_SHORT_TAG
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. Minimum is 4; maximum is 255.
SVsOUI	3 octets	An IEEE assigned Organizationally Unique Identifier
SvsData	0-251[a] octets	Vendor-specific data for extension.
[a] Limited by available space in physical or link-layer frame format.		

11.10.6.4 Vendor-specific extension, long format

The following extension is defined for all extensible subtypes.

Table 11-37 – Vendor-specific long format TLV extension

Field	Length	Value/Meaning
LETag	2 octets	VENDOR_LONG_TAG
LELength	2 octets	Total length of TLV extension excluding the tag and length octets. Minimum is 4; maximum is 65526.
LVsOUI	3 octets	An IEEE-assigned organizationally unique identifier
LvsData	0-65522 [a] octets	Vendor-specific data for extension.
[a] Limited by available space in physical or link-layer frame format.		

11.10.7 Subtype and tag assignments

Tables 11-38 and 11-39 list the current (and planned) assignment of LCP subtypes, and tag values for LCP extensions.

Table 11-38 – Subtype assignments

Subtype name	Value	Use
Reserved	0	Reserved
SUBTYPE_RATE	1	Rate request protocol
SUBTYPE_LINK	2	Link integrity protocol
SUBTYPE_CSA	3	Capabilities and status announcement protocol
SUBTYPE_LARQ	4	Limited automatic repeat request protocol
SUBTYPE_VENDOR_SHORT	5	Vendor-specific short format header
SUBTYPE_FRAME_BURSTING	6	Frame bursting protocol
SUBTYPE_master_SELECTION	7	Dynamic master selection protocol
SUBTYPE_TIMESTAMP_REPORT	8	Timestamp report indication
Reserved	9-127	Reserved/Unassigned
Reserved	128-255	Reserved for long message type
Reserved	32768	Reserved
SUBTYPE_VENDOR_LONG	32769	Vendor-specific long format subtype
SUBTYPE_CERT	32770	Certification protocol
SUBTYPE_RS	32771	Reed-Solomon header
SUBTYPE_MAP	32772	MAP synchronization protocol
SUBTYPE_REGISTRATION	32773	Network admission control (registration) protocol
SUBTYPE_FLOW_SIGNALLING	32774	Flow signalling protocol
Reserved	32775-65535	Reserved/Unassigned

Table 11-39 – Tag assignments

Tag name	Value	Use
LCP_EXT_PAD	0	Single octet (no length field), padding for alignment, all subtypes
VS_SHORT_TAG	1	Vendor-specific extension, short format, all subtypes
CSA_SUBTYPES_TAG	2	List of supported optional subtypes CSA only
CSA_PARAMS_TAG	3	Parameters for a subtype CSA only
RESERVED	4	Reserved for legacy systems
RRCF_RS_TAG	2	Reed-Solomon extension (see 11.11.7); Rate negotiation only
RRCF_CID_TAG	3	Logical channel ID extension (see 11.4.2); Rate negotiation only
FS_PARAMS_TAG	2	Flow parameters (see 11.16.1.1); Flow signalling only
FS_CLASSIFIER_TAG	3	Flow classification filter (see 11.16.1.2); Flow signalling only
VS_LONG_TAG	32769	Vendor-specific extension, long format, all subtypes

11.10.8 Reservation of LCP subtypes and TLV tags for experimental use

Small ranges of short and long format values for LCP subtypes and TLV extension tags should be reserved for experimental use. The suggest range for short format values is 124 through 126 (3 values), inclusive. The suggested range for long format values is 65280 through 65534 (255 values). These ranges apply to both subtypes and tags. These values are reserved exclusively for development purposes, and shall not be including as part of an HNT-compliant implementation.

11.11 Reed-Solomon coding with intra-frame interleaving (Optional)

This clause describes the use of an optional Reed-Solomon code and intra-frame interleaving of bytes.

11.11.1 Embedded Reed-Solomon codewords

The Reed-Solomon codeword check bytes are embedded within the HNT packet with as a tag-length-value (TLV) encapsulating header; the original payload shall be unchanged and will follow the check bytes. This maintains backward compatibility with HNT nodes; HNT nodes that do not perform RS decoding can ignore the encapsulating header and recover the original payload (assuming no transmission errors).

TLV extensions allow Reed-Solomon coding and decoding to be implemented in a device driver, as long as the receiver hardware still sends packets that fail the FCS and CRC-16 checks up to the logical layer in the driver for possible error correction.

If the RS decoding is performed in a device driver above HNT demodulator, the demodulator SHOULD also pass the FCS and CRC-16 values to the RS decoder in order to verify that the correction was successful. If the recalculated FCS and CRC-16 fail after the payload has been corrected via RS, the receiver may wish to flag the packet as uncorrected and ask for retransmission.

11.11.2 Reed-Solomon symbol size

The symbol size shall be 8 bits, resulting in a code based on GF(256). This limits the maximum codeword size to 255; a HNT packet may contain several codewords. The primitive polynomial and generator polynomials are identical to those used for ITU-T Rec. G.992.1.

The arithmetic is performed in the Galois Field GF(256), where α is a primitive element that satisfies the primitive binary polynomial $x^8 + x^4 + x^3 + x^2 + 1$. A data byte (d_7, d_6, ..., d_1, d_0) is identified with the Galois Field element $d_7\,\alpha^7 + d_6\,\alpha^6 \ldots + d_1\,\alpha + d_0$.

11.11.3 Generator polynomial

$G(X) = \Pi\,(X + \alpha^i)$, is the generator polynomial of the Reed-Solomon code, where the index of the product runs from $i = 0$ to $R - 1$. X is a unit byte delay, and R is the number of check bytes per codeword.

11.11.4 Number of check bytes per codeword: Range of values

R can be one of: 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, or 20. Implementations do not need to encode or decode all of these allowed values of R; when stations announce the ability to perform RS encoding and decoding, they shall also announce the set of R values they support for encoding and decoding.

11.11.5 Interleaving

Because the length of HNT packets can range from 64 to 1522 and beyond, a packet can contain several codewords. These codewords could be transmitted sequentially inside a single packet, but a better solution is to interleave the codewords within the packet, giving added protection from bursty errors.

Interleaving shall be applied only within a single packet rather than span across multiple packets. The interleaver resets at the beginning of each packet.

The range of interleaving depths D shall be 1, 2, 4, 8, 16, 32, 64; the interleaver depths vary by a factor of 2. An interleaving depth of 64 allows a packet length of slightly over 16,000 bytes, although the specification limits the packet length to 1024 * N octets where N is the number of bits per symbol (for 2-Mbaud modulation).

The interleaving method is a simple write-by-columns, code-by-rows block interleave method. The transmission order of the original payload bytes is not affected; the interleaving is used conceptually for the calculation of the redundant bytes.

Interleaving example

An example of interleaving is shown in the next few paragraphs using a packet payload of 15 bytes, with $R = 2$ and $D = 4$.

Below is the original payload containing 15 bytes S_1 through S_{15}.

S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}	S_{13}	S_{14}	S_{15}

Below is a representation of the payload as a two-dimensional array; the number of rows is equal to the interleave depth *D*.

S_1	S_5	S_9	S_{13}
S_2	S_6	S_{10}	S_{14}
S_3	S_7	S_{11}	S_{15}
S_4	S_8	S_{12}	

There are now 4 (= *D*) codewords. Each RS codeword now reads across rows; the first codeword consists of bytes S_1, S_5, S_9, and S_{13}. Each codeword has at most 4 bytes, which equals `ceil` (15/4) or `ceil` (K/D) where *K* equals the payload length, and `ceil` (x) is the minimum integer larger than *x*; the last payload has 3 bytes because *K* is not an integral number of *D*.

Below, the Reed-Solomon check bytes are appended to each codeword. These check bytes are labelled $C_{\text{codeword-index,checkbyte-index}}$.

S_1	S_5	S_9	S_{13}	$C_{1,1}$	$C_{1,2}$
S_2	S_6	S_{10}	S_{14}	$C_{2,1}$	$C_{2,2}$
S_3	S_7	S_{11}	S_{15}	$C_{3,1}$	$C_{3,2}$
S_4	S_8	S_{12}	$C_{4,1}$	$C_{4,2}$	

After the check bytes are calculated, the packet can be transmitted. As described earlier, the check bytes shall be transmitted separately in an encapsulating header which will be described in detail later. Notice that the payload is transmitted in its original byte ordering.

Payload transmission order for this example:

S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}	S_{13}	S_{14}	S_{15}

Check-byte transmission order for this example is:

$C_{1,1}$	$C_{2,1}$	$C_{3,1}$	$C_{4,1}$	$C_{1,2}$	$C_{2,2}$	$C_{3,2}$	$C_{4,2}$

Check-byte transmission order:

In general, check bytes shall be transmitted in the following order: $C_{i,j}$ where i is the codeword index, j is the check-byte index, and the i (codeword) index varies quickest. If the number of check bytes (= $R \times D$) is not a multiple of 4, then two zero bytes shall be appended to the check bytes so that the payload, or next TLV extension, starts on a 4-byte boundary.

11.11.6 Indicating the redundancy parameters *R* and *D*

Packet lengths can vary on a packet by packet basis, and the value of D may also have to vary in order to ensure that any single codeword does not exceed the 255 symbol limit. Furthermore, it is desirable to give implementors a wide range of flexibility in determining the amount of redundancy to provide. The above two considerations motivate a mechanism to allow a transmitter to vary R and D on a packet-by-packet basis; in general a transmitter may vary R and D, with D chosen to limit the codeword length, and R chosen to provide the desired redundancy.

The mechanism must transmit the R and D parameters in a robust manner, as any error in R or D will render the entire packet uncorrectable. The need for robustness is complicated by the fact that the R and D parameters will often be transmitted at a payload rate which is higher due to the increased SNR of RS coding; therefore the R and D parameters themselves will be redundantly transmitted. To avoid forcing a minimum RS decoding capability in all transceivers, these parameters, along with the Tag, Length, and Payload Length Values, are simply repeated 3 times; receivers can vote on the three received sets.

Format of Reed-Solomon protocol header

The length of the encapsulating header must be a multiple of 4 octets, measured from the SSType field through the Next Ethertype field inclusively. The header consists of 3 copies of the SSType, SSLength, SSVersion, and SSParams, followed in turn by a set of check bytes, followed by a Next Ethertype field. The set of check bytes shall be zero-padded, if necessary, to ensure that the length of the header will be a multiple of 4 bytes.

SSVersion has two fields. One field shall be the version of RS encoder being used (0 for this version of this Recommendation) and another field shall be the length of the packet modulo 16. The length encoded here is the sum of all bytes starting at the Reed-Solomon SSType and ending at the end of the payload that Reed-Solomon encoding covers, which would be the entire G.9954v2 packet excluding the FCS and CRC-16.

RSParams

The RSParams octet has the format as shown in Table 11-40.

Table 11-40 – RSParams octet format

Bit	7 (MSB)	4	3	(LSB) 0
	R field		D field	

The R field has the encoding as shown in Table 11-41.

Table 11-41 – R field encoding

Bits 7 6 5 4	R value
0 0 0 0	0
0 0 0 1	2
0 0 1 0	4
0 0 1 1	6
0 1 0 0	8
0 1 0 1	10
0 1 1 0	12
0 1 1 1	14
1 0 0 0	16
1 0 0 1	18
1 0 1 0	20

The D field has the encoding as shown in Table 11-42.

Table 11-42 – D field encoding

Bits 3 2 1 0	D value
0 0 0 0	1
0 0 0 1	2
0 0 1 0	4
0 0 1 1	8
0 1 0 0	16
0 1 0 1	32
0 1 1 0	64

Table 11-43 – Format of TLV header, long form

Field	Length	Value/Meaning
DA	6 octets	Destination address (from original Ethernet PDU)
SA	6 octets	Source address (from original Ethernet PDU)
Ethertype	2 octets	0x886c
LSType	2 octets	SUBTYPE_RS = 32771. Reed-Solomon encapsulating header type (provisional)
LSLength	2 octets	Number of additional octets in the RS header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field
RSVersion	1 octet	= 0-15, overloading the version to encode the payload length modulo 16
RSParams	1 octet	RS Redundancy Parameters (4-bits each for D, R as already proposed) This field has 2 replicated bytes.
LSType2	2 octets	Duplicate of LSType
LSLength2	2 octets	Duplicate of LSLength
RSVersion2	1 octet	Duplicate of RSVersion
RSParams2	1 octet	Duplicate of RSParams
LSType3	2 octets	Duplicate of LSType
LSLength3	2 octets	Duplicate of LSLength
SSVersion3	1 octet	Duplicate of RSVersion
RSParams3	1 octet	Duplicate of RSParams
RSCheckBytes	D*R octets (possibly padded to a multiple of 4)	The array of computed check bytes. Order of transmission: ($C_{1.1}$, $C_{2.1}$ .. $C_{D.1}$, $C_{1.2}$, $C_{2.2}$ .. $C_{D.2}$, … $C_{1.R}$ ..$C_{D.R}$) Two additional zero bytes may follow, to pad to a multiple of 4 bytes. The RS payload coding starts with the next field, typically "Next_Ethertype".
Next Ethertype	2 octets	From "original" Ethernet PDU (could be 886c, with a LARQ header)
Payload	Min tbd octets	From original Ethernet PDU payload
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

11.11.7 Receiver indication of desired encoding. TLV Extension to the LCP SUBTYPE_RATE subtype

It is the receiver that monitors the packet error ratio, and is therefore the best qualified to guide the selection of Reed-Solomon redundancy. A mechanism to have a receiver indicate the desired redundancy to a remote transmitter is described below. It is a TLV extension of the current rate request control frame.

Three additional parameters are included for each band: one is an enhanced payload rate (Bandn_EPR) and the other two parameters indicate a minimum redundancy that will allow transmission at that enhanced payload rate. The Band*n*_EPR format is the same as that for non-RS encoded Band*n*_PE.

The minimum redundancy is specified by two octets. The first octet specifies a desired number of redundant bytes per RS codeword; the remote transmitter should encode all payloads with this

number of redundant bytes. As the number of redundant bytes is a multiple of 2, this field shall be encoded as R/2.

The second octet that specifies the desired redundancy is a maximum payload size per codeword. The remote transmitter should restrict the payload to never exceed this length, by increasing D if necessary.

Table 11-44 – Rate request control frame definition with Reed-Solomon extension

Field	Length	Value/Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c
SSType	1 octet	= 1
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. The minimum value of SSLength is 8 for SSVersion 0.
SSVersion	1 octet	= 0
OpCode	1 octet	Operation code for this control message. See Table 11-6 for definitions.
NumBands	1 octet	Number of bands specified in this control
NumAddr	1 octet	Number of addresses specified in the payload of this control message. NumAddr may be zero.
Band1_PE	1 octet	The PE value that should be used to send data when the 2 MBaud band is selected
Band1_rank	1 octet	The rank order of the ReqDAs' preference for this band
…		
BandN_PE	1 octet	Optional, only present if NumBands ≥2.
BandN_rank	1 octet	Optional, only present if NumBands ≥2.
RefAddr1	6 octets	Optional. Present if NumAddr ≥1.
RefAddr2	6 octets	Optional. Present if NumAddr ≥2.
● ● ●		[additional instances of RefAddr, until the number of RefAddr fields equals NumAddr]
SETag	1 octet	= RRCF_RS_TAG (2), Optional RS values for rate negotiation
SELength	1 octet	Total length of option excluding the tag and length octets, and pad. Must be 2 + 4 × Numbands, Minimum is 6.
Band1_EPR	1 octet	Enhanced payload rate to use when Reed-Solomon coding is used at the target redundancy specified in the next field
Band1_RSR	1 octet	Number of redundant bytes per codeword when Reed-Solomon coding is used
Band1_Kmax	1 octet	Maximum payload size per codeword
Band1_Pad [suggest Band1_Rdesired]	1 octet	For alignment and possible extensions. = 0 if unused. [Desired upper limit for total number of total redundant bytes, allowing reduced redundancy per code-word for longer frames.]
● ● ●		[additional instances of RS coding params, if Numbands ≥2]
Pad	2 octets	Pad to make encapsulating header a multiple of 4 octets
Next Ethertype	2 octets	= 0

Table 11-44 – Rate request control frame definition with Reed-Solomon extension

Field	Length	Value/Meaning
Pad		To reach minFrameSize if required
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

11.11.8 Capabilities announcement

The ability of a station to encode and decode packets shall be transmitted in the CSA_SUBTYPES tag extension to the CSA frame; via member fields. An example of this extension is in Table 11-45.

Table 11-45 – Example TLV extension for CSA, announcing RS capability

Field	Length	Value/Meaning
SETag	1 octet	CSA_SUBTYPES_TAG.
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. 2 × number of advertised subtypes (n).
Subtype	2 octets	SUBTYPE_RS_LONG (32771)
Additional subtypes	2 × (n – 2) octets	Additional optional subtypes supported by the implementation

In addition to announcing support for the Reed-Solomon subtype, the explicit capabilities of a station announcing RS capability are sent in a CSA parameters extension. This indication in a CSA extension for subtype parameters. The format of this extension follows in Table 11-46.

Table 11-46 – RS subtype parameters TLV extension for CSA

Field	Length	Value/Meaning
SETag	1 octet	CSA_PARAMS_TAG
SELength	1 octet	= 6
Subtype	2 octets	SUBTYPE_RS (32771)
Supported encoding R value bit mask	2 octets	*First octet*: Bit R 0 2 1 4 2 6 3 8 4 10 5 12 6 14 7 16 *Second octet:* Bit R 0 18 1 20 2 through 7: Reserved
Supported decoding R value bit mask	2 octets	*First octet:* Bit R 0 2 1 4 2 6 3 8 4 10 5 12 6 14 7 16 *Second octet:* Bit R 0 18 1 20 2 through 7: Reserved
Parameter data	1+ octets	Implementation-specific data

11.12 Frame bursting protocol

The frame bursting protocol is required. The purpose of the protocol is to reduce the overhead associated with the physical layer framing format by concatenating frames that share the same DA/SA value with equal or greater priority.

The frame format is in Table 11-47:

Table 11-47 – Frame burst format

Field	Length	Value/Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_FRAMEBURST (6)
SSLength	1 octet	6
SSVersion	1 octet	= 0
Priority/FlowID	1 octet	Priority or Flow_ID associated with packet. Interpretation of field is Flow_ID if bit 7:7 is set. Otherwise, interpretation of field is priority.
Packet length	2 octets	Length in octets of the first packet, from the first octet following the packet length field through the last octet of the data preceding the FCS
Next Ethertype	2 octets	Pre-encapsulation Ethertype the first packet being bursted
Data#1	variable	Pre-encapsulation payload data from the frame #1 being bursted
FCS#1	4 octets	Frame check sequence
CRC-16#1	2 octets	Additional frame check sequence (includes the LLC header)
Control Info#2	4 or 24 octets	Control information for second packet according to Table 11-48.
Next Ethertype#2	2 octets	Pre-encapsulation ethertype of second packet being bursted
Data#2	Variable	Pre-encapsulation payload data of second packet being bursted
FCS#2	4 octets	Frame check sequence
CRC-16#2	2 octets	Additional frame check sequence from end of previous CRC-16
…		More bursted packets
Control Info#N	4-24 octets	Control information for *N*th packet according to Table 11-48
Next Ethertype#N	2 octets	Pre-encapsulation Ethertype from the frame #N being bursted
Data#N	variable	Pre-encapsulation payload data from the frame #N being bursted
FCS#N	4 octets	Frame check sequence
CRC-16#N	2 octets	Additional frame check sequence
Burst termination trailer	4 octets	0xFFFF. Burst termination trailer, indicates the end of the burst
Pad		To reach minFrameSize if required

Table 11-48 – Control information

Field	Length	Value/Meaning
FT	4 bits	The FT of the original packet being bursted. Encoding of the FT is as defined immediately below:
EID	3 bits	Extended identifier
RSVD	1 bit	Reserved. This field shall be set to zero by the transmitter and the receiver shall discard frames with non-zero values.
ID	4 bits	Identifier
SI	4 bits	Scrambler index of the original packet being bursted
PE	8 bits	Payload encoding of the original packet being bursted
HCS	8 bits	Header check sequence of the original packet being bursted
DA	6 octets	Destination address of original packet being bursted
SA	6 octets	Source address of original packet being bursted
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_FRAMEBURST (6)
SSLength	1 octet	6
SSVersion	1 octet	= 0
FLH pad	1 octet	Reserved; must be sent as 0 and ignored by receiver
Packet length	2 octets	The original packet length being bursted

The maximum length shall not exceed the maximum allowed time on the wire. The maximum size of the bursted frame shall be negotiated in CSA messages as described in clause 11.10.6. All frames in a burst shall have the same DA/SA value. When a transmitter is constructing a bursted frame, the priorities of each bursted subframe in an unmanaged network must be equal to or greater than the priority of the first frame. If the priority of a frame is less than the priority of the first subframe of a burst frame, it must not be concatenated into a burst frame; it shall start a new physical layer frame. In a managed network there shall be no limit on the bursted flows between the same DA/SA values.

The burst termination trailer shall be used to indicate the end of the burst.

11.13 MAC cycle synchronization

MAC cycle synchronization shall be performed using the master-generated media access plan (MAP). The MAP indicates the beginning of the MAC cycle and contains the media access plan for the following MAC cycle.

All G.9954v2 stations shall implement the MAC cycle synchronization function in order to implement the G.9954v2 media access method in a master-controlled network.

11.13.1 MAP control frame

In Table 11-49, horizontal shading is used to show the decomposition of a field (TXOP) into sub-fields. The decomposition of the sub-field (TXOPID) into bit-fields is shown by the transition from (horizontal) shading to clear (non-shaded) fields.

Table 11-49 – Map control frame

Field	Length	Value/Meaning
DA	6 octets	Destination address = 0xFF:FF:FF:FF:FF:FF
SA	6 octets	Source address of master device
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_MAP (32772)
LSLength	2 octets	Number of additional octets in the control header, starting with the LSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum LSLength is 22 for LSVersion 0.
LSVersion	1 octet	= 0
LSPad	1 octet	Ignored on reception
MAPHeader	12 octets	MAP header as described in Table 11-50
TXOP1	4 octets	Transmission opportunity described by the sub-fields immediately below and defined in Table 8-2
Reserved	1 bit TXOP[31:31]	Reserved for future use
Length	15 bits TXOP[30:16]	Length allocated to associated TXOP in TIME_SLOT units where the size of a TIME_SLOT is determined by a base TICK size multiplied by a constant factor defined in the MAP. A length of zero indicates a sub-burst slot transmission.
DeviceID	6 bits TXOP[15:10]	Device associated with transmission opportunity. DeviceID = 0 indicates a special MAP control directive. DeviceID > 0 identifies the associated device by DEVICE_ID.
GroupType	3 bits TXOP[9:7]	Group Type identifier Identifies the collection of devices/flows associated with the same group as well as any implicit scheduling policy that applies to the group
FSelector	1 bit TXOP[6:6]	Field selector used to determine the interpretation of the following fields: 0 – Priority interpretation (Reserved + Priority Field) 1 – FlowID interpretation (FlowID)
Reserved	3 bits TXOP[5:3]	Reserved for future use
Priority	3 bits TXOP[2:0]	Priority associated with transmission opportunity Defined in range 0-7 with priority 7 being the highest priority
FlowID	6 bits TXOP[5:0]	Identifies the flow associated with a transmission opportunity
…		Additional TXOPs
TXOP*N*	4 octets	
Next Ethertype	2 octets	= 0
Pad		Pad to reach minFrameSize if necessary
FCS	4 octets	Frame check sequence
CRC-16	2 octets	HNT frame check sequence

In Table 11-50, horizontal shading is used to show the decomposition of the control field into bit-fields.

Table 11-50 – MAP control header

Field name	Bit number	Field size [bits]	Description
Control Field1		32	Set of control fields used to control the behaviour of endpoint nodes. The encoding of this field is described immediately below:
Modified	31:31	1	Indicates that the TXOP table defined in this MAP is different from the TXOP table defined in the "previous" MAP where "previous" is defined as the MAP sent in the "previous" MAC cycle with *Sequence Number* one less than the "current" *Sequence Number (*accounting for modulo arithmetic). 0 – MAP is the same as "previous" cycle 1 – MAP changed since "previous" cycle This flag may be used by an endpoint for local optimization.
Reserved	30:27	4	Reserved for legacy systems.
SMAC_EXIT	26:26	1	Exit from synchronous MAC mode. The master shall subsequently cease sending MAPs. This flag is used as an indication for G.9954v2 devices to start the Automatic Master Selection.
Reserved	25:22	4	Reserved for legacy systems.
MAP_IFG_INCR	21:16	6	Increment added to CS_IFG (29 µs) in order to determine the size of MAP_IFG (inter-frame gap) planned between TXOPs by the master. MAP_IFG is defined by the relation: $MAP\_IFG = CS\_IFG + MAP\_IFG\_INCR$ MAP_IFG silence shall be guaranteed by each endpoint at the end of its TXOP. MAP_IFG_OFFSET is measured in 500-ns units.
Reserved	15:0	16	Reserved for future use. Shall be sent as 0 and ignored by the receiver.
ControlField2		32	Set of control fields used to control the behaviour of endpoint nodes. The encoding of this field is described immediately below:
Reserved	31:12	20	Reserved for future use
SyncDataCollection	11:11	1	Data sampling synchronization bit. When SyncDataCollection is set, devices shall schedule the collection of data at the start of the next MAC cycle immediately following the reception of the next MAP. The actual data collected is implementation dependent.
Reserved	10:7	4	Reserved for future use
TimebaseMultiplier	6:4	3	A constant shift factor applied to the TICK in order to determine the resolution of a TXOP TIME_SLOT. The resolution of a TIME_SLOT is determined as follows: TIME_SLOT = TICK << TimeBaseMultiplier
Reserved	3:0	4	Reserved for future use

Table 11-50 – MAP control header

Field name	Bit number	Field size [bits]	Description
SequenceNumber		16	MAP sequence number. Modulo counter that is incremented each MAC cycle.
Seq0	15:8	8	Modulo counter incremented each MAC cycle
Seq1	7:0	8	Modulo counter that is incremented each wrap around of Seq0 counter
NumTXOPs		16	Number of entries in allocation map. The minimum number of entries in a MAP is normally 2 (one entry for the MAP and the second entry for the UNALLOCATED TXOP). When the SMAC_EXIT flag is set, the number of entries in the MAP may be zero. The maximum number of entries is limited by the maximum size of the MAP control frame as described above.
BaseNumEnt	15:8	8	
ExtendNumEnt	7:0	8	Number of entries in allocation map is BaseNumEnt plus ExtendNumEnt multiplied by 256

11.13.2 Terms and parameters

11.13.2.1 Timers

SYNC_Timer is a free running timer with a period of 3 times the MAC cycle period in milliseconds.

This timer is used to detect loss of synchronization with the master-generated MAC cycle. The timer is activated upon entry to the SYNC state and cancelled upon leaving the SYNC state.

11.13.3 MAC cycle synchronization protocol

11.13.3.1 Receive MAP control frame

If the G.9954v2 device is currently in the NOSYNC state, the periodic SYNC_Timer should be armed and the system state changed to SYNC mode (see Figure 8-1).

If the G.9954v2 device is already in the SYNC state, the SYNC_Timer should be re-armed to count a new SYNC timeout period.

Control information communicated in the MAP should be used to update system state variables used by the MAC processor.

11.13.3.2 SYNC_Timer timeout

When a SYNC_Timer timeout occurs, this indicates that a MAP was not received for the SYNC_Timer period and that SYNC_LOSS has occurred.

The current state of the device should be changed to NOSYNC state, and system state variables updated.

11.14 Network admission control (Registration) protocol

In a master-controlled network, a G.9954v2 device is required to perform the following procedures in order to enter the network:

- Synchronization – Wait for periodic MAP transmissions from the master.
- Registration – Locate transmission opportunities in the MAP for the transmission of registration protocol messages and perform registration with the master.

The synchronization procedure involves waiting for the reception of a periodic MAP transmission from the currently assigned master. Once a MAP is received, a G.9954v2 device that wishes to join the network is able to locate available transmission opportunities and proceed with the registration procedure.

The registration procedure consists of a request-response sequence of transactions between the master and the registering device. The registration procedure is used to authenticate a device for network entry, to assign it a unique device identifier and to download network configuration information.

11.14.1 Registration opportunities

Once a device is synchronized with the MAC cycle, the device is required to locate transmission opportunities that will allow it to bootstrap the registration process. Such transmission opportunities are identified in the MAP by an unallocated TXOP (UTXOP) also known as a REGISTRATION TXOP.

The master guarantees to allocate a REGISTRATION TXOP at least once every REG_SLOT_PERIOD. The REGISTRATION transmission opportunity is used to advertise an intention to register. This intention is expressed by sending a REG_REQUEST message to the master.

Devices contend for access to the REGISTRATION transmit opportunities.

11.14.2 Registration and authorization control

Registration is the process performed to allow a G.9954v2 device to request media access bandwidth. Only after a device has registered with the master can it reserve bandwidth through explicit flow set-up requests with the master.

The registration procedure involves a request-response sequence, whereby a G.9954v2 device requests to be registered with the master by sending a REG_REQUEST message containing the device's MAC address as well as other identifying characteristics, such as authentication key and a set of capability parameters. Upon receiving a REG_REQUEST message, the master is responsible for authorizing the entry of the requesting device and, if authorization is successful, for allocating resources to the registered device.

Authorization is performed by checking that the device, identified by its MAC address and possibly other identifying information (e.g., authentication key), is valid and the device is authorized to join the home network controlled by the master. The details of the authorization procedure are implementation dependent.

Once a device is admitted to the network, it is assigned a unique Device ID. This Device ID is subsequently used as part of the addressing scheme used to allocate transmission opportunities to devices and flows in the media access plan.

The master responds to a REG_REQUEST with a REG_RESPONSE. The response contains a status flag that indicates whether the registration procedure was successful or not. If the procedure is successful, the master downloads network configuration data to the registering device.

If a REGISTRATION_RESPONSE message is not received from the master within the time interval REG_TIMEOUT (T0) period, the registering device should retry after backing off a random amount of time using the RetransmitTimer (see 11.14.7). If the registering device fails to receive a response after MAX_RETRIES, the device should be reinitialized and the sequence restarted.

The network admission protocol is illustrated in the sequence diagram in Figure 11-7.

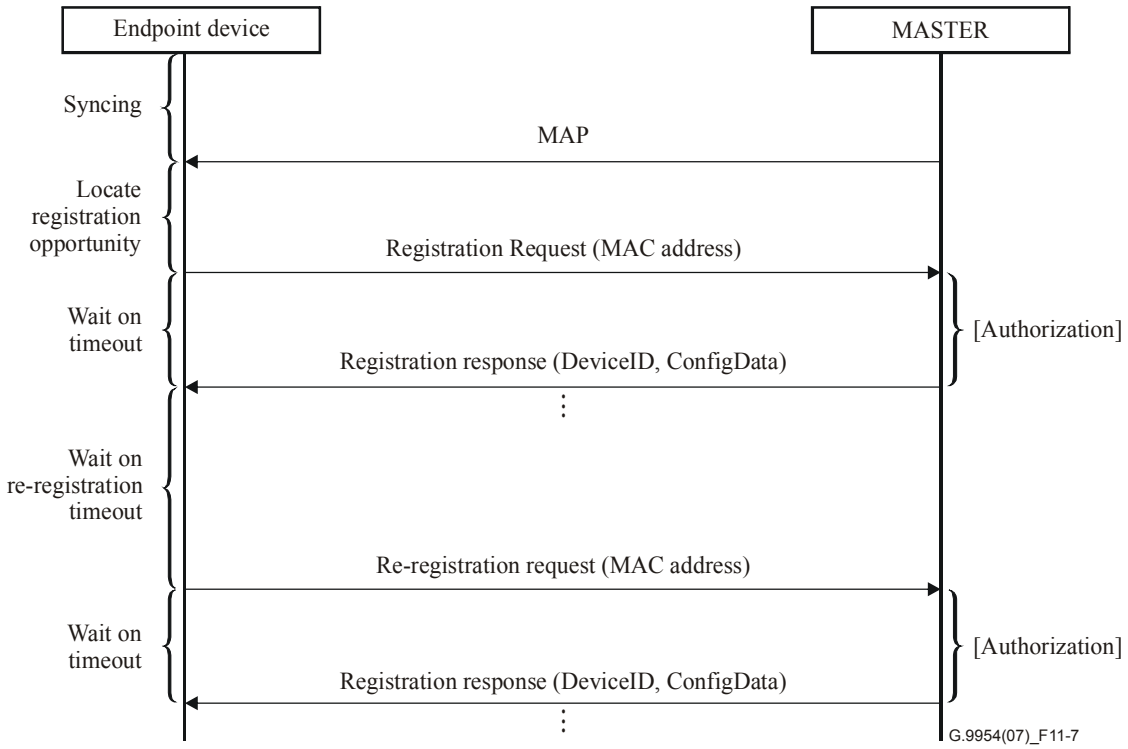

Figure 11-7 – Network admission protocol sequence diagram

11.14.3 Registration state machine

The following state diagram in Figure 11-8 gives a pictorial view of the state transitions in the registration process from the perspective of an endpoint device.

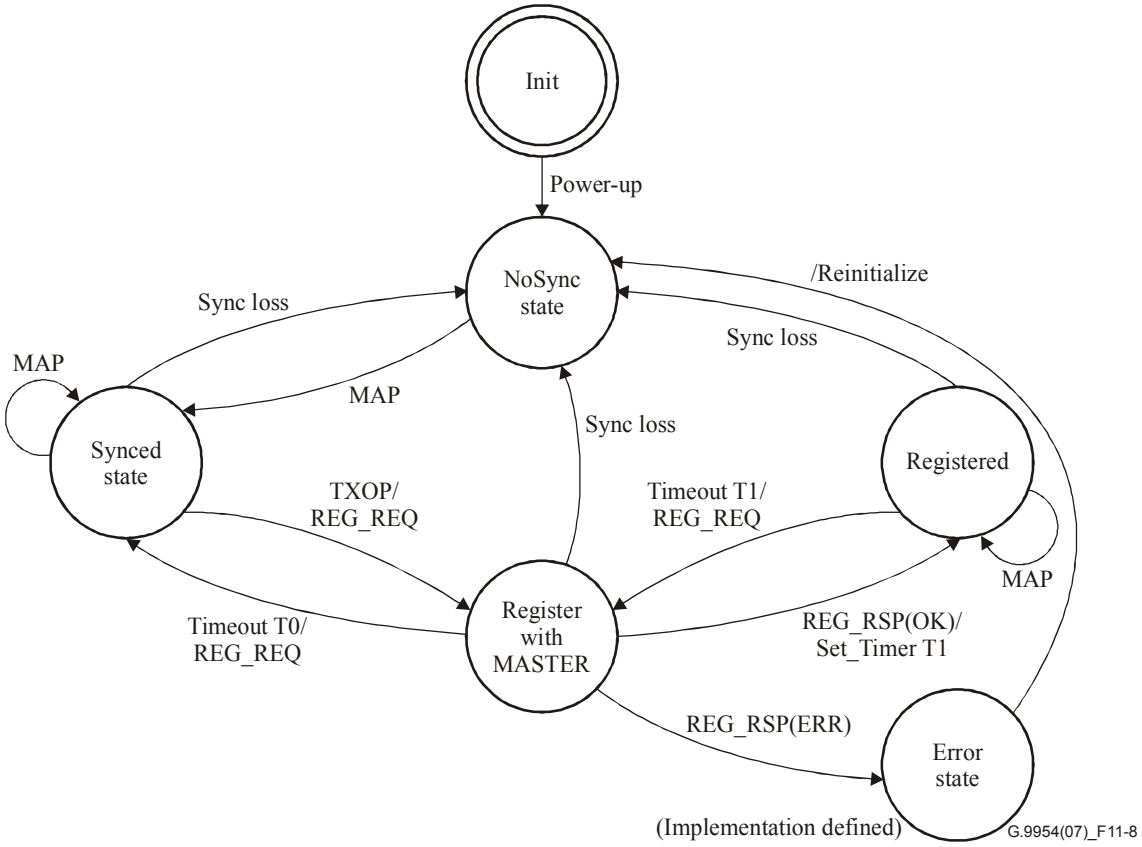

Figure 11-8 – Registration at endpoint device

The following SDL (specification and description language) diagrams (Figures 11-9 and 11-10) provide a complete description of the behaviour of endpoint and master devices during the registration protocol.

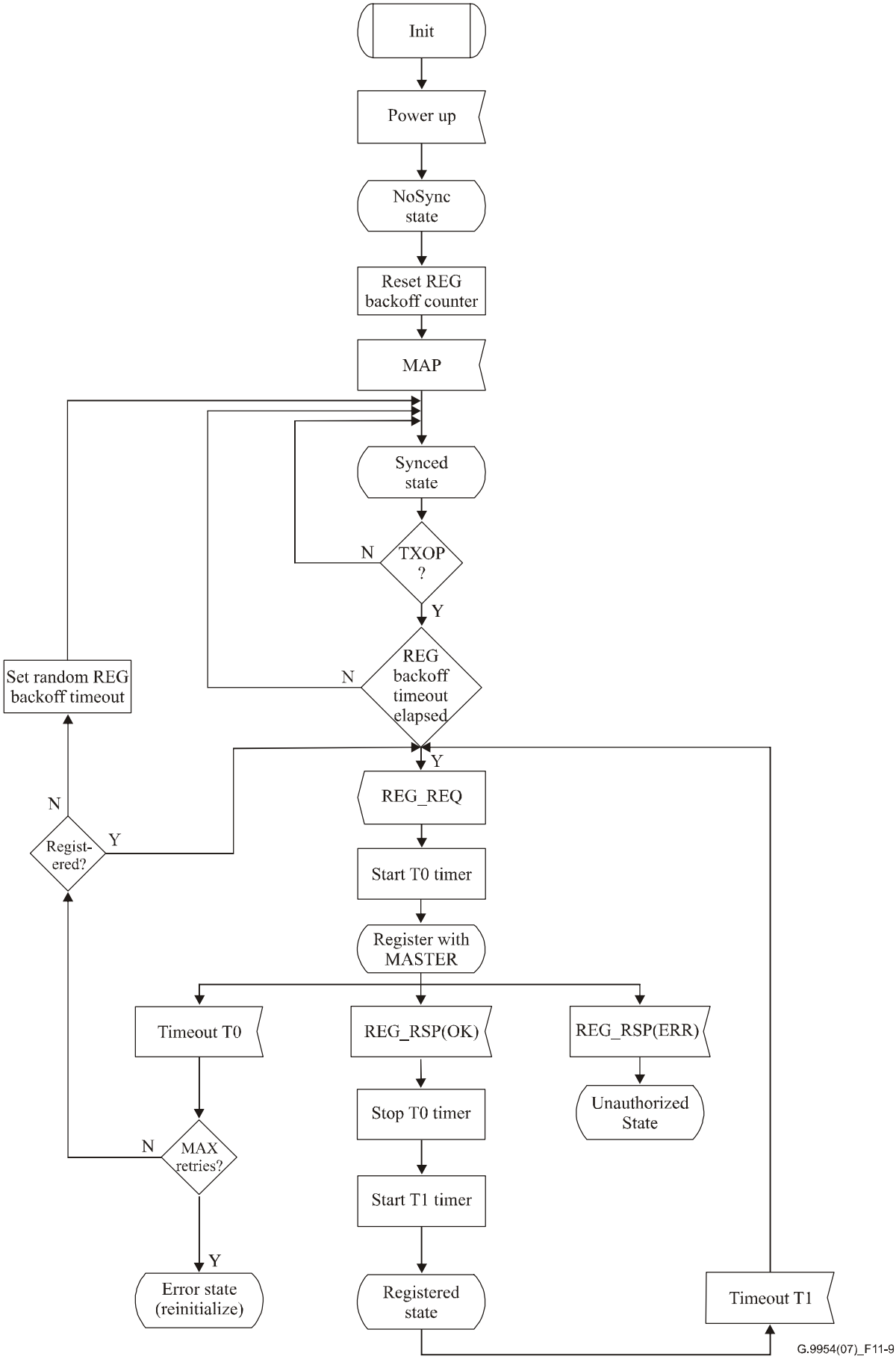

Figure 11-9 – End point registration sequence

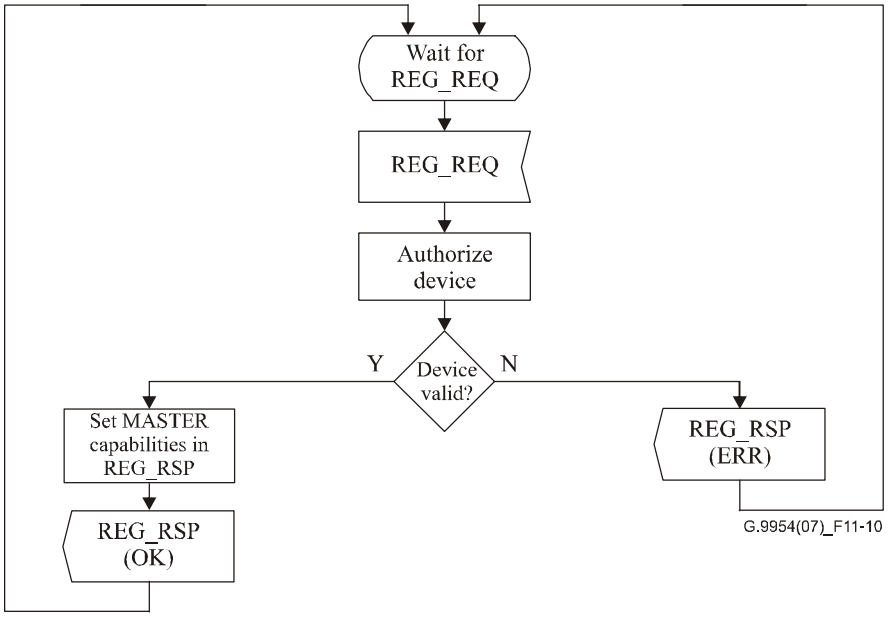

Figure 11-10 – Master registration sequence

11.14.4 Ageing-out registered devices

The master shall maintain an AgeingTimer and at the end of each AgeingTimer period, shall check that a CSA frame was received for each registered device. If a CSA frame was not received for a registered device within the AgeingTimer period, the device shall be de-registered and any associated resources removed.

For a definition of the AgeingTimer, see clause 11.14.7.1.

11.14.5 Periodic re-registrations

A G.9954v2 endpoint shall periodically re-register with the G.9954v2 master. Re-registration is performed each T1 period after receiving the last REG_RESPONSE message. Re-registration uses the same protocol sequence as registration except that the G.9954v2 endpoint need not transmit the registration request in the REGISTRATION opportunity, since it may be allocated sub-burst slots. In addition, the G.9954v2 endpoint already has an assigned Device_ID and this shall be communicated in the REG_REQUEST message.

11.14.6 Frame formats

Registration control frames should be sent using 2-Mbaud, 2-bits-per-symbol payload encoding (PE = 33). The format of registration control frames are described in Tables 11-51 and 11-53.

Table 11-51 – Registration request message

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address of device requesting registration
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_REGISTRATION (32773)
LSLength	2 octets	Number of additional octets in the control header, starting with the LSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum LSLength is 4 for LSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for Registration Request (0)
Registration data	0-65531 octets	Registration information sent by the device to the master includes device capabilities, identification information, etc. Registration data is optional and TLV encoded.
Next Ethertype	2 octets	= 0
Pad		Pad to reach minFrameSize if necessary
FCS	4 octets	Frame check sequence

A device generating a Registration message may include the parameters in Table 11-52 in the registration data.

Table 11-52 – Registration parameters

Field	Length	Meaning
SETag	1 octet	= 2, Device identity
SELength	1 octet	Total length of TLV extension excluding the tag and length octets (84 octets)
Primary_ID	4 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Subsystem_ID	4 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Vend_Date	4 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Manuf_Date	4 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Serial_Num	16 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Vend_Name	32 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Vend_Driver	16 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
OUI	3 octets	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Device_Type	1 octet	See CERT and DIAG Protocol, clause 11.9.6, Table 11-27
Vendor-specific	1+ octets	Vendor-specific TLV encoded extension
SETag	1 octet	= 3, Device capabilities
SELength	1 octet	Total length of TLV extension excluding the tag and length octets (3 octets)
Reserved	4 octets	Reserved for legacy systems
Capabilities	4 octets	Set of device capabilities. Described by the following sub-fields:
Reserved	8 bits	Reserved for future use

Table 11-52 – Registration parameters

Field	Length	Meaning
QoS_Support_Level	2 bits	Level of QoS support provided by G.9954v2 device as described in clause 10.9 0 – G.9954v1 QoS: 1 – Best-effort 2 – Priority-based 3 – Full QoS
Spectral mode	2 bits	The spectral mode supported by the G.9954v2 device (i.e., when Highest Version is G.9954v2). 0 – Spectral Mode A 1 – Spectral Mode B 2 – Spectral Mode C 3 – Spectral Mode D
Supports frame bursting	1 bit	This station supports frame bursting
Smallest sub-burst slot supported	2 bits	Size (duration) of smallest sub-burst slot supported. 0 – 8 microseconds 1 – 16 microseconds 2 – 32 microseconds 3 – 64 microseconds
Supports high-constellation encoding	1 bit	Supports high-round-constellation encodings of 8, 9 and 10 bits per symbol
Frame burst packet limit	3 bits	0 – No limit (actually limited by maximum link-level frame size in the highest PE) 1 – This station supports bursts of up to 16 frames. 2 – This station supports bursts of up to 32 frames. 3 – This station supports bursts of up to 64 frames. 4 – This station supports bursts of up to 128 frames. 5 – This station supports bursts of up to 256 frames.
Frame burst size limit	3 bits	0 – No limit (actually limited by maximum link-level frame size in the highest PE) 1 – This station supports bursts of up to 8 kbytes. 2 – This station supports bursts of up to 16 kbytes. 3 – This station supports bursts of up to 32 kbytes. 4 – This station supports bursts of up to 64 kbytes. 5 – This station supports bursts of up to 80 kbytes. For the purpose of burst size limitations, a burst consists of all the link-layer frames (i.e., all the frames excluding the physical layer preamble, frame-control, pad and EOF). For further information on frame bursting and aggregation, see clause 11.2.
Reserved	7 bits	Reserved for future use

Table 11-52 – Registration parameters

Field	Length	Meaning
Highest version	3 bits	This station's highest supported HNT version: 0x000 – Reserved 0x001 – Reserved for legacy usage 0x010 – Reserved for legacy usage 0x011 – G.9954v1 0x100 – G.9954v2 0x101-0x111 – Reserved for future use
Vendor-specific	1+ octets	Vendor-specific TLV-encoded extension

The Registration Response message shall be sent by the master to a device in response to a registration request. See Table 11-53.

Table 11-53 – Registration response message

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_REGISTRATION (32772)
LSLength	2 octets	Number of additional octets in the control header, starting with the LSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum LSLength is 6 for LSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for Registration Response (1)
DeviceID	1 octet	Device ID assigned to device by master
Status	1 octet	Status of registration request. 0 – OK. Device registered 1 – Error
Configuration data	0-65530 octets	Network configuration information returned by the master upon successful registration of device. This information is optional and TLV encoded.
Next Ethertype	2 octets	= 0
Pad		Pad to reach minFrameSize if necessary
FCS	4 octets	Frame check sequence

The master responds to a Registration Request with a Registration Response. The following information shall be returned in the Registration Response:

Status

A status return code indicating the success or failure of the registration request.

Device ID

A device identifier assigned by the master to the device with the specified MAC address.

Configuration data

Network configuration data is optional and may be vendor-specific. It may be used to communicate:

- network-wide configuration parameters;
- master capabilities;
- security information;
- service provisioning information.

Table 11-54 describes the values that may appear in the MsgType entry in the Registration Control Frame.

Table 11-54 – MsgType values

MsgType	Meaning
0	Registration Request
1	Registration Response
2-255	Reserved

11.14.7 Terms and parameters

- REG_PERIOD – The maximum amount of time between TXOPs that can be used for sending Registration Requests. The value of REG_PERIOD is 50 milliseconds.
- MAX_RETRIES – Number of times an endpoint should retry registration with the master before re-initializing the device. The value of MAX_RETRIES is 5.

11.14.7.1 Timers

- T0 – A one-shot timer set after the transmission of a REG_REQUEST message. Used to time out the expected REG_RESPONSE from the master before retrying the request. This timer is cancelled if a REG_RESPONSE is received. The value of the timer is 150 milliseconds.
- T1 – A one-shot timer set after the reception of a REG_RESPONSE message. Used to schedule the transmission of a re-registration request. The value of this timer is 10 seconds.
- RetransmitTimer – A one-shot timer, set to a random interval in the range 1 ms to 1000 ms, inclusive. Used to set the backoff time before resending a REG_REQUEST in case of a collision during the transmission within a REGISTRATION TXOP.
- AgeingTimer – Periodic timer with a period of 180 seconds used to determine which registered devices have been actively sending CSA frames.

11.15 Master selection protocol

A G.9954v2 network requires the existence of a network node that takes the role of master in order to coordinate and schedule media transmissions. Although a master is required for an operational G.9954v2 network, not all network nodes necessarily have the functionality to become a master. Amongst those that *do* have the required capabilities, any one of them can potentially become master.

A home network that contains more than one network node that is capable of becoming the master allows for quick recovery from master failure and is inherently more fault/failure tolerant. A master selection protocol shall be used to dynamically select a single master in the presence of multiple potential masters.

The protocol used for discovering and selecting a single master, known as the master Selection Protocol, is described in the following clause.

11.15.1 Detection of a managed network

Following power-up, a G.9954v2 device (configured for G.9954v2 mode) first tries to detect whether it is operating in a master-controlled network, by listening for MAP control frames and synchronizing with the MAC cycle. If no MAP frames are detected after master_DETECTION_TIMEOUT (T0) interval, the device can conclude that there is no master currently on the network. If the device is master-capable and is willing to become the master, it is able to offer up its candidacy as the network master. If a MAP control frame is received, the device shall synchronize with the advertised MAC cycle and proceed as a regular endpoint device.

11.15.2 The master selection procedure

If the network is determined to be unmanaged and a device is capable of and willing to become a master, it can offer up its candidacy by broadcasting a master_SELECTION Control Frame using the asynchronous transmission mode of G.9954v2. Since several master capable devices may be active on the network at the same time, the master selection procedure includes the mechanism to allow other potential masters to compete for selection as the network master.

Master selection shall be performed according to relative master priority. Each master-capable device shall be assigned a priority using configuration or management parameters. This priority together with the device's MAC address shall be advertised in the master_SELECTION Control Frame. Upon receiving a master_SELECTION Control Frame, G.9954v2 nodes that are capable themselves of becoming master, may compare the priority of the potential master candidate with its own assigned priority in order to determine whether it is a "better" candidate. If it is a "better" candidate and it wishes to compete for the role of master, it must broadcast a master_SELECTION Control Frame within master_SELECTION_TIMEOUT (T1) interval.

If an alternative "better" candidacy is not offered within master_SELECTION_TIMEOUT (T1) interval, the master candidate assumes the role of master and may commence the transmission of MAP control frames. If an alternate candidacy is offered, the master with the highest priority shall be assumed to be the master. If there are several candidates with the same priority, the device with the lowest MAC address shall be selected as the master.

All stations that have given up the chance of becoming master should be silent until the selected master's MAP control frame is received.

11.15.3 Detection of master failure and recovery

A master is determined to have "failed" if synchronization with the master is lost. Synchronization is lost when a MAP control frame is not received within master_DETECTION_TIMEOUT (T0) interval following the last MAP control frame. Upon detection of master failure, the master may perform an orderly shutdown by inviting a master selection process by sending a master_SELECTION Control Frame with a declared priority of zero.

11.15.4 Master selection state machine

Figure 11-11 gives a pictorial view of the state transitions with some minor loss of detail, including omission of events that do not cause state transitions (and have no associated actions), decision logic within a state that leads to the raising of an event and the representation of complex conditions as a high-level "logical" event.

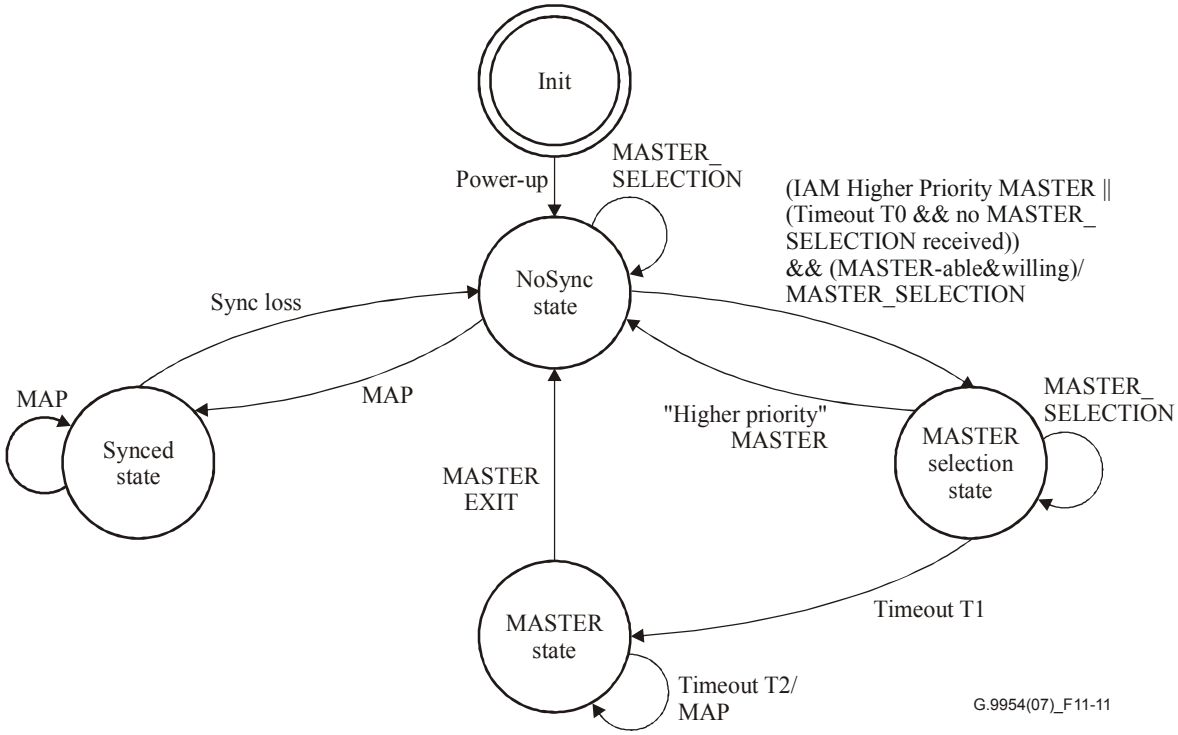

Figure 11-11 – Master-selection state diagram

Figures 11-12 and 11-13 provide a complete description of the master-selection protocol.

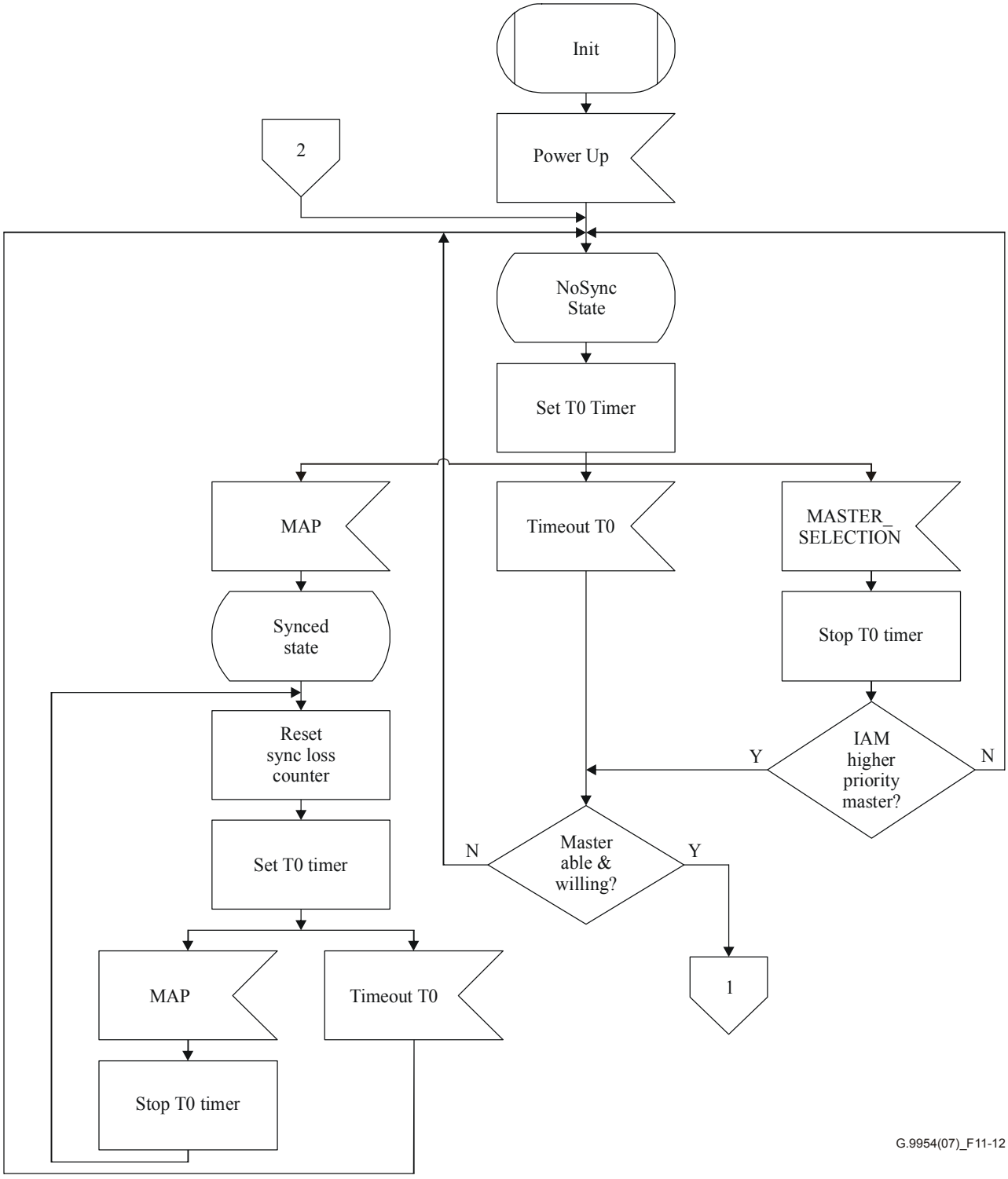

Figure 11-12 – SDL for master selection protocol

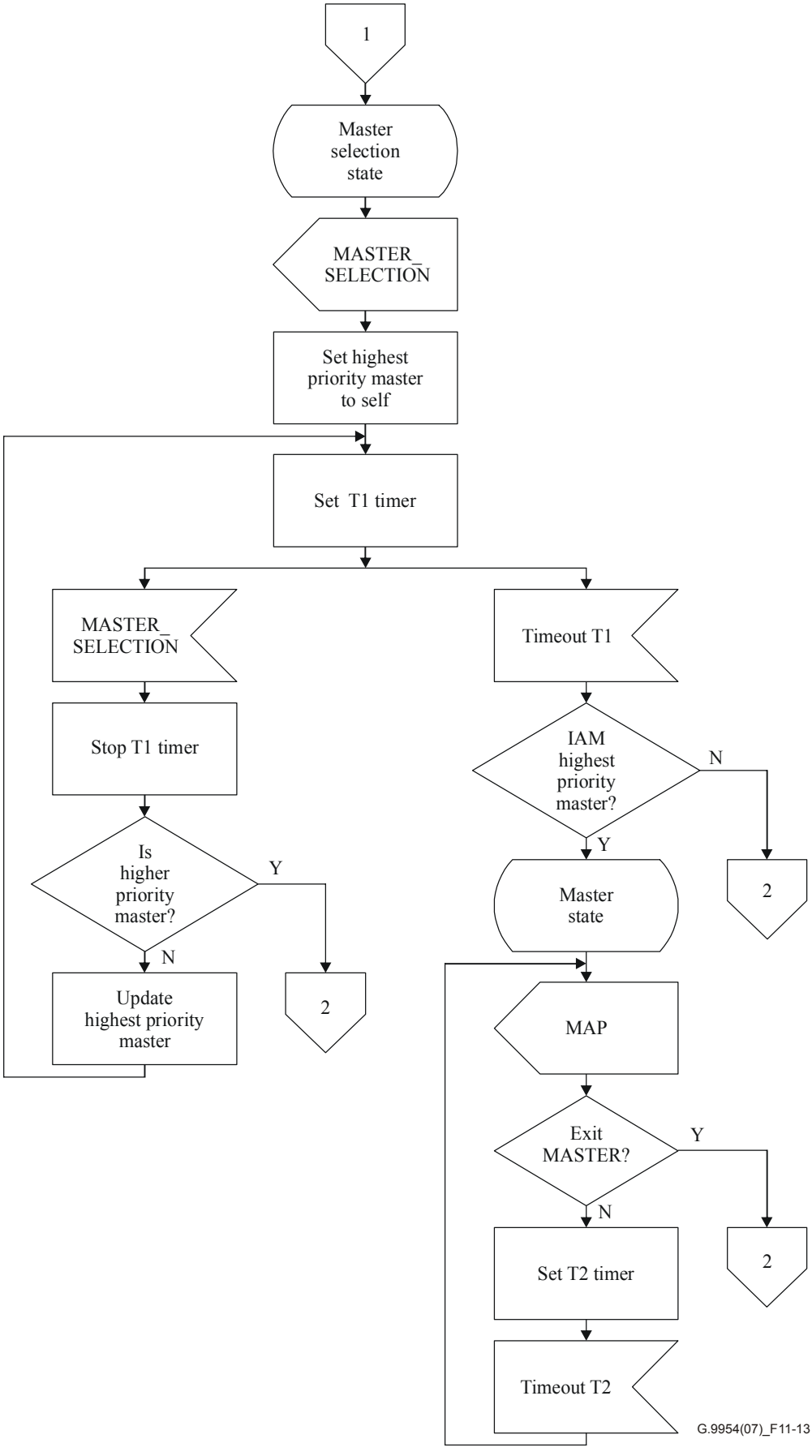

Figure 11-13 – SDL for master selection protocol (*cont.*)

11.15.5 Master selection protocol messages

See Table 11-55.

Table 11-55 – Master selection control frame

Field	Length	Meaning
DA	6 octets	Destination address (broadcast or multicast address)
SA	6 octets	Source address of the device requesting to become master
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_master_SELECTION (7)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength is 4 for SSVersion 0
SSVersion	1 octet	= 0
Priority	1 octet	The assigned master priority. Used to rank potential MASTERs into an order that supports priority selection. Priority values are from 0-255 with high numbers representing high priorities. Priority 0 is reserved and can be used by a master to broadcast a desire relinquish master control.
Next Ethertype	2 octets	= 0
Pad	40 octets	
FCS	4 octets	

11.15.6 Terms and parameters

11.15.6.1 Timers

- T0 – A one-shot timer, set to the value 150 ms, and used to detect the absence of a master on the network. The timer is set by a master-capable device upon entry to the *unsynchronized* state. A device is in *unsynchronized* state when it first wakes up and after MAC cycle SYNC_LOSS. The timer is cancelled upon the arrival of a MAP control frame. (See clause 11.13). If the timer T0 expires, a master-capable device may advertise its intention to become master.
- T1 – A one-shot timer set after the transmission or reception of a master SELECTION protocol message. This timer is used to open up a period of time for negotiation between master-capable devices for the role of master. After the T1 timer expires, a master-capable device can decide whether it is the selected master based on its priority and MAC address. The timer T1 is re-armed upon the arrival of a master-selection control frame. The value of T1 timer is 50 ms.
- T2 – A one-shot timer set by the master to measure the length of the MAC cycle. The value of T2 is variable and dependent upon the scheduler. When the T2 timer expires, the MAP for the next MAC cycle is sent.

11.16 Flow signalling protocol

The flow signalling protocol is used to dynamically establish and manage service flows with QoS parameters and traffic classification filters defined by upper-layer protocols. More specifically, the flow signalling protocol is used to perform the following flow-related functions:

- Set up a flow and traffic classification filters;
- Modify flow parameters and add or remove classification filters;
- Tear down flows;
- Query QoS parameters for a flow or Class-of-Service.

The flow signalling protocol shall be performed between G.9954v2 devices at the source and destination of a flow and will be used to establish QoS parameters for the flow. In a master-controlled network, flow signalling shall also be performed between the G.9954v2 device at the source of the flow and the master, if reserved bandwidth is required. The flow signalling protocol may be initiated by either source or destination devices involved in a unicast flow, or by the source device in a broadcast/multicast flow or by the master.

The flow signalling protocol, in general, involves a 3-way handshake. The handshake allows for negotiation of flow parameters between flow source and destination devices and between flow source and master devices.

Flow signalling with the master is used to reserve media bandwidth to a flow in order to contract QoS throughput, latency/jitter and BER parameters. The master shall be responsible for performing admission control on flow set-up requests in order to validate that sufficient media resources exist and the QoS specified by the flow parameters can be met. If the flow is admitted by the master, media transmission opportunities (TXOPs) shall be allocated in the media access plan for the exclusive use of the flow.

The destination of a flow may be a single device identified by a unicast destination address or it may be a group of devices, identified by a broadcast or multicast address. The flow signalling protocol for a group of devices does not require a 3-way handshake in the same manner as a unicast flow set-up. Rather, flow parameters are broadcast to the group and no response is required. Group members are always able to initiate an explicit request for flow parameters (from the flow source) for a flow to which they are actively listening.

The remainder of this clause describes the details of the flow signalling protocol and the flow signalling control frame formats.

11.16.1 Flow signalling control frames

The SETUP/MODIFY_FLOW_REQUEST control frame (see Table 11-56) is used to request the set-up or modification of a flow. The flow being set up or modified is identified using the { FS_SA, FS_DA, FS_FlowID } tuple. The Flow Set-up request is used to set up a flow with a defined set of QoS flow parameters. A Flow Modification request is used to modify a QoS flow parameter for an existing flow. In both cases, flow parameters are always defined for Set-up and Modify requests and appear in one of either of two forms as described in clause 11.16.1.1. Optionally, flow classifiers may be installed at a flow source using the *FlowClassifier* TLV structure (see clause 11.16.1.2).

Table 11-56 – Set-up/Modify flow request control frame

Field	Length	Meaning
DA	6 octets	Destination address. FS_DA or address of master
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_FLOW_SIGNALLING (32774)
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum SSLength is 58 for SSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for SETUP/MODIFY_FLOW_REQUEST (0,3) as defined in Table 11-63
Request_Key	2 octets	Unique request key used to correlate response/confirm protocol messages
FS_SA	6 octets	MAC address of station at source of flow. Does not necessarily correspond to SA
FS_DA	6 octets	MAC address of station at destination of flow. Does not necessarily correspond to DA
FS_FlowID	1 octet	Unique identifier of the flow between the flow source (FS_SA) and flow destination (FS_DA). The flow identifier is assigned locally by the device at the flow source. If the flow set-up request is not initiated by the flow source, the flow identifier shall be specified as NULL.
FS_DeviceID	1 octet	Device ID identifying the device requesting the flow set-up or modification. The Device ID is that assigned by the master during the registration process
FlowParameters	50 octets	QoS properties of flow to be set up. Flow properties are described by a TLV encoded structure as defined in Table 11-64.
FlowClassifiers	*N* octets	Specification of flow classifiers used to identify a packet belonging to flow. Flow classifiers are optional and described by a TLV encoded structure as defined in Table 11-67. More than one flow classifier may be defined.
Next Ethertype	2 octets	= 0
Pad	Variable	
FCS	4 octets	Frame check sequence

The SETUP/MODIFY_FLOW_RESPONSE control frame (see Table 11-57) shall be returned in response to a SETUP/MODIFY_FLOW_REQUEST. The response is associated with the corresponding request using the unique *Request Key* assigned by the requestor. The response contains a status indicating whether the request was successful and, in case the requested flow parameters need to be negotiated or modified from their requested values, the modified parameters are returned in a *Flow Parameters* TLV structure.

Table 11-57 – Set-up/Modify flow response control frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_FLOW_SIGNALLING (32774)
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum SSLength is 60 for SSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for SETUP/MODIFY_FLOW_RESPONSE (1,4) as defined in Table 11-63
Request_Key	2 octets	Key used to identify the request associated with the response
FS_SA	6 octets	MAC address of station at source of flow. Does not necessarily correspond to SA
FS_DA	6 octets	MAC address of station at destination of flow. Does not necessarily correspond to DA
FS_FlowID	1 octet	Unique identifier of the flow between the flow source (FS_SA) and flow destination (FS_DA). If the flow set-up request is not initiated by the flow source, the flow identifier shall be returned in the flow set-up response.
Status	1 octet	Status of flow set-up request
FS_TXOPID	2 octets	The identifier used to identify TXOPs reserved (allocated) by the master for flow transmissions. This field is assigned only by the master in response to a flow set-up request
FlowParameters	*N* octets	Flow parameters returned in response. The flow parameters returned are those that differ from the corresponding request parameters. Flow parameters are as defined in Table 11-66.
Next Ethertype	2 octets	= 0
Pad	Variable	Pad to reach minFrameSize if necessary
FCS	4 octets	Frame check sequence

The SETUP/MODIFY_FLOW_CONFIRM control frame (see Table 11-58) shall be used to complete the flow set-up/modify protocol. The flow setup/modify sequence is identified by the same *Request_Key* assigned during the request phase of the protocol. The *Confirmation* field is used to indicate acceptance or rejection of the flow signalling transaction.

Table 11-58 – Set-up/Modify flow confirm control frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_FLOW_SIGNALLING (32774)
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength is 8 for SSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for SETUP/MODIFY_FLOW_CONFIRM (2,5) as defined in Table 11-63
Request_Key	2 octets	Key used to identify the confirmation with request-response sequence
FS_SA	6 octets	MAC address of station at source of flow. Does not necessarily correspond to SA
FS_DA	6 octets	MAC address of station at destination of flow. Does not necessarily correspond to DA
FS_FlowID	1 octet	Flow identifier assigned by the flow source. If a flow set-up request is not initiated by the flow source, the flow identifier shall be returned in the flow set-up response.
Confirmation	1 octet	Confirmation code for the set-up flow protocol sequence
FS_TXOPID	2 octets	The identifier used to identify TXOPs reserved (allocated) by the master for flow transmissions. This field is assigned only by the master in response to a flow set-up request.
FlowParameters	*N* octets	Flow parameters, found in the set-up/modify flow response, and requiring re-negotiation. The flow parameters structure is optional and TLV encoded as described in Table 11-66.
Next Ethertype	2 octets	= 0
Pad	Variable	
FCS	4 octets	Frame check sequence

The FLOW_TEARDOWN_REQUEST control frame (see Table 11-59) shall be used to request the tear-down of a flow. The flow is identified by the { FS_SA, FS_DA, FS_FlowID } tuple. The flow tear-down transaction is ended by the reception of the FLOW_TEARDOWN_RESPONSE control frame (see Table 11-60).

Table 11-59 – Tear-down flow request control frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_FLOW_SIGNALLING (32774)
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength is 20 for SSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for TEARDOWN_FLOW_REQUEST(6) as defined in Table 11-63
Request_Key	2 octets	Key used to identify the tear-down request
FS_SA	6 octets	MAC address of station at flow source
FS_DA	6 octets	MAC address of station at flow destination
FS_FlowID	1 octet	ID of flow to be torn down
FS_Pad	1 octet	Ignored on reception
Next Ethertype	2 octets	= 0
Pad	24 octets	
FCS	4 octets	Frame check sequence

Table 11-60 – Tear-down flow response control frame

Field	Length	Meaning
DA	6 octets	Destination address
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_FLOW_SIGNALLING (32774)
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength is 8 for SSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for TEARDOWN_FLOW_RESPONSE(7) as defined in Table 11-63
Request_Key	2 octets	Key used to identify the tear-down request
FS_SA	6 octets	MAC address of station at flow source
FS_DA	6 octets	MAC address of station at flow destination
FS_FlowID	1 octet	ID of flow to be torn down

Table 11-60 – Tear-down flow response control frame

Field	Length	Meaning
Status	1 octet	Status of teardown request
Next Ethertype	2 octets	= 0
Pad	36 octets	
FCS	4 octets	Frame check sequence

The GET_FLOW_PARAMS_REQUEST control frame (see Table 11-61) shall be used to request the flow parameters for a given flow identified by { FS_SA, FS_DA, FS_FlowID }. The flow parameters are returned in the GET_FLOW_PARAMS_RESPONSE control frame (see Table 11-62).

Table 11-61 – Get flow parameters request control frame

Field	Length	Meaning
DA	6 octets	Destination address. FS_SA or address of master
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_FLOW_SIGNALLING (32774)
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum SSLength is 18 for SSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for GET_FLOW_PARAMS_REQUEST (8) as defined in Table 11-63
Request_Key	2 octets	Key used to identify the tear-down request
FS_SA	6 octets	MAC address of station at source of flow. Does not necessarily correspond to SA
FS_DA	6 octets	MAC address of station at destination of flow. Does not necessarily correspond to DA
FS_FlowID	1 octet	Identity of flow between FS_SA and FS_DA being queried
FS_pad	1 octet	Ignored on reception
Next Ethertype	2 octets	= 0
Pad	0 octet	
FCS	4 octets	Frame check sequence

Table 11-62 – Get flow parameters response control frame

Field	Length	Meaning
DA	6 octets	Destination address. FS_SA or address of master
SA	6 octets	Source address
Ethertype	2 octets	0x886c (HNT link control frame)
LSType	2 octets	= SUBTYPE_FLOW_SIGNALLING (32774)
LSLength	2 octets	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. Minimum SSLength is 50 for SSVersion 0.
LSVersion	1 octet	= 0
MsgType	1 octet	Message type for GET_FLOW_PARAMS_RESPONSE (9) as defined in Table 11-63
Request_Key	2 octets	Key used to identify the tear-down request
FS_DA	6 octets	MAC address of station at destination of flow. Does not necessarily correspond to DA
FS_FlowID	1 octet	Identity of flow between FS_SA and FS_DA being queried
Status	1 octet	Status of the get flow parameters request
FlowProperties	32 octets	QoS properties of the flow specified in the corresponding request control frame
Next Ethertype	2 octets	= 0
Pad	Variable	
FCS	4 octets	Frame check sequence

Table 11-63 describes the MsgType values used in the flow signalling control frame.

Table 11-63 – Flow signalling protocol message types

MsgType	Meaning
0	SETUP_FLOW_REQUEST
1	SETUP_FLOW_RESPONSE
2	SETUP_FLOW_CONFIRM
3	MODIFY_FLOW_REQUEST
4	MODIFY_FLOW_RESPONSE
5	MODIFY_FLOW_CONFIRM
6	TEARDOWN_FLOW_REQUEST
7	TEARDOWN_FLOW_RESPONSE
8	GET_FLOW_PARAMS_REQUEST
9	GET_FLOW_PARAMS_RESPONSE
10-127	Reserved
128-135	Reserved for master notification of setup, modify and teardown flow request, response and confirm messages

11.16.1.1 Flow parameters

Flow parameters are specified in the flow signalling control frames using one of two kinds of TLV encoded structures:

1) Flow specification structure (see Table 11-64);

2) Flow parameters structure (see Table 11-65).

The first structure (see Table 11-64), the *"Flow Specification"*, describes each QoS parameter in a flow specification and may be used by a station when setting up a flow or when responding to a GET_FLOW_PARAMS_REQUEST.

Table 11-64 – Flow specification TLV structure

Field	Length	Meaning
SETag	1 octet	= FS_PARAMS_TAG (2)
SELength	1 octet	Total length of TLV extension excluding the tag and length octets (= 30)
Subtype	2 octets	= Flow specification(0)
ControlWord#1	2 octets	See Table 11-66 item 2.
ControlWord#2	2 octets	See Table 11-66 item 3.
PacketSize	2 octets	See Table 11-66 item 4.
MaxPacketSize	2 octets	See Table 11-66 item 5.
MaxDataRate	2 octets	See Table 11-66 item 6.
AvgDataRate	2 octets	See Table 11-66 item 7.
MinDataRate	2 octets	See Table 11-66 item 8.
BER	1 octet	See Table 11-66 item 9.
PE	1 octet	See Table 11-66 item 10.
PacketTimeout	4 octets	See Table 11-66 item 11.
TXTimeslot	4 octets	See Table 11-66 item 12.
FlowTimeout	4 octets	See Table 11-66 item 13.

The second structure (see Table 11-65), the *"Flow Parameters"* structure, is an incremental structure that can be used to report individual QoS flow parameters or sets of parameters. It shall be used to notify of changes to specific QoS parameters or changes to a specific set of QoS parameters.

Table 11-65 – Flow parameters TLV structure

Field	Length	Meaning
SETag	1 octet	= FS_PARAMS_TAG (2)
SELength	1 octet	Total length of TLV extension excluding the tag and length octets. Minimum length is 3 and the maximum is 49.
Subtype	2 octets	= Flow parameters(1)
FPType	1 octet	See Table 11-66.
FPLength	1 octet	See Table 11-66.
FlowParameter	1-4 octets	See Table 11-66.
•••		[additional instances of flow parameters]

Table 11-66 describes the flow parameters used in the flow signalling control frames. Horizontal shading is used to show the decomposition of byte and word fields into bit-fields.

Table 11-66 – Flow properties

Nr.	Parameter name	FPType	FPLength [octets]	Values	Comments
1	Pad	00	1	0	
2	Control word #1	0x01	2		Control Word is decoded as shown immediately below.
	Priority		Bits 13:15	0-7	Priority assigned to the flow. May be used for G.9954v2 priority semantics
	Service type		Bits 10:12	0-3	Defines the type of service that the flow supports: 0 – CBR 1 – rt-VBR 2 – nrt-VBR 3 – BE 4~7 Reserved The service type should be used by the QoS scheduler as a recommendation for the bandwidth allocation model to apply to the flow.
	Max. latency		Bits 5:9	0-16	Maximum tolerable transmission and queuing delay according to Table 11-67. 17~31 – Reserved
	Max. jitter		Bits 2:4	0-3	Maximum delay variation according to Table 11-68. 5~7 Reserved
	Reserved		Bits 0:1	0	Must be set to zero by the transmitter and ignored by the receiver
3	Control word #2	0x02	2		A set of control fields controlling flow behaviour policy. The control word is decoded as shown immediately below:
	ACK policy		Bits 15:15	0-1	0 – None 1 – LARQ
	FEC policy		Bits 13:14	0-3	0 – None 1 – Reed-Solomon 2~3 – Reserved
	Aggregation policy		Bits 12:12	0-1	0 – None 1 – MAC-level aggregation

Table 11-66 – Flow properties

Nr.	Parameter name	FPType	FPLength [octets]	Values	Comments
	Checksum error handling policy		Bits 11:11	0-1	0 – Do not discard packets with checksum errors. 1 – Discard packets with checksum errors. A checksum error includes an error in the FCS or CRC-16 fields of the G.9954v2 link-layer frame or frame burst.
	Reserved		Bits 0:10	0	Must be set to zero by the transmitter and ignored by the receiver.
4	Nominal packet size	0x03	2	0-64 kbit/s	The nominal packet size in octets for packets associated with the service. A value of 0 indicates an unspecified or unknown value.
5	Max. packet size	0x04	2	0-64 kbit/s	The maximum packet size in octets for packets associated with the service. A value of 0 indicates an unspecified or unknown value. NOTE – Used by the scheduler to ensure that TXOPs are at least large enough to include a single packet.
6	Max. data rate	0x05	2	4 kbit/s-256 Mbit/s	Peak burst rate in 4-kbits-per-second units. Takes into account the net (payload) data rate
7	Average data rate	0x06	2	4 kbit/s-256 Mbit/s	Average required bit-rate required by the service in units of 4 kbit/s
8	Min. data rate	0x07	2	4 kbit/s-256 Mbit/s	Minimum required bit-rate in 4-kbit/s units for the service to operate. This number is expected to be different from zero only for real-time traffic requiring a minimum transmission delay (min ≤ *avg* ≤ max).
9	BER Word	0x08	1	10^{-10} to 10^{-5}	Service-level BER in the range $10^{-10} \leq BER \leq 10^{-5}$. BER is represented by two integer fields: mantissa, m, and exponent, e, such that: $BER = (8 + m) \times 2^{e-43}$ When CRC error handling policy is *discard packets with CRC Error*, the BER value is the PER divided by the mean number of bits per packet. For example, suppose the desired $PER = 10^{-2}$ and 1500-byte packets are used, then $BER = 10^{-2}/12000 \approx 10^{-6}$.
	Mantissa (m)		Bits 5:7	0-7	

Table 11-66 – Flow properties

Nr.	Parameter name	FPType	FPLength [octets]	Values	Comments
	Exponent (e)		Bits 0:4	7-24	
10	PE	0x09	1	0-255	Payload encoding used on logical channel. The value of PE should be derived by rate negotiation from BER requirements.
11	Packet timeout	0x0A	4	0-(2^32 – 1)	Amount of time in milliseconds a packet will remain queued before being deleted from the flow queue. A value of 0 indicates that the packet never times out.
12	Reserved	0x0B	4		Reserved for legacy systems
13	Flow inactivity timeout	0x0C	4	0-(2^32 – 1)	Amount of time in milliseconds a flow will remain "alive" in the absence of any traffic before the flow is automatically torn down and resources released. A value of 0 indicates that the flow is never torn down automatically.

Tables 11-67 and 11-68 list the possible values for the maximum latency and maximum jitter and their meaning.

Table 11-67 – Maximum latency values

Latency	Meaning
0	No limit
1	5 milliseconds
2	10 milliseconds
3	20 milliseconds
4	30 milliseconds
5	40 milliseconds
6	50 milliseconds
7	60 milliseconds
8	70 milliseconds
9	80 milliseconds
10	90 milliseconds
11	100 milliseconds
12	200 milliseconds
13	300 milliseconds
14	400 milliseconds
15	500 milliseconds

Table 11-68 – Maximum jitter values

Jitter	Meaning
0	No limit
1	5 milliseconds
2	10 milliseconds
3	20 milliseconds

11.16.1.2 Flow classifier

Flow classifiers are filter specifications that define the criteria by which the G.9954v2 convergence layer will classify packets and map them to flows. Table 11-69 describes the flow classifier TLV structure used in the SETUP/MODIFY_FLOW_REQUEST control frame.

Table 11-69 – Flow classifier data

Field	Length	Comments
SETag	1 octet	= FS_CLASSIFIER_TAG (Table 11-39)
SELength	1 octet	Total length of TLV extension excluding the tag and length octets
Priority	1 octet	Priority of classifier. Defines order in which classifiers are applied within a convergence layer. A higher value indicates a higher priority.
Opcode	1 octet	Classifier action to be applied: 0 – Add classifier 1 – Delete classifier
ClassifierParam		Classifier parameter
ClassifierTag	1 octet	Classifier tag identifier. For a description of classifier tag values, see Table 11-70. Values 0x0E~0xFF are reserved.
ClassifierLength	1 octet	Length of the classifier parameter
ClassifierParameter	Variable	A classification parameter whose structure is specific to the ClassifierTag as described in Table 11-70.

Table 11-70 – Classifier parameters

Classifier parameter	Classifier tag	Length [octets]	Comments
Flow ID	0x00	2	Flow ID of the flow to which an incoming packet has been determined to belong by higher protocol layers
Destination address	0x01	N * 6	A list of (N) Ethernet destination addresses
Source address	0x02	N * 6	A list of (N) Ethernet source addresses
EtherType	0x03	N * 2	A list of (N) EtherType values
TOS	0x04	3	IP Type of Service field: (tos_{low}, tos_{high}, tos_{mask})
Protocol	0x05	N * 1	List of protocols: $protocol_1$..$protocol_n$
IP source address	0x06	N * 8	A list of source IP (address,mask) tuples
IP destination address	0x07	N * 8	A list of (N) destination IP (address,mask) tuples

Table 11-70 – Classifier parameters

Classifier parameter	Classifier tag	Length [octets]	Comments
Source port range	0x08	N * 4	A list of (N) source IP port number ranges ($port_{low}$, $port_{high}$)…
Destination port range	0x09	N * 4	A list of (N) destination IP port number ranges ($port_{low}$, $port_{high}$)…
EtherType/802.2 DSAP	0x0A	N * 1	LLC DSAP address
EtherType/802.2 SSAP	0x0B	N * 1	LLC SSAP address
User priority	0x0C	2	A range of 802.1D user priority values pri_{low}, pri_{high}
VLAN ID	0x0D	2	The 802.1Q VLAN identifier. Only the leftmost 12 bits are significant.
Generic	0x0E	N * 16	A list of (N) generic classification rules composing a classification filter where a classification rule is defined by the following parameters:
Pattern		6	1-, 2-, 4- or 6-octet pattern
Mask		6	1-, 2-, 4- or 6-octet mask applied to pattern
Offset		2	Offset within packet for pattern matching where offset zero corresponds to the DA field in the Ethernet frame
Mask size		1	Size of the pattern/mask (i.e., 1, 2, 4 or 6)
Pad		1	Padding for alignment

11.16.2 Flow signalling transactions

Multiple flow signalling transactions may be initiated by a station simultaneously using a uniquely assigned *Request Key*. All protocol messages belonging to the same transaction shall use the same *Request Key*. The *Request Key* shall be assigned by the initiator of the flow signalling transaction.

11.16.3 Flow signalling protocol sequences

11.16.3.1 Flow Set-up protocol sequence

Flow set-up shall be performed between source and destination endpoints of a flow using the flow set-up protocol sequence. Either the source or destination stations may initiate the flow set-up.

The purpose of flow set-up signalling is to establish a set of well-defined and negotiated flow parameters between flow endpoints.

If reserved bandwidth (QoS contracts) is required for a flow, the master shall be informed of the flow set-up parameters, by the flow source, after flow parameters have been negotiated. Notification of flow set-up to the master and reservation of bandwidth shall be performed using the same 3-way handshake for flow set-up used between two endpoint nodes.

The master may also be the source or destination endpoint of a flow. This is a special case of the standard flow set-up protocol sequence.

NOTE – When the master is at the endpoint of a flow, no further master notification is required in order to reserve bandwidth beyond the original flow set-up signalling.

The different flow set-up protocol sequences are defined in the clauses below.

11.16.3.1.1 Source-initiated flow set-up procedure

To set-up a flow between two G.9954v2 devices on the network where the device initiating the flow set-up shall be the device at the source of the flow, the initiator shall send a SETUP_FLOW_REQUEST message to the device at the flow destination. The SETUP_FLOW_REQUEST message shall contain a *Request Key* assigned by the initiator and identifying the flow set-up transaction, the flow identity and flow QoS parameters. The flow identity shall be locally assigned by the initiator by assigning a flow identifier that will be unique within the context of the flow *source* and *destination* addresses.

After sending the SETUP_FLOW_REQUEST, the station shall set a timer and wait for up to FLOW_RESPONSE_TIMEOUT (T1) ms for a SETUP_FLOW_RESPONSE message. If no response is received within the timeout period, the request shall be resent using the same *Request Key*. This process shall be performed until the MAX_FLOW_SIGNALLING_RETRIES.

Upon receiving a SETUP_FLOW_REQUEST, the destination station shall set up the flow locally. It may offer suggested modifications to the flow parameters in order to better suit the flow to the endpoint's resource restrictions. Any modified parameters shall be returned in the SETUP_FLOW_RESPONSE. After sending a SETUP_FLOW_RESPONSE, the destination endpoint shall start a timer and wait for up to FLOW_CONFIRM_TIMEOUT (T2) ms for the SETUP_FLOW_CONFIRM message. If a SETUP_FLOW_CONFIRM is not received within this timeout period, a SETUP_FLOW_RESPONSE message shall be retransmitted. This procedure continues MAX_FLOW_SIGNALLING_RETRY times before the destination shall close the transaction.

If a SETUP_FLOW_RESPONSE is received, the station shall disable the timer (T1) and check the returned status and flow parameters. If flow parameters were modified by the destination station in its response, then the source shall adjust its flow parameters accordingly. If the return status in the SETUP_FLOW_RESPONSE is OK and the modified parameters are acceptable to the source, the source station shall return a FLOW_SETUP_CONFIRM message with a status of OK and the flow set-up transaction closed. If the offered flow parameters are rejected by the source station, it shall return a confirmation code of REJECT together with the rejected parameters.

Upon receiving a SETUP_FLOW_CONFIRM message, the station shall disable the timer (T2). If the *Confirmation Code in the* SETUP_FLOW_CONFIRM is OK, then the destination station may complete the flow set-up transaction. If the *Confirmation Code* is REJECT, the destination station may either end the flow set-up transaction or it may modify its offer using the same FLOW_SETUP_RESPONSE/CONFIRM cycle. If the flow cannot be successfully set up, a SETUP_FLOW_RESPONSE status of ERROR should be returned and the flow set-up transaction shall be closed at source and destination.

If a flow is not successfully set up between source and destination stations, flow data may be sent by the device in any available transmission opportunity assigned to it.

If a flow is successfully set up and the network is master-controlled, bandwidth may be reserved for the flow by signalling the flow set-up with the master. For more information on reserved bandwidth allocation for a flow, see clause 11.16.3.1.4.

Figure 11-14 illustrates the *Flow Set-up Signalling Protocol* used to set up a flow between devices A (the source) and device B (the destination) when the initiator of the *Flow Set-up* transaction in the example is Device A. This example, illustrates the timer periods (T1, T2) used in the flow signalling protocol as well as rate negotiation (RRCF) performed over the flow channel.

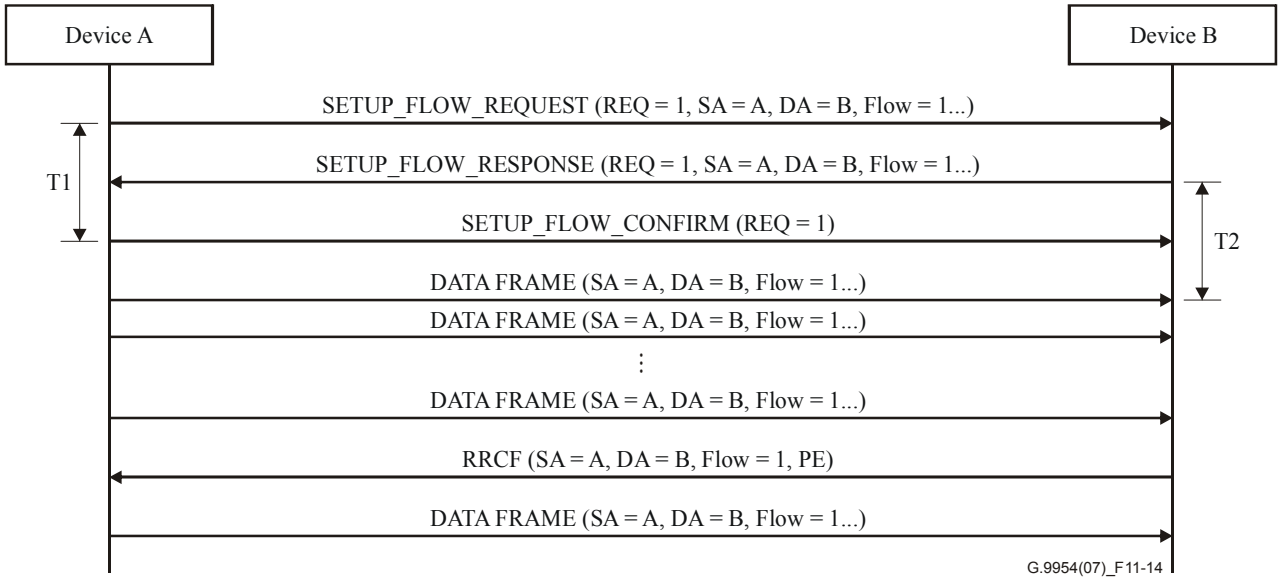

Figure 11-14 – Source-initiated flow set-up procedure

11.16.3.1.2 Destination-initiated flow set-up procedure

Flow set-up, when initiated by the flow destination is similar to the procedure described in clause 11.16.3.1.1. The difference between the sequences is as follows:

The Flow_ID specified in the FLOW_SETUP_REQUEST is NULL since the Flow_ID must be defined by the station at the source of the flow. The assigned Flow_ID is returned in the FLOW_SETUP_RESPONSE.

Flow parameter negotiation proceeds as for the case of the source-initiated flow set-up.

Figure 11-15 illustrates the *Flow Set-up Signalling Protocol* used to set up a flow between devices A (the source) and device B (the destination) when the initiator of the *Flow-Setup* transaction in the example is Device B.

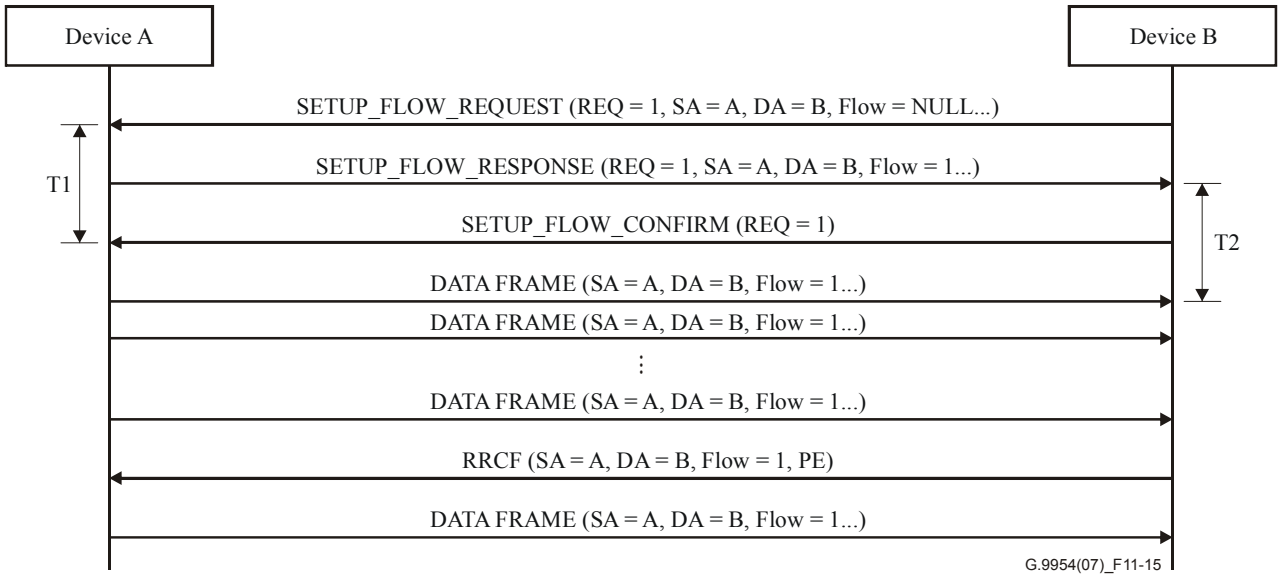

Figure 11-15 – Destination-initiated flow set-up protocol

11.16.3.1.3 Broadcast/Multicast Flow Set-up procedure

When setting up a broadcast/multicast flow, the *Flow Set-up Signalling Protocol* does not use the standard 3-way handshake to set up the flow since the initiator of the flow set-up cannot wait for a response from all broadcast/multicast group members. Rather, flow set-up shall be signalled by broadcasting the SETUP_FLOW_REQUEST without waiting for a response and without having to reply with a confirm. Flow parameters (except for payload encoding (PE)) cannot be negotiated for broadcast/multicast flows. Payload encoding shall be negotiated using the standard rate negotiation mechanism for broadcast/multicast channels as described in clause 11-4.

In order to allow a broadcast/multicast group member to acquire flow parameters at any time, in case the SETUP_FLOW_REQUEST was not received, or the broadcast/multicast group member came alive after the establishment of the flow, a station may make a request to *Get Flow Parameters* at any time using the GET_FLOW_PARAMS_REQUEST. The request is sent to the station at the flow source. The station at the flow source, upon receiving a GET_FLOW_PARAMS_REQUEST, shall return the parameters for the designated flow using the GET_FLOW_PARAMS_RESPONSE message.

The flow set-up protocol sequence in the case of broadcast/multicast flows is illustrated in Figure 11-16.

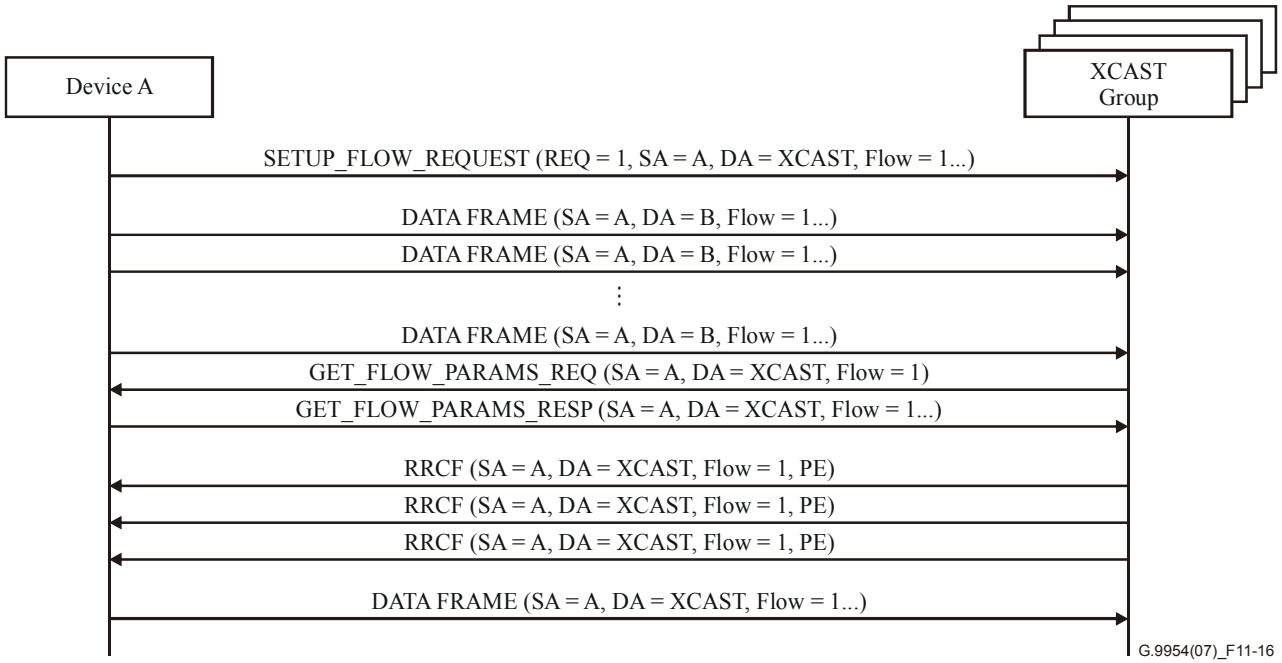

Figure 11-16 – Multicast flow set-up

11.16.3.1.4 Master flow set-up notification procedure

As described in clauses 11.16.3.1.1, 11.16.3.1.2 and 11.16.3.1.3, the flow set-up protocol shall be performed between flow source and destination devices, irrespective of whether the network is master-controlled or not. This allows the definition of flows with defined latency, rate and BER characteristics. This information may be used by transmitter and receiver devices to negotiate appropriate channel parameters for the flow (e.g., buffer requirements, payload encoding, etc.).

If the network is master-controlled, explicit TXOPs may be reserved for an established flow by signalling the master of the flow set-up using the regular flow set-up signalling protocol.

To signal flow set-up with the master, the protocol shall be initiated by the flow source. The same 3-way protocol handshake shall be used as for the regular flow set-up operations between source and destination devices. If the flow is admitted by the master, reserved TXOPs shall be allocated by

the master and assigned in the advertised master-generated MAP. The TXOPs shall be allocated by the master scheduler in such a manner and position so as to provide sufficient bandwidth and meet latency and jitter requirements defined for the flow in flow parameters.

Devices at the source of a flow shall be *registered* with the master in order to be able to request reserved bandwidth.

Figure 11-17 illustrates the *Flow Set-up Protocol Sequence* including master Flow Set-up Notification. The *Flow Set-up Protocol Sequence*, appearing between Device A and Device B (i.e., within the double-sided arrow), represents the protocol sequence as described in Figures 11-14, 11-15 and 11-16. The *Flow Set-up Protocol Sequence* between Device A and master represents the reserved bandwidth allocation request.

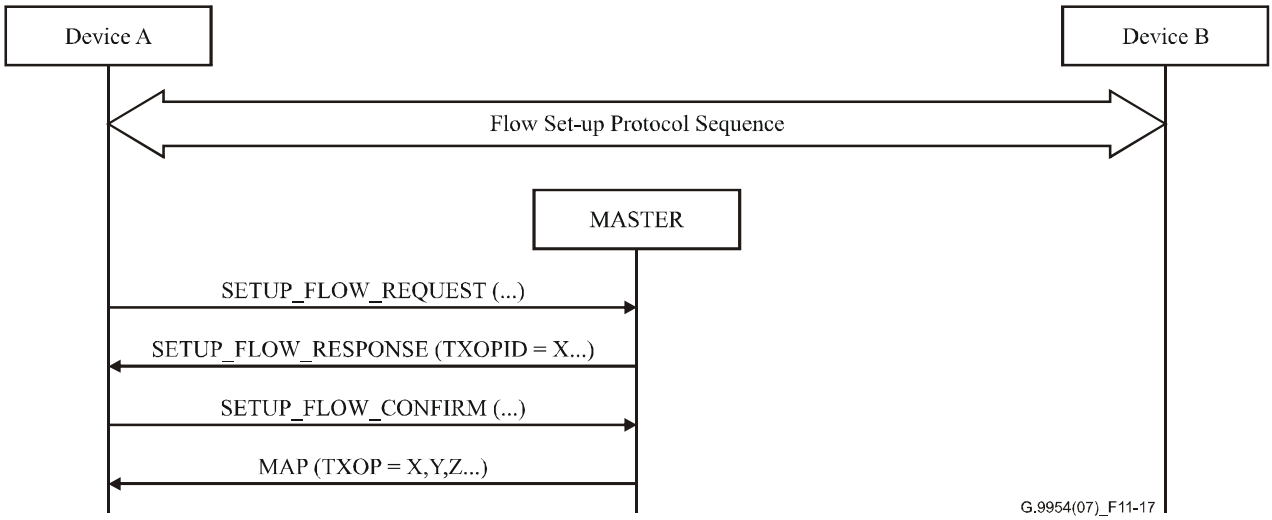

Figure 11-17 – Master flow set-up notification

11.16.3.1.5 Master-initiated and terminated flow set-up procedure

If the device initiating the flow set-up sequence is the master, the flow set-up sequence proceeds normally, as for the case of a regular endpoint station (see clauses 11.16.3.1.1 and 11.16.3.1.2). In this case, admission control may be performed by the master before the protocol sequence begins. Furthermore, the master need not be notified of the flow set-up in order to reserve bandwidth. This shall be performed automatically by the master for flows requiring reserved bandwidth.

Similarly, for flows whose endpoint terminates at the master the *Flow Set-up Protocol Sequence* proceeds as for the regular case and bandwidth reservation shall be performed automatically by the master as required.

NOTE – Bandwidth for a flow need not be allocated immediately by the master and may be deferred until flow channel parameters (e.g., payload encoding) have been determined.

11.16.3.2 Flow modification protocol sequence

The flow modification protocol sequence closely follows the *Flow Set-up Protocol*. It similarly involves a 3-way REQUEST-RESPONSE-CONFIRM protocol exchange sequence between flow source and destination devices and optionally between flow source and master device.

Flow modification can be initiated by flow source or destination devices. Similar to the flow set-up protocol, the master shall be informed of modifications to flows for which bandwidth has been explicitly reserved, if the modified parameters effect bandwidth reservation.

Modifications to the following parameters effect master bandwidth reservation:

- Data rate (Minimum, Average, Maximum);
- Maximum latency/jitter;
- Payload encoding;
- Nominal packet size.

11.16.3.2.1 Flow modification procedure

The device requesting the flow modification shall open a flow signalling transaction and send a MODIFY_FLOW_REQUEST message containing a specification of the flow parameters to be modified and/or optionally the traffic classification filters to be installed in the device at the source of the flow.

After sending the MODIFY_FLOW_REQUEST, the initiator shall set a timer and wait for up to FLOW_RESPONSE_TIMEOUT (T1) ms for a MODIFY_FLOW_RESPONSE. If the timer expires before the response is received, the MODIFY_FLOW_REQUEST shall be resent up to MAX_FLOW_SIGNALLING_RETRY times before the flow modification request shall be abandoned.

Upon receiving a MODIFY_FLOW_REQUEST message, the receiving device should look up the specified flow in its list of established flows and, if found, set up a new flow signalling transaction. Modified parameters should be checked and, if acceptable, the flow parameters should be updated accordingly. A MODIFY_FLOW_RESPONSE with a *Status* of OK should subsequently be returned within (T1)/2 ms from the time the MODIFY_FLOW_REQUEST was received. If the modified flow parameters are unacceptable, a MODIFY_FLOW_RESPONSE with a *Status* of REJECT should be returned. The rejected parameters should be returned in the response message.

The remainder of the protocol sequence, including renegotiation of flow parameters (if necessary), and the termination of the flow signalling transaction proceeds as for the case of *Flow Set-up*. This is illustrated in Figure 11-18.

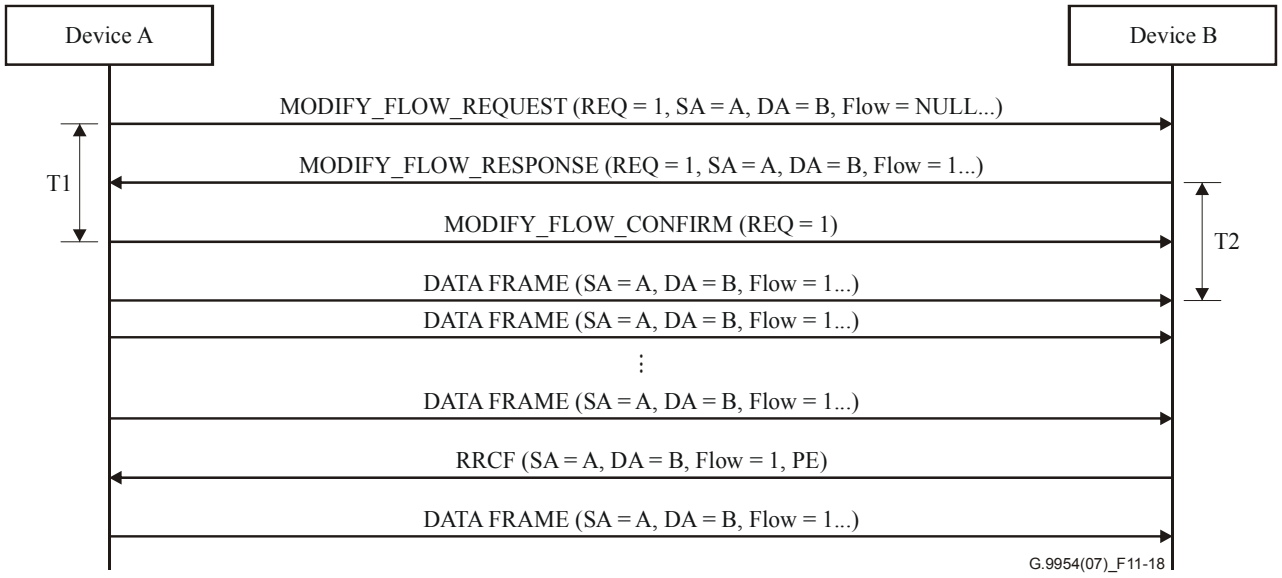

Figure 11-18 – Modify flow signalling protocol

11.16.3.2.2 Master notification and flow modification

If a flow that has bandwidth reserved by the master is modified, the master shall be notified of any modifications to flow parameters that effect bandwidth allocation. Notification shall be performed using the *Modify Flow Signalling Protocol*.

Flow parameters that may be modified and effect bandwidth reservation, are as defined in clause 11.16.3.2.

The *Modify Flow Signalling Protocol* between the device at the flow source and the master is the same as described in clause 11.16.3.2.1.

11.16.3.3 Flow tear-down protocol sequence

Flows are torn down using the *Flow Tear-down Protocol Sequence.* A flow may be torn down in response to an explicit request from an upper protocol layer or after a flow-parameter configurable period of inactivity (see *Flow Inactivity Timeout* flow parameter in clause 11.16.1.1).

The flow tear-down sequence is normally initiated by the device at the source of the flow after sensing a period of flow inactivity greater than or equal to the flow's *Flow Inactivity Timeout.* Flow tear-down may also be initiated by the device at the destination of a flow if it senses a period of inactivity greater than its *Flow Inactivity Timeout* parameter.

The *Flow Tear-down Protocol Sequence* involves a REQUEST-RESPONSE message sequence. The initiator shall identify the flow by *Source Address, Destination Address* and *Flow ID.* When a flow is torn down, the resources it binds shall be released.

If a flow that has bandwidth reserved to it by the master is torn down, the master shall be notified by the device initiating the *Flow Tear-down Sequence*.

If a registered device is no longer detected, as indicated by the absence of capability and status announcement (CSA) control frames, the master shall de-register the device and tear down all flows sourced at the device. Similarly, devices at the source of a flow shall detect the absence (using CSA timeout) of a device at the flow's destination and shall tear down such flows accordingly.

The *Flow Inactivity Timeout* at the source of a flow shall be greater than the *Flow Inactivity Timeout* at a flow's destination in order to eliminate flow tear down race conditions.

NOTE – The initiator of a flow set-up protocol sequence should guarantee this above requirement by specifying the desired flow inactivity timeout accordingly. This means that for a flow set-up initiated by the flow source, the flow inactivity timeout specified in the set-up request should be forced to be greater than the parameter at the flow source. Similarly, for a flow set-up initiated by the flow destination, the flow inactivity timeout specified should be less than the value used at flow destination.

11.16.3.3.1 Station-initiated flow tear-down procedure

Flow Tear-down Protocol signalling shall be performed between devices found at the endpoints of a flow or between the device at the source of a flow and the master. In either case, a device initiates the flow tear-down protocol sequence by sending a TEARDOWN_FLOW REQUEST message containing the identity of the flow to be torn down and a unique *Request Key* identifying the flow signalling transaction. The initiating device shall subsequently set a timer and wait for up to FLOW_RESPONSE_TIMEOUT (T1) ms for a TEARDOWN_FLOW_RESPONSE message before resending the tear-down request. This procedure shall be performed up to MAX_FLOW_SIGNALLING_RETRY times before the flow tear-down transaction shall be terminated and the flow torn down locally.

A device receiving a TEARDOWN_FLOW_REQUEST shall search for the identified flow in its database of active flows and, if found, the device should tear-down the flow locally and release resources bound to the flow. In all cases, a TEARDOWN_FLOW_RESPONSE should be returned within (T1)/2 ms.

The *Flow Tear-down Protocol* sequence is illustrated in Figure 11-19. The scenario described shows a flow tear-down sequence between devices at the endpoints of a flow and between the device at the flow source and the master.

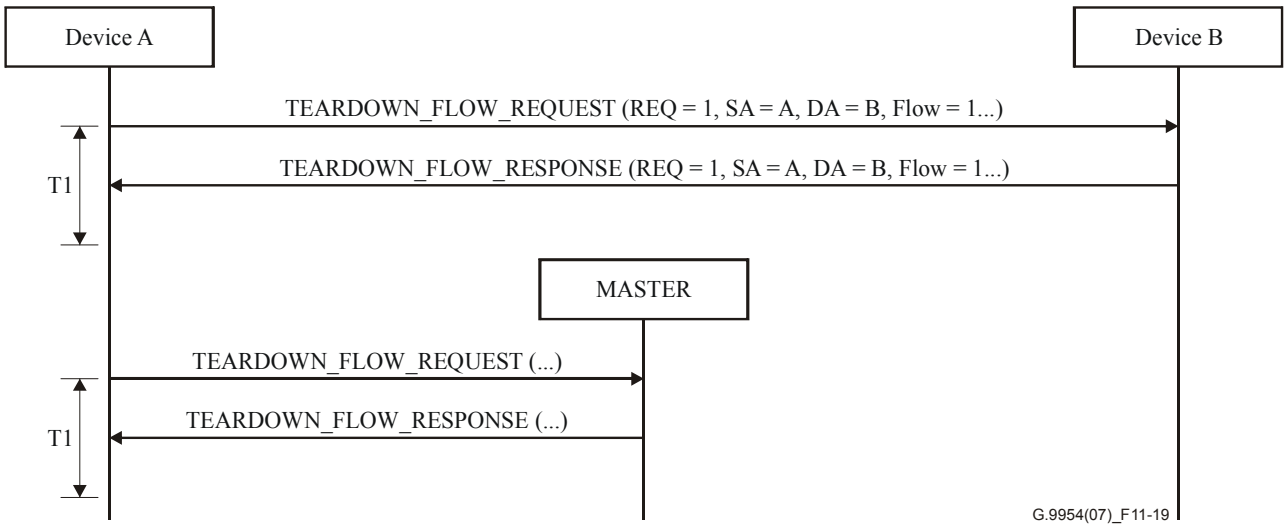

Figure 11-19 – Flow tear-down protocol

11.16.3.3.2 Flow tear-down signalling with the master

If a flow has bandwidth reserved to it by the master and the flow is torn down, the master shall be notified by the device found at the flow source. The master shall be notified using the *Flow Tear-down Protocol* sequence, the same as for devices found at the endpoints of a flow.

11.16.3.3.3 Broadcast and multicast flow tear-down

To tear down a broadcast or multicast flow, the TEARDOWN_FLOW_REQUEST shall be sent by the device at the source of the flow. The TEARDOWN_FLOW_REQUEST message shall be sent using the broadcast/multicast address. The initiating device shall not wait for a TEARDOWN_FLOW_RESPONSE and may end the transaction after sending the tear-down request.

If a broadcast/multicast group member does not receive the TEARDOWN_FLOW_REQUEST, the flow shall be timed out by each device using the standard *Flow Inactivity Timeout* mechanism.

11.16.4 Terms and parameters

- MAX_FLOW_SIGNALLING_RETRY – Number of times a device should retry. The recommended value of MAX_FLOW_SIGNALLING_RETRY is 3.

11.16.4.1 Timers

- T1 – A one-shot timer set after the transmission of a flow signalling_request message. Used to time out the expected flow signalling response before retrying the request. This timer is cancelled if a response is received. The recommended value of the timer is 50 milliseconds.
- T2 – A one-shot timer set after sending a flow signalling response message when a confirm message is expected. Used to schedule the re-transmission of a flow signal response message request if a confirmation is not received. The recommended value of this timer is 50 milliseconds.

11.17 Timestamp report indication message (optional)

Synchronization to a master clock reference may be required, by some endpoint devices, in order to synchronize sampling rates or to synchronize the allocation of media TXOPs with an external source.

To support synchronization with a master clock, a master clock reference device distributes its clock to all devices on the network.

Any device on the network may be a master clock reference to some group of clock slave devices. More than one master clock reference device may co-reside on the network simultaneously. Typically a clock slave device should synchronize to a single master clock reference. There is no requirement that the device acting as the master in the network be a master clock reference.

The timestamp reporting mechanism assumes the ability of a master clock reference to latch the transmission timestamp of a well-known message (the Timestamp Report message itself) and to send the latched timestamp value in the subsequent Timestamp Report Indication message. Furthermore, it assumes the ability of an endpoint device to latch the receive timestamp of the same message. The time difference between the latched receive time at the endpoint and the latched transmit time at the master-clock reference is used to adjust the clock at the endpoint to compensate for the calculated clock frequency error. This is illustrated in Figure 11-20.

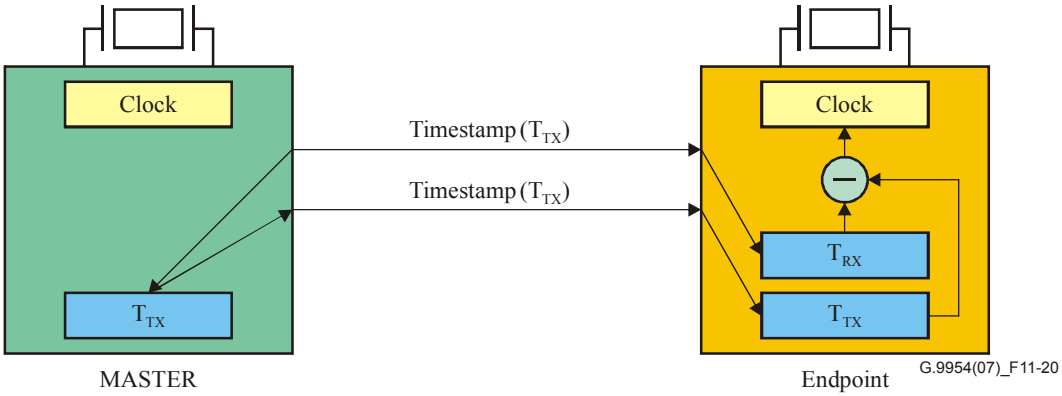

Figure 11-20 – Timestamp report indication

The master clock reference may transmit a timestamp report indication at any time. It should transmit pairs of these indications in successive frames. For each timestamp report indication message transmitted, the master clock reference shall increment the timestamp sequence number by one. The timestamp sequence number may start at any arbitrary value.

In measuring the start-of-transmission and start-of-reception times by the master clock reference and endpoint, respectively, the measurements must be defined with respect to a common point in the frame. That point is immediately following the MAC-layer source address field. A particular implementation may make its actual measurement with respect to other points in the frame, but in the following procedures below, it must correct the measured value so that the time corresponds to the specified point.

All endpoints that require data sampling synchronization are encouraged to receive the timestamp report indication and measure the start-of-reception time for received frames that contain this message. On reception of a timestamp report indication, the endpoint shall perform the following actions:

– Record the start-of-reception time of the current frame along with the timestamp sequence number and timestamp from the received timestamp report indication.

– Compare the timestamp sequence number parameter contained in the current frame with that of the most recently received Timestamp Report Indication. If the timestamps have a modulo difference of one, then continue. Otherwise, stop processing the message at this point.

– Calculate the relative frequency error of its internal clock by the following:

$$\text{Frequency error} = [(R_{(seqnum-1)} - R_{(seqnum-2)}) / (C_{seqnum} - C_{(seqnum-1)})] - 1$$

where:

$R_{(seqnum–1)}$ is the start-of-reception time of the frame containing the timestamp report indication with the previous sequence number, as measured by the endpoint's local clock.

$R_{(seqnum–2)}$ is the start-of-reception time of the frame containing the timestamp report indication with the sequence number two less (modulo) than that of the current frame, as measured by the endpoint's local clock.

C_{seqnum} is the timestamp value indicated in the timestamp report indication in the current frame (which corresponds to the start-of-transmission time of the frame containing the timestamp report indication with the previous sequence number, as measured by the master).

$C_{(seqnum-1)}$ is the timestamp value indicated in the timestamp report indication with the previous sequence number (which corresponds to the start-of-transmission time of the frame containing the timestamp report indication with the sequence number two less (modulo) than that of the current frame, as measured by the master clock reference).

– Adjust the local clock according to the determined frequency error using a locally defined algorithm.

The mechanism that the master clock reference or endpoints use to measure the frame start-of-transmit and start-of-receive time, respectively, is locally defined.

11.17.1 Timestamp report indication frame format

See Table 11-71.

Table 11-71 – Timestamp report indication frame format

Field	Length	Meaning
DA	6 octets	Destination address = FF:FF:FF:FF:FF:FF
SA	6 octets	Source address is that of the master clock reference
Ethertype	2 octets	0x886c (HNT link control frame)
SSType	1 octet	= SUBTYPE_TIMESTAMP_REPORT (8)
SSLength	1 octet	Number of additional octets in the control header, starting with the SSVersion field and ending with the second (last) octet of the Next Ethertype field. SSLength value is 8 for SSVersion 0
SSVersion	1 octet	= 0
Reserved	1 octet	Set to zero by sender and ignored by the receiver
TimestampSequenceNr	2 octets	A sequence number that increments by one each time a timestamp report indication is transmitted
Timestamp	4 octets	The time measured by the master of the start-of-transmission of the previous frame containing the timestamp report indication. The time is measured in units of ticks clocked at the frequency defined by ClockFrequency.
ClockFrequency	4 octets	Frequency of the clock used to clock the timestamp reference expressed in kHz. E.g., 8192 kHz for an 8.192-MHz clock with resolution of 2^{-13} ms.
Next Ethertype	2 octets	= 0
Pad	36 octets	
FCS	4 octets	

Annex A

Mechanical interface (MDI)

(This annex forms an integral part of this Recommendation)

A.1 RJ11 MDI connector

The wire connector mounted on the HNT device shall be an RJ11 female connector with the pin assignment of Table A.1.

Table A.1 – RJ11 MDI connector pin assignment

Contact	Signal
1	Not used
2	Not used
3	TX/RX (+)
4	TX/RX (–)
5	Not used
6	Not used

A depiction of the connector is shown in Figure A.1. The two pins labelled TX/RX(+) and TX/RX(–) constitute the HNT W1 interface to the phonewire network.

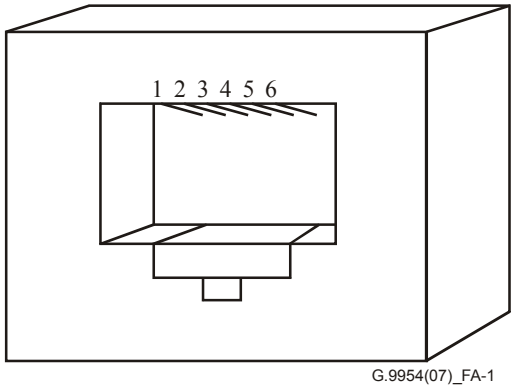

Figure A.1 – RJ11 female wire connector

A.2 F-type female connector

The wire connector mounted on the HNT device shall be an F-TYPE female connector.

A depiction of the connector is shown in Figure A.2. The centre and ground constitute the HNT W1 interface to the coaxial network.

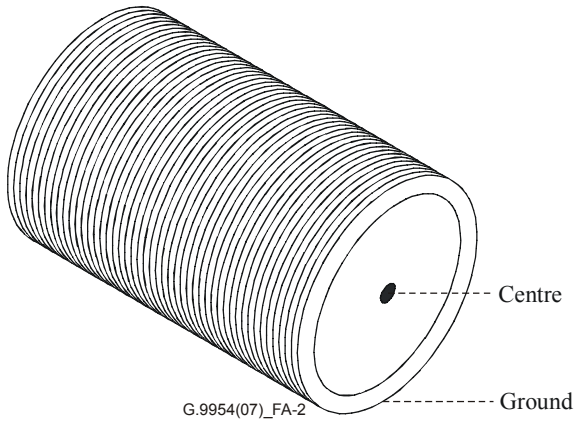

Figure A.2 – F-TYPE female wire connector

Annex B

Network test loops

(This annex forms an integral part of this Recommendation)

Ten test loops are defined for evaluating the performance of HNT receivers. This annex includes specification of the wire types and the topologies.

B.1 Wire model

The wire labelled "quad" in the following diagrams in Figures B.1 to B.10 is assumed to be Belden 1242A, or wire with equivalent primary parameters. The wire labelled "flat" is assumed to be Mouser flat 4-wire 26-AWG cable (stock number 172-UL4210), or wire with equivalent primary parameters. All other wire types are Belden UTP-5 of the specified gauge.

For simulations, the "BT #1" [ITU-T G.9954] model is used to generate primary parameters *R*, *L*, *G*, and *C* vs. frequency. This model is given as:

$$R(f) = \sqrt[4]{r_o^4 + a \cdot f^2}$$

$$L(f) = \frac{l_0 + l_\infty \cdot \left(\frac{f}{f_m}\right)^b}{1 + \left(\frac{f}{f_m}\right)^b}$$

$$G(f) = g_0 \cdot f^{g_e}$$

$$C(f) = c_\infty + \frac{c_0}{f^{c_e}}$$

The parameter set for each of the wire types used in the next clause is given in Table B.1. The assumption is that $R(f)$ is in units of ohms/mi., $L(f)$ is in units of mH/mi., $G(f)$ is in units of µmhos/mi., and $C(f)$ is in units of µF/mi.

Table B.1 – Model parameters for wires

Model parameter	Belden 1242A quad	Mouser flat 4-wire	Belden UTP-5 (24AWG)
r_0	406.65	643.4	277.2
A	0.2643	0.757	0.278
l_0	1.229	1.27	0.9863
B	0.794	0.654	0.83
l_∞	0.927	0.953	0.718
f_m	386e3	697e3	500e3
g_0	0.0432	0.519	0.000282
g_e	0.8805	0.7523	0.869
c_0	0.121	0.04	0
c_∞	0.071	0.06875	0.083
c_e	0.245	0.122	0

B.2 Test loops

See Figures B.1 to B.10.

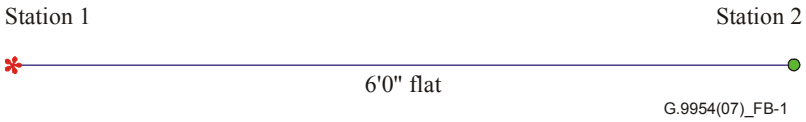

Figure B.1 – Test Loop Number 1

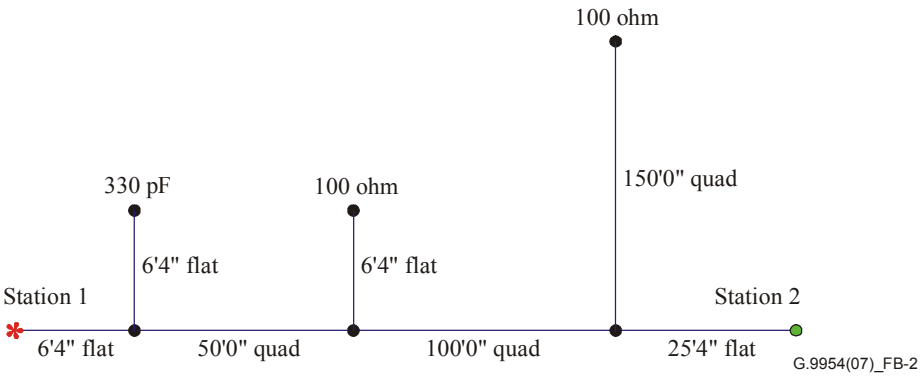

Figure B.2 – Test Loop Number 2

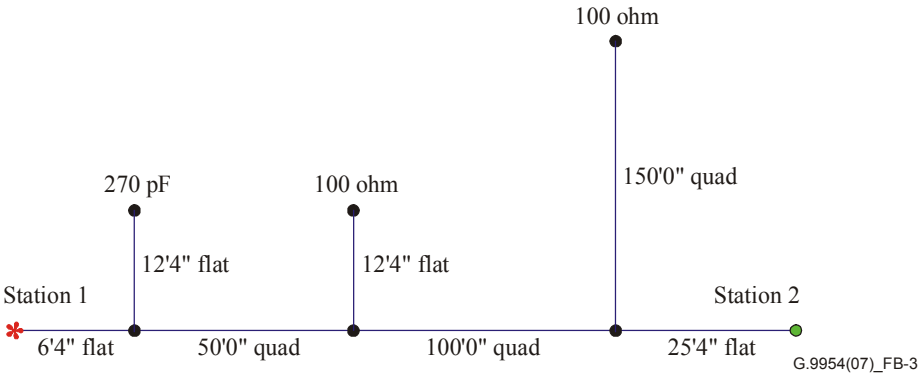

Figure B.3 – Test Loop Number 3

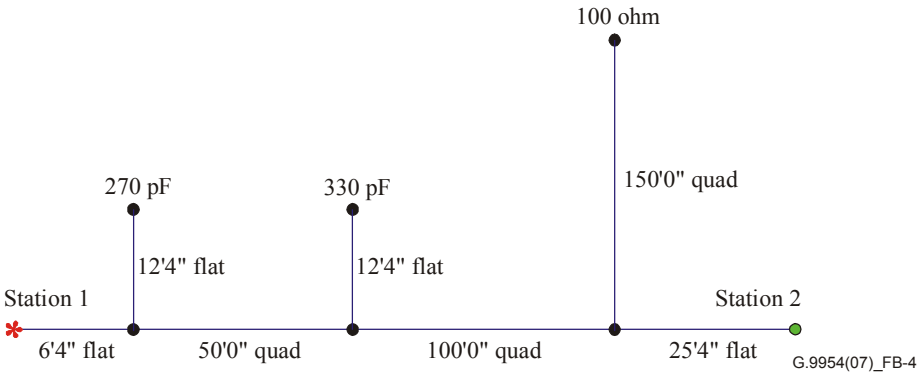

Figure B.4 – Test Loop Number 4

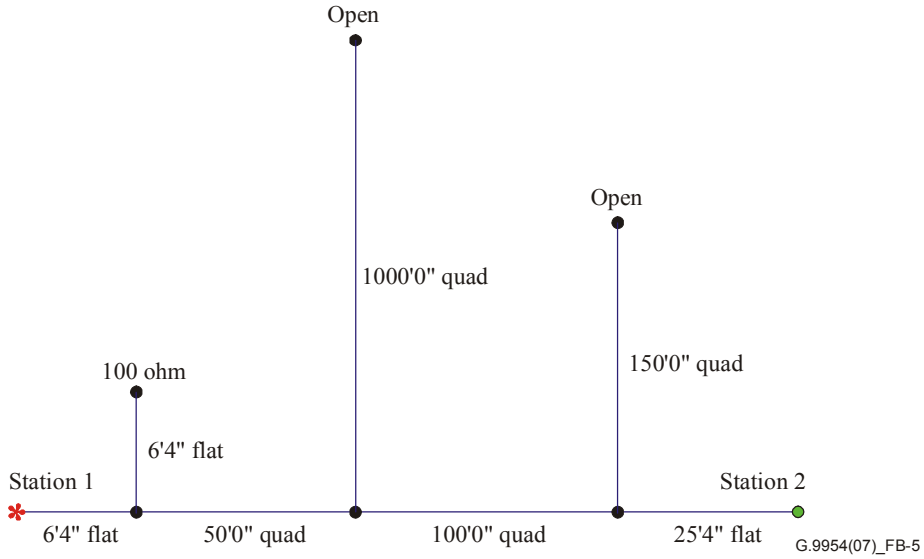

Figure B.5 – Test Loop Number 5

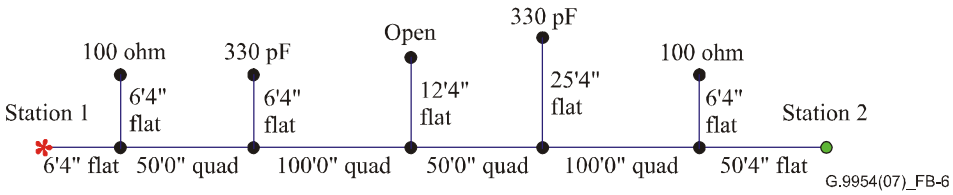

Figure B.6 – Test Loop Number 6

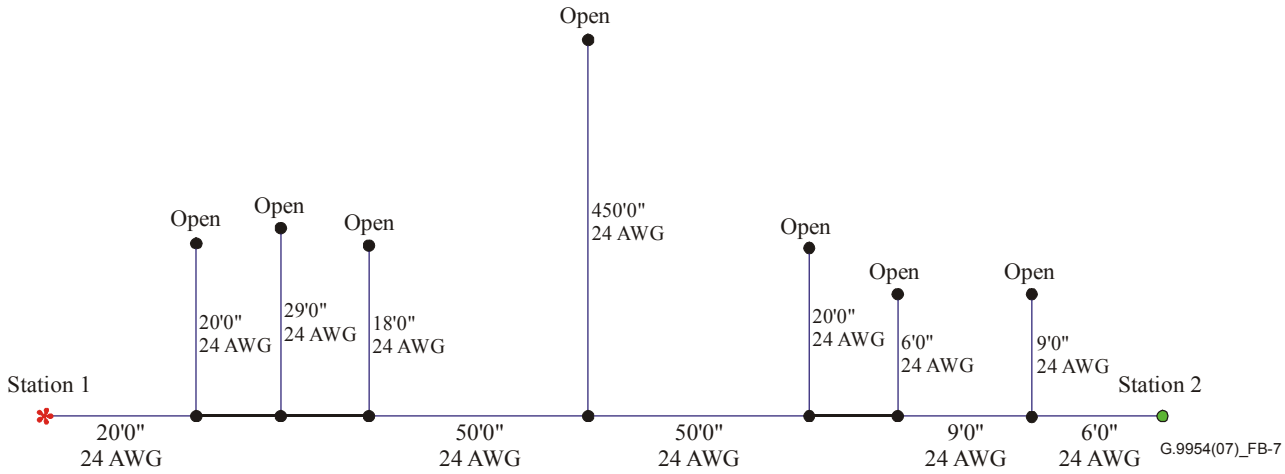

Figure B.7 – Test Loop Number 7

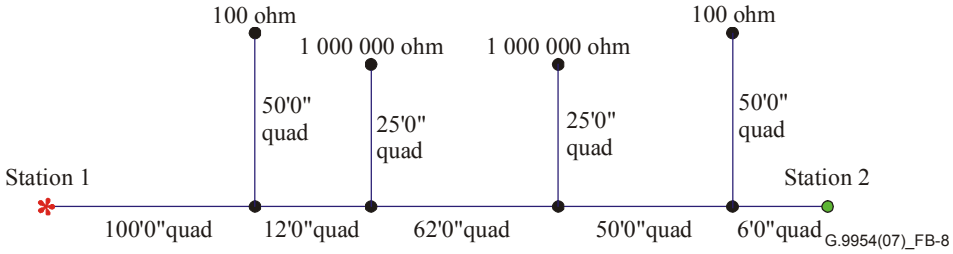

Figure B.8 – Test Loop Number 8

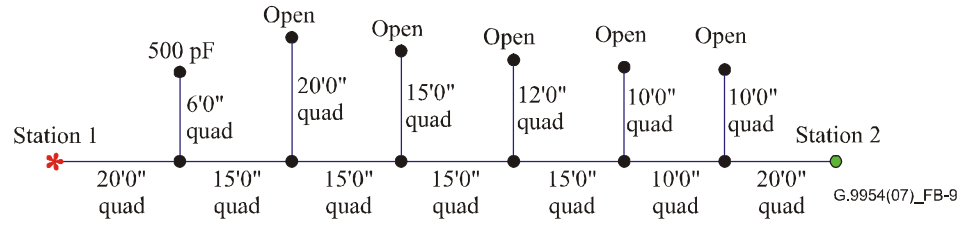

Figure B.9 – Test Loop Number 9

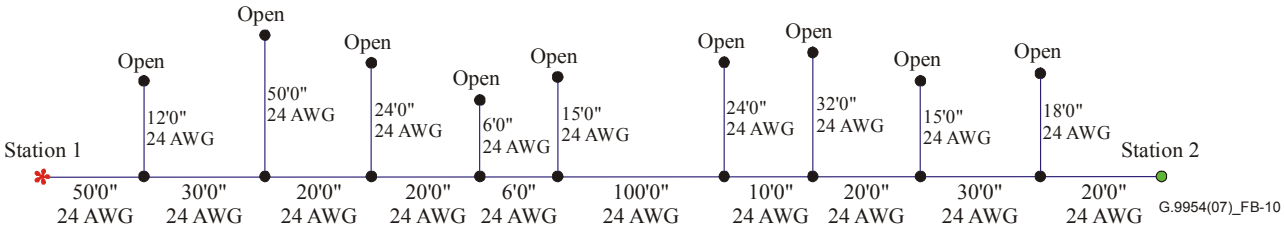

Figure B.10 – Test Loop Number 10

Appendix I

Convergence layers

(This appendix does not form an integral part of this Recommendation)

The convergence layer is a protocol-specific sublayer that maps various transport layer protocols into the native primitives of the LLC sublayer. The LLC sublayer provides a protocol-independent interface and a well-defined QoS framework. It is the responsibility of the convergence layer to translate the native protocol into this underlying framework.

This appendix describes the G.9954v2 convergence layer, its logical interfaces and general requirements for particular protocol-specific convergence layers. Since the logical interface between convergence and link layers are between protocol stack layers developed by the same vendor, there is no issue of interoperability between different vendor solutions. Consequently, the content of this appendix should be considered informative in nature and used only as a guideline for implementations.

I.1 Overview

The G.9954v2 protocol stack supports interfaces and bridging to external network protocols through the convergence layer. The protocol convergence sublayers available on a G.9954v2 device are advertised using the link-layer capability and status announcement protocol; see clause 11.6. By default, Ethernet and IP convergence layers are defined.

It is the responsibility of the protocol convergence layer to map data packets arriving from a particular interface onto the *flows* appropriate for the particular service. Flows defined for a particular convergence layer are set up by the convergence layer itself in an implementation-dependent way, possibly during initialization, on receipt of data from upper layers, on network admission or upon demand. The flow traffic and rate parameters for a flow may be also defined in an implementation-dependent way, perhaps by upper-layer protocols, or configured using management operations or configuration data held in non-volatile storage.

G.9954v2 convergence sublayers considered for the G.9954v2 protocol stack include the IEEE 802.3/Ethernet, IP protocols, USB and IEEE 1394. In addition, bridging interfaces to broadband access protocols, such as DOCSIS and wireless access protocols, such as IEEE 802.11 and IEEE 802.16, are envisioned, as are application-level convergence sublayers for applications for delivery of MPEG transport streams.

Protocol mapping and convergence at a well-defined level of the protocol stack enables a degree of synchronization between external and home protocols. Furthermore, given QoS defined in terms that are similar to those of the external network, this further supports the extension of QoS from external networks into the home network.

The convergence layer may perform the following functions:

- Interface to higher layer protocols and receives PDU from the upper layers.
- Signal the set-up of traffic flows and classifiers in local and peer MAC, link-layer and convergence layer entities.
- Classify upper-layer PDUs, using built-in knowledge of the protocols, and map the PDUs to underlying flows.
- Perform address bridging and translation functions.
- Perform any special PDU processing before passing them onto the Link/MAC layers (e.g., removal of payload header information).
- Send upper-layer PDUs to HNT Link/MAC layers.

- Receive PDUs transported by the HNT PHY/MAC layers and performs any protocol specific processing before delivery to upper protocol layers.
- Perform peer-to-peer convergence sublayer signalling.
- Perform data sampling and synchronization control.

No assumptions should be made as to the system partitioning of link and convergence layer functions as it is possible to implement both of these protocols both on-chip or in external host drivers.

I.2 Convergence-layer primitives

The following clause describes the convergence-layer interface to the lower layers of the G.9954v2 protocol stack. Since the details of the convergence layer to LLC interface are implementation dependent, this interface is described in terms of a set of primitives supported by the link-layer control service access point (LLC_SAP).

The following primitive types are defined:

- req (request) – Primitive used by the convergence sublayer to request a service from the LLC sublayer.
- cnf (confirm) – Primitive used by the LLC sublayer to confirm that a requested activity has been completed.
- ind (indication) – Primitive used by the LLC sublayer to notify the convergence sublayer of any specific service related activity.
- rsp (response) – Primitive used by the convergence sublayer to acknowledge the receipt of an indication primitive from the LLC sublayer.

The primitives and their relationships are illustrated below in Figure I.1.

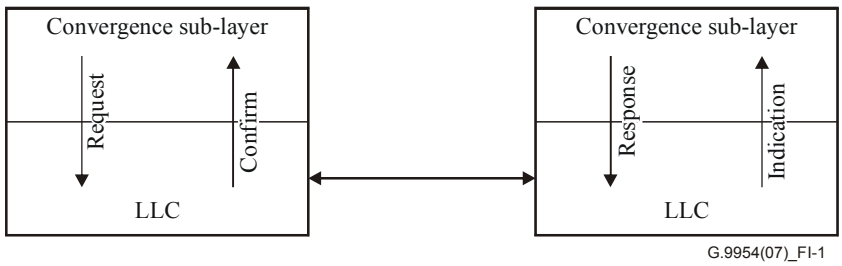

Figure I.1 – Service primitives

Figure I.2 illustrates the convergence layer to link layer interface.

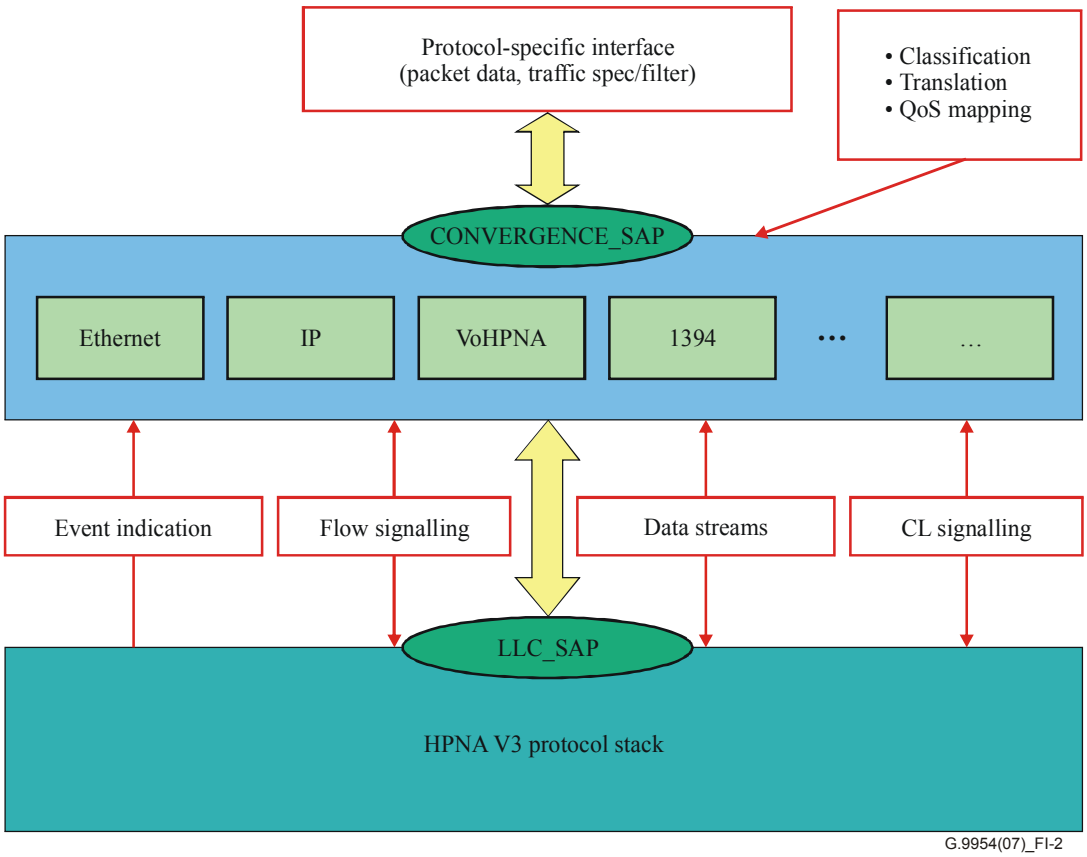

Figure I.2 – Convergence layer – Link-layer primitives

I.2.1 Flow signalling primitives

I.2.1.1 LLC_SETUP_FLOW (req, cnf, ind, rsp)

This primitive is used to set up a flow between a source and a single or multiple destinations on the network. It is protocol specific as to which event at the protocol level will cause the set-up of a flow and what are the flow characteristics.

The **request** primitive is used by the convergence layer to request the set-up of a flow with defined flow properties and traffic classifier specification (see clause 10.4). If the source of the flow is also the device requesting the flow set-up, the traffic classifier specification only has local significance. The **request** primitive is normally only generated at the source or destination of a flow although it is possible that it may be generated by the master.

The **indication** primitive is used to notify the convergence layer of the set-up of a flow. The flow properties and traffic classifier are passed to the convergence layer. The flow properties delivered to the convergence layer are after admission control and contain the offered QoS properties and assigned *Flow ID*. The **indication** primitive may be used to trigger signalling operations with the higher-layer protocols and to initialize, install or populate protocol-specific data-structures such as address translation and bridging tables.

The **response** primitive is used by the convergence layer to signal to the link layer of the status of the flow set-up request from the perspective of upper-layer protocols. It provides an opportunity for the upper-protocol layer to reject the flow set-up request or offered flow properties due to some protocol-specific consideration.

The **confirm** primitive is used to notify the requestor of the status of the **request** and to report back information concerning the flow, including the *Flow ID* and offered flow parameters. The actual (offered) flow parameters may vary from the original request due to resource limitations.

The parameters in Table I.1 are used in this primitive:

Table I.1 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
Flow properties	√	√	√	√
Traffic filter specification	√		√	
Status		√		√

where:

- Flow properties – Properties of flow to be set up (see QoS spec). The convergence sublayer participating in the interface is specified in the flow properties parameter as is the Flow ID assigned to the flow.
- Traffic filter specification – Filter specification as defined in QoS spec. Action specification for the filter is ADD.
- Status – Status of set-up request in confirm primitive type.

For further information, see clause 11.16.

I.2.1.2 LLC_MODIFY_FLOW (req, cnf, ind, rsp)

The **request** primitive is used to request the modification of a flow's properties or the associated traffic classifier filters. The flow is identified by *Flow ID* in the flow properties parameter.

The **indication** primitive is used to notify the convergence layer of the requested modifications. Flow properties are after admission control. The traffic classifier filter specification may indicate an add, modify or delete action. This primitive may trigger operations within the upper-layer protocol and may cause modifications of internal data structures.

The **response** primitive allows the convergence sublayer to accept or reject the modification request.

The **confirm** primitive is used to inform the convergence layer of the result of the request.

The parameters in Table I.2 are used in this primitive:

Table I.2 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
Flow properties	√	√	√	√
Traffic filter specification	√		√	
Status		√		√

where:

- Flow properties – Properties of flow to be modified (see QoS section). The Flow ID of the flow to be modified is encoded in the flow properties.
- Traffic filter specification – Specification of the filter used to map the flow. Actions defined for filter specification includes add, modify and delete a filter.
- Status – Status of modify request in confirm primitive type.

For further information, see clause 11.16.3.2.

I.2.1.3 LLC_TEARDOWN_FLOW (req, cnf, ind, rsp)

This primitive is used to tear down an existing flow identified by *Flow ID*.

The parameters in Table I.3 are used in this primitive:

Table I.3 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
Source MAC address	√	√	√	√
Destination MAC address	√	√	√	√
Flow ID	√	√	√	√
Status		√		√

where:

- Source MAC address – Address of device at source of the flow;
- Destination MAC address – Address of device at destination of flow;
- Flow – Identifies the flow to be torn down;
- Status – Status of modify request in confirm primitive type.

For further information, see clause 11.16.3.3.

I.2.2 Data stream primitives

I.2.2.1 LLC_DATA (req, cnf, ind)

This primitive is used to send packet data between peer convergence sublayer entities.

The **request** primitive is used to request the transfer of a protocol layer packet or convergence layer information to a peer convergence layer entity over a particular flow (identified by *Flow ID*) or using a particular *priority* (if operating in master-less mode).

The **indication** primitive is used to notify the convergence sublayer of the arrival of the convergence layer information. The notification includes the timestamp at the time of reception measured with reference to a common point in the transmission frame. The point defined is immediately following the SA in the frame in which the frame arrived.

The **confirm** primitive is used to notify the completion of the data transfer request. Parameters of the primitive include the *Status* of the request and the timestamp when the data was actually transmitted on the media.

The parameters in Table I.4 are used in this primitive:

Table I.4 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
FC	√		√	
DA	√			
SA			√	
EtherType	√		√	
MAC aggregation	√			
Payload length	√		√	
Payload	√		√	
FCS	√		√	
TX Timestamp		√		
RX Timestamp			√	
Status		√	√	

where:

- FC – is the Frame Control and includes the frame type and frame sub-type, priority/flow ID and PE;
- DA – is the destination address of the SDU;
- SA – is the source address of the SDU;
- EtherType – is the Ethernet type defined for the frame;
- MAC aggregation – indicates whether the packet should be aggregated by the MAC-layer with other packets belonging to the same priority or flow. This parameter is used to indicate either no aggregation should be performed (a value of 0), or the packet is a candidate for aggregation (a value of 1);
- Payload – is the payload data to be delivered by the protocol stack. This payload may come from the link layer or protocol convergence layers of the protocol stack. The payload frame format is not necessarily an Ethernet frame and may come from any convergence layer as indicated by the FT parameter;
- Payload length – is the length of the payload data;
- FCS – is an optional 32-bit frame checksum that may be supplied with the frame;
- TX timestamp – Timestamp of actual transmission. Time is specified in units of 2^{-13} ms;
- RX timestamp – Timestamp of actual reception. Time is specified in units of 2^{-13} ms;
- Status – is the data TX/RX status.

I.2.3 Event indication primitives

I.2.3.1 LLC_MAC_CYCLE (ind)

This primitive is used to notify the convergence layer of MAC cycle timing information and of media access planning (bandwidth allocations). The primitive provides information that enables convergence layers to synchronize upper-protocol layers with the G.9954v2 MAC cycle, synchronize sampling rates and use media resource allocation information for protocol-level signalling.

This primitive is intended for use in convergence layers that interface with upper-layer protocols that are synchronous in nature or support isochronous services and require some degree of synchronization. Examples of such protocols include IEEE 1394, USB, etc.

The parameters in Table I.5 are used in this primitive:

Table I.5 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
MAP			√	
Scheduled MAC cycle start time			√	
Actual MAC cycle start time			√	
Indication time			√	

where:

- MAP – is the MAP control frame;
- Scheduled MAC cycle start time – is the time when the MAC cycle was scheduled to start;
- Actual MAC cycle start time – is the time when the MAC cycle actually started. This may differ from Scheduled MAC cycle start time if jitter was introduced into the MAC cycle due to AMAC interference;
- Indication time – is the time when the indication was actually delivered to the convergence layer.

For a further description of the parameters used in the LLC_MAC_CYCLE primitive, see the description of the MAP in clause 11.13.1.

I.2.3.2 LLC_NETWORK_ENTRY (ind)

This primitive is used to notify the convergence layer of the registration of the device with the master and of the assigned Device ID.

The parameters in Table I.6 are used in this primitive:

Table I.6 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
Device ID			√	
802.3 MAC address			√	

where:

- Device ID – is the master-assigned device ID;
- 802.3 MAC address – is the 48-bit IEEE MAC address assigned to the node.

I.2.3.3 LLC_NETWORK_EXIT (ind)

This primitive is used to notify the convergence layer of the de-registration of a device with the master.

The parameter in Table I.7 is used in this primitive:

Table I.7 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
Device ID			√	

where:

- Device ID – is the master-assigned device ID

I.2.3.4 LLC_SYNC_EVENT (ind)

This primitive is used to notify the convergence layer of the synchronization of a G.9954v2 device with a master generated MAC cycle. (See Table I.8.)

Table I.8 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
Sync Event			√	

I.2.3.5 LLC_SYNC_LOSS_EVENT (ind)

This primitive is used to notify the convergence layer of the loss of synchronization with the master-generated MAC cycle. (See Table I.9.)

Table I.9 – Primitive parameters

Parameter	Request	Confirm	Indication	Response
Sync Loss Event			√	

I.3 Convergence layer architecture

The internal structure of the convergence layer component according to the model described above is illustrated in Figure I.3:

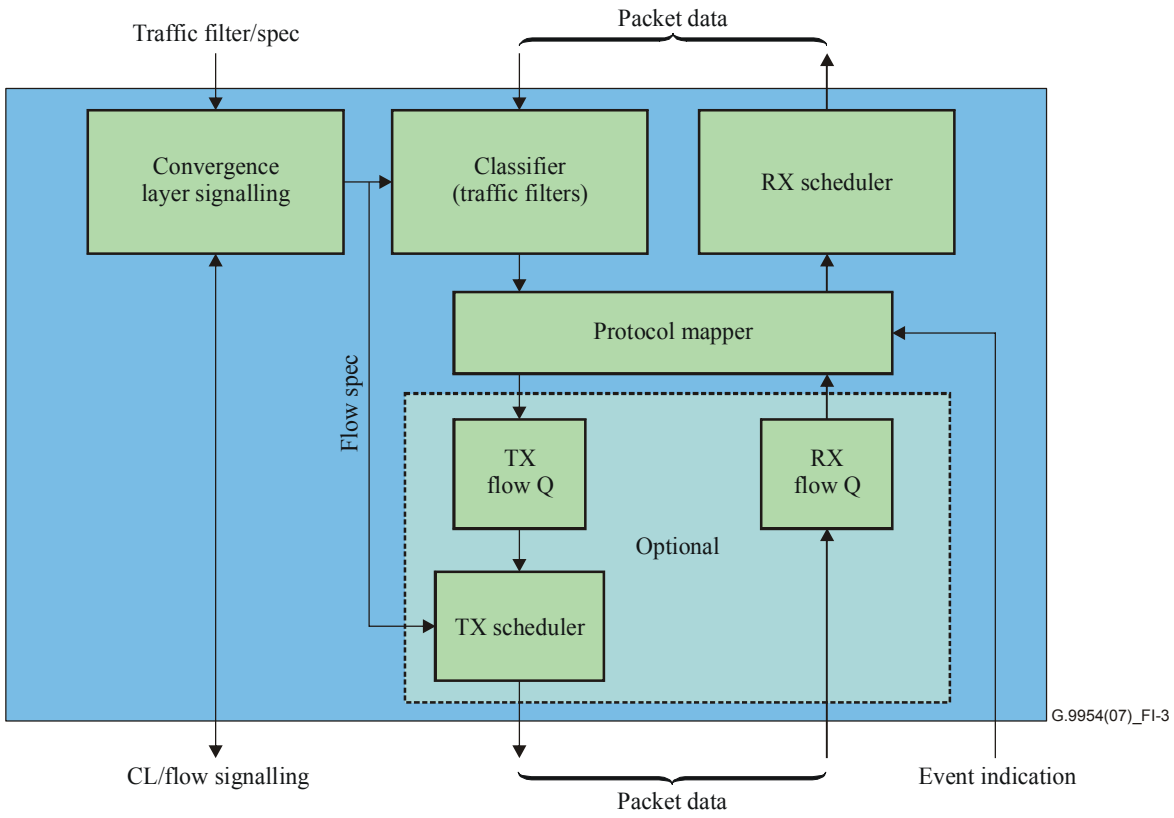

Figure I.3 – Convergence layer architecture

The components within the Convergence Layer block are responsible for the following functions:

- **Flow/Convergence layer signalling** – This component is responsible for performing flow set-up/tear-down signalling and peer convergence-sublayer signalling. It responds to flow set-up requests, originating from upper-protocol layers or from within the convergence layer itself, and manages convergence layer peer-level signalling. It communicates with the classifier in order to define traffic-filter specifications and with the TX scheduler in order to define traffic-rate specification.
- **Classifier** – The classifier is responsible for mapping incoming packet data to a flow using the traffic filter specification defined by the convergence layer signalling component.
- **Protocol mapper** – This component is an optional entity and may perform protocol-specific mapping functions.
- **Flow queues** – Flow queues are optional data-structures used to hold packets while they are waiting to be scheduled by the appropriate scheduler component. Flow queues on the TX side may be token buckets that are used for traffic shaping.
- **TX scheduler** – The TX scheduler is responsible for selecting packets from the TX flow queue and delivering it to the underlying network device. It may perform traffic-shaping functions. This function may be trivial in those implementations that do not require flow queues and shaping in the convergence layer.

- **RX scheduler** – The RX scheduler is responsible for delivering packets received from the network interface to the upper-protocol layers. Packets arriving from the network may be passed through the protocol mapper in order to perform the inverse protocol mapping function.

Convergence sublayers may need to maintain several data-structures in order to implement sublayer functions. Examples of such data-structures include traffic queues, used to shape traffic-according to rate parameters, buffers used to balance the differences in cycle frequencies between upper and lower-level protocols, address mapping tables used for bridging between networks, etc.

Since memory demands for certain convergence sublayers may be significant, it is possible that convergence sublayers may be implemented at the host driver level and not on-chip.

I.4 Flow set-up triggering

Flow set-up may be triggered by the following events:

- Registration of device with master;
- Arrival of an upper-layer service data unit (SDU);
- Upon request from upper-protocol layer;
- Management operations.

In the first case, when flow set-up is triggered by the registration process, the operation may be initiated in either the master or the endpoints. In both cases, the assumption is that the master and/or endpoint knows which flows need to be provisioned after registration and what are the flows' properties. This information may be built-in to the convergence layer or it may be attained from configuration parameters.

In the second case, when a flow is set-up upon arrival of an SDU, the assumption is that the convergence layer has traffic filters installed that allow it to classify an SDU upon arrival and to identify the properties of the flow that needs to be set up to handle traffic of this type. It then should initiate the flow set-up using the flow specification attached to the filter. The filters and their association with the flow property descriptor may be built into the convergence layer or it may be installed in a configuration data.

Upper-layer protocols may also initiate the set-up of a flow with specific properties. For example, applications may initiate flow set-up in response to handling RSVP or equivalent DOCSIS signalling messages.

Management operations, whether initiated from the local or remote sides of the device, may initiate the set-up of a flow with well-defined flow properties.

I.5 Classification

Classification is the process by which upper-layer PDUs are mapped to G.9954v2 flows. The classification process is protocol-specific and may include a set of classification rules that are processed in a particular priority ordering.

The classification rules that apply to a flow are part of the flow description. This model is consistent with the RSVP model that defines a *flow descriptor* as the composite of a *flow specification* (the traffic-related component) and a *filter specification.*

For a description of traffic classification filters, see clause 10.6.

I.6 Convergence layer interfaces to upper protocol layers

Each convergence sublayer provides its own protocol-specific interface to the upper layer. All interfaces provide a primitive (or primitives) for transporting and receiving the upper-layer protocol data units. The primitives in this interface are of the form:

- XXX_CSL_DATA.req – used to request the transmission of data.
- XXX_CSL_DATA.cnf – used to notify the upper layer of the status of the transmission request.
- XXX_CSL_DATA.ind – used to notify the upper layer XXX of the arrival of data.

I.7 Protocol-specific convergence layers

I.7.1 IP convergence

The IP convergence layer processing may use RSVP protocol packet filtering rules. These rules specify classification according to the following criteria:

- IP type of service (TOS) field;
- IP protocol number;
- IP source address;
- IP destination address;
- IP protocol source port number;
- IP protocol destination port number.

For further details on IP traffic classifiers, see clause 11.16.

I.7.2 Ethernet convergence

Ethernet convergence layer processing performs classification of PDUs based on the following criteria:

- Ethernet destination MAC address;
- Ethernet source MAC address;
- Ethernet type and 802.2 SAP;
- VLAN (802.1P) priority;
- VLAN (802.1Q) ID.

The following special Ethernet types in Table I.10 are recognized by the Ethernet convergence layer and result in PDUs being routed to the appropriate convergence layer component:

Table I.10 – Routed Ethernet types

Ethernet type	Description
0x0800	IP packet routed to IP convergence layer
0x0806	ARP packet routed to IP convergence layer
0x86DD	Ipv6 packet routed to Ipv6 convergence layer

For further details on Ethernet traffic classification filters, see clause 11.16.

Appendix II

Media Independent Interface (MII) Recommendations

(This appendix does not form an integral part of this Recommendation)

The media-independent interface (MII) as specified in IEEE Std 802.3-1998, clause 22, is a common interface found on many pieces of existing networking silicon. While there are many possible implementations for interfacing an G.9954v2 PHY to an existing Ethernet MAC via the MII, the following guidelines provide a reference for designing a PHY that is completely compatible with silicon complying with clause 22 of the IEEE standard.

Flow control is the major issue in using the MII interface. The MII specification calls for interface clocks to be fixed frequency of 25 MHz ±100 ppm, resulting in a data transfer rate of 100 Mbit/s. G.9954v2 provides for a wide range of bit rates ranging from 4 Mbit/s to 128 Mbit/s. For the PHY-to-MAC (receive) direction there is a rate mismatch between the PHY and MAC over this interface. This may result in some packet loss in the unlikely event that transmissions on the wire are all at full rate. In this case a receiver should limit the maximum size of its frame receive buffer in order to force transmitters to transmit shorter frames and to guarantee that the effective throughput does not exceed the MII 100 Mbit/s limit. For the MAC-to-PHY (transmit) direction, the PHY needs a method to hold off the MAC while previous data is being modulated and sent out on the wire.

This flow control should use the CRS signal in a "false carrier sense" mode to hold off the MAC transmitter with the deference mechanism. The details of this signalling are described below.

II.1 MII overview

See Figure II.1.

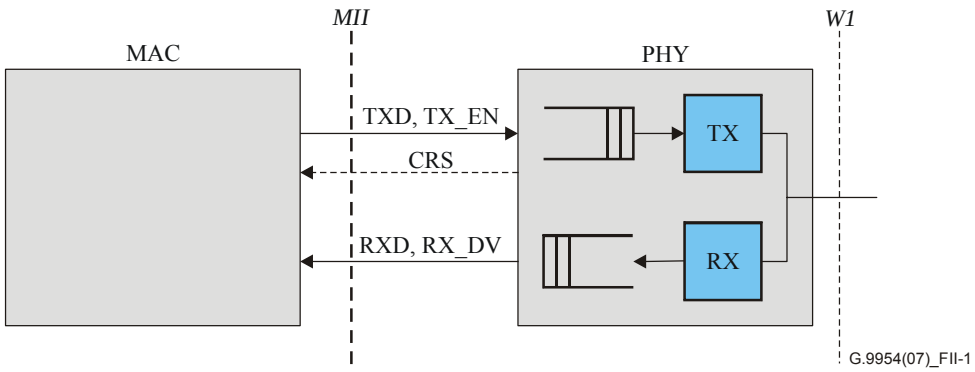

Figure II.1– MII interface

II.1.1 MII data path

The MAC/PHY interface consists of the 16 signals shown in Table II.1.

Table II.1 – MAC/PHY signals

Signal	Direction relative to PHY	Description
TX_EN	In	Transmit framing signal
TXD[3:0]	In	Four bits per clock of transmit data
TX_ER	In	Transmit error
TX_CLK	Out	Transmit clock (2.5 MHz or 25 MHz)
CRS	Out	Carrier sense
RX_DV	Out	Receive data valid
RXD[3:0]	Out	Four bits per clock of receive data
RX_CLK	Out	Receive clock
RX_ER	Out	Receive error
COL	Out	Collision

II.1.2 Transmission without collision

Shown in Figure II.2 is an example transfer of a packet from MAC to PHY.

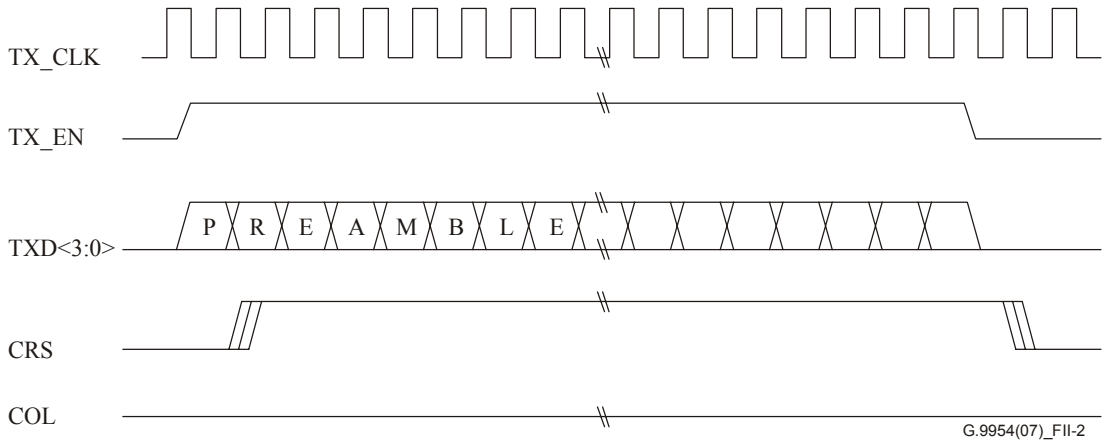

Figure II.2 – MAC-to-PHY packet transfer

II.1.3 Reception without error

Shown in Figure II.3 is an example of transfer of a packet from PHY to MAC.

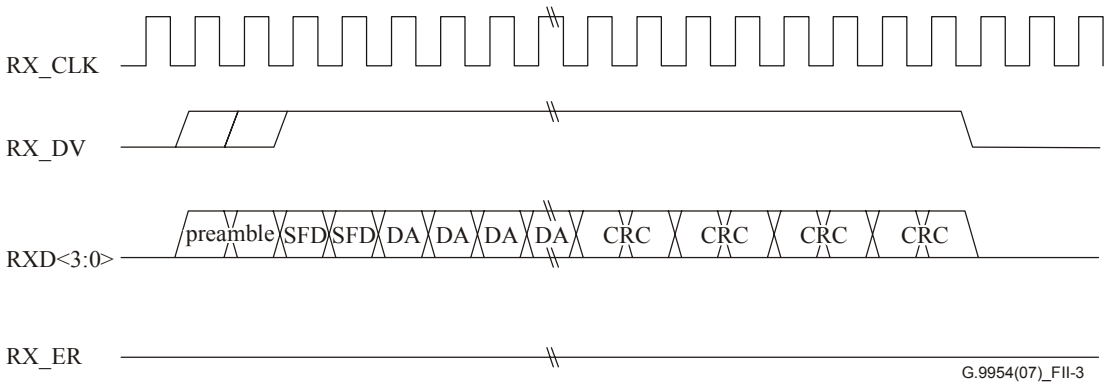

Figure II.3 – PHY-to-MAC packet transfer

II.1.4 MII management signals

There are two additional signals specified for management: MDIO (Management Data I/O) and MDC (Management Data Clock). Many, if not all, existing MACs will have MDIO/MDC interface pins, but these are vital only for management purposes in the case there are registers in the PHY that need to be accessed by the host. There is generally no requirement for MII-based management functions and, consequently, it is not necessary for HNT devices to implement MDIO/MDC. However, if there are registers in the PHY, then the protocol and signalling defined by MDIO/MDC in IEEE Std 802.3 clause 22 should be used.

II.2 G.9954v2 signalling recommendations

The following description references clause 22 of IEEE Std 802.3, Media-Independent interface specification, used in the 100 Mbit/s half-duplex mode. To account for physical layer differences between G.9954v2 and 100BASE-T Ethernet, the PHY is assumed to have an adaptation or reconciliation layer which handles all timing and data formatting issues. The MII is used as a data channel that transfers data back and forth in units of packets, flow controlled by the carrier sense (CRS) signal.

II.2.1 TX_CLK and RX_CLK

The PHY generates a stable, continuous 25-MHz square wave which is supplied to TX_CLK and RX_CLK. No "gapping" or other variable clocking method is used.

The frequency offset of the generated clock should be controlled to enable the use of all standard MAC implementations.

II.2.2 TX_ER and RX_ER

TX_ER is normally used in situations where the transmitter above the PHY has detected an error condition, but the transmission is currently in process. TX_ER indicates to a PHY that the current packet is errored and should be corrupted on the wire to ensure a receiver does not accept this as a valid packet. Normally, this is a condition that only applies to repeaters. Repeaters do not perform error checking on the complete packet. In the case of a DTE (sometimes referred to as a 'node'), the transmitter usually guarantees the frame to be without errors and there is no need for the TX_ER signal. Since G.9954v2 is based on bus topology wiring plants, no repeater is specified and use of the TX_ER signal is not anticipated. However, G.9954v2 PHYs may choose to respond to the TX_ER signal.

RX_ER is normally used in situations where the PHY detects an error in the receive stream as a result of decoding. G.9954v2 PHYs may assert this signal in the event that such an error is detected.

II.2.3 TX_EN

TX_EN from the MAC provides the framing for the Ethernet packet. TX_EN active indicates to the PHY that data on TXD[3:0] should be sampled using TX_CLK.

II.2.4 TXD[3:0]

TXD[3:0] contains the data to be transmitted and transitions synchronously with respect to TX_CLK. TXD[0] is the least significant bit. It is generally assumed that the data will contain a properly formatted Ethernet frame. That is, the first bits on TXD[3:0] correspond to the preamble, followed by SFD and the rest of the Ethernet frame (DA, SA, length/type, data, CRC).

The PHY strips the 802.3 preamble on MAC-to-PHY transfers.

II.2.5 RX_DV

RX_DV is asserted by the PHY to indicate that the PHY has decoded receive data to present to the MAC.

II.2.6 RXD[3:0]

RXD[3:0] contains the data recovered from the medium by the PHY and transitions synchronously with respect to RX_CLK; RXD[0] is the least significant bit. It is assumed that the PHY has properly formatted the frame such that the MAC will be presented with expected preamble plus SFD.

The TXD and RXD data paths are full duplex, although we use the MII interface in half-duplex mode. RX_DV is never asserted at the same time as TX_EN.

II.2.7 CRS

On transmit (see Figure II.4), the PHY asserts CRS some time after TX_EN comes true, and drops CRS after TX_EN becomes false AND when the PHY is ready to receive another packet. When CRS falls, the MAC times out an IFG (0.96 μs) and may assert TX_EN again if there is another packet to send.

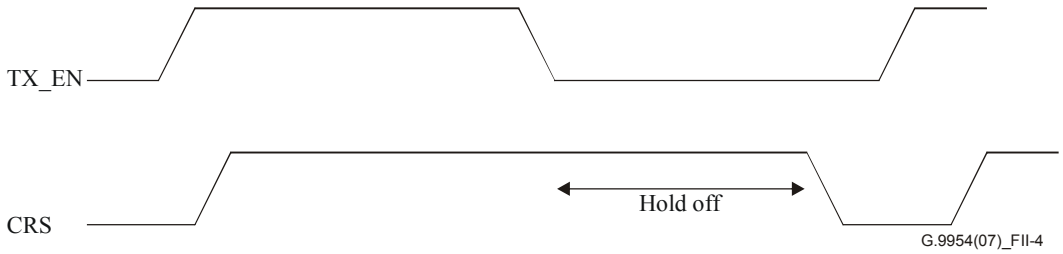

Figure II.4 – TX direction

This differs from nominal behaviour of CRS in that CRS can extend past the end of the packet by an arbitrary amount of time, while the PHY is gaining access to the channel and transmitting the packet. See Figure 6-2.

MACs in 100 Mbit/s mode do not use a jabber timeout, so there is no timing restriction on how long CRS can be asserted (other than sanity timeouts the PHY may implement).

Transmissions can "cut through" or begin to be modulated onto the wire as soon as the transfer begins, as the MII will fill the PHY buffer faster than data needs to be made available to the modulator. When a packet arrives at the PHY, it attempts to gain access to the channel using the

priority CSMA/CA algorithm described in clause 8.3. This may not happen before the entire packet is transferred across the MII interface, so the PHY will need to buffer at least one MTU to perform this rate adaptation.

On receive (see Figure II.5), when the PHY anticipates that it will have a packet demodulated, it raises CRS to seize the half-duplex MII channel, waits a short time (an IFG), then possibly defers to TX_EN (which may just have been asserted) plus an IFG, and then raises RX_DV to transfer the packet. At the end of the transfer, it drops CRS unless the transmit buffer is full or there is another receive packet ready to transfer. (See Figure 6-3, where one receive transfer is followed by a second which defers to TX_EN.)

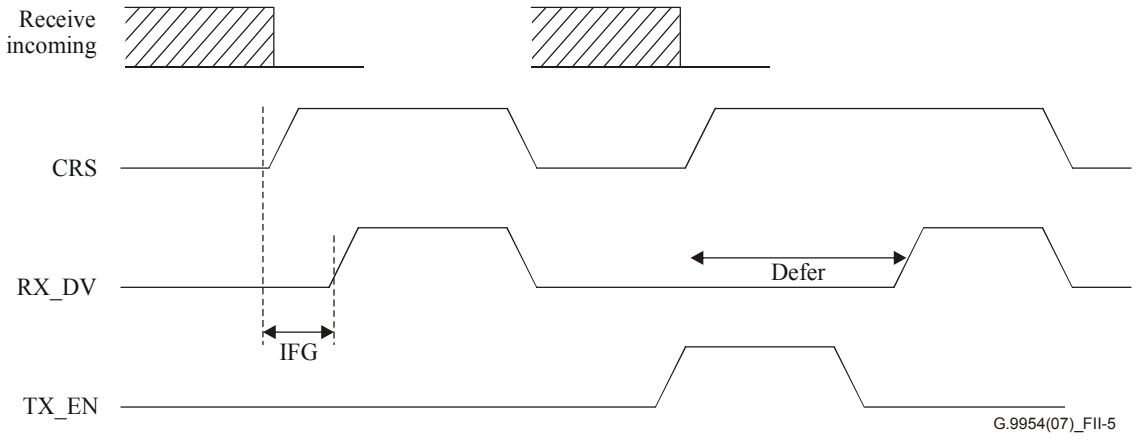

Figure II.5 – RX direction

RX_DV should not be asserted until the PHY is assured that the entire packet will be ready to transfer at the 100-Mbit/s rate. This implies some buffering on the receive side to do this rate adaptation. Once the MII burst transfer starts, new data can start filling the buffer, as the MII transfer is guaranteed to stay ahead of the data coming off the wire.

Receive direction transfers need to have priority over transmit direction to ensure that the buffer empties faster than packets arrive off the wire. The longest that the receiver needs to wait is the time to transfer one TX frame plus an IFG or approximately 134 µs. However, minimum size frames can arrive at a peak rate of one every 65 µs, so the receive side buffer has to accommodate multiple frames (but only little more than one MTU of data).

II.2.8 COL

COL is not used. The way the PHY manages the MII interface, collisions between receive and transmit direction transfers do not occur.

II.3 The "off-chip" G.9954v2 convergence layer

Interfaces to external MACs in G.9954v2 relies on the implementation of protocol-specific convergence layers. The separation of the convergence layer from the G.9954v2 link and MAC sublayer facilitates the tailoring of external protocols and interface implementations to G.9954v2. Furthermore, it lends itself to an off-chip solution where the convergence layer logic resides in host driver software. In such an environment, where memory requirements may be more relaxed, the convergence layer may be used to hide the complexity of the interface between an external MACs and the G.9954v2 device.

The following clauses describe the "off-chip" convergence layer architecture and how it can be transparently embedded in a software driver environment based on the NDIS or similar architecture. A discussion of MII interface implementation issues follows.

In configurations, where the complexity of the interface is limited or standard software drivers are used, convergence layer functions should be performed in an "intermediate software driver", running in the host operating system at a level between the host's "standard software driver" and the hardware interface. In such a configuration, the "intermediate software driver" should be made responsible for performing packet buffering and "traffic-shaping" in order to guarantee that packets are delivered to the hardware at a data rate that does not exceed the traffic specification of the active flows.

The architectural model that has the G.9954v2 convergence layer running off-chip in the "intermediate software driver" is described in Figure II.6.

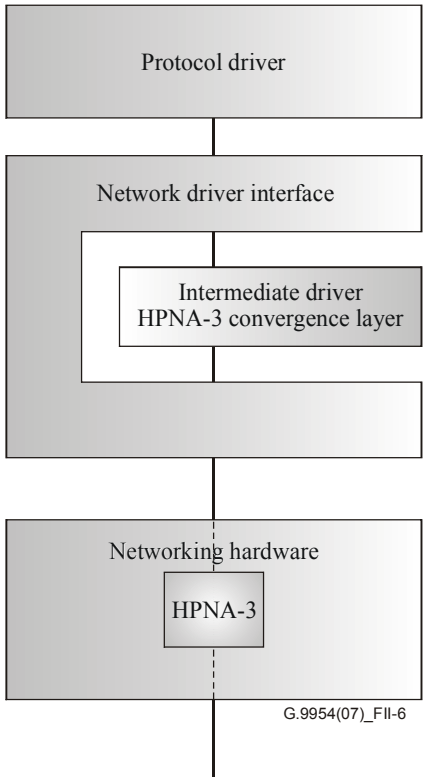

Figure II.6 – "Off-chip" convergence layer

This model assumes the existence of a network driver interface that is located between the protocol driver (e.g., 802.3 driver) and the actual networking hardware. It also assumes that there is a way to interface the intermediate software driver into the network driver interface in a transparent manner such that all packets that reach the network driver interface from the protocol driver or networking hardware are diverted through the intermediate driver.

The intermediate driver model is convenient for performing the following kinds of functions:

- Protocol translation – Map packets between protocol formats. May include bridging and address translation tables, etc.
- Packet filtering – A traffic shaper and/or scheduler may be used to buffer incoming packets and reorder their delivery to the underlying networking hardware.

Using this model, intelligence in the intermediate software driver allows the underlying interface to the G.9954v2 chip to be simple and standard, such as one based on the MII interface. Packets delivered to the MII interface can be safely blocked if no more memory resources exist since traffic shaping algorithms guarantee that data will not be delivered at a rate that is greater than the negotiated rate of the active flows.

The network driver interface specification (NDIS) driver model conforms to the above architecture.

Appendix III

End-to-end architecture

(This appendix does not form an integral part of this Recommendation)

III.1 G.9954v2-to-G.9954v2 protocol stack

Figure III.1 shows an end-to-end protocol stack involving two interconnected G.9954v2 devices. Each G.9954v2 device has a 48-bit MAC address. Each protocol layer exchanges protocol messages over a virtual link with HNT PHYs being connected physically over a phone-wire or cable media network.

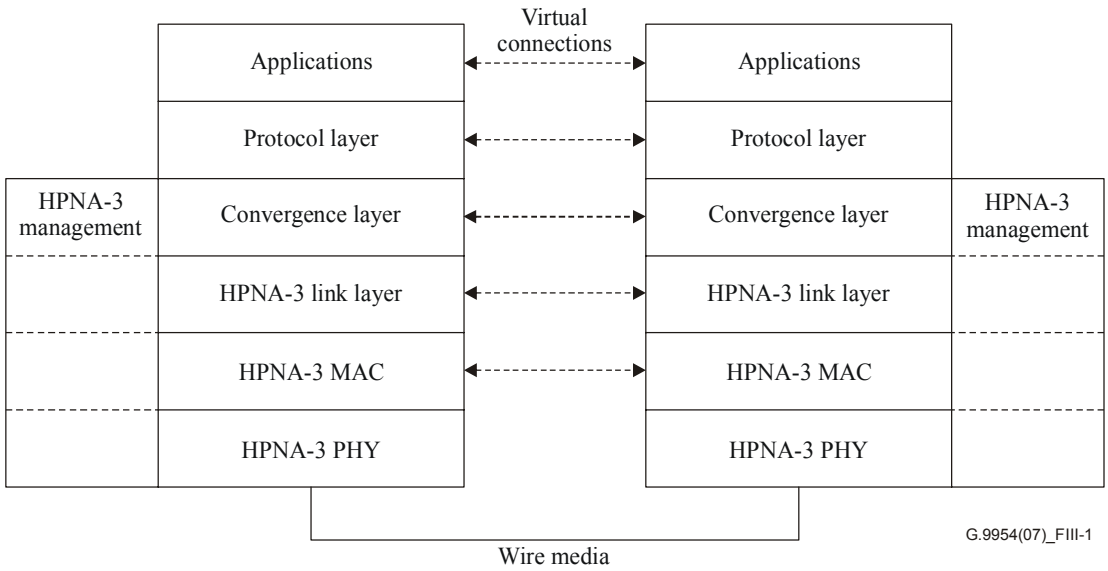

Figure III.1 – Communicating G.9954v2 protocol stacks

III.2 Ethernet-HNT interface

Ethernet is the natural protocol for transport over a HNT network. The HNT frame format is an extension to the Ethernet frame format and includes the entire Ethernet PDU within the frame.

G.9954v2 may interface with the Ethernet protocol in the following configurations:

- Ethernet PHY (MII Interface);
- Ethernet-HNT Bridge (MII Interface);
- Integrated Ethernet MAC-PHY (NIC card PCI or similar).

In the first configuration, G.9954v2 presents an MII interface and masquerades as an Ethernet PHY. This supports a glue-less connection to an external Ethernet MAC chip as illustrated in Figure III.2.

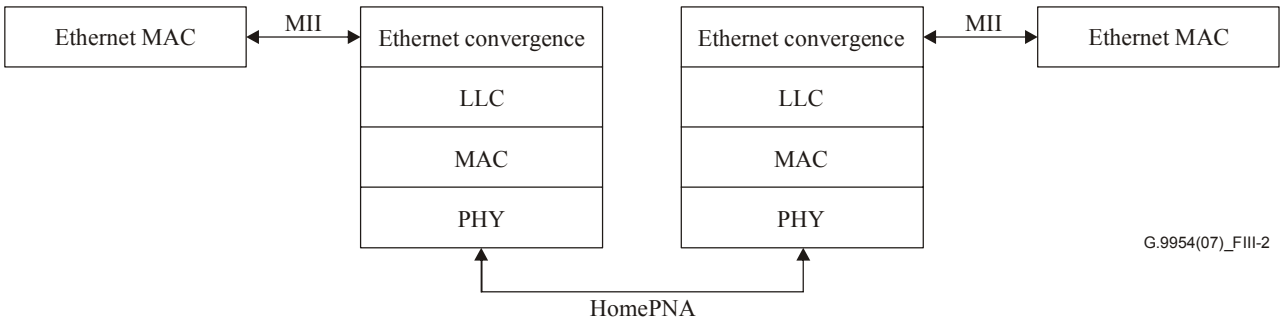

Figure III.2 – Ethernet PHY emulation

In another configuration, G.9954v2 provides an MII interface to an on-chip Ethernet MAC bridge. This interface is suitable for connecting to an Ethernet PHY in order to build an Ethernet-HNT bridge, as shown in Figure III.3.

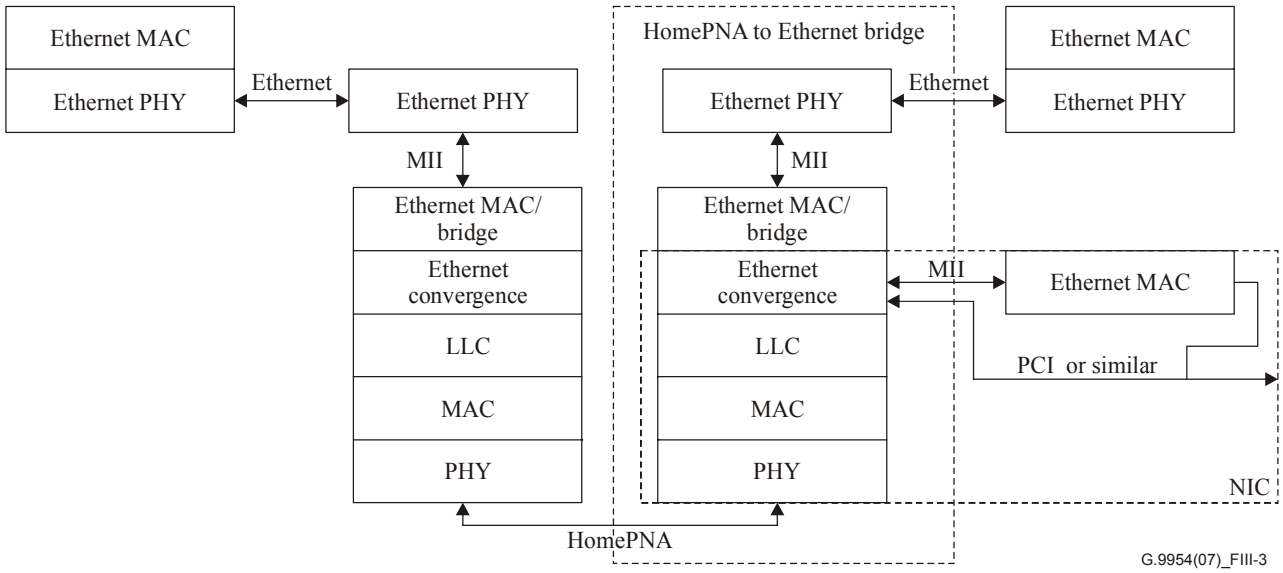

Figure III.3 – Ethernet-HNT bridge and NIC applications

III.3 USB-to-G.9954v2 protocol stack

A USB-to-G.9954v2 adapter (dongle) (see Figure III.4) is a USB device that provides a G.9954v2 connection to the host system. In this sense, it provides the same capability as a network interface card (NIC) except that the host PC connects to the network using the USB serial bus rather than the PCI bus.

USB is different from network protocols, such as Ethernet or IEEE 1394 in the sense that it is not an end-to-end network protocol but rather a bus protocol used to transfer data and control information from a host to the USB device. Data transfers, once they have arrived at the USB device, are removed from their USB wrappers, reconstructed into packets and transported over the HNT network. The USB wrappers themselves are discarded at the USB device endpoint.

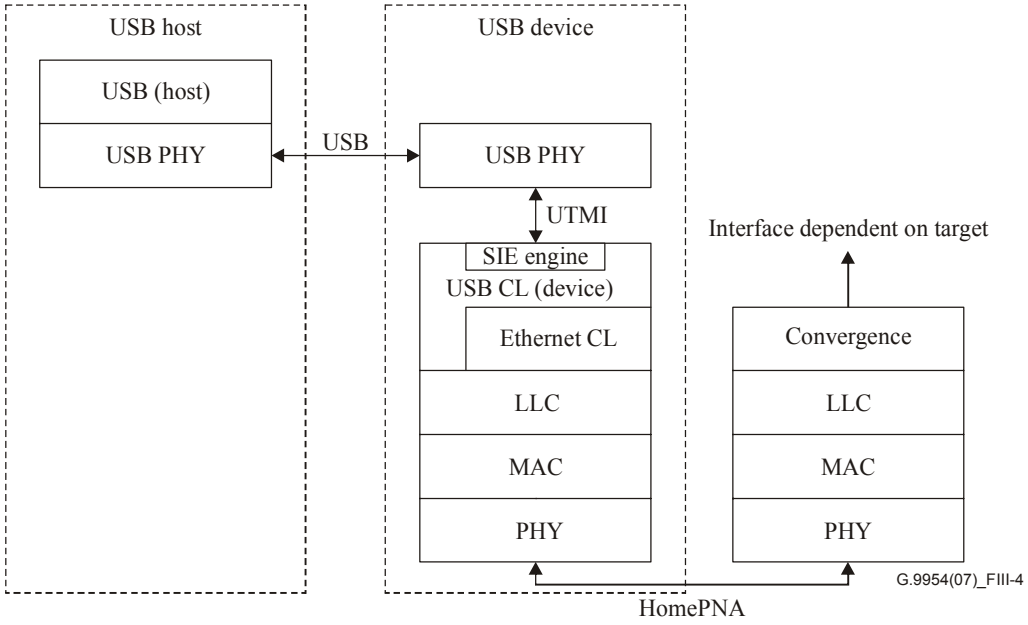

Figure III.4 – USB-to-G.9954v2 protocol adapter

III.4 IEEE 1394-to-G.9954v2 protocol stack

Two architectures incorporating IEEE 1394 and G.9954v2 are considered:

- IEEE 1394 over G.9954v2;
- IEEE 1394 – G.9954v2 bridge.

In the first architecture, the G.9954v2 device presents an IEEE 1394 link layer Interface to the IEEE 1394 protocol stack allowing IEEE 1394 applications to run over G.9954v2 in a transparent manner, as if they were running over an actual IEEE 1394 link and PHY layer. This implies that the 1394 convergence layer implements the standard IEEE 1394 link-layer primitives and maps these primitives to G.9954v2 link-layer functions. This is illustrated in Figure III.5.

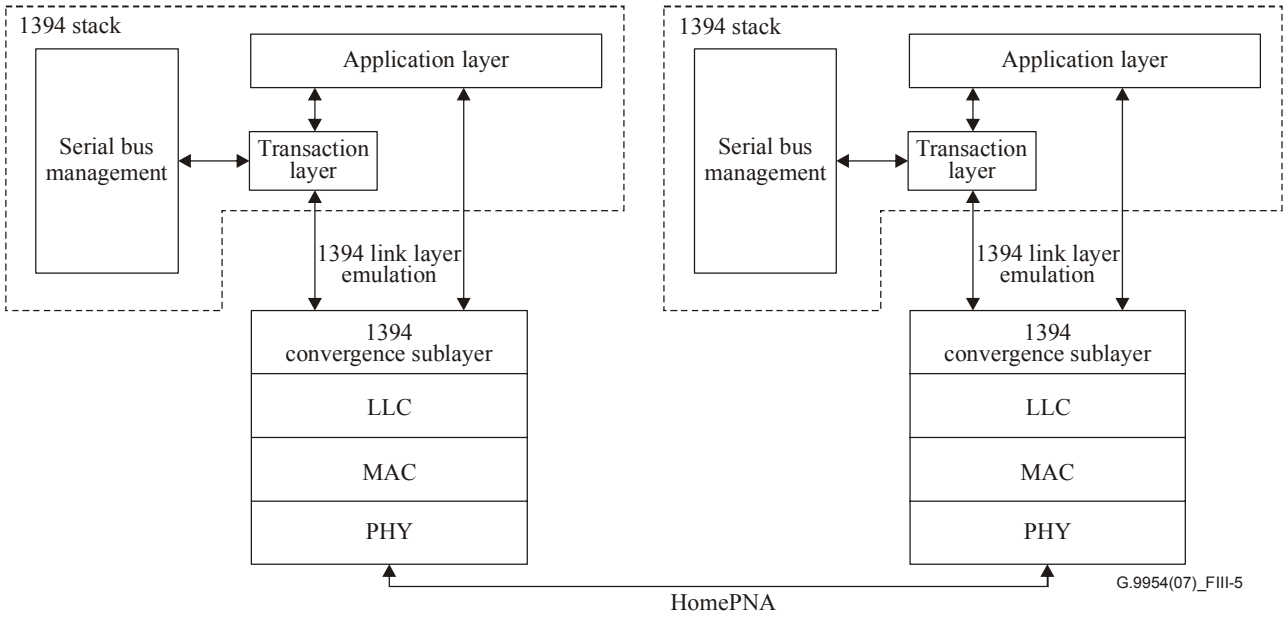

Figure III.5 – Transparent IEEE 1394 over G.9954v2

The second architecture is used to interconnect an IEEE 1394 bus with the G.9954v2 network using the P.1394.1 standard [b-IEEE P1394.1]. In this configuration, the G.9954v2 convergence layer includes IEEE 1394 bridging functions for asynchronous and isochronous data in addition to the IEEE 1394 convergence layer described above. This is illustrated in Figure III.6.

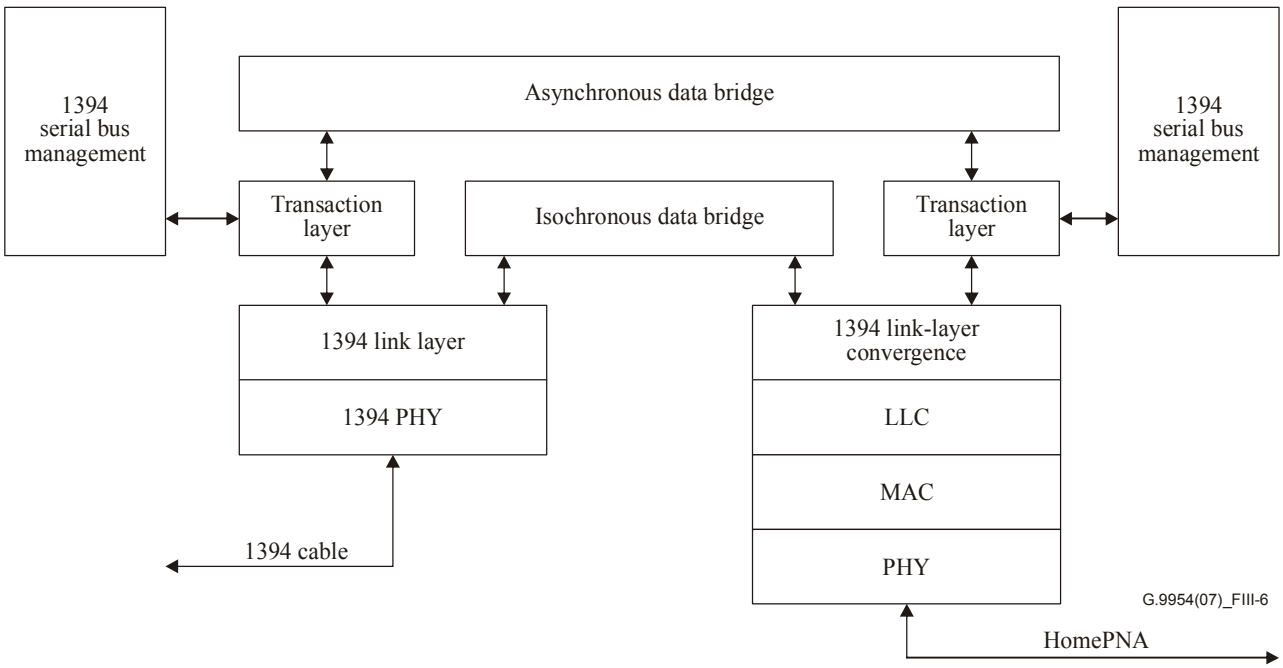

Figure III.6 – IEEE 1394 to G.9954v2 bridge

The details of this protocol bridge are for further study.

III.5 DOCSIS to G.9954v2 protocol stack

The protocol stack for a DOCSIS to G.9954v2 bridge described below is based on the DOCSIS specification for CPE-controlled cable modems defined in [b-DOCSIS-1] and the DOCSIS radio frequency interface specification in [b-DOCSIS-2].

The first specification assumes a Cable Modem device connected to a customer premises equipment (CPE) over an 802.3/Ethernet, USB or PCI PHY, which is used to transparently transport 802.3 MAC frames between cable modem and CPE devices. Since DOCSIS is defined as a system for the transparent transport of IP traffic over cable, the interface assumes that bridging between DOCSIS and other protocols, such as G.9954v2, is performed at the Ethernet/802.3 MAC frame level. This is illustrated in Figure III.7.

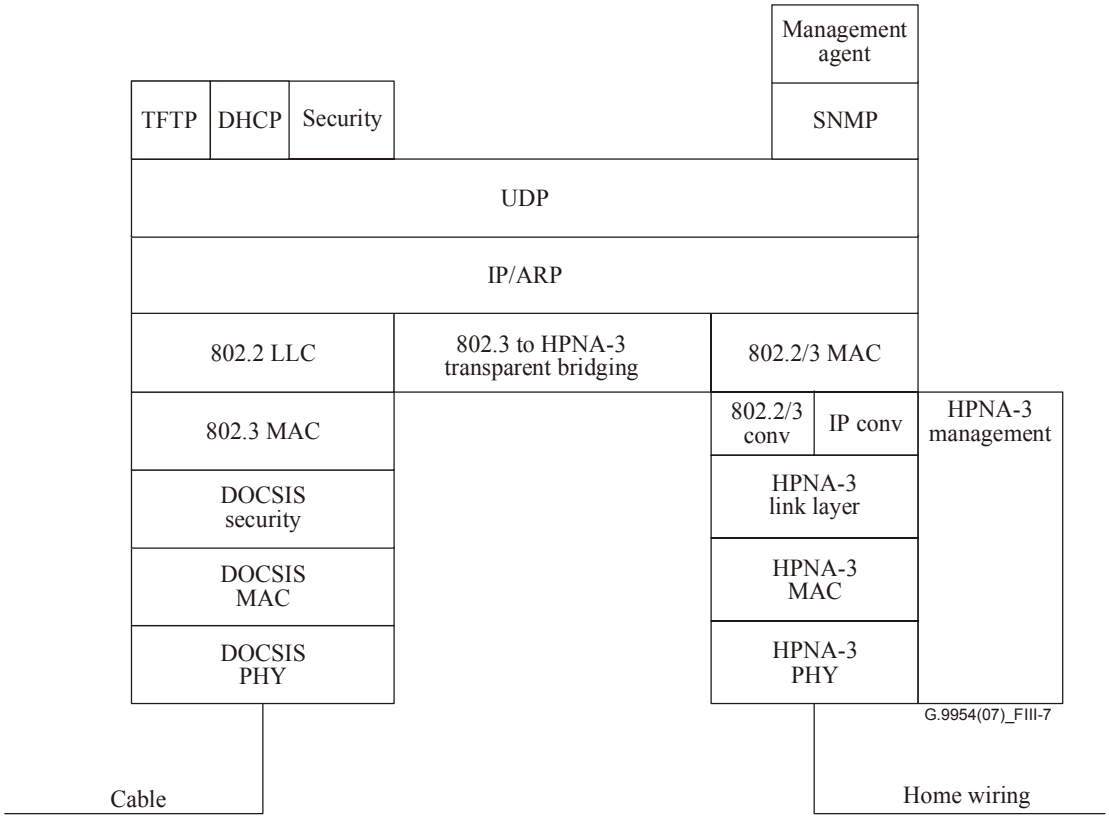

Figure III.7 – DOCSIS to G.9954v2 protocol stack

An additional configuration involves an direct interface to the DOCSIS MAC. This is a lower-level interface than the Ethernet/802.3 interface and provides access to elements in the MAC data service interface of DOCSIS, such as master CLOCK SYNCHRONIZATION, UPSTREAM GRANT SYNCHRONIZATION, that can be used to synchronize the G.9954v2 home network to the external DOCSIS network. This is illustrated in Figure III.8.

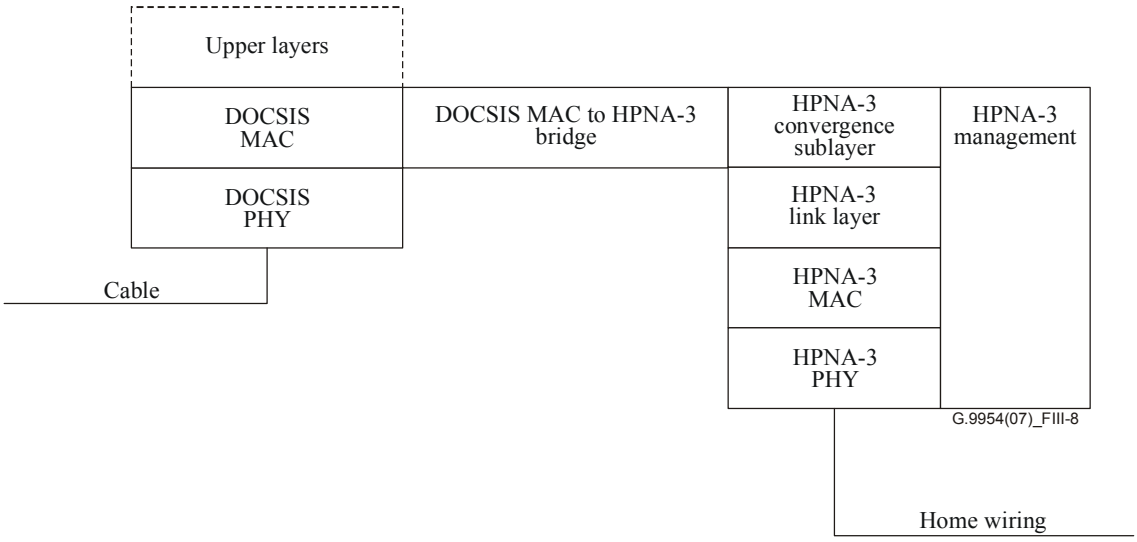

Figure III.8 – DOCSIS to G.9954v2 bridge

Appendix IV

Network synchronization

(This appendix does not form an integral part of this Recommendation)

The requirement to support synchronization to external network and protocols is derived from the types of services being delivered to the home and the networking technology and protocols used to transport these services. Given that some of these services, such as voice, audio and video are isochronous in nature and sensitive to latencies and jitter introduced by connecting networks, as well as to the differences in the clock frequencies between source and destination elements, in order to preserve the quality of a delivered service and to extend it into the home, a home networking technology should provide capabilities to allow synchronization of the home and external networks.

The G.9954v2 protocol supports several built-in mechanisms that, when used together, support end-to-end synchronization of home networks with an external synchronous network and services. These mechanisms and the manner in which they interoperate are described in the following clauses.

IV.1 Synchronization requirements

In order to synchronize elements connected to the home network to an external source or service, the following requirements must be addressed:

- Synchronization of data sampling rates – The frequencies of the clock used for data sampling at the source and destination of a service must be synchronized so as to guard against data underrun and overrun.
- Clock reference synchronization – Synchronization of clocks to common time reference may be required in order to relate to timestamp references that appear in sampled data or in protocol management messages.
- Synchronization to allocated timeslots and bandwidth grants – In order to reduce latency and jitter introduced by the home network, it is necessary to synchronize the allocation of timeslots on the home network with those on the external network used to deliver the service. The synchronization requirement is that data delivered to one network should only need to wait a minimal amount of time before gaining access to the other network.
- Quality of Service – Quality of Service mechanisms in the home network are required in order to guarantee timely access to the home network in accordance with QoS constraints of the delivered service.
- Protocol-awareness – In order to synchronize with external protocols, it is necessary to have protocol-specific knowledge of the elements used for synchronization. For example, knowledge of clock synchronization services in IEEE 1394 or time-synchronization and timeslot grant information in DOCSIS.

IV.2 The network synchronization model

The mechanisms used to support end-to-end synchronization to an external network are illustrated in Figure IV.1.

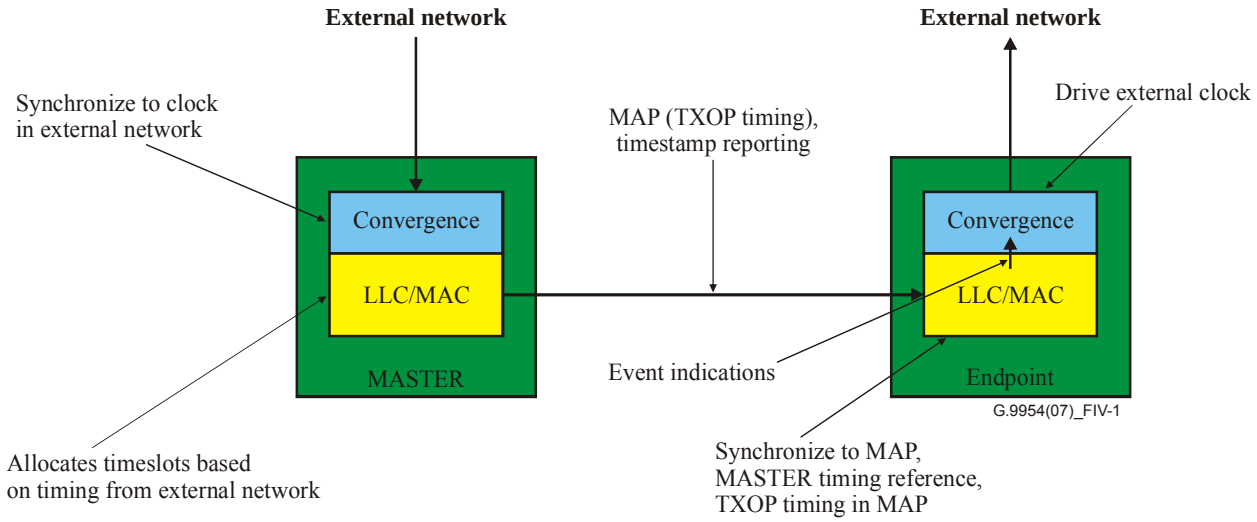

Figure IV.1 – Network synchronization model

The model describes a network based on a master connected to an external network that delivers synchronous services, such as telephony or video services, and one or more endpoint (SLAVE) devices connected to the master on the home network.

In this model, the convergence layer on the master side has protocol-specific knowledge of the connected external network and uses this knowledge to derive a clock reference from the external protocol. This may involve processing of protocol-specific messages, such as DOCSIS Time Synchronization (SYNC) messages or accessing protocol-specific registers that implement clock synchronization services, as defined in IEEE 1394. This timing information can be used to "drive" the G.9954v2 system clock and synchronize its time-reference to that of the external network. This allows time-references derived from the external network, such as timestamp information, to be easily interpreted within the context of the home network.

The convergence layer is further required to recognize the existence and timing of bandwidth grants, timeslots or channels associated with the transport of service and to map these services to the associated flows set up on the home network. Signalling protocol messages, derived from the external network, and associated with the set-up of delivered services, can be used to derive the QoS parameters for the service on the home network. Flows set up on the home network, by the convergence layer, will be set up using QoS and timing information derived directly from the external network. More specifically, the convergence layer will direct the bandwidth manager in the G.9954v2 stack, to allocate TXOPs at a time within the synchronous MAC cycle that is closely synchronized with the bandwidth grants on the external network. This is used to control service latency and jitter.

Once the master is synchronized with the external network, and timeslots (TXOPs) have been synchronized with the arrival of data from the external network, the synchronization of endpoint devices follows naturally from the synchronous protocol defined for G.9954v2. Endpoints can synchronize with the (synchronized) master clock reference through its distribution of periodic timestamp reports or from information contained in the periodic MAP message. Furthermore, endpoints naturally synchronize on the timing of allocated TXOPs described in the MAP. The event indication mechanism can be used to notify the convergence layer at the endpoint of expected or granted TXOP timing information associated with a service. For flows that have their timeslot event indication flags enabled, the G.9954v2 MAC will notify (using interrupt or similar mechanisms) the upper convergence layer of the planned arrival of timeslot (TXOP) grants or service data. This indication can be used to drive a clock at the endpoint and/or to drive the data sampling rate at the endpoint.

It is still possible to synchronize to an external network without synchronizing clock references or sampling clocks. If clocks in the master and external network are not synchronized, a service may experience a MAXIMUM TRANSMISSION DELAY that is a function of the length of the MAC cycle accounting for worst-case acquisition period and network access latency. Furthermore, a lack of synchronization of the data (sample) arrival time and the allocated TXOP on the home network may result in the familiar saw-tooth latency/jitter behaviour as illustrated in Figure IV.2.

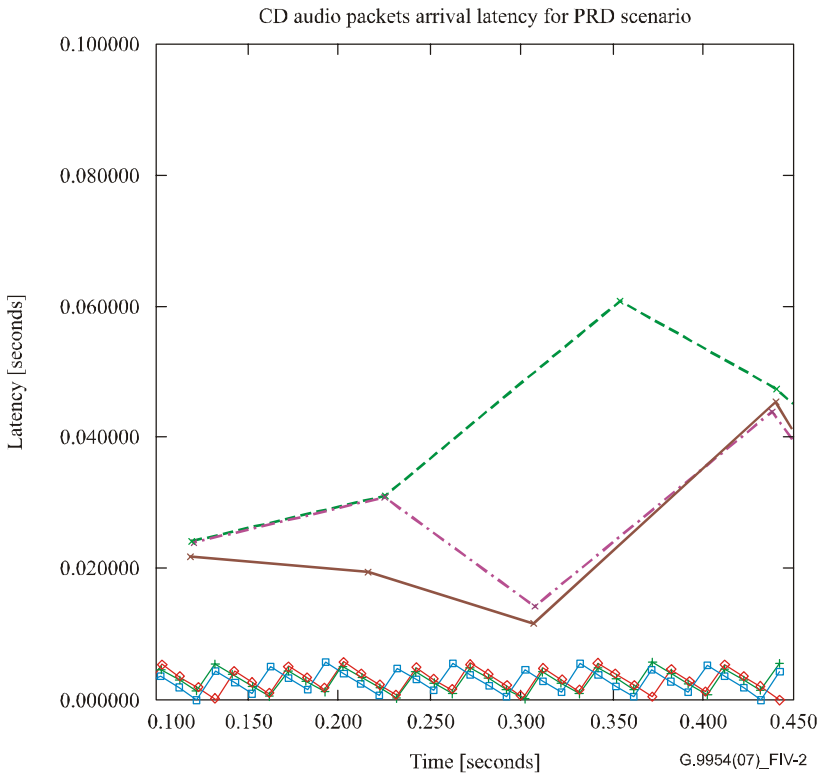

Figure IV.2 – Sawtooth latency/jitter behaviour

IV.3 Summary of synchronization mechanisms

Table IV.1 summarizes the set of synchronization mechanisms supported by the proposed G.9954v2 protocol.

Table IV.1 – Synchronization mechanisms summary

Synchronization mechanism	Purpose
Synchronous protocol	Supports synchronization with other synchronous protocols. Endpoint synchronization with master.
Clock synchronization	Synchronize clocks to a common time reference. Synchronize sampling rates.
MAC cycle indication	Synchronize external networks or protocols with MAC cycle.
Timeslot event indications	Synchronize to planned TXOP timing using information in MAP.
Timestamping stream data	Timestamp data using network clock reference.
Timeslot allocation control	Synchronize the allocation of timeslots on the home network with timeslot grants in an external network.
Protocol convergence layer	Supports protocol specific handling of synchronization methods from external networks.

Appendix V

Support for variable bit-rate (VBR) flows

(This appendix does not form an integral part of this Recommendation)

Variable bit-rate (VBR) flows can be handled using the following different bandwidth allocation strategies:

- Per-cycle bandwidth request;
- UGS + shared transmission opportunity;
- UGS + explicit bandwidth requests;
- UGS + spare bandwidth.

The content of this appendix should be considered informative in nature and used only as a guideline for implementations.

V.1 Per-cycle bandwidth request

This method requires that an explicit RTS request be issued each cycle. The amount of bandwidth requested each cycle is variable in accordance with the VBR behaviour of the service flow.

The bandwidth allocation method, although simple, may require some tight real-time control in order to ensure an endpoint node does not violate of traffic rate characteristics and that QoS constraints can be met.

V.2 UGS + shared transmission opportunity

The following method is suitable when there are several VBR flows active at one time. It is most appropriate when the source of all the VBR flows is the same station – i.e., there is no contention between VBR service flows, although it can also be used when the VBR flows originate from different stations.

This method assumes that a group of VBR flows will share the same transmission opportunity. The TXOP is allocated as for UGS service types (i.e., no explicit RTS is required); however, the amount of bandwidth allocated is calculated to be the cumulative average bit rates of all the flows sharing the same TXOP.

The method relies on the variable nature of VBR flows. It assumes that VBR flows will NOT all peak at the same time, but rather the bandwidth demands of all VBR flows approximately equals their cumulative average.

This method is illustrated in Figure V.1.

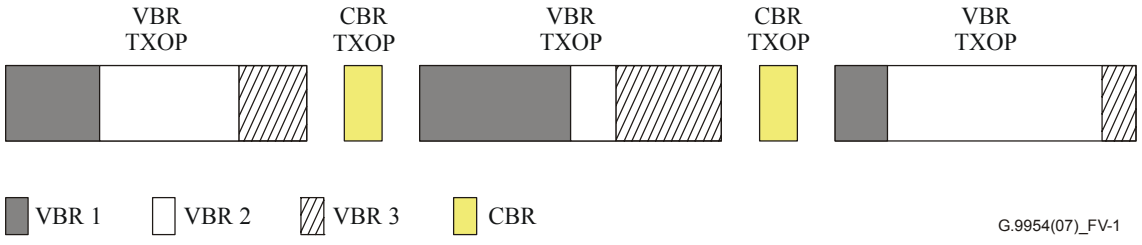

Figure V.1 – Variable bit-rate (VBR) bandwidth allocation

V.3 UGS + explicit bandwidth requests

The following method is a combination of the UGS and explicit bandwidth request methods. A VBR flow is treated like a CBR flow that may occasionally require some extra bandwidth to handle the variableness of the traffic. The basic data rate requested for the VBR flow is based on the flow's average bit-rate constraint.

The allocation of fixed size TXOPs for a VBR flow effectively shapes the flow traffic into a constant bit-rate (CBR) form. If the flow has sufficient buffers associated with it to handle the burstiness of the traffic, this should be sufficient to handle the VBR nature of the flow without explicit bandwidth requests. However, if sufficient buffer space is NOT available, an endpoint node can explicitly request extra bandwidth in order to temporarily relieve the traffic backlog.

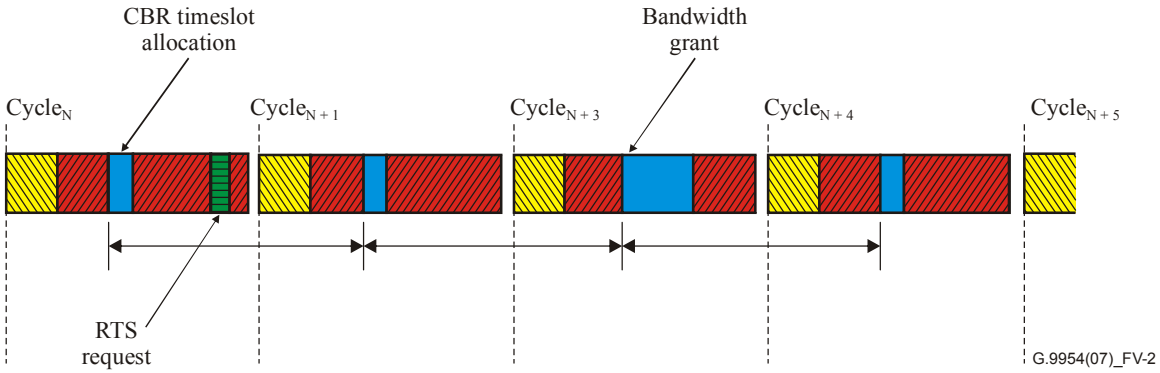

Figure V.2 – VBR using CBR + explicit bandwidth requests

V.4 UGS + spare bandwidth

Yet another method for handling VBR services involves using spare (unallocated) bandwidth to handle traffic bursts that exceed the traffic rate defined by the CBR TXOPs allocated for the flow. This is illustrated in Figure V.3.

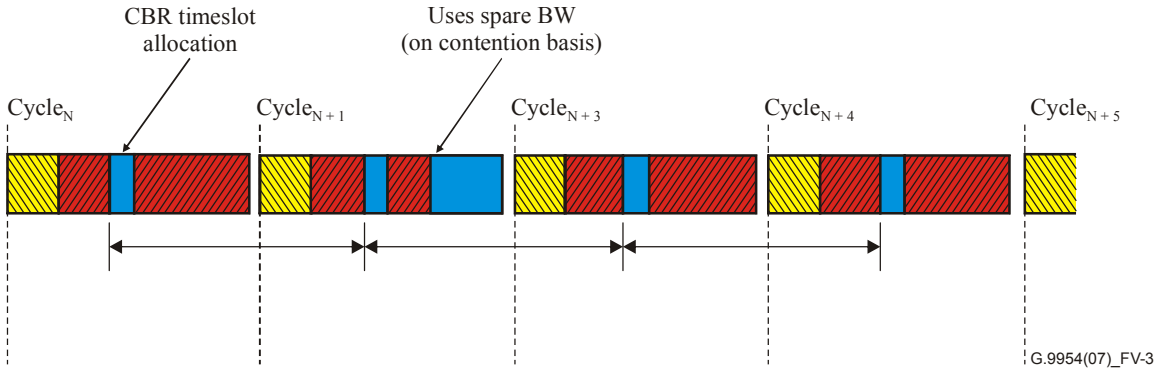

Figure V.3 – VBR using CBR + spare bandwidth

Extra bandwidth may also be allocated to the TXOPs of a VBR service such that there is sufficient extra media time for the transmission of at least a whole (extra) packet. Using this method, the master scheduler should allocate a little more than the average bit-rate requirements, relying on the extra bandwidth to be used occasionally to empty traffic queues.

Appendix VI

Quality of service (QoS) parameters

(This appendix does not form an integral part of this Recommendation)

This Recommendation supports all services described in Table VI.1. In addition, it should simultaneously support all services in Table VI.2.

Table VI.1 – Standard services QoS requirements[1]

Service	Relative priority	MAC payload rate (per stream)	Payload definition	Min. simultaneous streams	Max. bit-error rate	Max. latency	Max. jitter
Voice services							
High-quality narrowband voice telephony	High	32-64 kbit/s	Voice payload[a)]	8[b)]	1e-6	5 ms nominal; 10 ms max.	±5 ms
Lower-quality narrowband voice telephony	Low to Medium	6-16 kbit/s	Voice payload	8	1e-6	10 ms nominal; 30 ms max.	±10 ms
Time-critical packet service (e.g., video conferencing)	High	4-13 kbit/s for voice; 0.032-1.5 Mbit/s for audio/video	Voice payload for voice, MPEG-TS[c)] payload for audio/video	4 (2 conversations; 2 streams per conversation)	1e-8	5 ms nominal; 10 ms max. for full duplex services	±5 ms
High-speed data services							
Best-effort service	Low	Up to maximum physical layer rate	data packet[d)]	N/A	1e-6	500 ms	N/A
QoS (SLA[e)]) service	Medium to High	10 Mbit/s	data packet	2	1e-8	10 ms nominal; 30 ms max.	±10 ms
IP media streaming							
Standard audio	Low to Medium	96-256 kbit/s	MPEG-TS	3	1e-6	200 ms	±20 ms
CD-quality audio	Medium	192-256 kbit/s (stereo)	MPEG-TS	3	1e-8	100 ms	±10 ms
Lower-quality streaming video	Medium to High	64-500 kbit/s	MPEG-TS	3	1e-6	100 ms	±10 ms
Home theatre audio[f)]	High	6 Mbit/s	MPEG-TS	1	1e-8	100 ms	±10 ms
Higher-quality streaming video	High	1.5-10 Mbit/s	MPEG-TS	1	1e-8	50 ms	±10 ms
Digital video disk[g)]		3.0-20 Mbit/s	MPEG-TS	2	1e-8	100 ms	±10 ms

1 Source: CableLabs "Home Networking Requirements for Cable-Based Services," Vendor Release 1.0 dated June 9, 2000. Copyright Cable Television Laboratories, Inc. 2001. All rights Reserved. Reprinted with permission (except as noted).

Table VI.1 – Standard services QoS requirements[1]

Service	Relative priority	MAC payload rate (per stream)	Payload definition	Min. simultaneous streams	Max. bit-error rate	Max. latency	Max. jitter
Broadcast quality video							
SDTV	High	3-7 Mbit/s		2	1e-8	90 ms nominal;	Interpacket ±10 ms
HDTV	High	19.68 Mbit/s		1	1e-8	90 ms nominal;	Interpacket ±10 ms

a) Voice payload: Variable size depending on codec, considering the end-to-end latency budget. For example, G.711 μ-law encoding specifies frames of four samples where each audio sample is encoded as an 8-bit value (i.e., 32-bit).

b) The protocol developed for this technology must be able to support a minimum of four concurrent off-hook devices. With network connect rates greater than or equal to equivalent 10Base-T, the protocol shall support eight concurrent off-hook devices.

c) MPEG-TS: Audio/Video payload assumes an MPEG Transport Stream (TS) of size 188 bytes.

d) Data Packet: Ethernet payload including TCP/IP headers but excluding the Ethernet header and Ethernet CRC.

e) SLA used in this context implies "Service Level Agreement" and refers to a minimum quality of service that the service is committed to receiving. In this context, the SLA is taken to mean the "Committed Information Rate".

f) Home Theatre Audio encompasses 5.1 channels of simultaneous audio. Note this is not included in the CableLabs document. It is assumed that the AC-3 Dolby digital format is multiplexed in an MPEG-2 TS.

g) Digital Video Disk encompasses two SDTV streams. Note this is not included in the CableLabs document.

Table VI.2 – Additional standard services QoS requirements

Service	Relative priority	MAC payload rate (per stream)	Min. simultaneous streams	Max. bit-error rate	Max. latency	Max. jitter
Voice services						
High-quality narrowband voice telephony	High	32-64 kbit/s	6 (3 conversations; 2 streams per conversation)	1e-6	5 ms nominal; 10 ms max.	±5 ms
Time-critical packet service (e.g., video conferencing)	High	4-13 kbit/s for voice; 0.032-1.5 Mbit/s for audio/video	2 (1 conversation; 2 streams per conversation)	1e-8	5 ms nominal; 10 ms max. for full duplex services	±5 ms
High-speed data services						
Best-effort service	Low	Up to maximum physical layer rate	N/A	1e-6	500 ms	N/A
IP media streaming						
CD-quality audio	Medium	192-256 kbit/s (stereo)	3	1e-8	100 ms	±10 ms
Any 2-stream combinations of the following:						
Higher-quality streaming video	High	1.5-10 Mbit/s	1	1e-8	50 ms	±10 ms
Home theater audio	High	6 Mbit/s	1	1e-8	100 ms	±10 ms
Digital video disk		3.0-20 Mbit/s	1	1e-8	100 ms	±10 ms

Table VI.2 – Additional standard services QoS requirements

Service	Relative priority	MAC payload rate (per stream)	Min. simultaneous streams	Max. bit-error rate	Max. latency	Max. jitter
Broadcast quality video						
SDTV	High	3-7 Mbit/s	2	1e-8	90 ms nominal	Interpacket ±10 ms
HDTV	High	19.68 Mbit/s	1	1e-8	90 ms nominal	Interpacket ±10 ms

Appendix VII

Simultaneous applications test profiles

(This appendix does not form an integral part of this Recommendation)

Test profile 1 (see Figure VII.1) describes a home network consisting of a residential gateway (RG) providing access to telephony and Internet services and a second gateway or server source providing access to video-related services. The RG and possibly the video/TV server are connected to broadband pipes. Furthermore, the home network profile consists of clients that consume broadband services as well as those that interact directly with peers on the same home network.

The network throughput requirements for test profile 1 are described in Table VII.1. It is assumed the network configuration is a star network, with a 6-foot wire attached to each.

Table VII.1 – Network throughput requirements

Service	Quantity	Rate [Mbit/s]	Throughput requirement
HQ-voice	6	0.064	0.384
Video conference	2	1.5	3
Best effort	1	Up to physical limit	Up to physical limit
CD	3	0.256	0.768
SDTV	2	3	6
HDTV	1	19.68	19.68
Home theatre	2	5.76	11.52
TOTAL			41.352

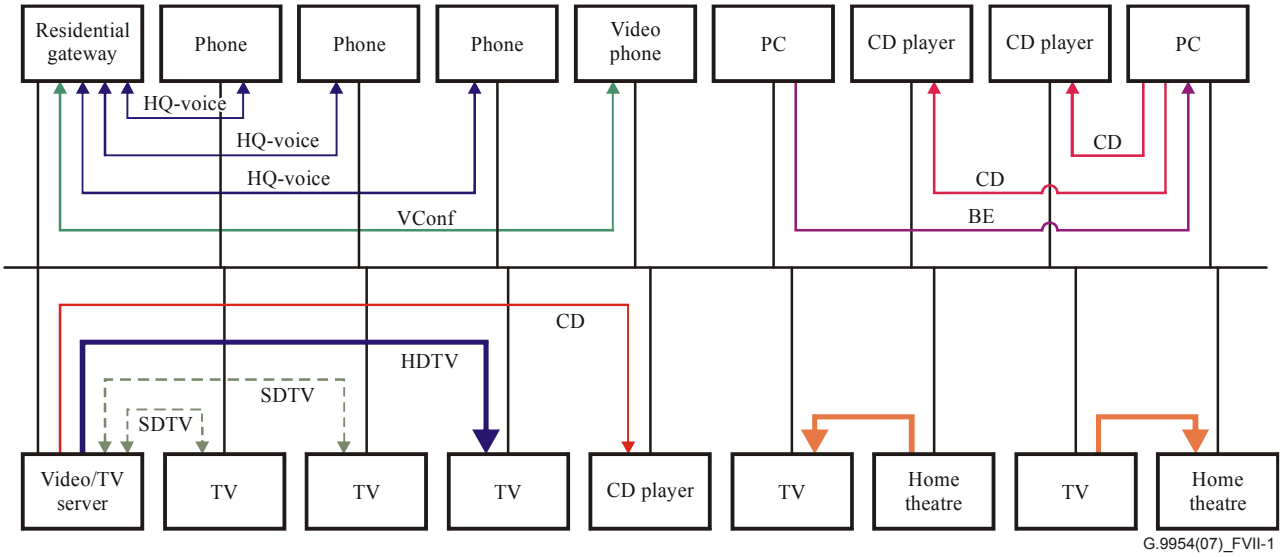

Figure VII.1 – Test profile 1

Appendix VIII

Media access planning guidelines

(This appendix does not form an integral part of this Recommendation)

Media access planning is a scheduling activity whose goal is to produce a media access plan (MAP) that satisfies the QoS constraints of all contending flows in the network. The scheduling algorithm executes entirely in the master node and takes into consideration the available media bandwidth and the QoS constraints of the entire network.

Although the specification of scheduling algorithms employed by a G.9954v2 master is beyond the scope of this Recommendation, it is expected that a G.9954v2 master scheduler support the following set of basic functional capabilities:

- Resource management;
- Media resource allocation and assignment;
- Burst size management;
- MAC cycle length management;
- Traffic policing and shaping;
- Latency and jitter control;
- Collision management strategy assignment;
- Bandwidth request management;
- MAP generation.

VIII.1 Resource management

The master should manage state information about the allocation of media resources in the home network and maintain an allocation map that describes allocated and free media resources and their sizes. The allocation map is used by the bandwidth allocation function when performing admission control for service requests.

VIII.2 Media resource allocation and assignment

Given the availability of sufficient media resources to service a bandwidth request, the master should allocate TXOPs to the specific flow. The allocated TXOP is subsequently described in the MAP.

VIII.3 Burst size management

In order to use the media more efficiently and to reduce protocol overheads, it is desirable to aggregate upper-level packets originating from a single source or flow into single PHY layer bursts (frames). The length of the burst depends on a number of factors including the length of the TXOP, flow latency requirements, BER characteristics, etc.

The master scheduler should to try to concentrate TXOPs assigned to the same source such that an endpoint can maximize the length of the bursts while still meeting flow QoS latency and jitter constraints.

VIII.4 MAC cycle length management

Each MAP frame implicitly defines the extent (in time) of the media access plan. This provides the infrastructure to support MAC cycles that are variable in length and that may even change dynamically from cycle to cycle.

The master scheduler is responsible for selecting the appropriate size of the MAC cycle. The guidelines used in the selection process require the scheduler to select a cycle length that balances the periodicity requirements of the active flows with protocol overhead considerations introduced by the transmission of the MAP frame.

VIII.5 Traffic policing and shaping

To ensure the conformance of a flow to its negotiated traffic parameters, the master scheduler should police and shape traffic such that the network will not suffer in case a traffic source starts to generate traffic in a non-conformant manner. Traffic policing and shaping is done by allocating TXOPs in a manner that meets traffic specifications.

For a G.9954v2 endpoint node that assigns packets to TXOPs in accordance with the description of the MAP, this will inherently shape the endpoints' traffic into the form intended by the master. This has the effect of reducing the potential complexity of endpoint nodes by centralizing the traffic policing and shaping algorithms in the master while also ensuring that endpoint nodes do not generate traffic in a manner that violates their negotiated agreement.

VIII.6 Latency and jitter control

The master scheduler is responsible for performing latency and jitter control by guaranteeing that TXOPs are allocated to flows at the required frequency, size and interval that allows them to meet flow latency and jitter requirements.

Consider two examples allocations of TXOPs over time (see Figure VIII.1), relative to the arrival time of the packets from the input source. In example 1, TXOPs are allocated such that they provide zero jitter. In example 2, the latency variance causes jitter as seen in the Latency/Jitter graph in Figure VIII.2.

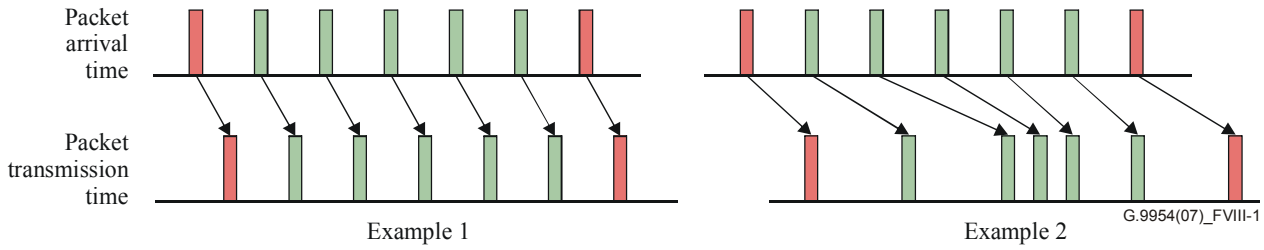

Figure VIII.1 – Latency/Jitter examples

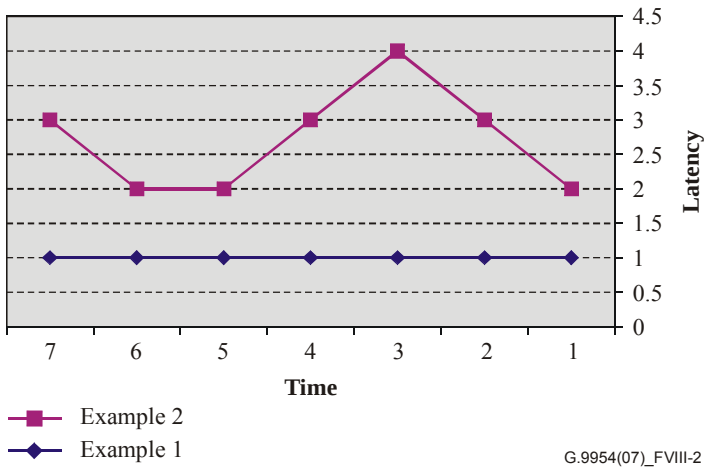

Figure VIII.2 – Latency/Jitter graph

VIII.7 MAP generation

The output of the master's media access planning is the MAP frame. The master is responsible for generating the periodic MAP control frame that contains the results of the processes and decisions described above.

Bibliography

[b-DOCSIS-1] Data-Over-Cable Service Interface Specifications SP-CMCI-I05-001215, *Cable Modem to Customer Premise Equipment Interface Specification*, July 14, 2000.

[b-DOCSIS-2] Data-Over-Cable Service Interface Specifications SP-RFIv1.1-I06-001215, *Radio Frequency Interface Specification*, December 15, 2000.

[b-IEEE-P1394.1] IEEE P1394.1, *Draft Standard for High Performance Serial Bus Bridges*, 0.16, March 29, 2001.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	**Transmission systems and media, digital systems and networks**
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2008