International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.9902

(10/2012)

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

Access networks – In premises networks

Narrowband orthogonal frequency division multiplexing power line communication transceivers for ITU-T G.hnem networks

Recommendation ITU-T G.9902

ITU-T

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
MULTIMEDIA 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.9900–G.9999**

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

Recommendation ITU-T G.9902

Narrowband orthogonal frequency division multiplexing power line communication transceivers for ITU-T G.hnem networks

Summary

Recommendation ITU-T G.9902 contains the physical layer (PHY) and the data link layer (DLL) specifications for the ITU-T G.9902 narrowband orthogonal frequency division multiplexing (OFDM) power line communication transceivers, operating over alternating current and direct current electric power lines over frequencies below 500 kHz.

This Recommendation uses material from Recommendations ITU-T G.9955 and ITU-T G.9956; specifically material from the main body and the annexes pertaining to the main body. New technical material has not been introduced in this version.

The control parameters that determine spectral content, power spectral density (PSD) mask requirements and the set of tools to support the reduction of the transmit PSD can be found in Recommendation ITU-T G.9901.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T G.9902	2012-10-29	15

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). 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/.

Table of Contents

Page

Recommendation ITU-T G.9902

Narrowband orthogonal frequency division multiplexing power line communication transceivers for ITU-T G.hnem networks

1 Scope

Recommendation ITU-T G.9902 contains the physical layer (PHY) and the data link layer (DLL) specifications for the ITU-T G.9902 narrowband orthogonal frequency division multiplexing (OFDM) power line communication transceivers operating over alternating current and direct current electric power lines over frequencies below 500 kHz. This Recommendation supports indoor and outdoor communications over low voltage-lines, medium-voltage lines, through transformer low-voltage to medium-voltage and through transformer medium-voltage to low-voltage power lines in both urban and long-distance rural communications. This Recommendation addresses grid to utility meter applications, advanced metering infrastructure (AMI) and other 'Smart Grid' applications such as the charging of electric vehicles, home automation and home area network (HAN) communications scenarios.

This Recommendation does not contain the control parameters that determine spectral content, power spectral density (PSD) mask requirements and the set of tools to support a reduction of the transmit PSD; all of which are detailed in [ITU-T G.9901].

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.9901] Recommendation ITU-T G.9901 (2012), *Narrowband orthogonal frequency division multiplexing power line communication transceivers – Power spectral density specification.*

[IEEE 802.3] IEEE 802.3 (2005), *Local and metropolitan area networks – Specific requirements. Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access method and Physical Layer specifications*.

[IETF RFC 791] IETF RFC 791 (1981), *Internet Protocol, DARPA Internet Program, Protocol Specification.*

[IETF RFC 2460] IETF RFC 2460 (1998), *Internet Protocol, Version 6 (IPv6) Specification.*

[IETF RFC 4861] IETF RFC 4861 (2007), *Neighbor Discovery for IP version 6 (IPv6)*.

[IETF RFC 4944] IETF RFC 4944 (2007), *Transmission of IPv6 Packets over IEEE 802.15.4 Networks.*

[IETF RFC 6282] IETF RFC 6282 (2011), *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks.*

[NIST PUB 197] NIST FIPS PUB 197 (2001), *Advanced Encryption Standard (AES)*.

[NIST SP 800-38C] NIST SP 800-38C (2004), *Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality*.

[EUI-64] EUI-64™ (in force), *Guidelines for 64-bit Global Identifier, IEEE.* <http://standards.ieee.org/develop/regauth/tut/eui64.pdf>

3 Definitions

3.1 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.1.1 alien domain: This is any group of non-ITU-T G.9902 nodes connected to the same or different medium (wired or wireless) operating in close proximity. These domains can be used as backbones to the ITU-T G.9902 network or as separate networks. The L3 bridging function to an alien domain, as well as coordination with an alien domain to avoid mutual interference is beyond the scope of this Recommendation.

3.1.2 advanced metering infrastructure (AMI): The primary means for utilities to interact with meters on customer sites. In addition to basic meter reading, AMI provides two-way communication which allows energy usage data to be collected and analysed and it enables the interaction with advanced devices such as electricity meters, gas meters, heat meters, and water meters, through various communications media.

3.1.3 bandplan: This is a specific range of the frequency spectrum that is defined by a lower frequency and upper frequency.

3.1.4 baseband: This is a frequency band defined by an up-convert frequency FUC = 0 and an up-shift frequency FUS = FSC×N/2 (see Table 8-27).

3.1.5 bridge to alien domain/network: An application device implementing an L2 or L3-bridging function to interconnect an ITU-T G.9902 domain to an alien domain (or alien network). Bridging to alien domains/networks is beyond the scope of this Recommendation.

3.1.6 broadcast: This is a type of communication where a node sends the same frame simultaneously to all other nodes in the home network or in the domain.

3.1.7 carrier sense (CRS): This is generated by the receiver and indicates that the medium is busy, i.e., a PHY frame or sequence of PHY frames, or a special signal (e.g., INUSE, PR) is currently being transmitted on the medium by another node. CRS may be either a physical carrier sense signal or a virtual carrier sense indicator.

– Physical carrier sense signal: generated by analysing physical signals that are present on the medium.
– Virtual carrier sense signal: generated based on the information on the PHY frame duration or PHY frame sequence duration derived from the frame header or communicated to a node by other means (e.g., in another frame).

3.1.8 *ceiling(x)*: A function that returns the minimum integer value bigger than or equal to x.

3.1.9 CENELEC band: Frequency band between 3 kHz and 148.5 KHz which is allowed to be used for power line communications. Four CENELEC (European Committee for Electrotechnical Standardization) bands are defined: A: 3-95 kHz, B: 95-125 kHz, C: 125-140 kHz and D: 140-148.5 kHz. For more details, see [ITU-T G.9901].

3.1.10 channel: A transmission path between nodes. One channel is considered to be one transmission path. Logically, a channel is an instance of where a communications medium is used for the purposes of passing data between two or more nodes.

3.1.11 coding overhead: Part of the overhead used to carry coding redundancy (such as redundancy bits of error correction coding or CRC).

3.1.12 data: Bits or bytes transported over a medium or via a reference point that individually conveys information. Data includes both user (application) data and any other auxiliary information (overhead, including control, management, etc.). Data does not include bits or bytes that, by themselves, do not convey any information, such as preamble.

3.1.13 data rate: The average number of data elements (bits, bytes, or frames) communicated (transmitted) in a unit of time. Depending on the data element, data bit rate, data byte rate, and symbol frame rate may be used. The usual unit of time for data rate is 1 second.

3.1.14 domain: A part of an ITU-T G.9902 home network comprising a domain master and all those nodes that are registered with this same domain master. In the context of this Recommendation, use of the term 'domain' without a qualifier means 'ITU-T G.9902 domain', and use of the term 'alien domain' means 'non-ITU-T G.9902 domain'.

3.1.15 domain access point (DAP): The unique node in a centralized mode (CM) that supports relay functionality through which all nodes communicate.

3.1.16 domain ID: A unique identifier of a domain.

3.1.17 domain master (DM): A node that manages (coordinates) all other nodes of the same domain. Domain master is a node with extended management capabilities that enables the forming, controlling and maintaining of the nodes associated with its domain.

3.1.18 end-node: A node that is not a domain master; all nodes in the domain except the domain master are end-nodes.

3.1.19 FCC band: FCC (Federal Communications Commission) frequency band between 9 kHz and 490 KHz allowed to be used for power line communications. For more details, see [ITU-T G.9901].

3.1.20 *floor(x)*: A function that returns the maximum integer value smaller than or equal to x.

3.1.21 global master (GM): A function that provides coordination between different domains of the same network (such as communication resources, priority settings, policies of domain masters, and interference mitigation). A GM may also convey management functions initiated by the remote management system. Detailed specification and use of this function is for further study.

3.1.22 guard interval (GI): The time interval intended to mitigate the corruption of data carried by the symbol due to ISI from the preceding symbols. In this Recommendation, the guard interval is implemented as a cyclic prefix.

3.1.23 hidden node: A node that cannot communicate directly with some other nodes within a domain. A hidden node may be able to communicate with another node using a relay node.

3.1.24 home area network (HAN): A network at customer premises that interconnects customer-owned devices for energy management and communication with the utility.

3.1.25 inter-domain bridge (IDB): A bridging function which interconnects nodes of two different domains.

3.1.26 inter-network bridge (INB): A bridging function which interconnects nodes of two different ITU-T G.9902 networks.

3.1.27 latency: A measure of the delay from the instant when the last bit of a frame has been transmitted through the assigned reference point of the transmitter protocol stack to the instant when a whole frame reaches the assigned reference point of a receiver protocol stack. Mean and maximum latency estimations are assumed to be calculated on the 99th percentile of all latency measurements. If retransmission is required for a frame, then the retransmission time is a part of the latency for the protocol reference points above the MAC.

3.1.28 logical (functional) interface: An interface in which the semantic, syntactic, and symbolic attributes of information flows are defined. Logical interfaces do not define the physical properties of signals used to represent the information. It is defined by a set of primitives.

3.1.29 management overhead: A part of the traffic used for management purposes (any used bandwidth that does not contain application data).

3.1.30 medium: A wire-line facility allowing the physical connection between nodes. Nodes connected to the same medium may communicate on the physical layer and may interfere with each other unless they use orthogonal signals (e.g., different frequency bands, different time periods).

3.1.31 ***mod(a,b)***: A function that returns the remainder when a is divided by b.

3.1.32 multicast: A type of communication when a node sends the same frame simultaneously to more than one node in the network.

3.1.33 net data rate: The data rate at the A-interface of the transceiver reference model.

3.1.34 network: Two or more nodes that can communicate with each other either directly or through a relay node at the physical layer, or through an inter-domain bridge above the physical layer.

3.1.35 node: Any network device that contains an ITU-T G.9902 transceiver. In the context of this Recommendation, use of the term 'node' without a qualifier means 'ITU-T G.9902 node', and use of the term 'alien node' means 'non-ITU-T G.9902 node'. Additional qualifiers (e.g., 'relay') may be added to either 'node' or 'alien node'.

3.1.36 node ID: A unique identifier allocated to a node within the domain.

3.1.37 peer-to-peer communication: A type of communication within a domain in which direct signal traffic is established between nodes with no relay nodes.

3.1.38 physical interface: An interface defined in terms of the physical properties of the signals used to represent the information transfer. A physical interface is defined by signal parameters like power (power spectrum density), timing, and connector type.

3.1.39 primitives: Variables and functions used to define logical interfaces and reference points.

3.1.40 priority: Medium access (MA) priority is a value assigned to a particular frame(s) that determines the relative importance of transmitting the frame(s) during the upcoming opportunity to use the medium. The protocol data unit (PDU) priority is a value assigned to a particular protocol data unit that determines the relative order of queueing of this PDU for transmission over the medium.

3.1.41 quality of service (QoS): A set of quality requirements on the communications in the network.

3.1.42 reference point: A location in a signal flow, either logical or physical, that provides a common point for observation and/or measurement of the signal flow.

3.1.43 registration: The process used by a node to join the domain.

3.1.44 relayed communication: A type of communication within a domain, in which a node can communicate with another node through a relay node. A relay node receives a signal from a node and forwards it to the addressee node.

3.1.45 relay node: A node supporting relay functionality that acts as an intermediary node, through which other nodes of the same domain can pass their signal traffic (data, control, or management).

3.1.46 subcarrier (OFDM subcarrier): The centre frequency of each OFDM sub-channel onto which bits may be modulated for transmission over the sub-channel.

3.1.47 subcarrier spacing: The difference between frequencies of any two adjacent OFDM subcarriers.

3.1.48 sub-channel (OFDM sub-channel): A fundamental element of OFDM modulation technology. The OFDM modulator partitions the channel bandwidth into a set of non-overlapping sub-channels.

3.1.49 symbol (OFDM symbol): A fixed time-unit of an OFDM signal. An OFDM symbol consists of multiple sine-wave signals or subcarriers. Each subcarrier can be modulated by a certain number of data bits and transmitted during the fixed time called symbol period.

3.1.50 symbol frame: A frame composed of bits of a single OFDM symbol period. Symbol frames are exchanged over the δ-reference point between the PMA and PMD sublayers of the PHY.

3.1.51 symbol rate: The rate in symbols per second, at which OFDM symbols are transmitted by a node onto a medium. Symbol rate is calculated only for time periods of continuous transmission.

3.1.52 throughput: The amount of data transferred from the A-interface of a source node to the A-interface of a destination node over a time interval; expressed as the number of bits per second.

3.1.53 transmission overhead: A part of the overhead used to support transmission over the line (e.g., samples of cyclic prefix, inter-frame gaps, and silent periods).

3.1.54 unicast: A type of communication when a node sends the frame to another single node.

3.1.55 utility access network (UAN): A power line communications network that operates under the control of the electric utility over the utility-owned electricity distribution lines, and provides communication between the utility and the utility-controlled devices and network infrastructure at customer premises.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AC	Alternating Current
ACK	Acknowledgement
ADM	Addressing Mode
ADP	Application Data Primitives
AE	Application Entity
AES	Advanced Encryption Standard
AFE	Analogue Front End
AMI	Advanced Metering Infrastructure
AMM	Automated Meter Management
APC	Application Protocol Convergence
APDU	Application Protocol Data Unit
ARF	Address Resolution Function
ARQ	Automatic Repeat request
ASC	Active Subcarriers
BAT	Bit Allocation Table
BC	Back-off Counter
BER	Bit Error Rate

BPR	Beacon Proxy
BPSK	Binary Phase Shift Keying
CC	Convolutional Code
CCR	Convolutional Code Rate
CCM	Counter with Cipher block chaining-Message authentication code
CCMP	CCM Mode Protocol
CCMPI	CCMP header presence Indicator
CM	Centralized Mode
CP	Contention Period
CRC	Cyclic Redundancy Check
CRS	Carrier Sense
CSMA-CA	Carrier Sense Multiple Access-Collision Avoidance
CW	Contention Window
DA	Destination Address
DAP	Domain Access Point
DDID	Destination node DOMAIN_ID
DLL	Data Link Layer
DM	Domain Master
DNI	Domain Name Indicator
DNID	Destination node DOMAIN_ID
DSL	Digital Subscriber Line
EMS	Energy Management System
ESI	Energy Service Interface
EV	Electric Vehicle
EVCF	Electric Vehicle Charging Facility
EVSE	Electric Vehicle Supply Equipment
FCS	Frame Check Sequence
FEC	Forward Error Correction
FFT	Fast Fourier Transform
FN	Frame Number
GF	Galois Field
GI	Guard Interval
GM	Global Master
HAN	Home Area Network
HCS	Header Check Sequence
HopsLft	number of Hops Left
IDB	Inter-Domain Bridge

IDFT	Inverse Discrete Fourier Transform
IFFT	Inverse Fast Fourier Transform
IID	IPv6 Interface Identifier
INB	Inter-Network Bridge
IPv4	Internet Protocol, version 4
IPv6	Internet Protocol, version 6
ISC	Inactive Subcarriers
ISI	Inter-Symbol Interference
LCDU	Link Control Data Unit
LFH	LLC Frame Header
LFSN	LLC Frame Sequence Number
LFSR	Linear Feedback Shift Register
LISN	Line Impedance Stabilization Network
LLC	Logical Link Control
LLCFT	LLC Frame Type
LPCS	LPDU Check Sequence
LPDU	LLC Protocol Data Unit
LPH	LPDU Header
LPRI	LLC frame Priority
LSB	Least Significant Bit
LSPL	Last Segment Pad Length
MAC	Medium Access Control
MDI	Medium-Dependent Interface
MHCS	MPH Check Sequence
MIB	Management Information Base
MIC	Message Integrity Check
MIH	Mesh Header presence Indicator
MM	Management Message
MMH	Management Message Header
MMPL	Management Message Parameter List
MPCS	MPDU Check Sequence
MPDU	MAC Protocol Data Unit
MSB	Most Significant Bit
MSC	Masked Subcarriers
MSDU	MAC Service Data Unit
NACK	Negative Acknowledgement
NLPDU	Number of LPDUs

OFDM	Orthogonal Frequency Division Multiplexing
P2P	Peer-to-Peer communication
PCS	Physical Coding Sublayer
PEV	Plug-in Electric Vehicle
PFH	PHY Frame Header
PHY	Physical layer
PLC	Power Line Communications
PMA	Physical Medium Attachment
PMD	Physical Medium Dependent
PMI	Physical Medium-Independent interface
PMSC	Permanently Masked Subcarriers
PPDU	PHY Protocol Data Unit
PPM	Parts Per Million
PRBS	Pseudo-Random Binary Sequence
PSC	Pilot Subcarriers
PSD	Power Spectral Density
PSDU	PHY Service Data Unit
PST	Programmable Smart Thermostat
QoS	Quality of Service
RCM	Robust Communication Mode
RELI	Relaying presence Indicator
RMS	Root Mean Square
RMSC	Regionally Masked Subcarriers
RS	Reed-Solomon
RX	Receiver
SA	Source Address
SDID	Source node DOMAIN_ID
SN	Sequence Number
SNID	Source node NODE_ID
SNR	Signal to Noise Ratio
SR	Source Routing
SRI	Source Routing header presence Indicator
SSC	Supported Subcarriers
TS	Time Slot
TX	Transmitter

UAN	Utility Access Network
UM	Unified Mode
UTCE	Unable To Comply Extension

5 Network architecture and reference models

5.1 Network architecture and topology

5.1.1 Basic principles of ITU-T G.9902 networking

The following are the basic principles of ITU-T G.9902 network architecture:

1. The network is divided into domains:
   - The division of a physical network into domains is logical; no physical separation is required, so domains may fully or partially overlap, i.e., some nodes of one domain may directly (on the physical layer) communicate with some nodes of another domain.
   - The number of domains within the physical network may be up to N.
   - Each domain is identified by a domain ID that is unique inside the network.
   - Nodes of different domains can communicate with each other via inter-domain bridges (IDB). The IDB functions are provided by one or more nodes dedicated to operate as an IDB.
   - A network may include alien domains, as well as ITU-T G.9902 domains. The connection between ITU-T G.9902 domains and alien domains is via L3 bridges.
   - The operation of different domains in the same network may be coordinated by the global master (GM). The function of the GM is associated with one of the nodes in one of the network's domains.
2. The domain is a set of nodes connected to the same medium:
   - One node in the domain operates as a domain master.
   - Each domain may contain up to M nodes (including the domain master).
   - Each node in the domain is identified by a node ID that is unique inside the domain.
   - All nodes that belong to the same domain are indicated by using the same domain ID. A particular single node can belong to only one domain.
   - Nodes of the same domain can communicate with each other either directly or via other nodes of the same domain called relay nodes. Domains where not all nodes can directly communicate with each other are called "partially connected".
3. Nodes of different ITU-T G.9902 networks:
   - Can communicate via inter-network bridges (INB). The INB function is an L3 bridging function associated with one or more dedicated nodes of network domains.

The generic network architecture of an ITU-T G.9902 network is presented in Figure 5-1.

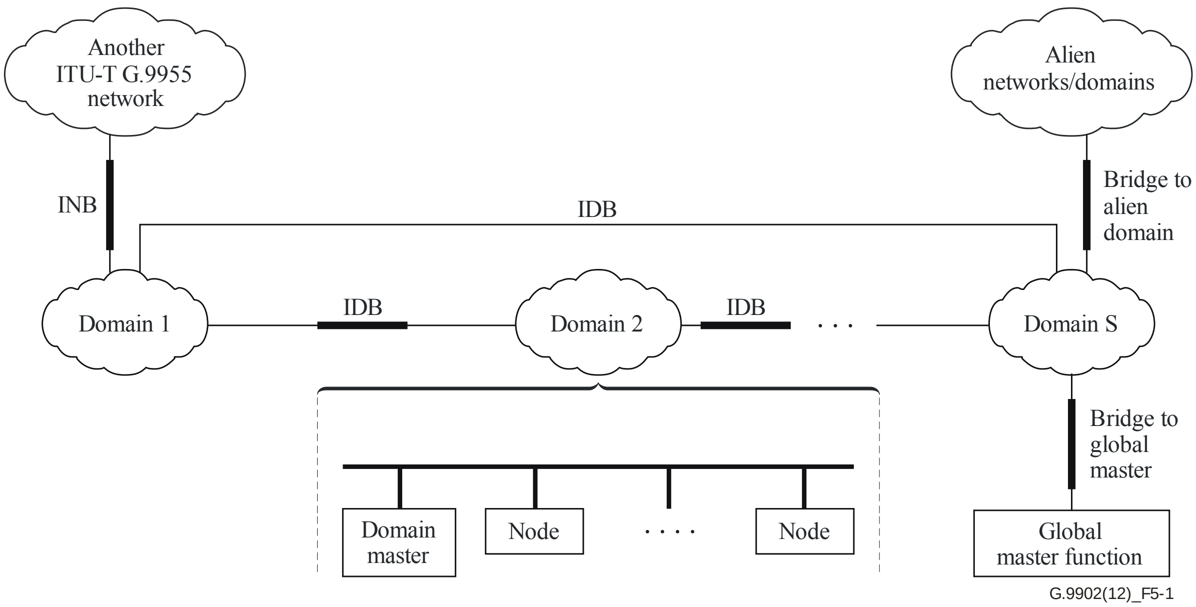


Figure 5-1 – Generic network architecture

The details of domain operation rules, the types of communication inside a domain and the functionalities of the domain master and endpoint nodes are described in clause 5.1.3. An ITU-T G.9902-based network supports a mesh topology that allows each node to communicate with any other node either directly, via one or more relays, or via relays and IDBs. This allows any type of network topology to be supported, such as star, tree, multiple trees and others. The maximum number of domains, N, and maximum number of nodes per domain, M, depends on the particular type of network.

Alien domains and bridges to alien domains are beyond the scope of this Recommendation. This Recommendation defines all the necessary means to support the IDB and INB functionality plus the exchange of any relevant information.

The scope of this Recommendation is limited to the PHY layer of ITU-T G.9902 transceivers capable of operating either with extended capabilities (e.g., domain master, relay node, or combinations thereof) or without extended capabilities as end-nodes.

5.1.2 Energy management network architecture and topology

An example architectural model of the EM network is presented in Figure 5-2. It contains the utility head-end, the multi-domain utility access network (UAN) and the energy-management home area networks (EM-HAN) at customer premises. Each EM-HAN can include one or more domains (not shown in Figure 5-2 – see clause 5.1.2.2 for EM-HAN architecture).

The domains of the UAN incorporate all devices that are owned and physically belong to the UAN (e.g., meters), while the HAN includes all customer-owned and some utility-owned devices related to energy management (e.g., home appliances, PSTs, EVSEs) that reside at customer premises. In this example, each HAN is connected to a UAN via an INB; the INB function is implemented by the energy service interface (ESI).

NOTE – This architectural model is exclusively for reference purposes and does not limit the use of ITU-T G.9902 transceivers for other network configurations.

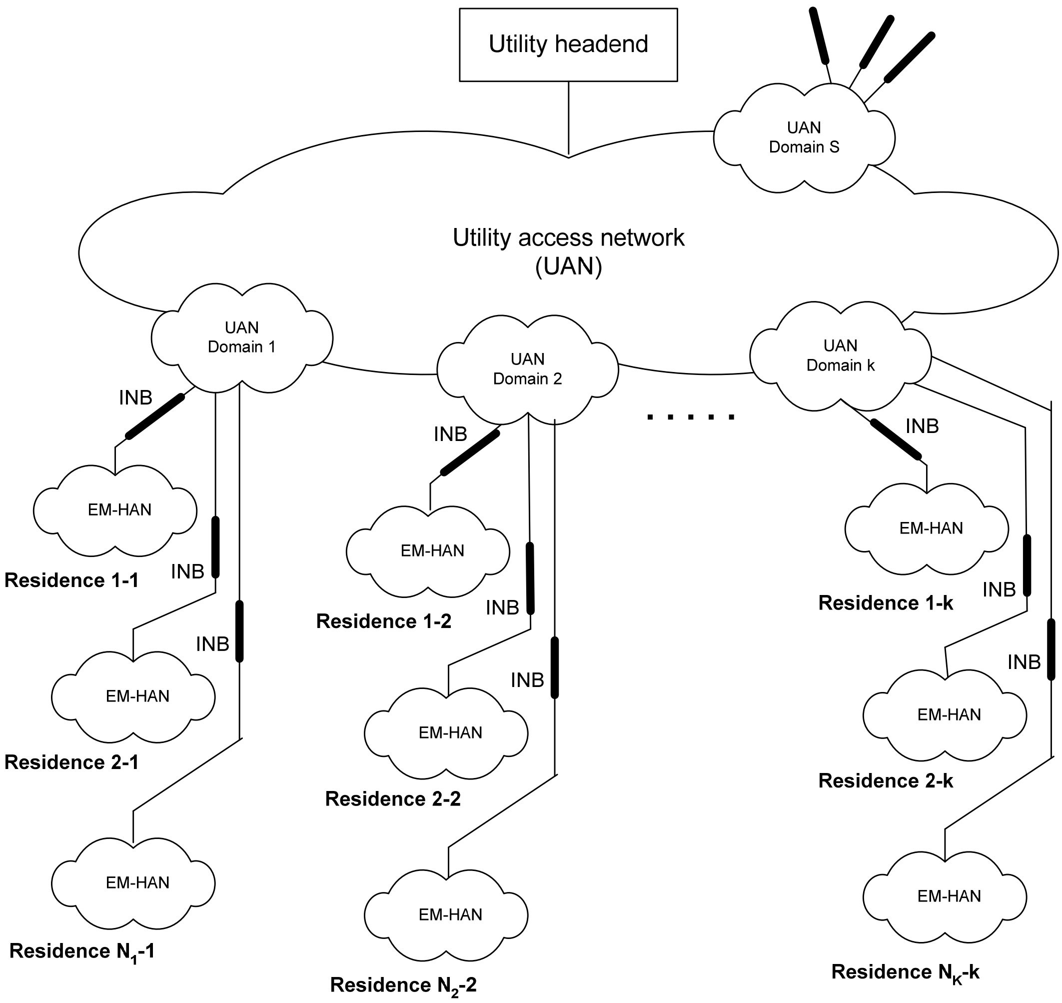


Figure 5-2 – Generic EM network architecture

5.1.2.1 Generic UAN architecture

The UAN is logically divided into domains. Each domain is associated with a particular set of ITU-T G.9902-based nodes connected to the same medium (usually, a power line). A particular node can only belong to one domain (this does not preclude a physical device incorporating multiple logical nodes belonging to different domains).

All nodes of a UAN domain are controlled by a domain master; other nodes are called end-nodes.

Nodes of the same UAN domain can communicate with each other directly or via other nodes of the same domain (relay nodes). Two or more UAN domains may overlap: nodes of overlapping domains may "see" transmissions of each other and thus may interfere with each other.

The UAN domains may be connected to each other by one or more IDBs (see example in Figure 5-5); this allows nodes of each domain to at least get connected to the utility head-end. Nodes of different UAN domains may communicate with each other using one or multiple IDBs. The GM function of the UAN coordinates the operation of all UAN domains (resources, priorities, operational characteristics) via the corresponding domain masters. This high-level management function can be performed by one of the nodes of the UAN.

NOTE – The typical structure of a UAN is a tree (see Figure D.3) and the utility head-end functions as a global master of the UAN. There might be one or more nodes per CP, including a node implementing ESI to connect between the UAN and EM-HAN.

In addition to ITU-T G.9902 domains, a UAN may also include alien domains. These domains are established by non-ITU-T G.9902 technologies, both wired and wireless. Alien UAN domains can be bridged to the ITU-T G.9902 domains using L3 bridges. The specification of bridges to alien UAN domains is beyond the scope of this Recommendation.

5.1.2.2 Generic HAN architecture

The EM-HAN (further called "HAN") is logically divided into domains. Each domain shall be associated with a particular set of ITU-T G.9902 nodes. A particular node can only belong to one domain. Nodes of the same HAN domain communicate via the medium over which the domain is established. The nodes of different HAN domains communicate with each other via IDBs. The HAN is connected to the UAN (if necessary) via an INB which is a part of the gateway between the HAN and the UAN. The interface between the UAN and the HAN is called an ESI.

The domains of the HAN are established using in-home wiring, usually power lines, but may also use other types of wired media. One of the HAN domain devices is the domain master, while all other nodes are called end-nodes. Two or more HAN domains may overlap: nodes of overlapping domains may "see" transmissions of each other and may interfere with each other.

In addition to ITU-T G.9902 domains, a HAN may include alien domains. These domains can be established using in-home wireline or wireless media. Alien HAN domains can be bridged to the ITU-T G.9902 domains using L3 bridges. The specification of bridges to alien HAN domains is beyond the scope of this Recommendation.

When coordination between domains of the HAN (resources, priorities, operational characteristics) is required, it is provided by the GM function of one of the nodes, which is a high-level management function that may also convey the relevant functions initiated by a remote management system.

The generic architecture of a HAN containing both ITU-T G.9902 domains and alien domains is presented in Figure 5-3.

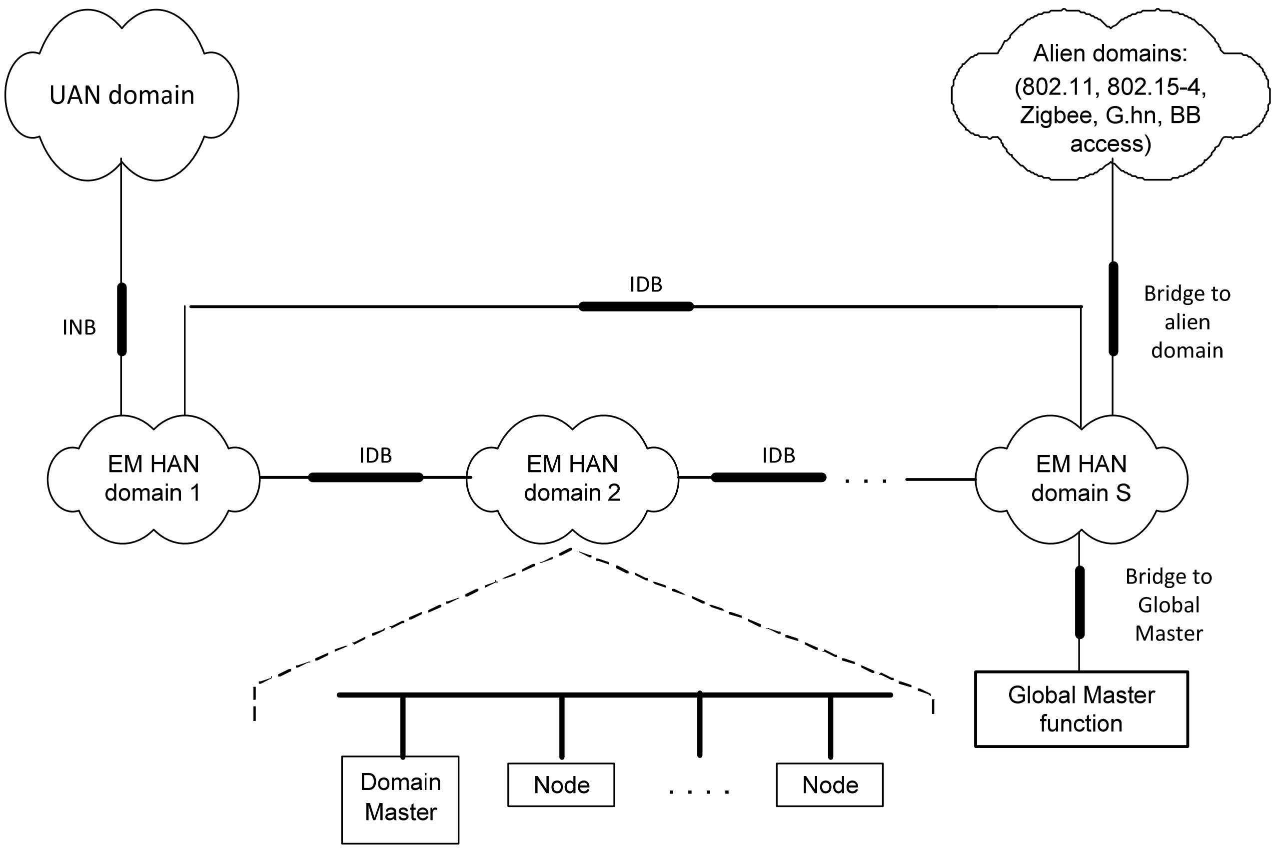


Figure 5-3 – Generic architecture of EM-HAN

NOTE 1 – It is not necessary for all IDBs presented in Figure 5-3 to be used. Depending on the application, domains could be daisy-chained, star-connected, or could use another connection topology. Support of multi-route connections between domains is for further study.

NOTE 2 – The end-nodes of the HAN are also those operating in the resident electric vehicle charging facility (EVCF), both in its stationary part – the electric vehicle supply equipment (EVSE) and in the plug-in electric vehicle (PEV).

An example of a HAN containing one ITU-T G.9902 domain and one alien domain is presented in Figure 5-4. The nodes of the ITU-T G.9902 domain include one which is installed in an EVSE and another which serves to connect the in-home energy management system (EMS). The alien domain is bridged to the ITU-T G.9902 domain via L3 IDB.

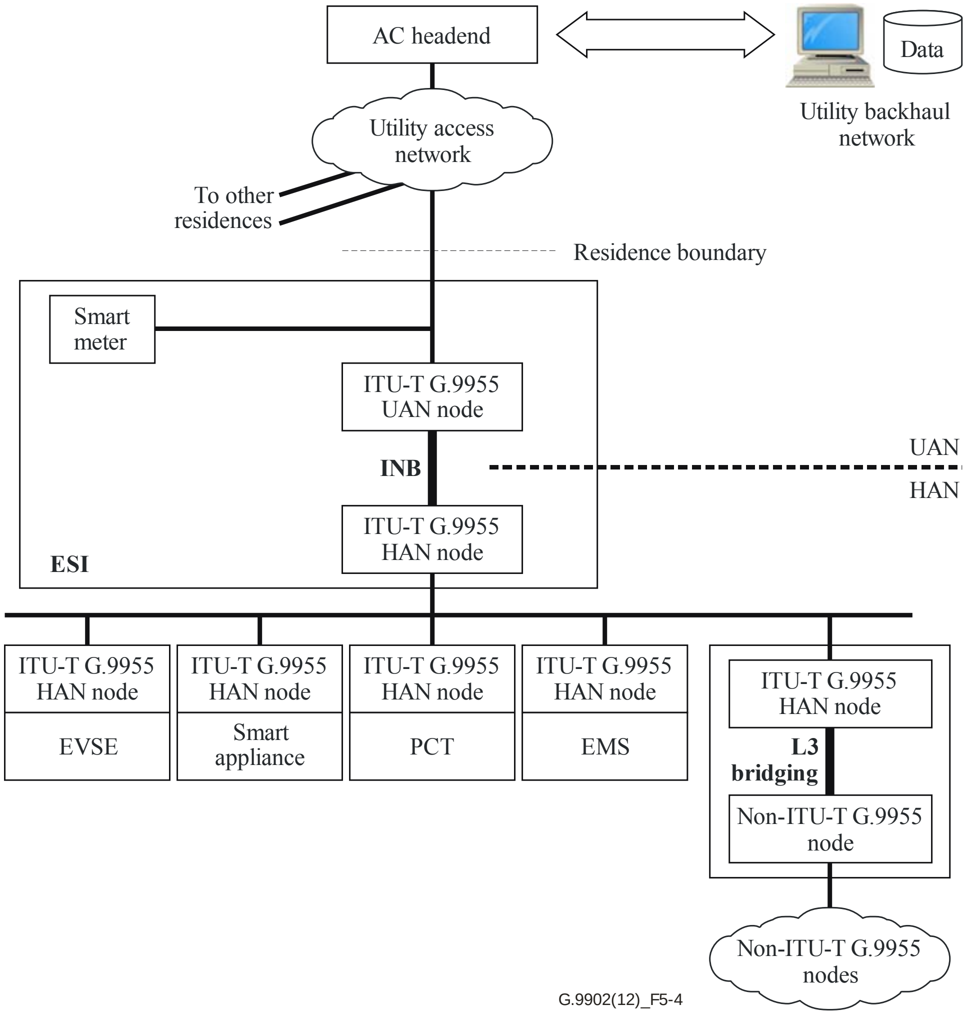


Figure 5-4 – Example of a functional diagram of an EM-HAN connected to the utility

5.1.2.3 Coexistence with other PLC networks

There are three coexistence mechanisms defined below which allow coexistence with other PLCs operating in the same frequency range:

– Frequency division (FD) coexistence mechanism – allows the suppression of interference from ITU-T G.9902 into a particular frequency band or bands by using non-overlapping ITU-T G.9902 bandplans (see clause 8.5). Flexible use of different bandplans provides an opportunity to separate systems operating over the same medium in non-overlapping bandplans. The FD coexistence mechanism can provide coexistence with both the narrowband FSK/PSK PLC systems and wideband PLC systems.

– Frequency notching coexistence mechanism – shall be used to suppress interference from ITU-T G.9902 into a particular (relatively narrow) frequency range by notching out one or more subcarriers (see clause 8.6.1). Frequency notching allows ITU-T G.9902 to coexist with the existing narrowband FSK/PSK systems operating over the same frequency band.

– Preamble-based coexistence mechanism – shall be used by ITU-T G.9902 to fairly share the medium with other types of PLC technologies operating over the same frequency band (and utilizing this coexistence mechanism). The definition of this coexistence mechanism is for further study. This same coexistence mechanism also facilitates coexistence between the ITU_T G.9902 implementations using different overlapping bandplans.

The above coexistence mechanisms can be applied simultaneously, enabling ITU-T G.9902 coexistence with multiple PLC technologies operating over the same medium.

5.1.3 Domain

5.1.3.1 General rules of operation

Both HAN and UAN domains, as depicted in Figure 5-5 of this Recommendation, comprise of a domain master and one or more end-nodes. Nodes may be of basic capabilities (usually, a majority of end nodes) and of extended capabilities such as a relay or a domain master, or a domain access point (DAP).

The following rules apply for any domain:

1. The function of the domain master is to manage and coordinate the operation of all nodes in its domain. The specific functions of the domain master and details of the particular management procedures are for further study.
2. There shall be one and only one active domain master per domain at a given time.
3. Nodes of the same domain communicate with each other using general rules defined in clause 5.1.1.

   NOTE – Nodes of different domains can communicate if applications employing these nodes have compatible security requirements. Details are for further study.
4. Domains of independent HANs or UANs may interfere with each other. Methods of coordination between domains of independent HANs or UANs are for further study.
5. Nodes are not required to be domain master capable or relay-capable or DAP-capable. That is, some nodes may not support the functionality necessary to become a domain master or a relay or a DAP.
6. All nodes within the same domain shall indicate their domain ID in all communications. Each node of a domain shall keep track of the domain IDs and shall discard frames received from all domains other than its own, except those assigned for inter-domain communication. The details are for further study.
7. Broadcast transmissions shall be supported in any domain.

5.1.3.2 Unicast communication types

Nodes of a domain can use three types of communication: peer-to-peer communications (P2P), centralized mode communications (CM), or unified mode communications (UM).

With P2P communications direct signal traffic is established between two communicating nodes. Figure 5-5 shows the use of P2P between nodes A and B. Frames addressed to nodes outside the domain are sent to the node associated with the corresponding IDB (node C in Figure 5-5).

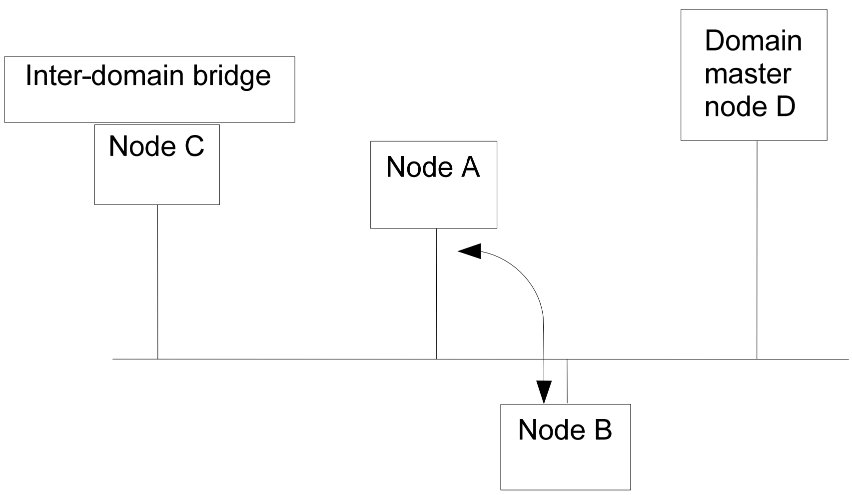


Figure 5-5 – Example of a P2P communication

For CM, only relayed communications are used. Thus, any node of the domain can communicate with another node through the DAP node only. The DAP receives frames from all nodes of the domain and further forwards them to the corresponding addressee nodes. Frames addressed to nodes outside of the domain are forwarded by the DAP to the node associated with the corresponding IDB (node C in Figure 5-6). Usually, but not necessarily, the DAP also serves as a domain master (Figure 5-6). If a particular node has no direct connection to the DAP (it is hidden from the DAP), this node may use other nodes as relays to reach the DAP (if a direct connection between nodes C and D shown in Figure 5-6 is impossible, node A may relay the frame from C to D).

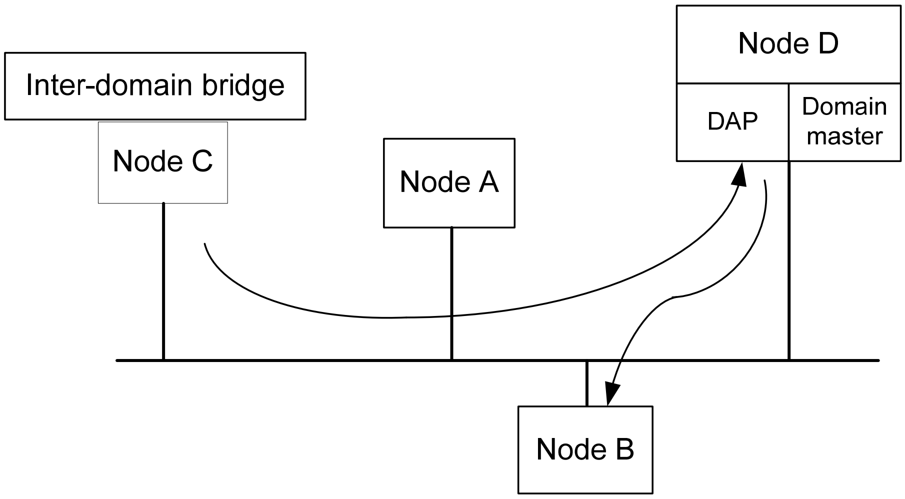


Figure 5-6 − Example of a CM communication

In the case of DAP failure, no communication between nodes in the domain is allowed in CM.

For a UM communication, a node can communicate with another node either directly or through one or more relay nodes as shown in Figure 5-7. In the example, two nodes within the same domain (nodes C and H) that are hidden from each other, communicate with each other via the relay node (node A). Both nodes are managed by the domain master (node D) and can communicate directly with all other nodes. Frames addressed to nodes outside the domain are sent to the node associated with the IDB (node C in Figure 5-7).

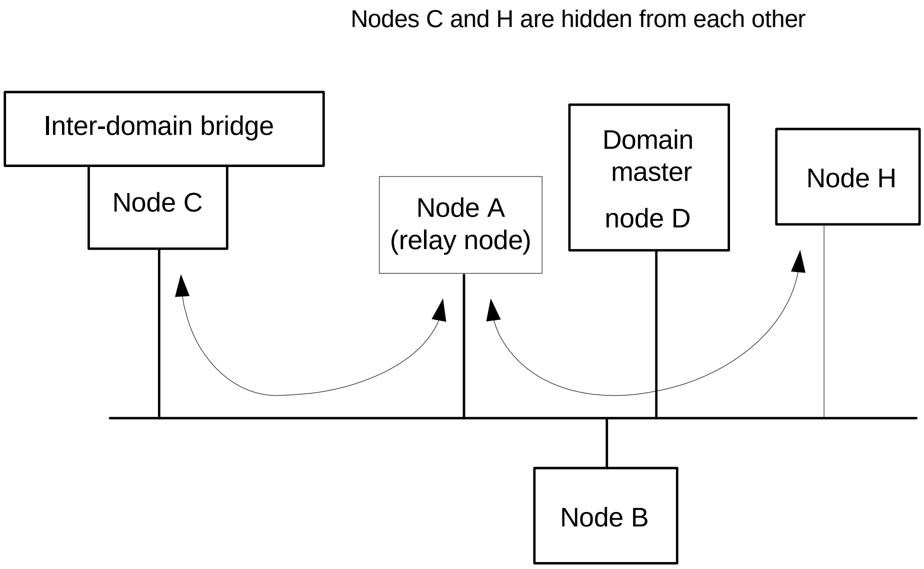


Figure 5-7 − Example of a UM communication

5.1.3.3 Management of the domain

A domain master manages the domain by transmission beacon frames and a set of control/management messages. Beacon frames carry information to be distributed to all nodes of the domain, thus describing for all nodes of the domain the specifics of the domain operation, such as bandplan, domain name, domain ID, the security mode to be used, spectral compatibility requirements (frequency band and maximum transmit power, if applicable), mode of operation, and others (see clauses 9.4.2 and 9.5.4). Nodes that join the domain shall detect the beacon and setup the required parameters to comply with the domain operation requirements prior to starting the transmission. Beacons also inform all operating nodes on changing the status of the domain or operation rules of the domain.

The domain master also manages domain topology; it may request topology reports from the nodes of the domain, store, process and distribute relevant topology information back to the nodes. If L2 routing is set for the domain, the domain master assigns one or more nodes as relays. The domain master shall also assign beacon proxies (BPR). The BPR re-generates the beacon to extend the coverage of the network.

The domain master is also responsible to collect statistics on the performance of the domain and provide domain management information to the nodes of the domain.

The main management functions provided by the domain master are summarized in Table 5-1; the detailed description of these functions is in clause 9.5.

Table 5-1 – Domain master management functions

Function	Description
Indication of presence	– Periodically communicate beacon to all nodes in the domain.
Determine the domain operation rules	– Inform all nodes on current operation modes and settings in the domain and provide their modification if necessary.
Admission control	– Register new nodes into the domain. – Facilitate the resignation of nodes from the domain. – Expel nodes from the domain in the case of domain policy violation.
Topology management	– Collect and maintain topology information from nodes of the domain. – Process and distribute the topology information. – Facilitate routing in the domain (assigns relay nodes and BPRs).
Statistics collection	– Collect statistics on performance of the nodes. – Store, process, and communicate statistics to the management centres.

5.1.4 Quality of service (QoS)

Quality of service (QoS) defines the quality of service delivery to the clients in the network that places requirements on the transmission and queueing of the traffic. All ITU-T G.9902 transceivers shall support priority-based QoS. Support of parameter-based QoS is for further study.

The QoS requirements are supported between nodes inside the same domain and between nodes connected to different domains (if services are communicated between nodes that belong to different domains). In the latter case, inter-domain bridges are expected to facilitate the provisioning of QoS between nodes connected to different domains and not to compromise the QoS requirements (such as latency). This provisioning is beyond the scope of this Recommendation.

With priority-based QoS, the ITU-T G.9902 transceiver assigns each protocol data unit, incoming from the application entity or management entity, to a certain priority queue, based on the QoS-related parameters indicated by the primitives of this incoming data unit or by QoS-related management primitives. Furthermore, the defined priority-based access method determines the order in which frames from each queue shall be sent to the medium and in which order frames will be processed (and possibly dropped), based on the medium access (MA) priorities assigned to these queues (see clause 8.2.1). The number of supported MA priority queues may be 4, 3 or 2, depending on the profile (see clause 7): standard profile supports 4 queues. Three priority levels associated with the regular incoming application data (at A-interface) are defined, denoted by 0 (lowest level), 1, and 2 (highest level). The priority level 3 is defined as an "over-ruling" priority for pre-defined emergency data transfers and asynchronous beacons only, and it is not intended for regular application data. Management communications use priority level 2 (highest level of application data priority).

5.1.5 Security

ITU-T G.9902 security is designed to address operations over shared media, such as power lines. It uses an advanced encryption mechanism based on AES-128 [NIST PUB 197] and the counter mode with the cipher block chaining message authentication code algorithm (CCM), according to [NIST SP 800-38C]. The defined node authentication and key management procedure (AKM) provides authentication of each node registering into the domain and can establish point-to-point security, with unique encryption keys assigned for each pair of communicating nodes. AKM also includes encryption key update and periodical re-authentication of all nodes in the domain.

ITU-T G.9902 security settings are configurable: the domain may be set to a secure mode with different levels of message authentication or to a non-secure mode, without encryption and message authentication. All nodes of the domain are set to the same security level, indicated by the domain master. Security of a network containing more than a single domain is provided by setting all the domains of the network in secure mode. If a domain operates in non-secure mode or has an insufficient level of security or message authentication, its communication with secure domains shall be restricted; communication between authenticated nodes of a secure domain with nodes of insecure domains shall not be authorized.

This Recommendation defines security on layer 2 (DLL). If security is provided by layers above L2, fully or partially, a non-secure mode or a mode with reduced security for L2 may be appropriate.

Security procedures are user-friendly, but may require the user to establish a password for each node prior to installation. The rest of the procedures necessary to establish and maintain security do not require any involvement of the user. Nodes with no appropriate user interface may use a manufacturer-set password (blind admission).

Confidentiality between clients associated with the same node is supposed to be resolved at the higher layers of the client protocol stack and is beyond the scope of this Recommendation. Details of security are defined in clause 9.6.

5.2 Reference models

5.2.1 Protocol reference model of transceiver

The protocol reference model of a transceiver is presented in Figures 5-8 and 5-9. They include three main reference points: application interface (A-interface), physical medium-independent interface (PMI), and medium-dependent interface (MDI). Two intermediate reference points, x1 and x2, are defined in the data link layer and two other intermediate reference points, α and δ, are defined in PHY layer. The shaded part of the reference model in Figure 5-8 is defined in clause 8; the shaded part of the reference model in Figure 5-9 is defined in clause 9. The layers above the data link layer (above the A-interface) are beyond the scope of this Recommendation.

The MDI is a physical interface defined in terms of the physical signals transmitted over a medium and the mechanical connection to the medium (clause 5.2.2.3).

The PMI is both medium independent and application independent. The A-interface is network layer (Layer 3) protocol specific (e.g., Ethernet, IP). Both the PMI and A-interface are defined as functional interfaces, in terms of sets of primitives exchanged across the interface.

All intermediate reference points are medium independent and are defined as functional (logical) interfaces in terms of the logical primitives exchanged across these reference points.

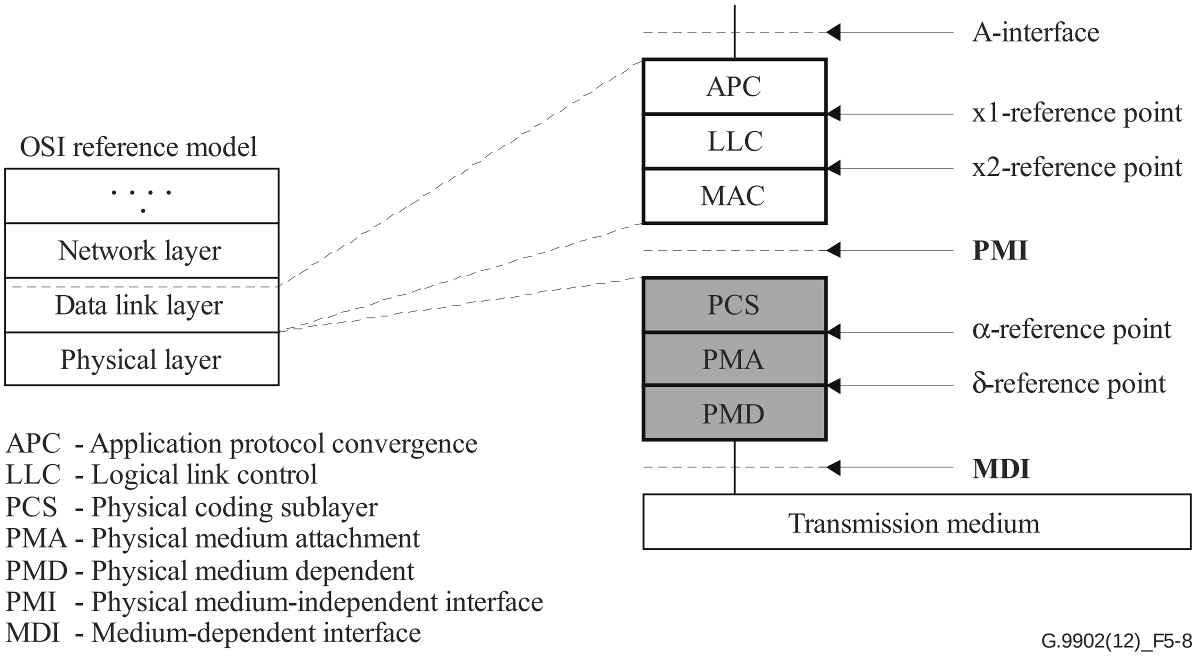


Figure 5-8 – Protocol reference model of an ITU-T G.9902 transceiver (PHY)

The application protocol convergence sublayer (APC) provides an interface with the network layer (layer 3), also called the application entity (AE), which operates with an application-specific protocol, such as IP. The APC also provides the bit rate adaptation between the AE and the transceiver.

The logical link control sublayer (LLC) coordinates the transmission of nodes in accordance with the rules of operation in the domain. In particular, it is responsible for establishing, managing, resetting and terminating all connections of the node with other nodes in the domain. The LLC also facilitates any quality of service (QoS) constraints defined for its established connections.

The medium access control sublayer (MAC) controls access of the node to the medium using medium access protocols defined in clause 8.4 of the Recommendation.

The physical coding sublayer (PCS) provides bit rate adaptation (data flow control) between the MAC and PHY and encapsulates transmit MPDUs into the PHY frame and adds a PHY-related control and management overhead.

The physical medium attachment sublayer (PMA) provides forward error correction encoding and interleaving of the PHY frame content (header and payload) for transmission over the medium.

The physical medium dependent sublayer (PMD) modulates encoded PHY frames for transmission over the medium using orthogonal frequency division modulation (OFDM). In the receive direction, PMD demodulates PHY frames incoming from the medium.

The functionality of the DLL and PHY is the same for any type of medium (e.g., utility access LV and MV wires, in-home power line wires, in-home phone wires, or similar) or any application, although their parameters may be medium-specific or application-specific. With appropriate parameter settings (determined by the transceiver management functions), operation of a single node and all nodes in the domain can be configured to fit the type of medium or a particular application.

Partitioning into data and management functions are not presented in Figure 5-9 and described in clause 5.2.2.

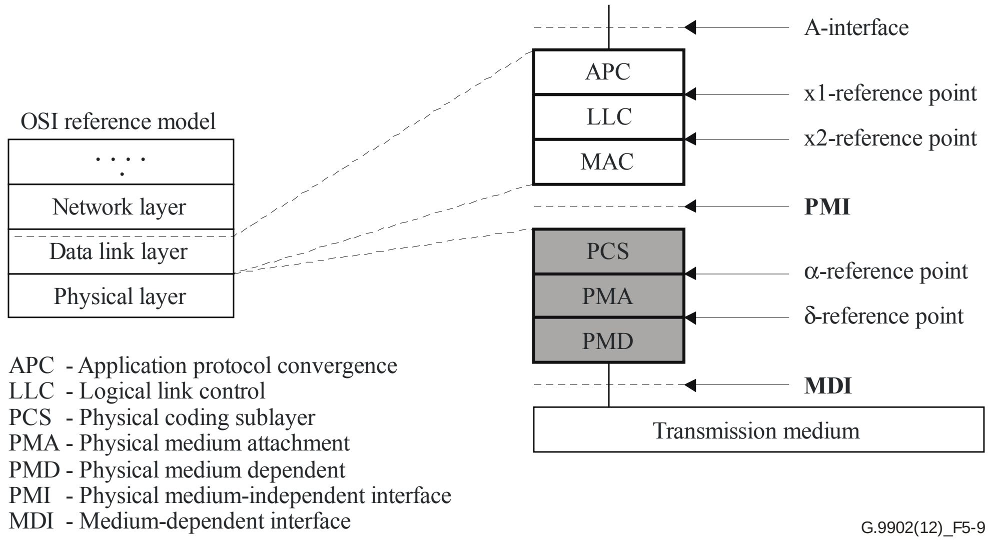


Figure 5-9 – Protocol reference model of an ITU-T G.9902 transceiver (DLL)

5.2.2 Functional description of interfaces

This section contains the functional description of the ITU-T G.9902 transceiver interfaces (reference points) based on the protocol reference model presented in Figures 5-10 and 5-11. The interfaces shown in Figure 5-10 are defined in clause 8; the interfaces shown in Figure 5-11 are defined in clause 9.

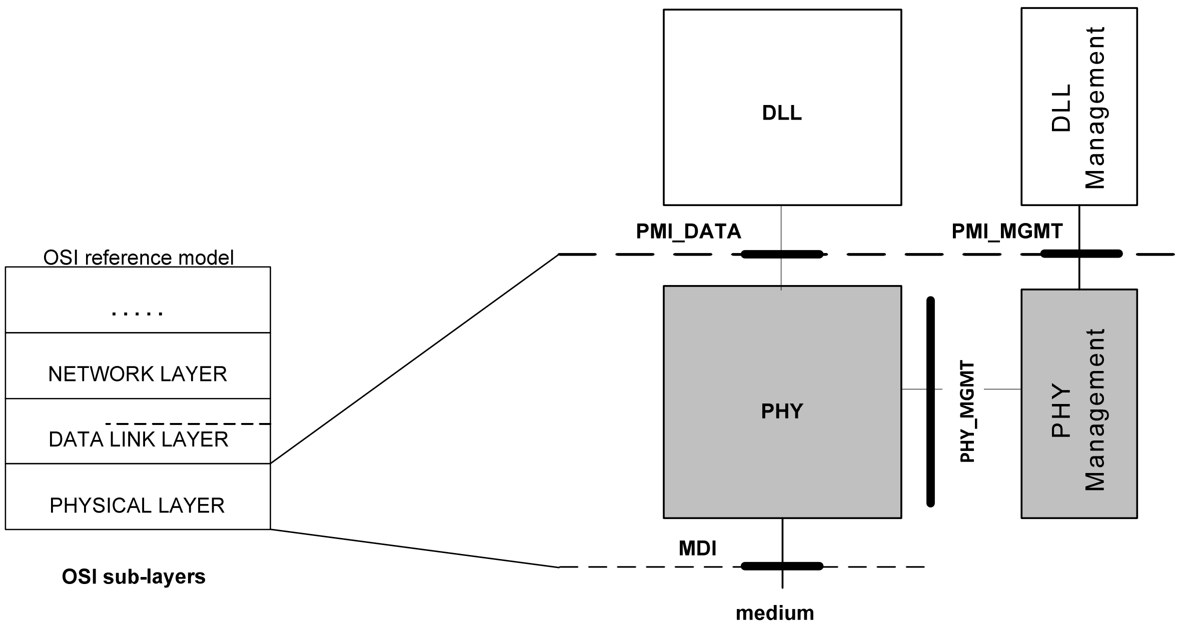


Figure 5-10 − Transceiver reference points related to PHY

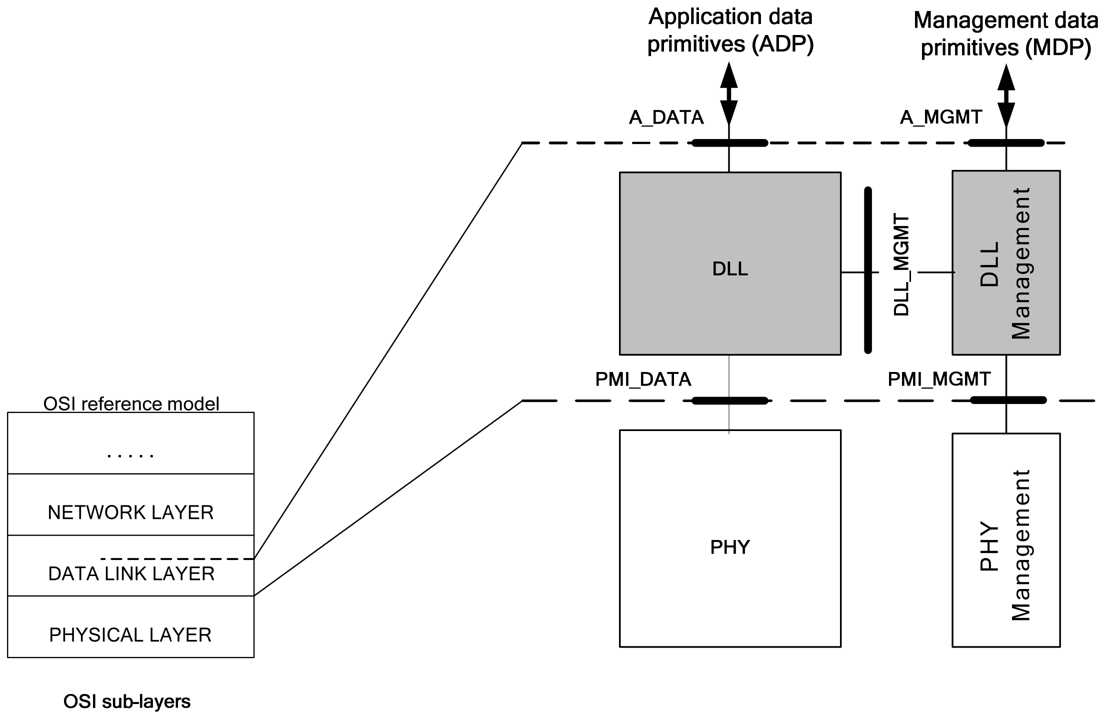


Figure 5-11 − Transceiver reference points related to DLL

The model in Figure 5-10 shows interfaces related to the application data path (PMI_DATA, and MDI), the management data path (PMI_MGMT), and management interfaces between data and management plains of the PHY (PHY_MGMT). The model in Figure 5-11 shows interfaces related to the application data path (A_DATA, PMI_DATA), the management data path (A_MGMT, PMI_MGMT), and management interfaces between data and management plains of the DLL (DLL_MGMT). All interfaces are specified as reference points in terms of primitive flows exchanged between the corresponding entities. The description does not imply any specific implementation of the transceiver interfaces.

5.2.2.1 A-interface

The A-interface is described in terms of primitives exchanged between the AE and DLL. There are six general types of A-interface primitives as shown in Table 5-2. Each primitive type may consist of one or more primitives, related to control or data respectively. Data primitives (A_DATA) represent the data path of the A-interface, while management primitives (A_MGMT) represent the management and control path. The format of the application data primitives (ADPs) and management data primitives (MDP) is application specific, determined by the AE. The definition of the A-interface management/control and data primitives of each type of the AE are defined in Annex B. The A-interface management primitives that are not AE-specific are defined in clause 9.4.3.4.

Table 5-2 – A-interface primitive type summary

Primitive type	Direction	Description
A-interface data primitives		
A_DATA.REQ	AE → DLL	Data from AE to DLL
A_DATA.CNF	DLL →AE	Data confirmation from DLL to AE
A_DATA.IND	DLL → AE	Data from DLL to AE
A-interface management and control primitives		
A_MGMT.REQ	AE → DLL	Control from AE to DLL
A_MGMT.CNF	DLL → AE	Control confirmation from DLL to AE
A_MGMT.IND	DLL → AE	Control from DLL to AE
NOTE – Primitives presented in this table are exclusively for descriptive purposes and do not imply any specific implementation.		

5.2.2.2 Physical medium-independent interface (PMI)

The PMI is described in terms of primitives exchange between the DLL and PHY layer presented in Table 5-3; the direction of each primitive flow indicates the entity originating the primitive. Both transmit and receive data primitives are exchanged by MAC protocol data units (MPDUs). The details of the PMI_DATA and PMI_MGMT primitives are defined in clause 8.8.

Table 5-3 – PMI primitive description

Primitive	Direction	Description
PMI-interface data primitives		
PMI_DATA.REQ	DLL → PHY	DLL requests the PHY to transmit an MPDU or an ACK frame
PMI_DATA.CNF	PHY → DLL	PHY frame transmission status (transmission complete, not complete, failed)
PMI_DATA.IND	PHY → DLL	Received frame passed by the PHY to the DLL
PMI-interface management and control primitives		
PMI_MGMT.REQ	DLL → PHY	Transmission and configuration parameters asserted by the DLL
PMI_MGMT.CNF	PHY → DLL	Confirms transmission and configuration parameters asserted by DLL (accepted or rejected)
PMI_MGMT.IND	PHY → DLL	Transmission parameters of the received frame payload and channel characteristics reported by the PHY
PMI_MGMT.RES	DLL → PHY	Acknowledges transmission parameters of the received frame and channel characteristics reported by the PHY
NOTE – Primitives presented in this table are exclusively for descriptive purposes and do not imply any specific implementation.		

5.2.2.3 Medium-dependent interface (MDI)

Functional characteristics of the MDI are described by two signal flows:

- Transmit signal (TX DATA) is the flow of PHY frames transmitted onto the medium.
- Receive signal (RX DATA) is the flow of PHY frames received from the medium.

Electrical characteristics of the MDI are described in clause 8.7.

5.2.2.4 Peer interfaces between data and management paths

5.2.2.4.1 PHY_MGMT reference point

This reference point defines control and management primitives related to all sublayers of the PHY (PCS, PMA, PMD), as defined in Figure 5-8. These primitives (PCS_MGMT, PMA_MGMT, and PMD_MGMT) are shown in the DLL functional model, clause 8.1, and defined in clause 8.8.

5.2.2.4.2 DLL_MGMT reference point

This reference point defines management and control primitives related to all sublayers of the DLL (APC, LLC, MAC), as defined in Figure 5-4. These primitives (APC_MGMT, LLC_MGMT, and MAC_MGMT), are shown in the DLL functional model, clause 9.1, and defined in clause 9.4.3.

5.2.3 Functional model of transceiver

The functional model of a transceiver is presented in Figures 5-12 and 5-13. It addresses nodes without extended capabilities as well as nodes with extended capabilities such as the domain master. The shaded part of the functional model in Figure 5-12 is addressed in clause 8; the shaded part of the functional model in Figure 5-13 is addressed in clause 9.

The detailed description of the functional model of the PHY layer is presented in clause 8.1.

The detailed description of the functional model of the DLL is presented in clause 9.1.

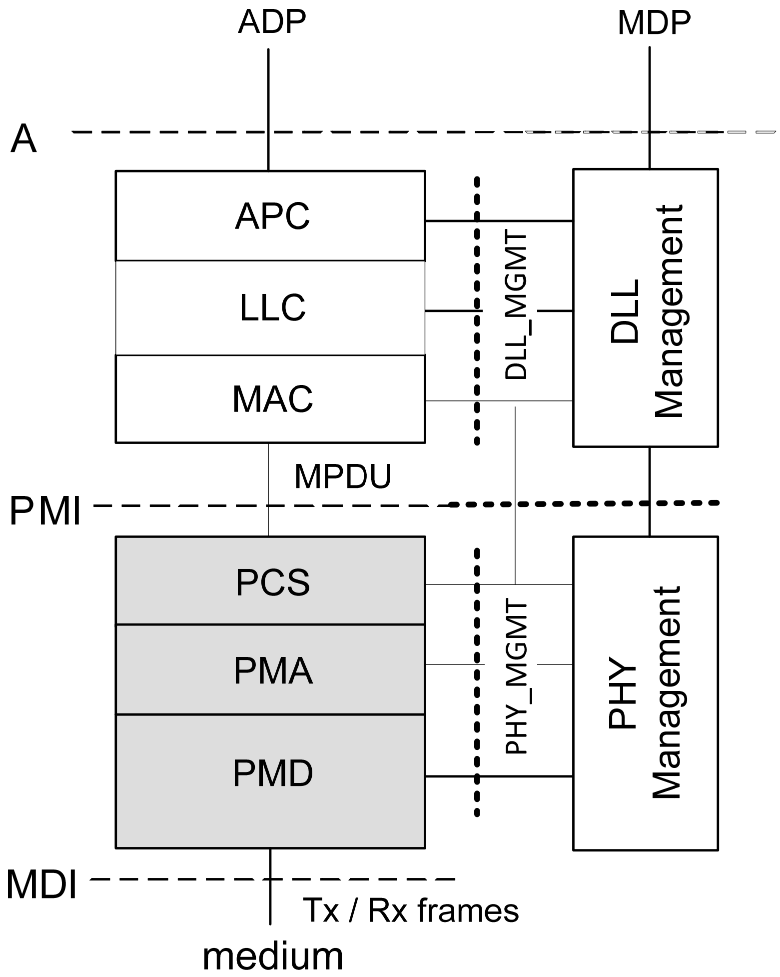


Figure 5-12 − Functional model of an ITU-T G.9902 transceiver (PHY)

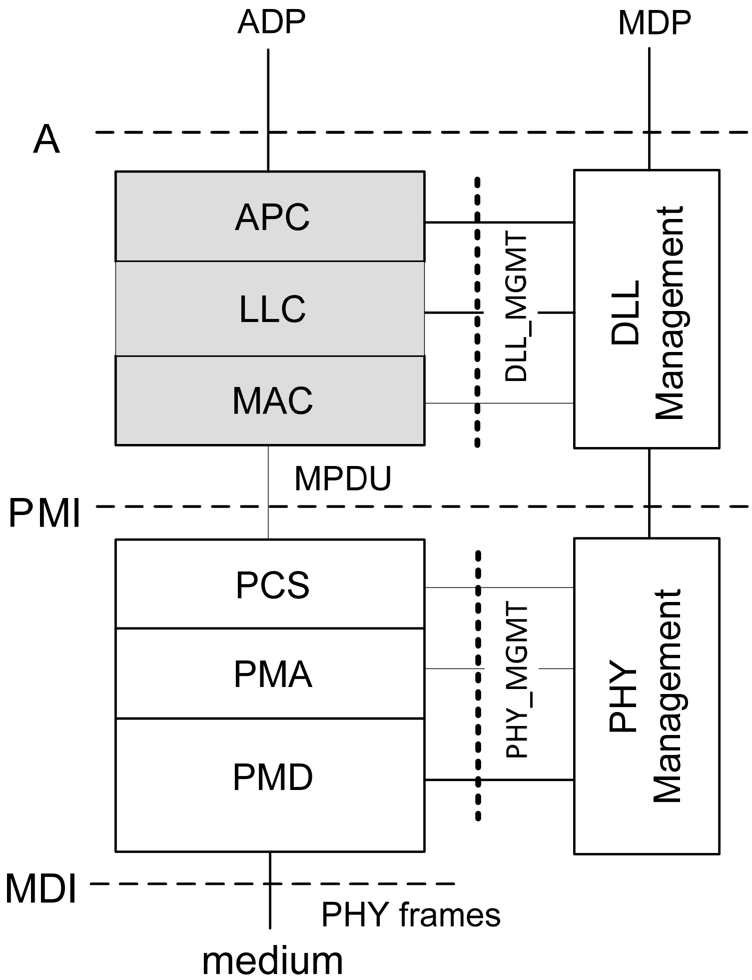


Figure 5-13 − Functional model of an ITU-T G.9902 transceiver (DLL)

6 Conventions

6.1 Bit ordering convention

A block of data composed of multiple octets shall be ordered by octet numbers in ascending order: 'octet 0' for the first octet, 'octet 1' for the second octet, and so on. If a block of data is segmented into multiple fields, the size of each field shall be expressed in terms of bits. The field may not be of an integer number of octets. The location of each field within a block of data shall be described as follows:

- The octets of an *N*-octet data block are ordered with numbers from 0 (first octet) to *N*–1 (last octet).
- The block is divided into non-overlapping groups of octets. Each group contains an integer number of consecutive octets, numbered from *J* to *J*+*V*–1, where *V* is the size of the group, and is described as a bit string with 'bit 0', the LSB of the octet with the smallest number (*J*), and 'bit (8×*V*–1)', the MSB of the octet with the largest number (*J*+*V*–1).
- Each group is divided into one or more fields, where the boundaries of each field are determined by the LSB and the MSB of the bits of the group that contains this field.

Any block of data or part of it shall be passed over the protocol stack with the octet having the smallest number first, i.e., octet 0 shall be the first octet of the block to be passed. Within each group of octets, the LSB (bit 0) of each octet shall be passed first.

Table 6-1 shows an example of a field description used throughout this Recommendation. The 'Octet' column represents the octet numbers for a group of octets to which a specific field belongs, and the 'Bits' column represents the bit location within this group of octets. In the presented example, there are 4 groups of octets:

- Group 1 = Octet 0, fields A, B, C, D
- Group 2 = Octets 1 and 2, fields E, F
- Group 3 = Octet 3, field G
- Group 4 = Octets 4 to 7, field H.

Figure 6-1 illustrates a mapping of these fields onto corresponding octets based on the example given in Table 6-1.

Table 6-1 – Example of field description

Field	Octet	Bits	Description
A	0	[2:0]	…
B	0	[3]	…
C	0	[4]	…
D	0	[7:5]	…
E	1	[1:0]	…
F	1-2	[15:2]	…
G	3	[7:0]	…
H	4-7	[31:0]	…

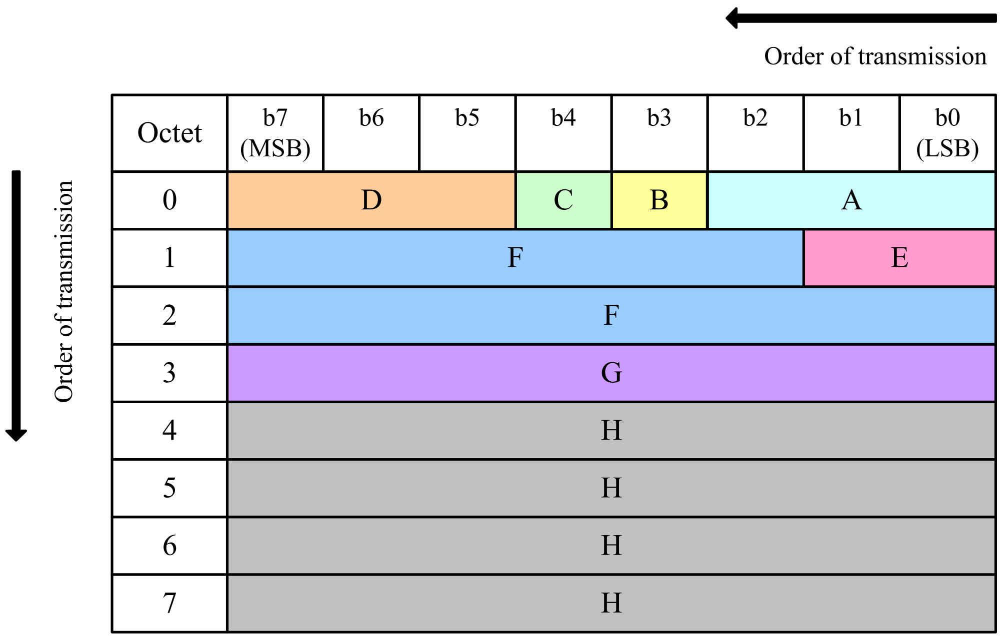


Figure 6-1 – Example of mapping fields onto groups of octets

6.2 Message nomenclature convention

The nomenclature for management messages defined in this Recommendation follows the convention described in this clause. The generic nomenclature format of the management message contains the name and the type: *Name.Type*.

Four message types are defined: Request (*req*), Confirm (*cnf*), Indication (*ind*) and Response (*rsp*). The *.req* and *.ind* messages are intended to initialize the message exchange. The *.cnf* message shall be sent only in response to the received *.req* message. The *.rsp* message shall be sent only in response to the received *.req* message. An *.ind* message may be sent in response to a *.req* message. A generic description of the message sequence chart is presented in Figure 6-2.

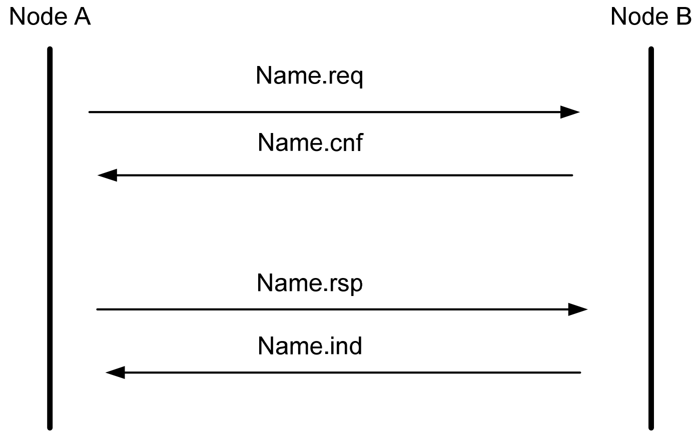


Figure 6-2 – Generic message sequence chart

The name has the following structure: [*Category*][*Source*][*Function*] with the following meaning: *Category*: the message application category, such as admission, beacon, channel estimation;

Source: the source of the message, such as end-node or domain-master;

Function: generic function of the message, such as parameter request, parameter report, specific command, and similar.

"Source" and "function" parts are optional, although at least one of them shall be present. The DLL uses management messages presented in Table 9-19.

7 Profiles

Profiles are intended to specify nodes with significantly different levels of complexity and functionality. A more complex profile is a superset of a less complex profile and shall interoperate with that profile. Less complex profiles target specific deployment scenarios or regional variations and are identified by a reduced list of supported features and parameter values compared to a standard profile, which is a superset of all less complex profiles.

A node shall be classified into a particular profile according to its degree of complexity and functionality. For compliance with this Recommendation, a node is required to support at least one profile. Profiles are summarized in Table 7-1.

Table 7-1 – Profiles

Profiles	Description	Note
Standard profile	Implements all requirements of the Recommendation	
Low complexity profile	Reduced functionality, as per Table 7-2	

7.1 Low complexity profile (LCP)

Table 7-2 describes the valid values of parameters for the LCP that differentiate from the standard profile.

Table 7-2 – Reduced features and valid values of parameters

Parameter/feature	Description	Notes
DLL-related		
Domain master functionality	Not required	
DAP functionality	Not required	
Data relaying	Not required	
Beacon relaying	Not required	
Selective acknowledgement	Not required	
Multi-segment MPDU transmission	Not required	
Supported MA priorities	0, 2	
PHY-related		
Modulation	2 bits per sub-carriers	
Code rate	½	
EVM	12 dB	Corresponds to QPSK
Maximum FEC block size	128 bytes	

8 Physical layer (PHY) specification

8.1 Functional model of the PHY

The functional model of the PHY is presented in Figure 8-1. The PMI and MDI are respectively, two demarcation reference points between the PHY and MAC and between the PHY and transmission medium. Internal reference points δ and α show separation between the PMD and PMA, and between the PCS and PMA respectively. The data primitives and management primitives at the PMI reference point and MDI reference point are defined in clauses 8.8.1 and 8.8.2, respectively. The electrical characteristics of the MDI are defined in clause 8.7.

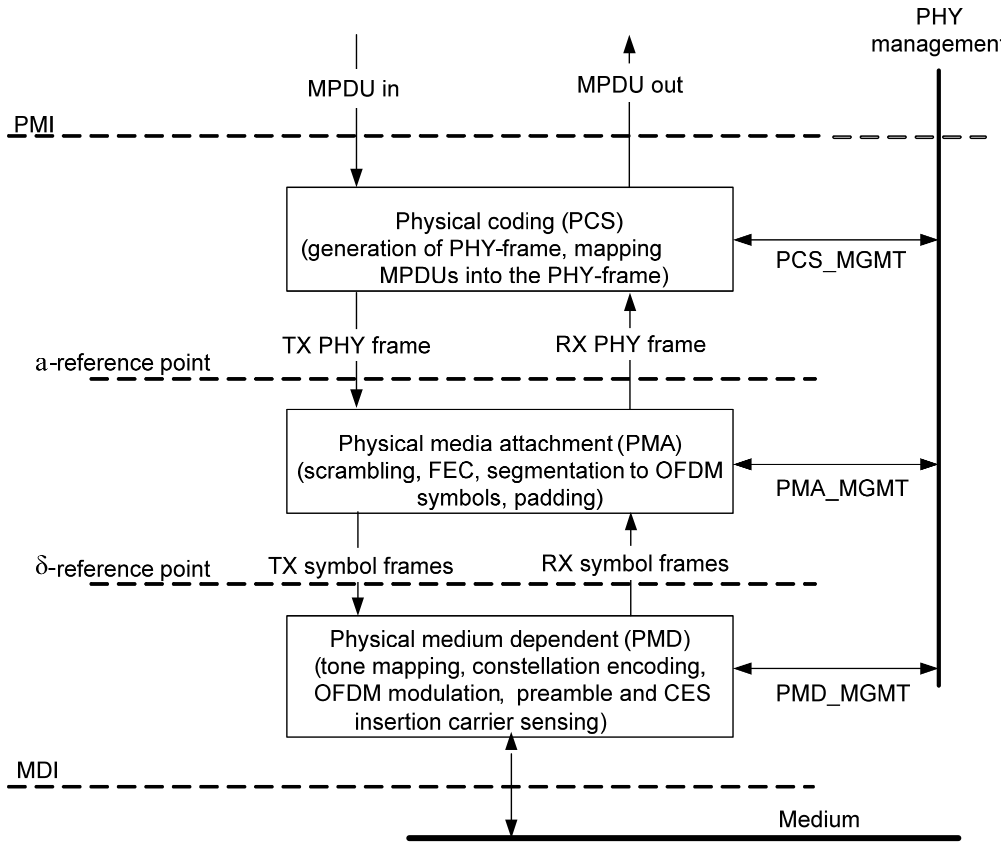


Figure 8-1 – Functional model of PHY

In the transmit direction, data enters the PHY from the MAC via the PMI by blocks of bytes called MAC protocol data units (MPDUs). The incoming MPDU is mapped into the PHY frame originated in the PCS, scrambled and encoded in the PMA, modulated in the PMD, and transmitted over the medium using OFDM modulation with relevant parameters. In the PMD, a preamble and the channel estimation symbols (CES) are added to assist synchronization and channel estimation in the receiver.

In the receive direction, a frame entering from the medium via the MDI is demodulated and decoded. The recovered MPDU is forwarded to the MAC via the PMI. The recovered PHY frame header (PFH) is processed in the PHY to extract the relevant frame parameters specified in clause 8.2.3.

8.2 Physical coding sublayer (PCS)

The functional model of the PCS is presented in Figure 8-2. It is intended to describe in more detail the PCS functional block presented in Figure 8-1.

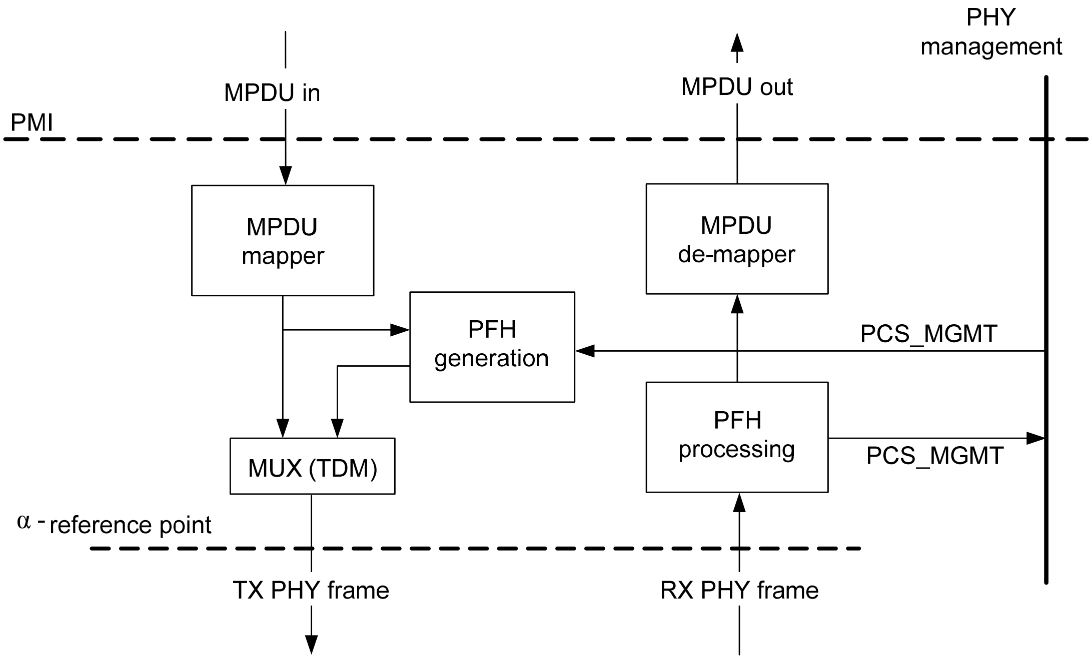


Figure 8-2 – Functional model of PCS

In the transmit direction, the incoming MPDU is mapped into a payload field of a PHY frame (clause 8.2.1) as described in clause 8.2.2. Furthermore, the PFH is generated and added to form a TX PHY frame. The TX PHY frame is passed via the α-reference point for further processing in the PMA.

In the receive direction, the decoded PHY frame payload and header are processed and the originally transmitted MPDU is recovered from the payload of the received PHY frame (RX PHY frame) and submitted to the PMI. Relevant control information conveyed in the PFH is processed and submitted to the PHY management entity, Figure 8-2.

The management primitives of the PCS (PCS_MGMT) are defined in clause 8.8.2.

8.2.1 PHY frame format

The format of the PHY frame is presented in Figure 8-3. The PHY frame includes preamble, PFH, channel estimation symbols (CES) and payload. Preamble and CES are added to the PHY frame in the PMD. The PFH and the payload are generated and formatted in the PCS. Preamble and CES do not carry any data and are intended for synchronization and initial channel estimation only. The structure of the preamble and its parameters are specified in clause 8.4.5, and for the CES, the parameters are defined in clause 8.4.6.

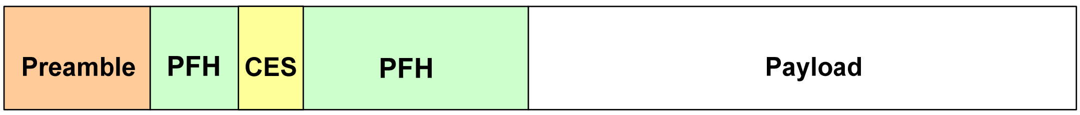


Figure 8-3 – Format of the PHY frame

All components of the PHY frame (preamble, PFH, CES, and the payload) consist of an integer number of OFDM symbols.

The number of symbols of the PFH depends on the applied bandplan, as described in Table 8-16. All symbols in the PFH for a particular bandplan are transmitted using a predefined set of coding and modulation parameters, as defined in clause 8.3.2.3, clause 8.4.2.5 and clause 8.4.7).

The length of the payload may vary from frame to frame; payload may also be of zero length. For payload, different coding and modulation parameters (including the number of repetitions, tone masking, and bit loading) can be used in different PHY frames depending on the channel and noise characteristics of the medium. The coding and modulation parameters of the payload are defined in the PFH, as described in clause 8.2.3.2.

PHY frames are divided into several types, depending on their purpose. The type of the PHY frame is indicated in the PFH. The types of PHY frames specified in this Recommendation are summarized in Table 8-1. The format of the PHY frame of each type is defined in clause 8.2.3.1.

Table 8-1 – PHY frame types

Frame type	Payload	Description
Type 1 frame	√	A PHY frame carrying a payload field with user data or management data
Type 2 frame	√	Reserved by ITU-T (Note)
Type 3 frame	None	A PHY frame containing no payload field
Type 4 frame	√	Reserved by ITU-T (Note)
NOTE – Upon receipt of a frame with a type defined as "reserved" (i.e., frame type 2 or 4) for the current revision of the Recommendation, a node shall: – discard the received PHY frame; – apply medium access rules based on the value of the duration field indicated on the PFH (as specified in clause 8.2.3.2.2).		

8.2.2 MPDU mapping

MPDUs are passed to the PHY as an ordered sequence of bytes which are processed as an ordered stream of bits from the LSB to MSB within each byte. The first bit of the MPDU shall be the first transmitted bit of the payload.

The only valid sizes of an MPDU are those that meet the representation presented in Table 8-5. Padding of the MPDUs to match the valid values indicated in Table 8-5 shall be done by the DLL, as defined in clause 9.1.3.1. Incoming MPDUs of invalid values shall be dropped.

8.2.3 PHY frame header (PFH)

The PFH is PHY_H bits long and comprise of a common part and a variable part. The common part contains fields that are common for all PHY frame types. The variable part contains fields according to the PHY frame type. The type of the PHY frame is indicated by the FT field. The content of the PFH is protected by a 12-bit header check sequence (HCS). The PFH format is defined in Table 8-2. The size of the variable field depends on the bandplan as specified in Table 8-2.

Table 8-2 – PFH format

Field	Number of bits	Description	Comment
FT	2	Frame type	Common part
FTSF	Variable	Frame-type specific field	For FCC and FCC-2 bandplans, the FTSP field is 60 bits For CENELEC and FCC-1 bandplans, the FTSP field is 28 bits
HCS	12	Header check sequence (12 bits)	Common part

The ordering of bits and bytes of the PFH is detailed in clause 8.2.3.3.

8.2.3.1 Common part fields

8.2.3.1.1 Frame type (FT)

The frame type (FT) field is a 2-bit field which indicates the type of PHY frame as described in Table 8-3.

Table 8-3 – Encoding of the FT field

Frame type	Value
Type 1 frame	00
Type 2 frame	01
Type 3 frame	10
Type 4 frame	11

8.2.3.1.2 Header check sequence (HCS)

The HCS field is intended for PFH verification. The HCS is a 12-bit cyclic redundancy check (CRC) and shall be computed over all the fields of the PFH in the order they are transmitted, starting with the LSB of the first field of the PFH (FT) and ending with the MSB of the last field of the FTSF.

The HCS shall be computed using the following generator polynomial of degree 12:

$$G(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$$

The value of the HCS shall be the remainder after the content of the HCS calculation fields (treated as a polynomial where the first input bit is associated with the highest degree, $x^{PHY_H\text{-}13}$, where PHY_H is the PFH length in bits, and the last input bit is associated with x^0) is multiplied by x^{12}, then XOR'd with a value of all-ones ($0FFF_{16}$), and then divided by *G(x)*.

The HCS field shall be transmitted starting with the coefficient of the highest order term (i.e., with x^{11}).

8.2.3.2 Variable part fields

The content of the variable part of the PFH depends on the frame type (FT field value) and shall be as shown in Figure 8-4 and further described in Table 8-4.

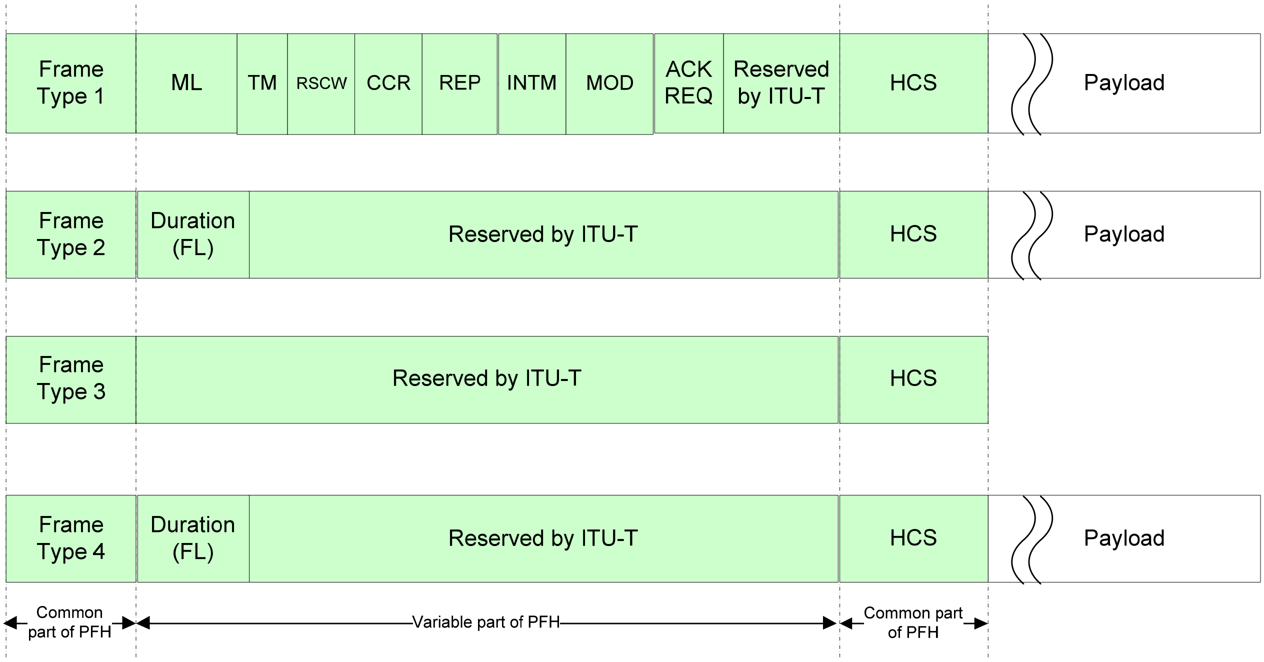


Figure 8-4 – Content of the PFH depending on the frame type field

Table 8-4 – Fields comprising the variable part of the PFH

Field	Number of bits		Description	Reference
	CENELEC, FCC-1	FCC, FCC-2		
MPDU Length (ML)	8	8	Indicates the length of the payload in bytes expressed using logarithmic scale	Clause 8.2.3.2.1
Duration (FL)	7	10	Indicates the duration of the PHY frame sequence expressed in OFDM symbols	Clause 8.2.3.2.2
Tone mask (TM)	8	40	Defines the tone mask used to transmit the payload	Clause 8.2.3.2.3
RS code word size (RSCW)	1	1	Indicates the maximum value of the RS code word size to be used for payload encoding	Clause 8.2.3.2.4
CC Rate (CCR)	1	1	Indicates the coding rate of the convolutional code used to transmit the payload	Clause 8.2.3.2.5
Repetitions (REP)	3	3	Indicates number of repetitions used to transmit the payload	Clause 8.2.3.2.6
Interleaving Mode (INTM)	1	1	Indicates the interleaving mode used to transmit the payload	Clause 8.2.3.2.7
Modulation (MOD)	2	2	Indicates the modulation used to transmit the payload	Clause 8.2.3.2.8

Table 8-4 – Fields comprising the variable part of the PFH

Field	Number of bits		Description	Reference
	CENELEC, FCC-1	FCC, FCC-2		
Acknowledgement request (ACK REQ)	2	2	Indicates whether the receiver should respond with an ACK to indicate MPDU reception status	Clause 8.2.3.2.9
Reserved by ITU-T	FT dependent	FT dependent	Reserved bits for future use by ITU-T	Clause 8.2.3.2.10

8.2.3.2.1 MPDU length (ML)

This 8-bit field indicates the number of bytes in the MPDU. The number of bytes is represented based on a mapping between the unsigned integer value in the ML field and the MPDU size in bytes as shown in Table 8-5.

Table 8-5 – Mapping of the ML field to MPDU size

From ML_{10} value	To ML_{10} value	Mapped MPDU [bytes]
0	63	ML10+1
64	127	$65+2\times(ML_{10} - 64)$
128	191	$193+8\times(ML_{10} - 128)$
192	255	$697+16\times(ML_{10} - 192)$
NOTE – ML10 is a decimal representation of ML field.		

8.2.3.2.2 Duration (FL)

This 7-bit/10-bit unsigned integer field indicates the duration of the PHY frame sequence, excluding the duration of the PFH and the preamble of the transmitted frame, represented in multiples of K_{Dur} OFDM symbols as specified in Table 8-6.

NOTE 1 – The duration of the preamble and the PFH is the same for all frames transmitted by nodes of the same domain (see clause 8.8).

NOTE 2 – The duration indicated in the FL field is counted from the beginning of the first symbol of the transmitted frame to the end of the last symbol of the last frame in the frame sequence (the last symbol of the ACK frame, if requested). More details are described in clause 9.3.3.1.

This field is used with frame types 2 and 4 only.

Table 8-6 – K_{Dur} value per bandplan

Band	K_{Dur}
CENELEC	4
FCC-1	8
FCC, FCC-2	1

8.2.3.2.3 Tone mask (TM)

This 8-bit/40-bit field is a bitmap that indicates whether a particular subcarrier group is active (i.e., is from the ASC set) or inactive (i.e., is from the ISC set), as defined in clause 8.4.2.1. The actual band is divided into groups of G tones according to the applied bandplan, as specified in clause 8.4.2.4, and each bit in the TM bitmap shall indicate whether the G consecutive tones are active (the respective bit in the TM field equals 1) or inactive (the respective bit in the TM field equals 0). The LSB of the TM field corresponds to the first group of subcarriers (with the lowest indices).

This TM field size and the value of G for different bandplans shall be as specified in Tables 8-4 and 8-7, respectively.

Table 8-7 – Value of G for different bandplans

Bandplan	G
CENELEC A	4 (See Note)
CENELEC B	2
CENELEC CD	2
FCC-1	4
FCC, FCC-2	4
NOTE – The tone mask settings of the last tone (#33) shall be the same as the value in bit b_7 (i.e., masked if set to 0, and unmasked if set to 1).	

To indicate the use of BAT type 0, BAT type 1, and BAT type 5, the TM field shall be set to all-zeros, and the MOD field value shall be set to 00 to indicate the use of BAT type 0, set to 01 to indicate the use of BAT type 1, set to 10 to indicate the use of BAT type 5 and set to 11 to indicate the use of BAT type7.

8.2.3.2.4 RS code word size (RSCW)

This 1-bit field indicates the value to use as the maximum RS code word size for dividing the MPDU into code words (as specified in clause 8.3.3).

If the maximum RS code word size of 239 bytes is used, the field shall be set to 0.

If the maximum RS code word size of 128 bytes is used, the field shall be set to 1.

8.2.3.2.5 CC rate (CCR)

This 1-bit field indicates whether the CC rate of 1/2 or 2/3 for convolutional encoding is used in the payload.

If the CC rate of 1/2 is used, the field value shall be set to 0.

If the CC rate of 2/3 is used, the field value shall be set to 1.

8.2.3.2.6 Repetitions (REP)

This 3-bit field indicates the number of repetitions used in the payload (value of the R for the payload encoding specified in clause 8.3.3).

The mapping of the field values to the values of the R parameter of the FRE is given in Table 8-8.

Table 8-8 – Encoding of the REP field

REP field value	R parameter of the FRE
000	1
001	2
010	4
011	6
100	12
101-111	Reserved by ITU-T

8.2.3.2.7 Interleaving mode (INTM)

This 1-bit field indicates whether the IoF or IoAC interleaving mode is used in the payload.

If the IoF mode is used, the field shall be set to 0.

If the IoAC mode is used, the field shall be set to 1.

8.2.3.2.8 Modulation (MOD)

This 2-bit field indicates the modulation used to transmit the payload, as specified in clause 8.4.3.

The mapping of the field values to the modulation used for payload transmission is given in Table 8-9.

Table 8-9 – Encoding of the MOD field

Mod field value	Modulation used
00	1-Bit
01	2-Bit
10	3-Bit
11	4-Bit

8.2.3.2.9 Acknowledgement request (ACK REQ)

This 2-bit field indicates to the receiver whether the transmitter requires it to respond with an ACK frame and indicates the type of the ACK frame as follows:

– 00 – no ACK frame is requested
– 10 – a regular Imm-ACK frame is requested
– 01 – an extended Imm-ACK frame is requested
– 11 – reserved by ITU-T.

The formats of the Imm-ACK frame and the extended Imm-ACK frame are defined in clause 9.3.3.1.1.

8.2.3.2.10 Reserved by ITU-T

The bits reserved by ITU-T are for further study. These bits shall be set to zero by the transmitter and ignored by the receiver.

The field size in bits depends on the frame type.

8.2.3.3 Ordering of the bits and bytes of the PFH

The ordering of the bits and bytes of the PFH (per frame type and bandplan) is shown in Tables 8-10 to 8-13.

Table 8-10 – Ordering of the bits and bytes of the PFH for frame type 1

Field	CENELEC, FCC-1	FCC, FCC-2	Description
	Bits	**Bits**	
FT	[1:0]	[1:0]	Clause 8.2.3.1.1
ML	[9:2]	[9:2]	Clause 8.2.3.2.1
TM	[17:10]	[49:10]	Clause 8.2.3.2.3
RSCW	[18]	[50]	Clause 8.2.3.2.4
CCR	[19]	[51]	Clause 8.2.3.2.5
REP	[22:20]	[54:52]	Clause 8.2.3.2.6
INTM	[23]	[55]	Clause 8.2.3.2.7
MOD	[25:24]	[57:56]	Clause 8.2.3.2.8
ACK REQ	[26]	[58]	Clause 8.2.3.2.9
Reserved by ITU-T	[29:27]	[61:59]	Clause 8.2.3.2.10
HCS	[41:30]	[73:62]	Clause 8.2.3.1.2

Table 8-11 – Ordering of the bits and bytes of the PFH for frame type 2

Field	CENELEC, FCC-1	FCC, FCC-2	Description
	Bits	**Bits**	
FT	[1:0]	[1:0]	Clause 8.2.3.1.1
FL	[8:2]	[11:2]	Clause 8.2.3.2.2
Reserved by ITU-T	[29:9]	[61:12]	Clause 8.2.3.2.10
HCS	[41:30]	[73:62]	Clause 8.2.3.1.2

Table 8-12 – Ordering of the bits and bytes of the PFH for frame type 3

Field	CENELEC, FCC-1	FCC, FCC-2	Description
	Bits	**Bits**	
FT	[1:0]	[1:0]	Clause 8.2.3.1.1
Reserved by ITU-T	[29:2]	[61:2]	Clause 8.2.3.2.10
HCS	[41:30]	[73:62]	Clause 8.2.3.1.2

Table 8-13 – Ordering of the bits and bytes of the PFH for frame type 4

Field	CENELEC, FCC-1	FCC, FCC-2	Description
	Bits	**Bits**	
FT	[1:0]	[1:0]	Clause 8.2.3.1.1
FL	[8:2]	[11:2]	Clause 8.2.3.2.2
Reserved by ITU-T	[29:9]	[61:12]	Clause 8.2.3.2.10
HCS	[41:30]	[73:62]	Clause 8.2.3.1.2

8.3 Physical medium attachment sublayer (PMA)

The functional model of the PMA is presented in Figure 8-5. It is intended to describe in more detail the PMA functional block presented in Figure 8-1.

In the transmit direction, the PFH and payload of the incoming PHY frame at the α-reference point has a format as defined in clause 8.2.1. Both the PFH bits and the payload bits of the incoming frame are scrambled as described in clause 8.3.1. The PFH bits of the incoming frame are further encoded as described in clause 8.3.4. The payload bits are encoded, as described in clause 8.3.3. The parameters of the payload encoder are controlled by the PHY management entity (PMA_MGMT primitives). The parameters of the PFH encoder are predefined for each particular bandplan to facilitate interoperability.

After encoding, the PFH and payload are each mapped into an integer number of symbol frames as described in clause 8.3.6. The obtained symbol frames of the PFH and the payload are submitted to the PMD (at the δ-reference point) for modulation and transmission over the medium.

In the receive direction, all necessary inverse operations of decoding, and de-scrambling are performed on the received symbol frames. The recovered PFH and payload are submitted to the α-reference point for further processing in the PCS.

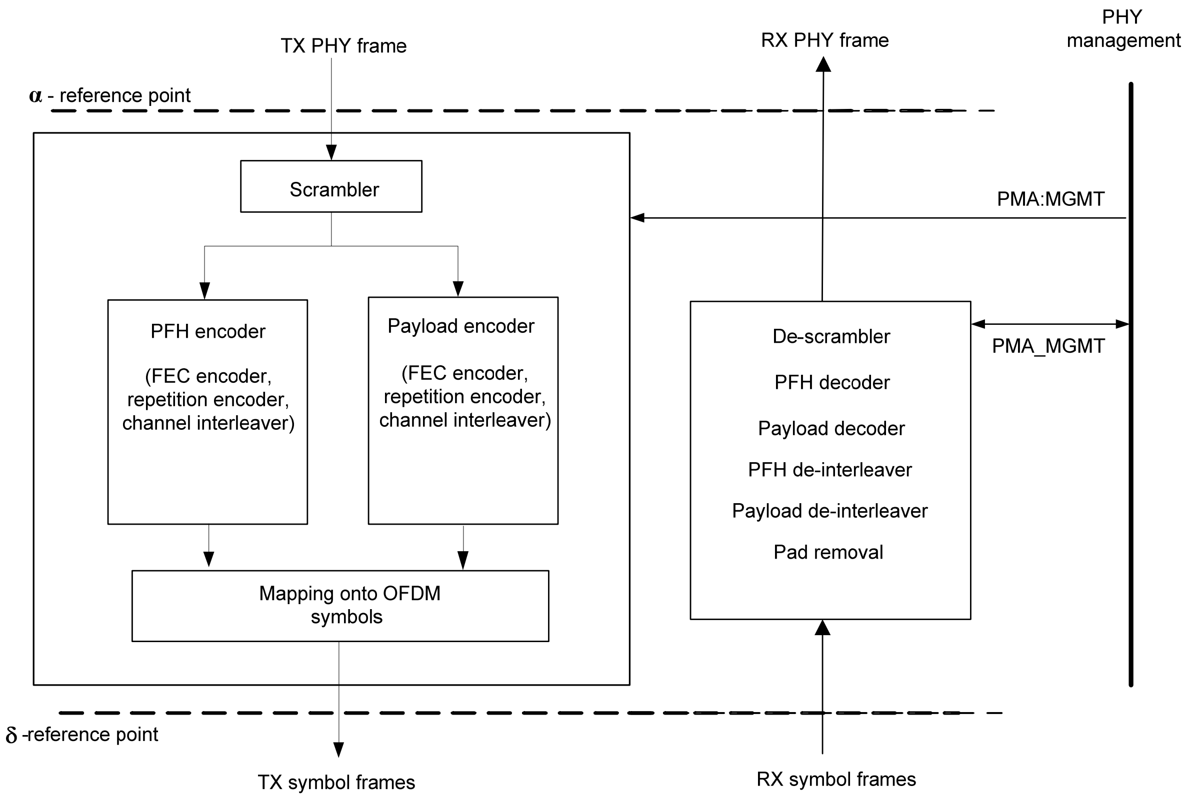


Figure 8-5 – Functional model of the PMA

The management primitives of the PMA (PMA_MGMT) are defined in clause 8.8.2.3.

8.3.1 Scrambler

All data bits, starting from the first bit of the PFH and ending by the last bit of the payload, shall be scrambled with a pseudo-random sequence generated by the linear feedback shift register (LFSR) with the polynomial $p(x) = x^7 + x^4 + 1$, as shown in Figure 8-6.

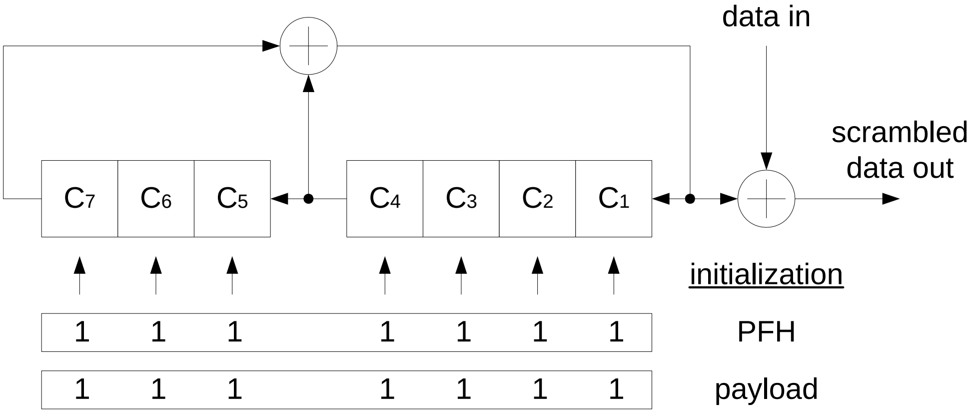


Figure 8-6 – Scrambler

The LFSR shall be initialized at the first bit of the PFH with the initialization vector equal to 0x7F (where the LSB corresponds to C1); this initialization is used for scrambling the PFH data. A second initialization shall be performed for payload data, immediately after the last bit of the PFH is read out from the scrambler and before the first bit of the payload is read out from the scrambler. For the second initialization, the initialization vector shall be set to 0x7F.

8.3.2 FEC encoder

The FEC encoder is shown in Figure 8-7. It consists of an inner convolutional encoder and the outer Reed Solomon (RS) encoder. The parameters of the FEC encoder are:

- the number of incoming RS information blocks, $m \geq 1$
- the number of bytes, K, in the incoming RS information blocks
- the number of RS parity-check bytes, R
- the number of bits incoming the inner encoder, k_I
- the inner code rate, r_I.
- the number of output bits, N_{FEC}, (the FEC codeword size depends on the overall code rate).

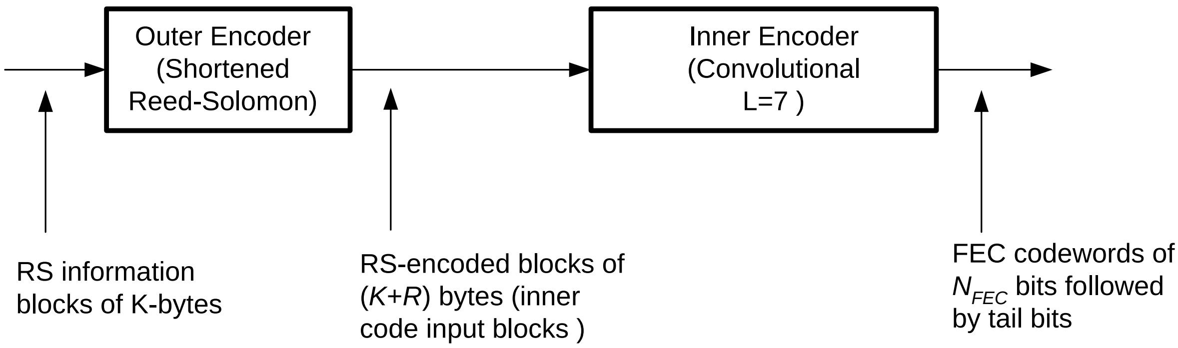


Figure 8-7 – FEC encoder

The incoming MPDU shall be first divided into RS information blocks. The number of RS information blocks, m, depends on the size of the MPDU and is determined by the PMI_DATA_REQ primitive (see clause 8.8.1.1). The size of each information block, K_l, where $l = 1, 2, ...m$, shall be an integer number of bytes and shall be computed for the given value of m as follows:

– The size of the first RS information block shall be 16 bytes (the size of the MPH, see clause 9.1.3.1.1).

– The m1 following RS information blocks shall be of the size $K_L = floor[(N_{MPDU}\text{-}16)/(m\text{-}1)] +1$ bytes, where $m_1 = mod[(N_{MPDU}\text{-}16)/(K_L\text{-}1)]$ and N_{MPDU} is the size of MPDU in bytes.

– The remaining $m\text{-}m_1\text{-}1$ information blocks shall be of the size $K_S = K_L\text{-}1$ bytes.

The valid values of other parameters of the FEC for the payload and the PFH are specified in Table 8-14 and Table 8-15, respectively. The m output FEC codewords followed by tail bits generated by the inner encoder shall be concatenated into an FEC codeword block. The order of the FEC codewords in the FEC codeword block (at the output of the FEC encoder) shall be the same as the order of the corresponding RS information blocks at the input of the FEC encoder.

The PFH shall be encoded as one single codeword. Encoding of the extended Imm-ACK frame is for further study.

8.3.2.1 Reed-Solomon encoder

The outer code shall use a standard byte-oriented Reed-Solomon code. The encoded RS block shall contain $N = K+R$ bytes, comprised of R check bytes $c_0, c_1, ..., c_{R-2}, c_{R-1}$ appended to the K bytes $m_0, m_1, ...,m_{K-2}, m_{K-1}$ of the input information block. The check bytes shall be computed from the information bytes using the equation:

$$C(D) = M(D)D^R \bmod G(D)$$

where:

$M(D) = m_0 D^{K-1} \oplus m_1 D^{K-2} \oplus ... \oplus m_{K-2} D \oplus m_{K-1}$ is the polynomial representing the input block,

$C(D) = c_0 D^{R-1} \oplus c_1 D^{R-2} \oplus ... \oplus c_{R-2} D \oplus c_{R-1}$ is the check polynomial, and

$G(D) = \prod_{i=1}^{R} \left(D \oplus \alpha^i\right)$ is the generator polynomial of the RS code.

The polynomial C(D) is the remainder obtained from dividing $M(D)D^R$ by $G(D)$. The arithmetic shall be performed in the Galois field GF(256), where α is a primitive element that satisfies the primitive binary polynomial $x^8 \oplus x^4 \oplus x^3 \oplus x^2 \oplus 1$. Bits $(d_7, d_6, ..., d_1, d_0)$ of data byte D are identified by the Galois field element $d_7\alpha^7 \oplus d_6\alpha^6 \oplus ... \oplus d_1\alpha \oplus d_0$.

With the above definitions, an input block size of (255-R) bytes can be corrected up to $t = R/2$ erroneous bytes. A t-error correcting code for all smaller input block sizes shall be obtained by using the following procedure:

- the input block is substituted by appending zeros to the size 255-2t
- the 2t parity bytes are computed as defined above
- the output block is formed by appending the 2t parity bytes to the input block.

The maximum value of t shall not exceed 8 and the maximum input block size shall not exceed 239 bytes. The output block size shall be configurable to have any integer value in the range from 25 bytes to 255 bytes, inclusive. For input blocks shorter than 25 bytes, the RS encoder shall be bypassed. The valid values of an error-correcting capability, $t = R/2$, for different input block size are defined in Table 8-7.

8.3.2.2 Convolutional encoder

Each RS information block encoded by the outer encoder shall be converted to a bit stream (LSB first) to form the inner input block of $k_I = 8 \times (K+R)$ bits. Inner input blocks shall be concatenated in the same order as the corresponding RS information blocks at the input of the FEC encoder. The last inner block shall be appended by six zeros (tail bits). The concatenated inner blocks shall be input to the inner convolutional encoder shown in Figure 8-8. The inner convolutional encoder shall have a mother code rate of 1/2 and constraint length $L = 7$, and code generator polynomials $G1=1111001_2 = 171_8$ and $G2=1011011_2 = 133_8$. The convolutional encoder state shall be set to zero before the first bit of the first inner block enters the encoder. The six zeros appended to the last incoming inner block are to flush the encoder.

For the mother code of rate, $r_I = 1/2$, all X and Y bits generated by the encoder (see Figure 8-8) shall be output in the order: $X_0Y_0X_1Y_1..X_kY_k$ …

For a code rate of $r_I = 2/3$, the puncturing of the output bits of the convolutional encoder shall be applied according to the pattern [1 1; 0 1], i.e., every alternate X output shall be punctured to yield the output bit stream in the order: $X_0Y_0Y_1\,X_2Y_2Y_3..X_{2k}Y_{2k}Y_{2k+1}$ …

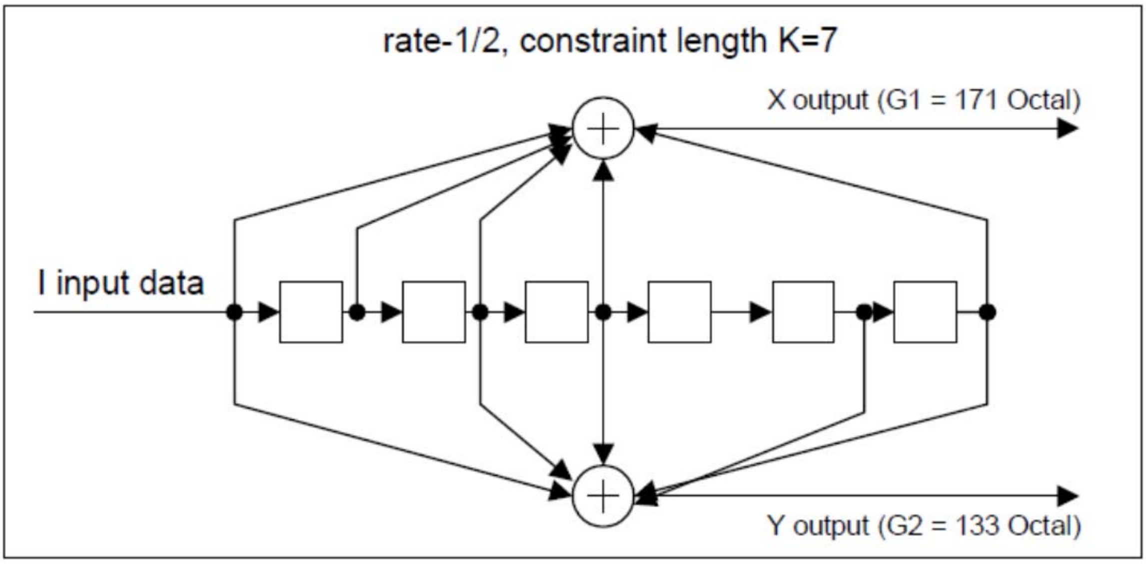


Figure 8-8 – Inner convolutional code encoder

The output bits of the inner encoder corresponding to the same inner input block form the output FEC codeword. The length of the FEC codeword can be computed as: $N_{FEC} = k_I / r_I$ bits.

8.3.2.3 FEC encoding parameters

The summary of valid FEC encoding parameters is specified in Table 8-14.

Table 8-14 – Valid values of payload FEC encoding parameters

RS information block size K, bytes	Valid inner code rate, r_I	RS parity check $R = 2t$, bytes
≤25	1/2, 2/3	0
26-50	1/2, 2/3	4
51-75	1/2, 2/3	8
76-100	1/2, 2/3	12
101-239	1/2, 2/3	16

The output FEC codeword size, NFEC, for the given values of K, r_I, and R presented in Table 8-14 can be computed as: $N_{FEC} = (8\times(K+R)) / r_I$ bits.

For the PFH, the outer encoder shall be bypassed. The inner encoder block size shall be $k_I = PHY_H$ bits (see clause 8.2.3) and the code rate shall be 1/2, as presented in Table 8-15. The output FEC codeword size is $(k_I + 6)/r_I$ bits.

Table 8-15 – Valid values of PFH FEC encoding parameters

Bandplans	Inner encoder input block, k_I, bits	Inner code rate, r_I
CENELEC, FCC-1	42	1/2
FCC, FCC-2	74	1/2

The total number of bits in an FEC codeword block corresponding to m input information blocks can be computed as:

$$N_{FECB} = 6/r_I + \sum_{l=1}^{m} N_{FEC,l} = \left[6 + 8 \times \sum_{l=1}^{m} (K_l + R) \right] / r_I$$

NOTE – The overall coding rate of the FEC encoder can be computed as:

$$r = \left[8 \times \sum_{l=1}^{m} K_l \right] / N_{FECB}$$

8.3.3 Payload encoder

The functional diagram of the payload encoder is presented in Figure 8-9. It contains an FEC encoder, an aggregation and fragmentation block (AF), a fragment repetition encoder (FRE), and an interleaver. The FRE is to support a robust communication mode (RCM) and is bypassed in the case of a normal mode of operation (no repetitions).

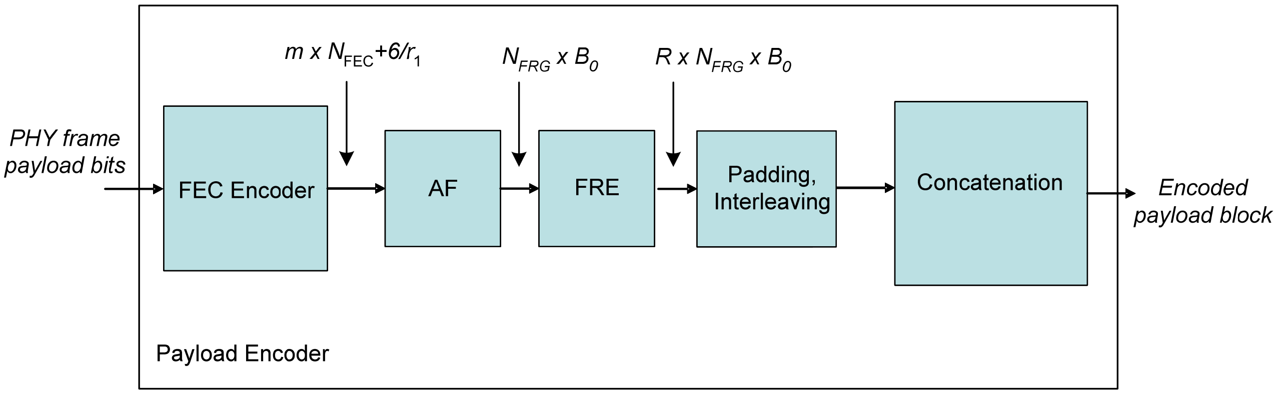


Figure 8-9 – Functional diagram of the payload encoder

The incoming PHY frame payload bits shall be divided into m sequential information blocks of K_l bytes per block, $l = 1, 2, ...m$, and each information block shall be encoded by the FEC encoder, as described in clause 8.3.2. The valid values of FEC parameters K, R, and r_I, and the coded block size N_{FEC} are presented in clause 8.3.2.3. The bytes in each information block shall be in the same order as they are in the corresponding MPDU.

The AF first collects the FEC codeword block of N_{FECB} bits generated by the FEC for the encoded payload. Furthermore, the FEC codeword block is partitioned into fragments of the same size B_0 bits each (e.g., $B_1 - B_4$ in Figure 8-10). The number of fragments is $N_{frg} = ceiling(N_{FECB}/B_0)$. To obtain an integer number of fragments, the FEC codeword block shall be padded with up to $B_P = B_0 \times N_{frg} - N_{FECB}$ bits.

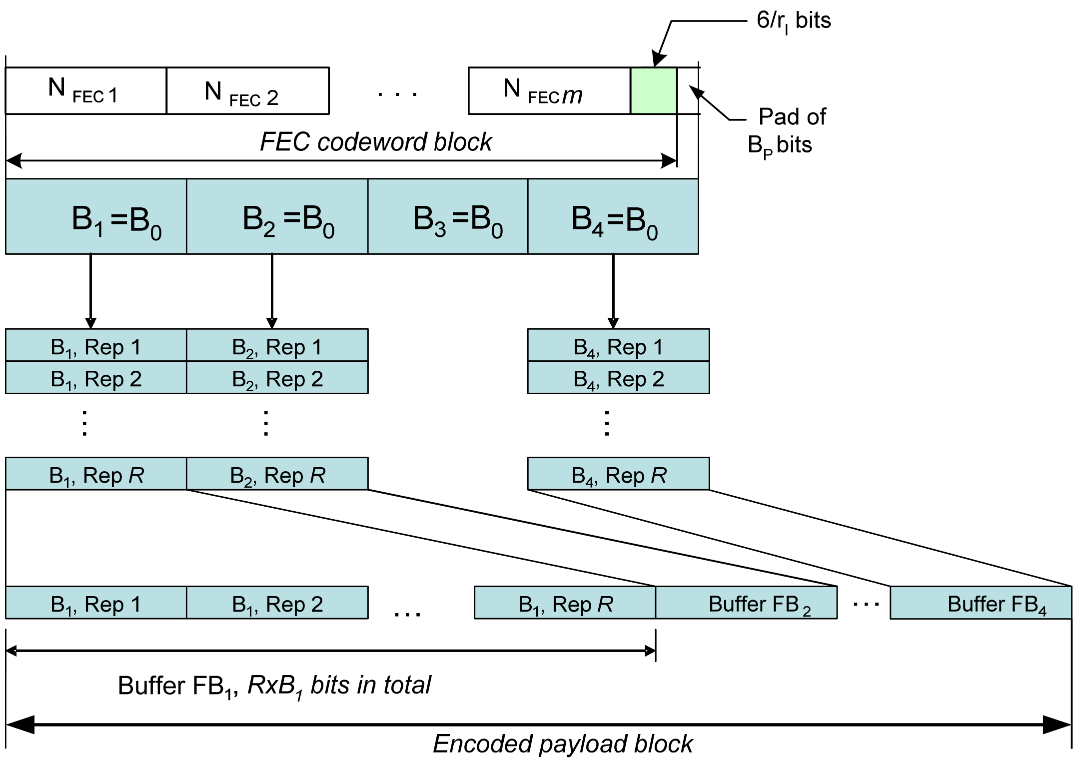


Figure 8-10 – Generation of the encoded payload block (case N_{frg} = 4, cyclic shifting, interleaving and padding of fragments when in IoAC mode is not shown)

The value of B_0 shall be calculated as an integer divisor of the total number of bits in the FEC codeword block and then increased to fit an integer number of symbols. This shall be the maximum divisor for which the value is less than or equal to the minimum of:

- the total number of input bits, N_{FECB}, in the FEC codeword block;
- the total number of bits, N_{ZC}, loaded on the symbols that span at least 10 ms for the case of 50 Hz AC lines, and at least 8.33 ms for the case of 60 Hz AC lines or lines with no AC;
- the maximum fragment size of B_{max} = 3072 bits.

The number of bits used to fit B_0 to an integer number of symbols shall not exceed the number of bits loaded onto a symbol (k_p) minus 1.

With the definitions above, the fragment size, B_0, and the number of pad bits, B_P, can be computed using the following steps:

- find the upper limit of the fragment size: $P = min(N_{FECB}, N_{ZC}, B_{max})$
- find the number of fragments: $N_{frg} = ceiling(N_{FECB}/P)$
- find the fragment size: $B_0' = ceiling(N_{FECB}/N_{frg})$; $B_0 = k_p \times ceiling\ (B_0'/\ k_p)$
- find the number of pad bits $B_P = B_0 \times N_{frg} - N_{FECB}$,

where k_p is the number of bits loaded onto a symbol. The pad bits, B_P, shall be generated by continuously extracting the MSB from the LFSR as shown in Figure 8-17 until the pad has filled up. The generation polynomial shall be as defined in clause 8.4.2.6. The LFSR initialization shall be all-ones as shown in Figure 8-17 prior to the first pad bit being extracted. The number of pad bits shall be less than $N_{frg} \times k_p$.

The FRE provides repetitions of fragments with the repetition rate of R. Each fragment shall be copied R times and all copies shall be concatenated into the fragment buffer, FB, so that the first bit of each copy follows the last bit of previous copy, see Figure 8-10. The total size of the FB is $B_0 \times R$ bits. The FRE shall support the values R = 1, 2, 4, 6, 12 (value of R = 1 corresponds to the normal mode of operation). If R = 1, an FB, accordingly, shall contain a single fragment of B_0 bits.

All fragments and their copies of each FB shall be interleaved. The interleaving method and parameters of the interleavers are defined in clause 8.3.5 and are the same for all valid values of R. Two modes of interleaving are defined:

- interleave-over-fragment (IoF)
- interleave-over-AC-cycle (IoAC).

The mode of interleaving is indicated in the PFH, as defined in clause 8.2.3.2.7 and shall be selected at the discretion of the transmitter. In both modes, for each fragment, prior to interleaving, the bits of each fragment copy starting from the second copy ("Rep 2" in Figure 8-10) shall be cyclically shifted by M = $ceiling\ (B_0/R_T)$ bits relative to the previous copy in the direction from LSB to MSB, i.e., the copy "Rep(d+1)" shall be shifted by d × M bits relative to copy "Rep 1" so that the LSB of copy "Rep 1" will have bit number (d × M) in the copy "Rep(d+1)". The value of $R_T \geq R$ is the total number of repetitions, including padding; it depends on the mode of interleaving.

If the IoF mode is set, each fragment of the FB shall be interleaved separately. After the interleaving of all copies of the fragment, the FB shall be passed for concatenation. The value of R_T shall be set equal to R.

If the IoAC mode is set, each FB (containing R copies of the fragment) shall be padded to the closest integer number of symbols that is equal or more than the closest integer number of N_{ZC}, Figure 8-11. The pad shall be generated by cyclical repeating of the bits of this same FB, starting from its first bit: the first bit of the pad shall follow the last bit of the FB and shall be the repetition of the first bit of the same FB.

Furthermore, all copies of the fragment, both original and padded, shall be interleaved as defined in clause 8.3.5 for the payload interleaver. The total number of interleaved copies is RT = $ceiling(ceiling((B0 \times R)/N_{ZC}) \times N_{ZC}/B_0)$. From the last copy, only the symbols that fill up the padded FB, as shown in Figure 8-11, shall be taken from the interleaver. After interleaving of all copies of the fragment, the padded FB shall be passed for concatenation.

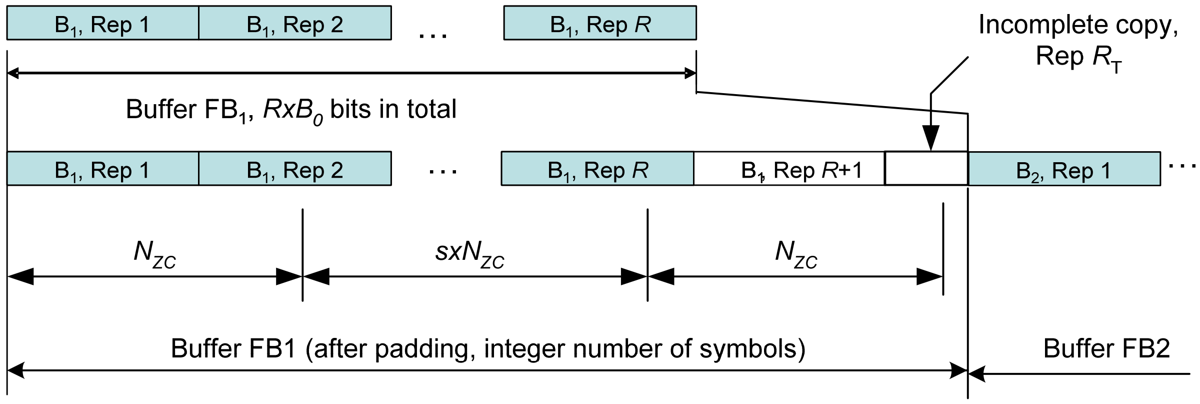


Figure 8-11 – Padding of the FB in IoAC mode

The FBs processed as described above shall be concatenated into an encoded payload block, in the order of the sourcing fragments, as shown in Figure 8-10.

The encoded payload block is passed for mapping into symbol frames (see clause 8.3.6).

8.3.4 PFH encoder

The functional diagram of the PFH encoder is presented in Figure 8-12, where all functional blocks operate as described in clause 8.3.3.

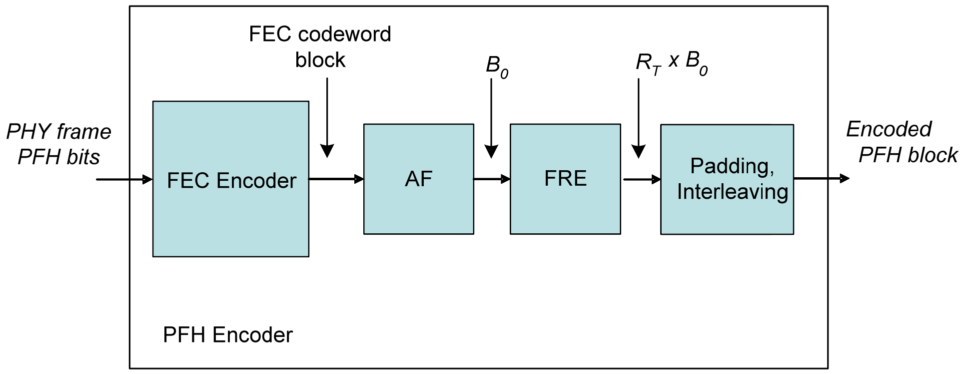


Figure 8-12 – Functional diagram of the PFH encoder

The bits of the PFH shall input the PFH FEC encoder in their original order and encoded as described in clause 8.3.2. The parameters of the PFH FEC encoder shall be as specified in clause 8.3.2.3, Table 8-15. The FEC codeword block at the output of the FEC encoder contains one FEC codeword and is $2 \times (PHY_H + 6)$ bits long, where PHYH is defined in clause 8.2.3.

Generation of an encoded PFH block is presented in Figure 8-13. The value of B_0 shall be equal to the FEC codeword block. The number of repetitions, RT, depends on the bandplan used and is determined by the number of symbols to carry the PFH, NS_H, and shall be computed as: $R_T = ceiling((NS_H \times k_H)/B_0)$, where k_H is the number of bits loaded onto a symbol. Two values of NS_H are defined for each bandplan: normal and robust, as presented in Table 8-16. The particular value of NS_H is determined by the PMI_MGMT.REQ primitive (see clause 8.8.2.1).

Table 8-16 – Number of symbols in the encoded PFH for 50Hz and 60Hz mains

Bandplan	Number of symbols, NSH		
	Normal	Robust	
	50 Hz, 60 Hz	50 Hz	60 Hz
CENELEC A	15	30	25
CENELEC B	30	45	50
CENELEC CD	45	45	50
FCC	8	30	25
FCC-1	19	30	25
FCC-2	10	30	25

The block of bits B_0 shall be copied R_T times and copies shall be concatenated in numerical order and divided into fragments of NS_I symbols, starting from the first symbol of the first copy, as presented in Figure 8-13. The size of the fragment shall be set as:

$$NS_I = min(\text{floor}(B_{max}/k_H), \text{ceiling}(N_{ZC}/k_H), NS_{H\_Normal})$$

where values B_{max} and N_{ZC} shall be as defined in clause 8.3.3, and N_{SH_Normal} is the normal value of NSH and shall be as defined in Table 8-16. The total number of fragments will be $R_F = \text{ceiling}(NS_H/NS_I)$. If the number of bits in the RT copies is insufficient to complete an integer number of fragments, the last fragment shall be completed by adding more copies of the block B_0.

Each fragment, starting from the second one ("Fragment 2" in Figure 8-13) shall be cyclically shifted by $M = ceiling((NS_I \times k_H)/R_F)$ bits relative to the previous copy, as described in clause 8.3.3. After cyclic shifting, all fragments shall be interleaved as defined in clause 8.3.5, for the PFH interleaver. If the last fragment is incomplete, only bits for the first symbols that are required to fit the size NS_H of the encoded PFH block shall be read out from the interleaver, as shown in Figure 8-13.

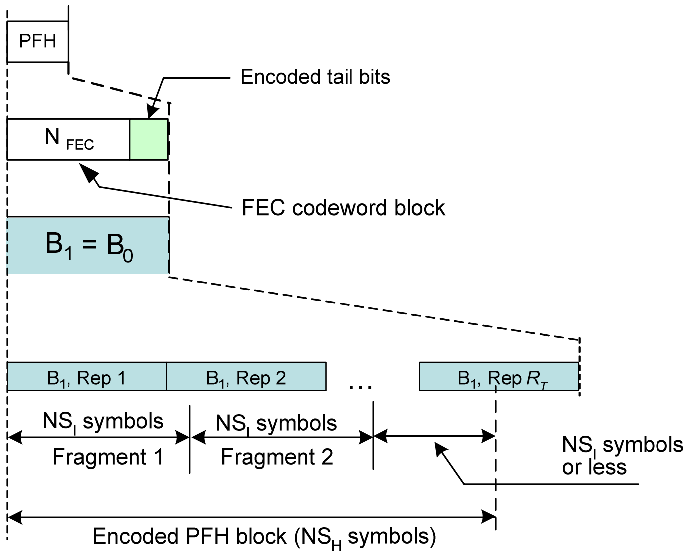


Figure 8-13 – Generation of the encoded PFH block

8.3.5 Channel interleaver

The channel interleaver interleaves a block of B_I bits (see clause 8.3.3, clause 8.3.4), based on the number of subcarriers per symbol frame that are loaded bits, denoted in this section by *m*. For the payload, these subcarriers are those identified in the TM field of the PFH, except the subcarriers from the PMSC, RMSC (unless BAT type 0 is used), and PSC sets. For the PFH, these are all subcarriers from the RMSC set and all subcarriers from the SSC set, except those from the PSC set (see clause 8.4.2.1, clause 8.4.2.2, clause 8.4.2.5).

For the payload encoder $B_I = B_0$, for the PFH encoder $B_I = NS_I \times k_H$. The interleaver is only defined for values of B_I that are multiples of *m*, i.e., $n = B_I / m$ is an integer. The B0I input bits shall be written into the permutation matrix with *n* rows and *m* columns. The insertion of the bits into the matrix shall be performed using the equations below:

$$q = floor(p/(k \times m))$$

$$r = mod(p, k \times m)$$

$$i = floor(r, k)$$

$$j = k \times q + mod(r, k)$$

where:

p is the sequential number of the bit in the input sequence (input vector), in the range from 0 to BI-1

k is the modulation used (k=1 for 1-bit modulation, k=2 for 2-bit modulation, etc.)

i is the index of the column and j is the index of the row in the permutation matrix in the range from 0 to m-1 and from 0 to n-1, respectively (m columns by n rows).

Figure 8-14 shows the insertion of the bits into a matrix when the equations are used with *k*=2. Each box in the Figure 8-14 represents a bit. The number in the box indicates the position of the bit in the input bit sequence (input vector) and in the output bit sequence (output vector), respectively.

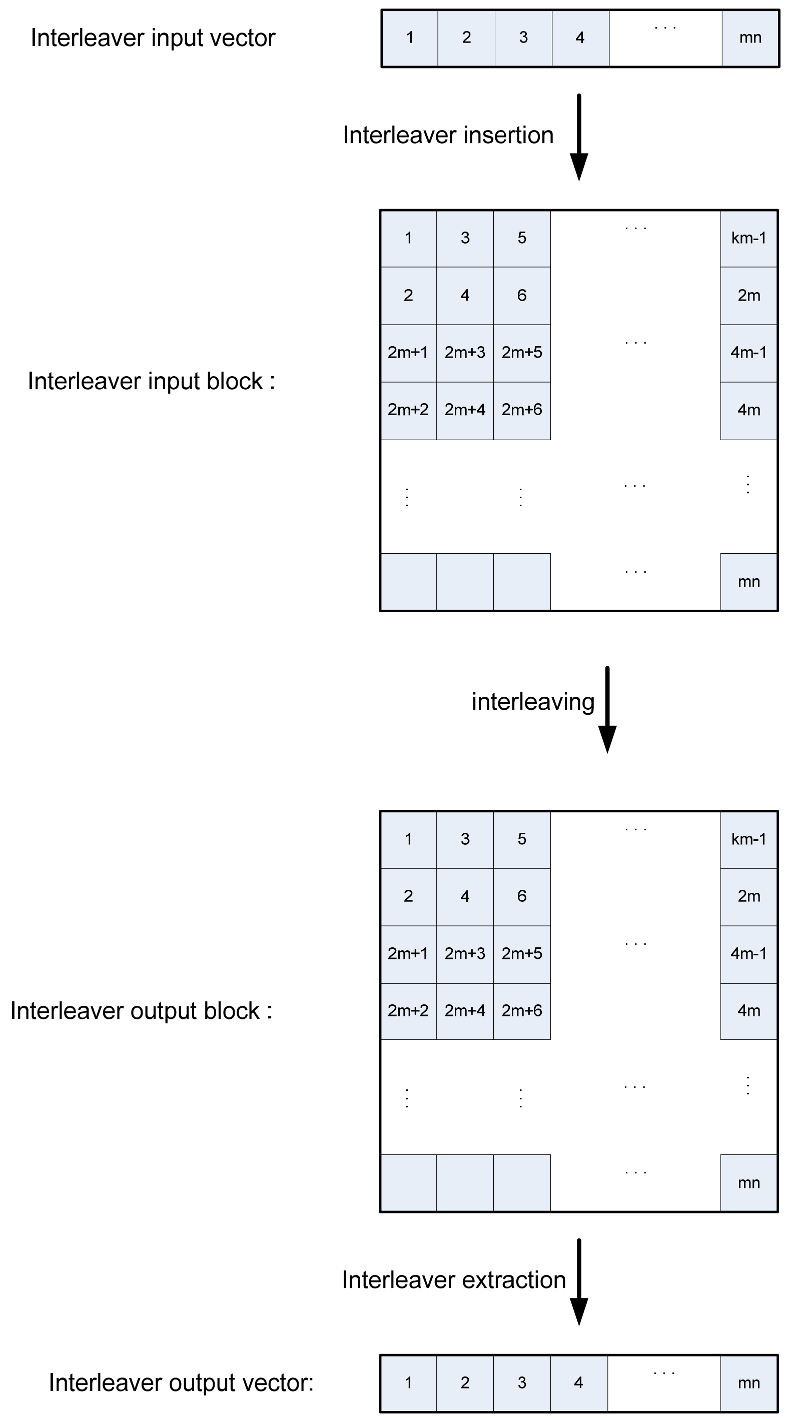


Figure 8-14 – Order of writing in and reading out of the permutation matrix

The entries of the $n \times m$ matrix shall be permuted. The relation between input and output bit indices shall be determined from the following equations: for the bit with the original position (i, j), where $i = 0,1,..., m-1$ and $j = 0,1,..., n-1$, the interleaved bit position (I, J) shall be:

$$J = (j \times n\_j + i \times n\_i) \bmod n$$

$$I = (i \times m\_i + J \times m\_j) \bmod m$$

where m_i, m_j, n_i, and n_j are selected based on the values of m and n, under the constraint that

$$m\_i, m\_j, n\_i, n\_j > 2$$

$$\mathrm{GCD}(m\_i,m) = \mathrm{GCD}(m\_j,m) = \mathrm{GCD}(n\_i,n) = \mathrm{GCD}(n\_j,n) = 1$$

where GCD stands for the greatest common divisor.

The values of n_i, n_j and m_i, m_j shall be computed as follows: For a given value of n, all the co-prime numbers of n except numbers 1 and 2 shall be sorted in ascending order; then, n_i shall be the first co-prime element above $n/2$ in that co-prime number set, and n_j shall be the next element to n_i. Same steps shall be applied to compute m_i and m_j, for a given value of m.

The following is an example for co-prime selection for $n = 8$, $m = 10$:

- Since $n = 8$, the co-prime numbers for 8 except 1 and 2 are: 3, 5, 7.

  The first co-prime number above $n/2$ is 5, so $n\_i = 5$; and the next co-prime is 7, so $n\_j = 7$.
- Since m = 10, the co-prime numbers for 10 except 1 and 2 are: 3, 7, 9.

  The first co-prime number above $m/2$ is 7, so $m\_i = 7$; and the next is 9, so $m\_j = 9$.

After permutation, bits shall be extracted from the permutation matrix in the same order that they were written into the matrix. An example for 2-bit modulation (k=2) is given in Figure 8-14.

8.3.6 Mapping onto symbol frames

The encoded payload block from the output of the payload encoder and the encoded PFH block from the output of the PFH encoder shall be mapped onto symbol frames. The number of bits in the symbol frame shall be equal to k_P for payload symbol frames and to k_H for PFH symbol frames. Payload and PFH symbol frames shall be passed to the PMD, as described in Figure 8-5.

8.3.6.1 Payload mapping

The encoded payload block shall be mapped onto one or more symbol frames. The number of symbol frames, M, shall be equal to the minimum number needed to accommodate all bits of the encoded payload block as defined in clause 8.3.3.

NOTE – The number of bits in the encoded payload block is always a multiple of k_P.

The first symbol frame shall contain the first k_P bits of the encoded payload block, the second frame shall contain the second k_P bits of the encoded payload block and so on, until the last symbol frame needed to accommodate the encoded payload block.

The payload mapping procedure is presented in Figure 8-15, showing also the convention for the start of the symbol frame further used for reference purposes (the start of the first frame is the LSB of octet 0 of the payload, the start of the second frame is bit with the number (k_P +1) of the payload block and so on).

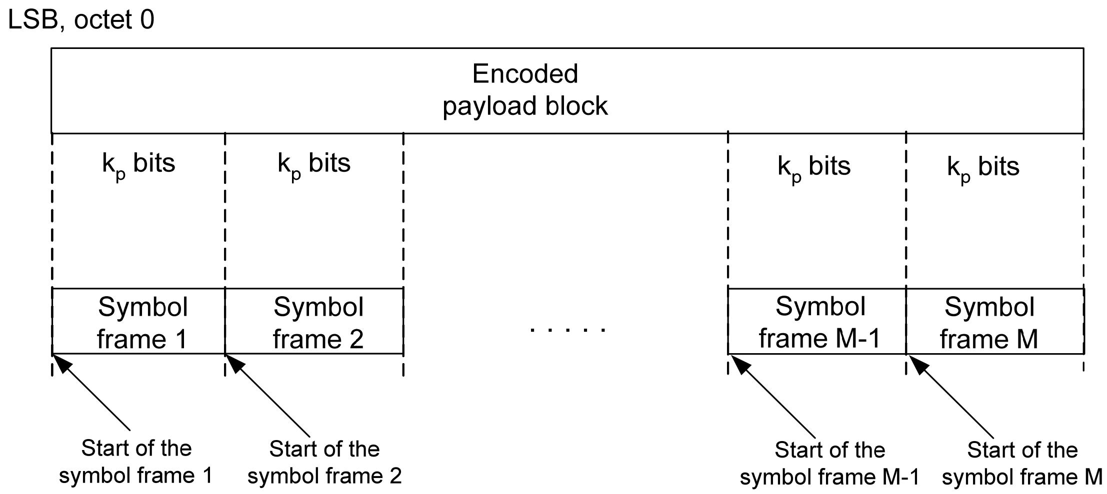


Figure 8-15 – Payload mapping

8.3.6.2 PFH mapping

The encoded PFH block shall be segmented into one or more symbol frames using the same convention as the payload block (the number of bits in the encoded PFH block is integer number of k_H, see clause 8.3.4).

8.4 Physical medium dependent sublayer (PMD)

The functional model of the PMD is presented in Figure 8-16. In the transmit direction, the tone mapper divides the incoming symbol frames of the PFH and the payload into groups of bits and associates each group of bits with a specific subcarrier onto which this group shall be loaded, as specified in clause 8.4.2. The constellation encoder converts each group of incoming bits into a complex number that represents the constellation point for this subcarrier. The constellation mapping process is described in clause 8.4.3. The unused subcarriers and pilot subcarriers are modulated by pseudo-random bit sequences generated as described in clauses 8.4.2.6 and 8.4.2.7, respectively.

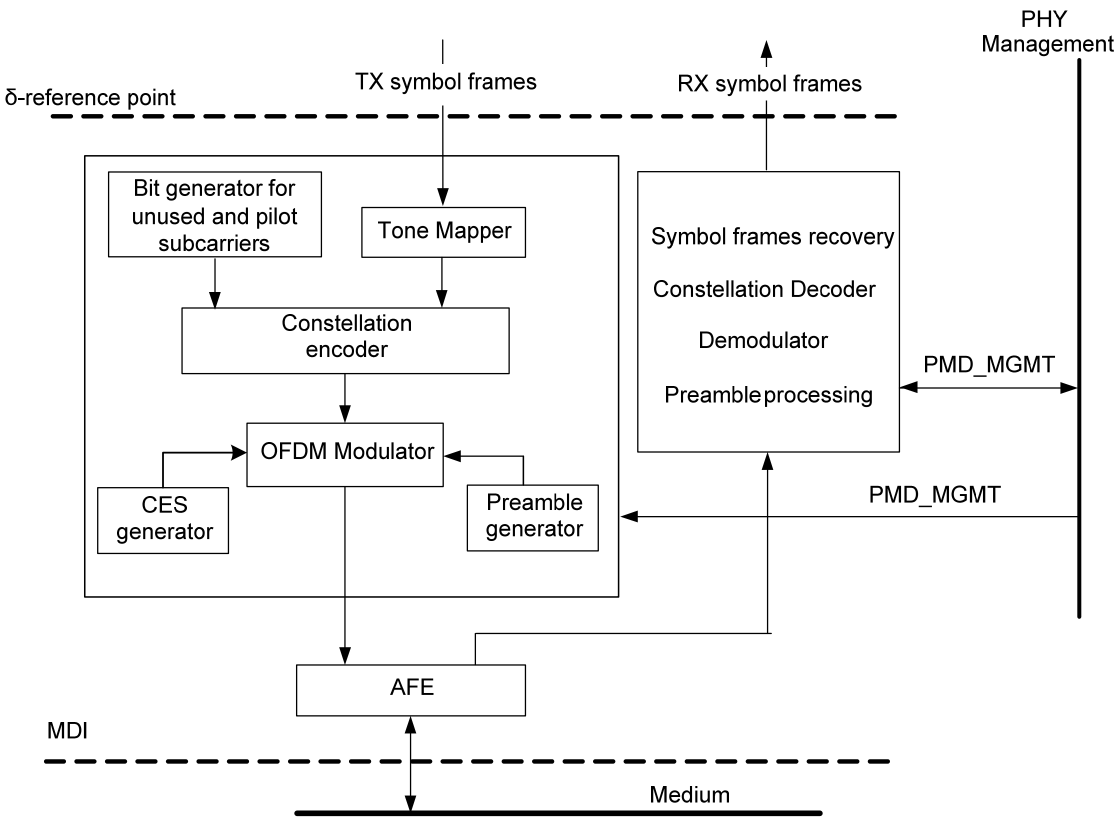


Figure 8-16 – Functional model of PMD

The OFDM modulator (see clause 8.4.4) converts the incoming stream of the N complex numbers into a stream of N complex time-domain samples. After adding the preamble and CES, the transmit signal is sent to the medium via the analogue front end (AFE). Parameters of the preamble defined in clause 8.4.5 are determined by the PHY management primitive PMD.MGMT.REQs.

In the receive direction, frames incoming from the medium are demodulated and decoded. The recovered symbol frames are transferred to the PMA via δ-reference point. The preamble and CES are processed and the processing results are passed to the PHY management entity.

The management primitives of the PMD (PMD_MGMT) are defined in clause 8.8.2.4.

8.4.1 Subcarrier spacing and indexing

The subcarrier spacing F_{SC} is the frequency spacing between any two adjacent subcarriers. Valid values of subcarrier spacing are presented in Table 8-6.

The subcarrier index i corresponds to the order of subcarriers in frequency: the subcarrier with index i shall be centred at frequency $f = F_{US} – (N/2 – i) \times F_{SC}$. The range of index i is from 0 to N-1. The subcarrier index is also referred to as subcarrier number.

Some subcarriers may not be used for data transmission. Some of these unused subcarriers may be switched off. This function is performed by subcarrier masking (see clause 8.6.1).

NOTE – The particular subcarriers used for data transmission between two particular nodes may depend on channel characteristics, such as loop attenuation and noise, and on the specific spectrum-use requirements, such as the notching of specific frequency bands to share the medium with other services.

8.4.2 Tone mapper

The tone mapper divides the incoming symbol frames of the PFH and payload into groups of bits, according to the used bit allocation table (BAT) and associates these groups of bits with specific subcarriers onto which these groups of bits shall be loaded. This information is passed to the constellation encoder.

8.4.2.1 Summary of subcarrier types

For the purpose of tone mapping, the following types of subcarriers are defined.

1) Masked subcarriers (MSC) are those on which transmission is not allowed, i.e., the gain on this subcarrier (see clause 8.4.3.3) shall be set to zero. Two types of MSC are defined:
   - Permanently masked subcarriers (PMSC) – those that are forbidden for transmission in all regions. Data bits shall not be mapped on a PMSC.
   - Regionally masked subcarriers (RMSC) – those that are forbidden for transmission in some regions, while may be allowed in other regions, and for some applications. The list of RMSCs depends on the region or application or both.

   The number of MSC, #MSC = #PMSC + #RMSC.

2) Supported subcarriers (SSC) are those on which transmission is allowed under restrictions of the relevant PSD mask. Three types of SSC are defined:
   - Active subcarriers (ASC) – those that are loaded bits ($b \geq 1$) for data transmission. ASC are subject to constellation mapping and scaling as described in clause 8.4.3. Data bits shall be mapped on ASCs as described in clause 8.4.2.2.
   - Inactive subcarriers (ISC) – those that are loaded pseudo-random bits instead of data bits. ISC can be used for measurement purposes or other auxiliary purposes. The modulation of ISC is defined in clause 8.4.2.6.

   NOTE – Using zero transmit power with ISCs provides tone masking capabilities on a per connection basis rather than static masking provided by the MSC set.

   - Pilot subcarriers (PSC) – those that carry pilots instead of data bits. PSC can be used for timing recovery, channel estimation, or other auxiliary purposes. The modulation of PSCs is defined in clause 8.4.2.7.

   The number of SSC, #SSC = #ASC + #ISC + #PSC. The SSC are subject to transmit power shaping by using gain scaling (see clause 8.4.3.3).

All subcarriers belong to either MSC or SSC. That is, #MSC + #SSC = N.

8.4.2.2 Bit allocation table (BAT)

Tone mapping is defined by a BAT that associates subcarrier indices with the number of bits to be loaded on the subcarrier. The subcarrier indices in a BAT shall be in ascending order, from the smallest index to the largest index. Bits of the TX symbol frame shall be loaded onto the subcarriers as defined in clause 8.4.3, in the order of subcarrier indices in the BAT.

The BATs used by the node to transmit the particular PHY frame shall be indicated to the receiving node(s) in the PFH, as described in clause 8.2.3.2.2. Up to 16 BATs, with BAT ID values in the range from 0 to 15 can be defined. The assignment of BAT IDs shall be as described in Table 8-17.

Table 8-17 – Assignment of BAT_ID

BAT_ID	Type of BAT	Reference
0	Type 0	Clause 8.4.2.2.1
1	Type 1	
2	Type 2	
3	Type 3	
4	Type 4	
5	Type 5	
6	Type 6	
7	Type 7	
8-15	Reserved by ITU-T for other BATs	

Every node shall at least support BATs of type 0, 1, 2, 4, 5, 6 and 7.

8.4.2.2.1 Predefined BATs

The following BATs are predefined:

1. BAT type 0: uniform 2-bit loading on all subcarriers except the PMSC and PSC sets.
2. BAT type 1: uniform 2-bit loading on all subcarriers except the PMSC, PSC, and RMSC sets (i.e., loaded onto all subcarriers of the SSC set except the PSC).
3. BAT type 2: uniform 2-bit loading on a particular ASC set.
4. BAT type 3: uniform 3-bit loading on a particular ASC set.
5. BAT type 4: uniform 4-bit loading on a particular ASC set.
6. BAT type 5: uniform 1-bit loading on all subcarriers except the PMSC, PSC, and RMSC sets (i.e., loaded onto all subcarriers of the SSC set except the PSC).
7. BAT type 6: uniform 1-bit loading on a particular ASC set.
8. BAT type 7: uniform 1-bit loading on all subcarriers except the PMSC and PSC sets

   NOTE – BAT types 0, 1, 5, and 7 may be used when channel characteristics are unknown (i.e., no knowledge is available on whether particular subcarriers could be loaded with bits or not). If the SNR is below the level to provide reliable detection of 1-bit or 2-bit loading, repetition encoding should be used, as defined in clause 8.3.3.

The particular ASC set to be used in conjunction with BATs of types 2, 4 and 6 shall be defined as a particular subcarrier mask associated with the communication channel by using the TM field of the PFH, while the total number of loaded bits per symbol can also be derived from the PFH fields TM and MOD, as defined in clause 8.2.3.2.2.

8.4.2.3 Transmitter-determined and receiver-determined mapping

Two types of tone mapping are defined: transmitter-determined and receiver-determined. With transmitter-determined mapping, the BAT is defined by the transmitter and shall be either a predefined BAT or it shall be communicated using the BAT communication protocol to all destination nodes prior to transmission. With receiver-determined mapping, the BAT is determined by the receiver of the destination node and communicated to the transmitter. The type of mapping to use is determined by the transmitter. If a transmitter selects to use receiver-determined mapping, the BAT is communicated from the receiver to the transmitter as part of the channel estimation protocol defined in clause 9.5.4.

8.4.2.4 Subcarrier grouping

With subcarrier grouping, the entire bandplan used is divided into groups of consecutive subcarriers, with G subcarriers in each group. The value of $G = 1$ corresponds to no grouping. If grouping is used ($G > 1$), all subcarriers of the same group shall use the same bit loading and the same gain value. The valid values of G are 2, 4 and 8 subcarriers; the eligible values depend on the bandplan and are defined in clause 8.2.3.2.3.

The first group shall include G subcarriers in ascending order of subcarrier indices starting from the smallest index of the used bandplan, as defined in clause 8.5. The second group includes G subcarriers in ascending order of subcarrier indices starting from the smallest index that is bigger than indices of the first group, and so on. If a group includes subcarriers that are masked (e.g., PMSC or RMSC), or are from the PSC set, or extend beyond the upper subcarrier index of the bandplan, the node shall apply the bit loading and gain assigned for this group only to its active subcarriers. The default group index G for a particular bandplan is defined in Table 8-7. Using of more than one (default) value of G for a particular bandplan is for further study.

8.4.2.5 Special mappings

8.4.2.5.1 Tone mapping for PFH

The PFH shall use a uniform loading of 2 bits per subcarrier on all subcarriers except the PMSC set and PSC set (BAT type 0).

8.4.2.5.2 Tone mapping for RCM

Payload transmission in robust communication mode (RCM) shall use a uniform loading of 2 bits per subcarrier (BAT type 0 or BAT type 1).

8.4.2.5.3 Assignment of pilot subcarriers

The PSCs shall be assigned in all symbols of the PFH and in all symbols of the PHY frame payload. Each symbol of the PFH and of the payload shall be assigned the same number of PSCs.

For PSC assignment, the subcarrier indices of a symbol shall be enumerated sequentially over all the subcarriers of the SSC set excluding those of the ISC set, starting from 0 (subcarrier with the lowest frequency) to M-1 (subcarrier with the highest frequency), where M is equal to the difference between the number of SSC and the number of ISC.

The number of PSCs in a symbol, p, shall be computed as:

$$p = \begin{Bmatrix} floor(M/n), & if\ \mathrm{mod}(M,n) \prec k \\ ceiling(M/n), & if\ \mathrm{mod}(M,n) \geq k \end{Bmatrix}$$

where:

- n is the number of subcarriers between adjacent PSC (PSC spacing); the value of n shall be set to 12 for all bandplans;
- k is an index shift between PSC indices of adjacent symbols and shall be set to 3.

The indices of the PSCs in a symbol with sequential number j, j = 1, 2, … s, assigned with p PSCs, shall be equal to:

$$d_x = \mathrm{mod}(\mathrm{mod}(M, n) + (j\text{-}1) \times k + (x\text{-}1) \times n, M), \text{ for } x = 1, \ldots, p.$$

where $\{d_x\}$ is the set of indices of the PSC taken from the set of M subcarriers defined above, where the first subcarrier index of the symbol is 0. The value of j = 1 corresponds to the first symbol of the PFH, and the value $j = s$ corresponds to the last symbol of the payload.

An example of the values of d_x for the set of parameters: M = 36 and n = 12 (that correspond to: mod(M,n) = 0 and p = 3) for the first 6 OFDM symbols is given in Table 8-18.

Table 8-18 – Values of dx for 6 OFDM symbols, using M=36 and n=12

x	Symbol (j)	Pilot tone position (d_x)
1	1	0
2	1	12
3	1	24
1	2	3
2	2	15
3	2	27
1	3	6
2	3	18
3	3	30
1	4	9
2	4	21
3	4	33
1	5	12
2	5	24
3	5	0
1	6	15
2	6	27
3	6	3

8.4.2.6 Modulation of inactive subcarriers

Inactive subcarriers (ISC) shall be loaded with a pseudo-random binary sequence (PRBS) defined by the LFSR generator with the polynomial $p(x) = x^7 + x^4 + 1$ shown in Figure 8-17. The LFSR generator shall be initialized at the beginning of the first payload OFDM symbol with a seed 0x7F (bit C_1 in Figure 8-17 is LSB).

The LFSR shall be advanced by two bits for each inactive subcarrier of each symbol of the payload.

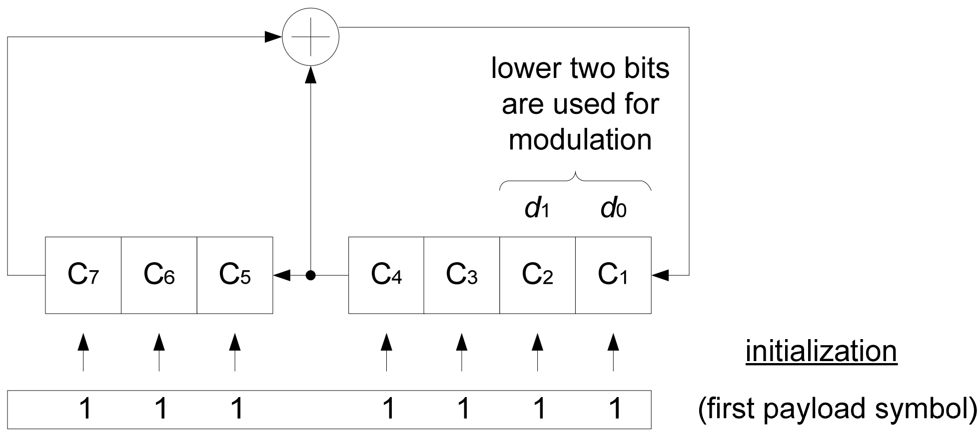


Figure 8-17 – LFSR for the modulation of inactive subcarriers

The modulation of an ISC shall start from the first payload OFDM symbol; each subcarrier from the ISC set shall be modulated with the two bits which are the LSBs of the LFSR, d_0, and d_1 (as presented in Figure 8-17), using 2-bits constellation mapping defined in clause 8.4.3.

Bits from the LFSR shall be loaded on subcarriers from the ISC set in ascending order of subcarrier indices according to the subcarrier indexing defined in clause 8.4.1. The modulation of subcarriers shall start from the ISC with the lowest index of the first payload symbol, continue in ascending order of subcarrier indices till the ISC with the highest index of the first payload symbol, continue with the ISC with the lowest index of the second payload symbol, continue in ascending order of indices till the ISC with the highest index of the second payload symbol, and so on till the ISC with the highest index of the last payload symbol.

8.4.2.7 Modulation of pilot subcarriers

Pilot subcarriers shall be modulated with 2-bit modulation where the bits shall be generated using an LFSR initialized with all ones at the beginning of the PFH, prior to the transmission of the first PSC. The generation polynomial shall be as defined in clause 8.4.2.6.

The modulation of the PSC shall start from the first PFH symbol; each subcarrier from the PSC set shall be modulated with the two bits which are the LSBs of the LFSR, d_0, and d_1 (as presented in Figure 8-17), using the 2-bits constellation mapping defined in clause 8.4.3.

8.4.3 Constellation encoder

The constellation encoder divides the symbol frame (of the PFH or of the payload, see clause 8.3.6) into sequential groups of bits $\{d_{b-1}, d_{b-2}, \ldots, d_0\}$ and maps each group on the corresponding subcarrier. The number of bits in each group and the order of subcarriers is determined by the BAT, as defined in clause 8.4.2.2.

Groups of bits for encoding shall be taken from the symbol frame in sequential order, starting from the first bit of the symbol frame (as bit d_0 of the first group) and ending with the last bit of the symbol frame (bit d_{b-1} of the last group). Groups shall be loaded on subcarriers in the order they are taken from the symbol frame, in ascending order of subcarrier indices (i.e., starting from the ASC with the lowest index and ending by the ASC with the highest index, running sequentially through all subcarrier indices defined in the BAT). Bit assignment for unloaded subcarriers (ISC and PSC) is defined in clauses 8.4.2.6 and 8.4.2.7.

Constellation mapping associates every group of bits to be loaded onto a subcarrier, with the values of *I* (in-phase component) and *Q* (quadrature component) of a constellation point. Each incoming group of b bits $\{d_{b-1}, d_{b-2}, \dots d_0\}$ shall be associated with specific values of *I* and *Q* computed as described in this section. The output of the constellation encoder for a subcarrier *i* is represented as a complex number Z_i and passed to the modulator (see clause 8.4.4). Z_i is derived from Ii and Qi as defined in clause 8.4.3.3.

8.4.3.1 Constellations for an even number of bits

If the number of bits, b, loaded onto the subcarrier is even (i.e., 2 or 4), square-shaped constellations with mappings described in this section shall be used. Support of the 2-bit constellation is mandatory at both the transmitter and the receiver. Support of 4-bit constellation is mandatory at the transmitter and optional at the receiver.

Constellation mapping for $b = 2$ shall be as presented in Figure 8-18 and described in Table 8-19.

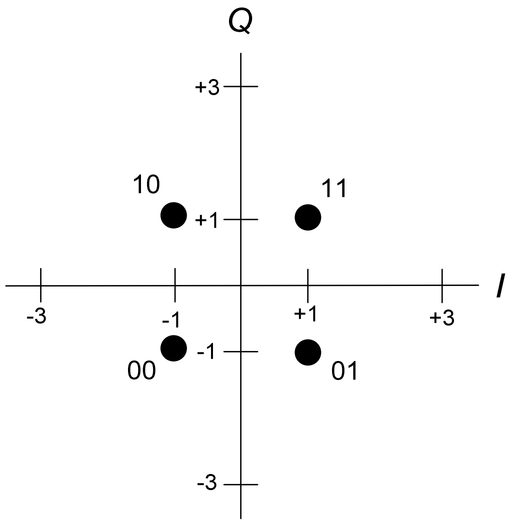


Figure 8-18 – Constellation mapping for b = 2 (d1d0)

Table 8-19 – Mapping for b = 2 (QPSK)

Bit d_0	*I*
0	–1
1	1

Bit d_1	*Q*
0	–1
1	1

Constellation mapping for $b = 4$ shall be as described in Table 8-20. The first quadrant of the mapping is presented in Figure 8-19.

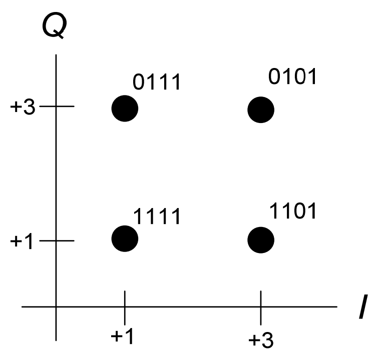


Figure 8-19 – Constellation mapping for *b* = 4 ($d_3d_2d_1d_0$, first quadrant)

Table 8-20 – Mapping for *b* = 4 (16-QAM)

Bits [d_1d_0]	*I*
00	–3
10	–1
11	1
01	3

Bit [d_3d_2]	Q
00	–3
10	–1
11	1
01	3

8.4.3.2 Constellations for odd number of bits

If the number of bits, *b*, loaded onto the subcarrier is odd (i.e., 1 or 3) constellations with mappings described in this section shall be used. Support of the 1-bit constellation is mandatory at both the transmitter and the receiver. Support of 3-bit constellation is mandatory at the transmitter and optional at the receiver.

Constellation mapping for *b* = 1 shall be as presented in Figure 8-20 and Table 8-21.

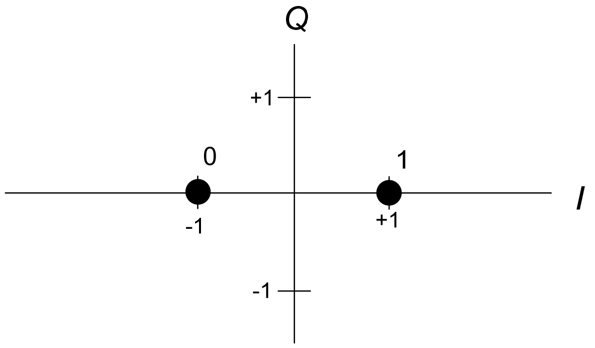


Figure 8-20 – Constellation mapping for *b* = 1 (d_0)

Table 8-21 – Mapping for
***b* = 1 (BPSK)**

Bit d_0	*I*
0	–1
1	1

Constellation mapping for *b* = 3 is for further study.

8.4.3.3 Constellation scaling

Each constellation point (I_i, Q_i) for a subcarrier *i*, corresponding to the complex value $I_i + jQ_i$ at the output of the constellation encoder, shall be scaled by the gain scaling factor *g* and power normalization factor $\chi(b)$ where *b* denotes the number of bits loaded onto a subcarrier. The output of the constellation encoder Z_i shall be:

$$Z_i = g \times \chi(b) \times (I_i + jQ_i)$$

8.4.3.3.1 Power normalization

The power normalization scaling provides all constellations, regardless of their size, having the same average transmit power. The required power normalization scaling, $\chi(b)$, for a subcarrier loaded with *b* bits depends only on the value of b and shall be set as presented in Table 8-22.

Table 8-22 – Power normalization factor

Number of bits loaded (b)	**$\chi(b)$ (linear scale)**
1	1
2	$1/\sqrt{2}$
3	for further study
4	$1/\sqrt{10}$

8.4.3.3.2 Gain scaling

The gain scaling, *g*, provides power shaping by applying a certain average power on different subcarriers. The average transmitted power of a particular subcarrier is controlled by setting an appropriate gain. The following rules shall apply for any frame:

– subcarriers with the same indices of all preamble symbols and all CES symbols shall have the same gain factor;

– subcarriers with the same indices of all PFH symbols shall have the same gain factor;

– subcarriers with the same indices of all payload symbols shall have the same gain factor.

Furthermore, the gain of subcarriers of the same symbol in the preamble, header, and payload shall comply with the rules defined in Table 8-23.

Table 8-23 – Gain factor of different subcarrier sets

Case	**ASC**	**PSC**	**ISC**	**MSC**
Preamble and CES	$GN_0 \times GB_P$	N/A	N/A	0
Header	$GN_0 \times GB_H$	$GN_0 \times GB_H$	N/A	0
Payload	GN_0	GN_0	0 to GN_0	0
NOTE 1 – The GN_0 stands for the nominal gain and GB stands for gain boost. NOTE 2 – The selection of the ISC gain in the assigned range is vendor discretionary.				

The nominal gain (payload gain) GN_0 and the gain boosts GB_P (of the preamble) and GB_H (of the PF_H) relative to the payload gain GN_0 shall be set so that the transmit power limits defined in clause 8.7 shall not be violated during preamble, PFH, and payload.

The maximum value of either GB_P or GB_H is for further study, but shall not exceed 1.41 (3 dB boost). The default value GB is 1 (no boost for preamble and header).

By default, the nominal gain, GN_0, is the same for all subcarriers. Use of different values of GN0 for subcarriers with different indices (spectrum shaping) is for further study.

8.4.4 OFDM modulator

The OFDM modulator consists of the following major parts: IDFT, cyclic extension, windowing, overlap and add, and frequency up-shift. The incoming signal to the modulator at the l-th OFDM symbol in the present frame for a single subcarrier with index i, is a complex value $Z_{i,l}$ generated by the constellation encoder, as described in clause 8.4.3 (for symbols of the PFH and the payload), or by the preamble generator, as described in clause 8.4.5 (for symbols of the preamble), or by the CES generator, as described in clause 8.4.6 (for CES symbols). Time-domain samples generated by the IDFT, after adding the cyclic prefix and windowing, are frequency up-shifted by F_{US}. The functional diagram of an OFDM modulator is presented in Figure 8-21.

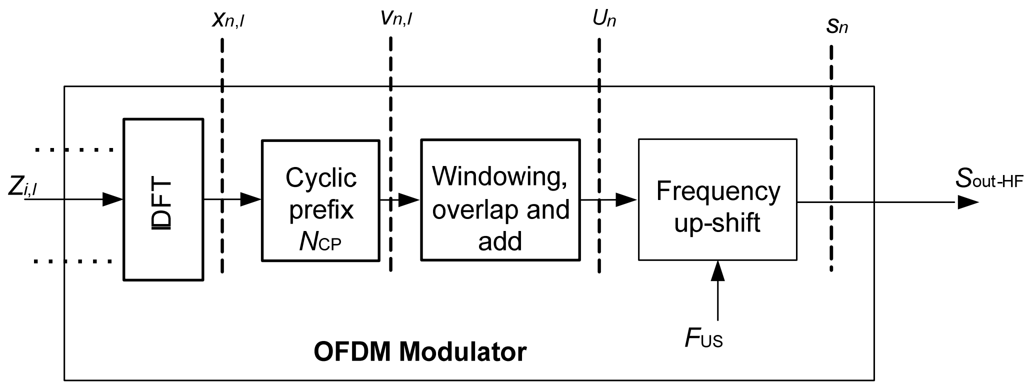


Figure 8-21 – Functional diagram of the OFDM modulator

The presented functional diagram and other figures presented in this section do not imply any specific implementation. All aspects of signal processing used in the modulator shall comply with equations and textual descriptions.

8.4.4.1 IDFT

The IDFT converts the stream of the N complex numbers $Z_{i,l}$ at its input into the stream of N complex time-domain samples $x_{n,l}$. The input values represent the N mapped blocks of data, where the i-th block of data represents the complex value $Z_{i,l}$ of the i-th modulated subcarrier of the OFDM signal, where i = 0, 1, … N-1 is the subcarrier index and l is the sequential number of the OFDM symbol within the current frame, excluding the preamble. The conversion shall be performed in accordance with the equation:

$$x_{n,l} = \sum_{i=0}^{N-1} \exp\left(j \cdot 2\pi \cdot i \frac{n}{N} \right) \cdot Z_{i,l} \qquad \text{for } n = 0 \text{ to } N-1, \quad l = 0 \text{ to } M_F - 1$$

where M_F denotes the total number of OFDM symbols in the current frame excluding the preamble symbols, and the value of N represents the maximum number of possibly modulated subcarriers in the OFDM spectrum and shall be either 128 or 256 (see Table 8-27). The value of $Z_{i,l}$ for all masked subcarriers shall be set to 0. For non-masked subcarriers with indices i < N that are from the ISC and PSC subcarrier sets), the corresponding values of $Z_{i,l}$ shall be generated as described in clauses 8.4.2.6 and 8.4.2.7, respectively.

8.4.4.2 Cyclic extension and OFDM symbol

The cyclic extension provides a guard interval between adjacent OFDM symbols. This guard interval is intended to protect against inter-symbol interference (ISI).

The guard interval of the l-th OFDM symbol in the frame shall be implemented by prepending the last $N_{CP}(l)$ samples of the IDFT output (called cyclic prefix) to its output N samples, as presented in Figure 8-22. The order of samples in the symbol shall be as follows:

- the first sample of the symbol is the IDFT output sample $N–N_{CP}(l)$;
- the last sample of the cyclic prefix is the IDFT output sample N–1; the next sample is the IDFT output sample 0.

The l-th OFDM symbol consists of N IDFT samples and $N_{CP}(l)$ cyclic extension, samples, in total:

$$N_W(l) = N + N_{CP}(l) \text{ [samples]}.$$

After the cyclic extension described above, time-domain samples at the reference point $\upsilon_{n,l}$ in Figure 8-21 shall comply with the following equations:

$$\upsilon_{n,l} = x_{n-N_{CP}(l),l} = \sum_{i=0}^{N-1} Z_{i,l} \times \exp\left(j \cdot 2\pi \cdot i \frac{n - N_{CP}(l)}{N} \right) \quad \text{for } n = 0 \text{ to } N_W(l) - 1 = N + N_{CP}(l) - 1$$

The number of IDFT samples, N, and the number of windowed samples, β, shall be the same for all symbols of the same PHY frame.

8.4.4.3 Symbol timing

The PHY frame consists of a preamble followed by an integer number, M_F, of OFDM symbols. The first symbol following the preamble (the first symbol of the PFH) shall have the symbol count 0, and the last symbol of the frame shall have the symbol count $M_F - 1$. The time position of each symbol in the frame is defined by a sample count. The first sample of the symbol with the symbol count 0 shall have the sample count $M(0)=N_{pr}-\beta$, where N_{pr} is the number of samples in the preamble. The count of the first sample of the l-th symbol ($l = 1, 2, \ldots, M_F - 1$) in the frame shall be:

$$M(l) = N_{\text{pr}} - \beta + \sum_{k=0}^{l-1} N_S(k)\,,$$

where $N_S(k) = N + N_{CP}(k) - \beta$ and $N_S(k)$ are different for symbols of the PFH and the, payload, as described in clause 8.4.7.

8.4.4.4 Windowing, overlap and add

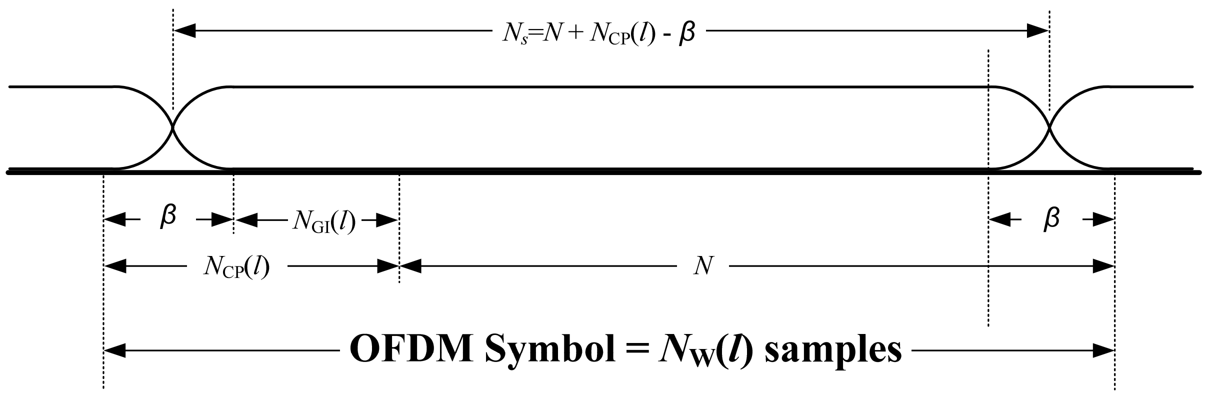


Figure 8-22 – Structure of an OFDM symbol

The first β samples of the cyclic prefix and the last β samples of the IDFT output shall be used for shaping the envelope of the transmitted signal (windowing). The window function facilitates PSD shaping: it allows sharp PSD roll-offs used to create deep spectral notches and a reduction of the out-of-band PSD. The number of windowed samples, β, shall be the same for all of the payload symbols, PFH symbols, CES, and preamble symbols of the same frame.

The windowed samples of adjacent symbols shall overlap, as shown in Figure 8-22. The value of $N_{CP}(l) - \beta = N_{GI}(l)$ forms the guard interval. The number of samples in the l-th OFDM symbol is thus $N_S(l) = N + N_{CP}(l) - \beta$.

After windowing, overlap and add, the time-domain samples at the reference point u_n in Figure 8-21 shall comply with the following equations:

$$u_n = u_n^{(pr)} + \sum_{l=0}^{M_F-1} w(n - M(l), l) \times \upsilon_{n-M(l),l} \quad \text{for } n = 0 \text{ to } M(M_F - 1) + N_W(M_F - 1) - 1$$

where $u_n^{(pr)}$ is the n-th sample of the preamble, as defined in clause 8.4.5 (the signal $u_n^{(pr)}$ already includes windowing), and $w(n,l)$ is the windowing function defined on $N_W(l)$ samples of the OFDM symbol in the following way:

$$w(n,l) = \begin{cases} w_\beta(n) & 0 \le n < \beta \\ 1 & \beta \le n < N_W(l) - \beta \\ w_\beta(N_W(l) - 1 - n) & N_W(l) - \beta \le n < N_W(l) \\ 0 & \text{otherwise} \end{cases}$$

where $w_\beta(n)$ is the function describing the roll-off section of the window. The roll-off function $w_\beta(n)$ shall be determined by the vendor; however, it shall comply with the following rules:

- $w_\beta(n) + w_\beta(\beta\text{-}n\text{-}1) = 1$ for $0 \le n < \beta$.
- $0 \le w_\beta(n) \le 1$.

The symbol period T_{OFDM} for the given value of N_{CP} and β shall be computed, respectively, as:

$$T_{OFDM} = \frac{N + N_{CP} - \beta}{N \times F_{SC}}$$

8.4.4.5 Frequency up-shift

The frequency up-shift offsets the spectrum of the transmit signal shifting it up by F_{US}. The value of F_{US} shall be a multiple of the subcarrier frequency F_{SC}:

$$F_{US} = m \times F_{SC},$$

where m is an integer and $m \ge N/2$. The valid values of m are specified in clause 8.4.7, Table 8-27.

The real and imaginary components of the signal after frequency up-shift (reference point s_n in Figure 8-21) shall be as follows:

$$s_n = u_{n/p} \times \exp\left(j\frac{2\pi mn}{Np} \right) = \text{Re}(s_n) + j\,\text{Im}(s_n) \qquad \text{for } n = 0 \text{ to } [M(M_F - 1) + N_W(M_F - 1)] \times p - 1;$$

$$\text{Re}(s_n) = \text{Re}(u_{n/p})\cos\left(\frac{2\pi mn}{Np} \right) - \text{Im}(u_{n/p})\sin\left(\frac{2\pi mn}{Np} \right)$$

$$\text{Im}(s_n) = \text{Re}(u_{n/p})\sin\left(\frac{2\pi mn}{Np} \right) + \text{Im}(u_{n/p})\cos\left(\frac{2\pi mn}{Np} \right)$$

where $u_{n/p}$ is u_n after interpolation with factor p. The interpolation factor p is determined by the vendor, and shall be equal to or higher than 2.

NOTE – The minimum value of p sufficient to avoid distortions depends on the ratio between the up-shift frequency F_{US} and the bandwidth of the transmit signal BW = $N \times F_{SC}$. It is assumed that an appropriate low-pass filter is included to reduce imaging.

NOTE – The phase of the up-shift should be initialized to zero at the first sample of the preamble and be advanced by $\frac{2\pi m}{Np}$ per each sample (after interpolation).

8.4.4.6 Output signal

The output signal of the modulator shall be the real component of s_n:

$$S_{out\text{-}HF} = \text{Re}(s_n)$$

8.4.5 Preamble

8.4.5.1 General preamble structure

The preamble shall be prepended to every PHY frame as defined in clause 8.2.1. It is intended to assist the receiver in detecting the presence of the frame, synchronizing to the frame boundaries, and acquiring the physical layer parameters such as channel estimation and OFDM symbol alignment. The preamble shall meet the same transmit signal limits as the PFH and the payload symbols of the PHY frame, as defined in clause 8.7.

Table 8-24 presents the general structure of the ITU-T G.9902 preamble. The preamble comprises of two sections. Each section I (I = 1, 2) comprises NI repetitions of an OFDM symbol S_I employing all subcarriers of the SSC set (with subcarrier spacing F_{SC}). Each preamble section shall be windowed in order to comply with the transmit signal limits using the windowing mechanism defined in clause 8.4.4.4. The general preamble structure is illustrated in Figure 8-23, and the relevant parameters N_1 and N_2 are defined in Table 8-24.

Table 8-24 – Structure of the preamble

Parameter	1st section	2nd section
Number of symbols (N_I)	N_1 (Note 1)	$N_2 = 1$
Subcarrier spacing	F_{SC}	F_{SC}
Type of symbol (S_I)	S_1	$S_2 = -S_1$ (Note 2)
NOTE 1 – The valid values of N_1 are 8 and (8 + *ceiling*[(AC_Cycle/4)/T_{OFDM}], where AC_Cycle = 20 ms for 50 Hz mains and 16.67 ms for 60 Hz mains. Other valid values of N_1 are for further study. The value of N_1 to be used is determined by the PMD_MGMT.REQ primitives (see clause 8.8.2.4). NOTE 2 – The OFDM symbol of the 2nd section shall be an inverted time-domain waveform of the symbol used in the first section.		

Figure 8-23 shows the ITU-T G.9902 preamble waveform.

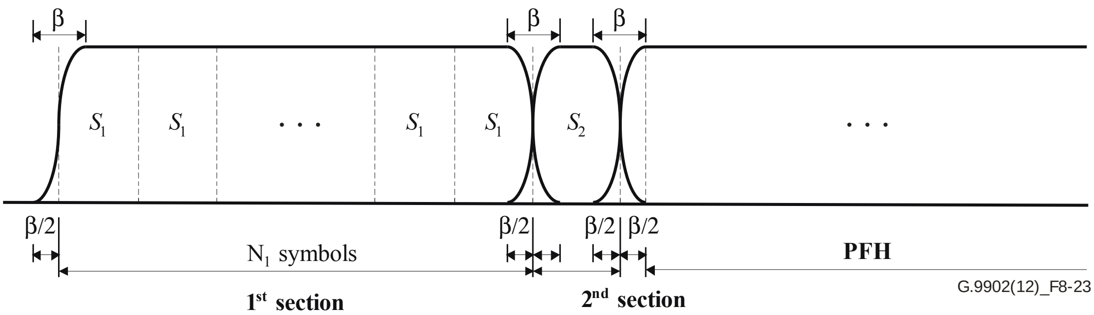


Figure 8-23 − Preamble structure (N_2 = 1)

8.4.5.2 Preamble generation

The preamble generation method described in this section is applicable to all frequency bands.

8.4.5.2.1 Frequency-domain symbol generation

The preamble generator shall output complex values Z_i for each subcarrier i in the range from $i = 0$ to $i = N$-1. These values shall be modulated onto corresponding subcarriers of the symbols of the preamble in accordance with the relevant subcarrier mask (i.e., modulated onto all subcarriers, except those from the PMSC and RMSC shall be masked out), as defined in clause 8.4.4.

The values of Z_i shall be generated by the constellation encoder for 2-bit constellation, as defined in clause 8.4.3.1, fed by the pseudo-random binary sequence (PRBS) generator, as shown in Figure 8-24.

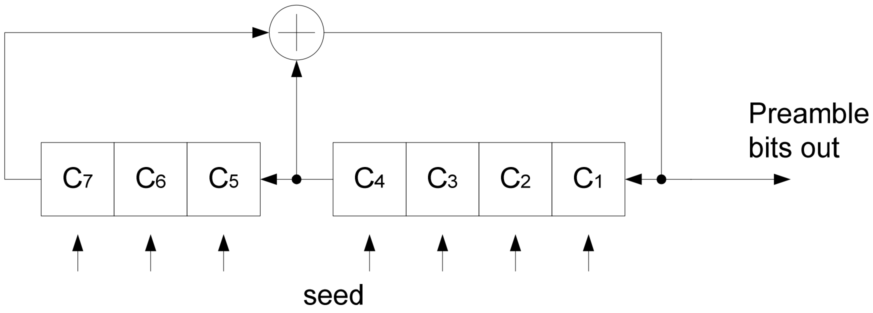


Figure 8-24 – PRBS generator

The PRBS generator shall be initialized at the beginning of each symbol to a seed. The default value of the seed shall be as specified in Table 8-25. In Figure 8-24, C_1 is the LSB of the seed. Other values of the seed are for further study.

Table 8-25 – Default seed value of the PRBS that generates the preamble

Bandplan	Seed value
CENELEC A	29_{16}
CENELEC B	23_{16}
CENELEC CD	50_{16}
FCC	$4C_{16}$
FCC-1	63_{16}
FCC-2	$0E_{16}$

The PRBS generator shall implement the polynomial $g(x) = x^7 + x^4 + 1$. The PRBS shall be advanced by 2 bits for each subcarrier (either masked or not; the shift of the PRBS for subcarrier index k shall be $2k+2$). The output bits of the PRBS generator shall be taken as the input bits of the constellation encoder, $\{d_0, d_1\}$, where d_0 corresponds to C_1 and d_1 corresponds to C_2 of the PRBS generator. Bits shall be assigned to subcarriers in ascending order of their indices, starting from index $i = 0$.

8.4.5.2.2 Time-domain symbol generation

To form a section of a preamble, the output preamble symbol shall be repeated N_I times.

The first and second sections of the preamble shall be windowed, overlapped and added as described below:

1. First section:
   a. The first symbol of the first section is cyclically extended by prepending the last β/2 samples of the symbol S_1.
   b. The last symbol of the first section is cyclically extended by appending the first β/2 samples of the symbol S_1.
   c. The first and last β samples of the extended first section are windowed with a window function $w_\beta(n)$ and $w_\beta(\beta-n-1)$ respectively.

2. Second section:

a. The symbol of the second section is cyclically extended by prepending the last β/2 samples of the symbol S_2 and further cyclically extended by appending the first β/2 samples of the symbol S_2.

b. The first and last β samples of the extended second section are windowed with a window function $w_\beta(n)$ and $w_\beta(\beta\text{-}n\text{-}1)$ respectively.

3. Overlap and add:

a. The β windowed samples at the end of the first section and at the beginning of the second section are overlapped and added.

b. The β windowed samples at the end of the second section are overlapped and added with the β windowed samples at the beginning of the PFH as described in clause 8.4.4.4.

The window shaping function $w_\beta(n)$ shall comply with the rules specified in clause 8.4.4.4.

Assembling of the OFDM symbols in the preamble is illustrated in Figure 8-25.

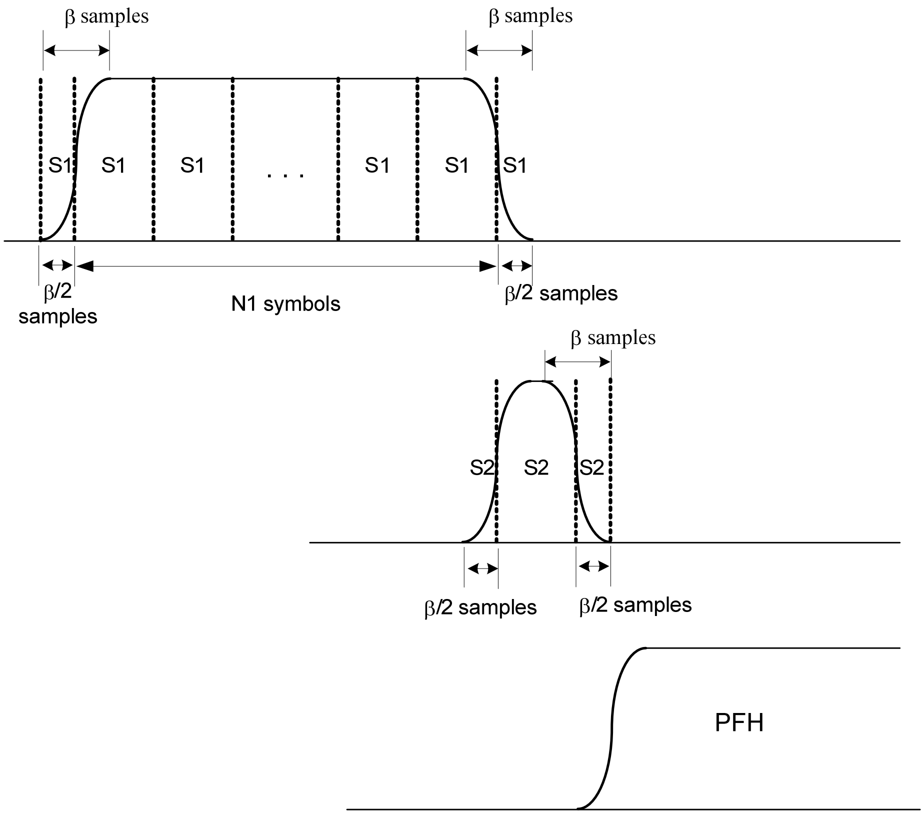


Figure 8-25 – Preamble time-domain generation

The total number N_{pr} of samples in the preamble can be computed as:

$$N_{pr} = \beta + N_1 \times N + N_2 \times N = \beta + N \times (N_1 + 1)$$

8.4.6 Channel estimation symbols

The channel estimation symbols (CES) shall be transmitted using BAT type 0. The modulation parameters of the CES shall be the same as for PFH symbols, as defined in clause 8.4.6.The windowing shall be as used for PFH symbols.

The CES shall be transmitted after N_{OCES} PFH symbols, using the same signal levels as symbols of the preamble and meet the transmit signal limits defined in clause 8.7. The value of NOCES depends on the bandplan and shall be as defined in Table 8-26. If the number of symbols in the PFH (see Table 8-16) is less than the value of N_{OCES} shown in Table 8-26, the CES symbols shall follow the PFH.

Table 8-26 – CES offset value for different bandplans

Bandplan	NOCES
CENELEC A, B, CD (50 Hz)	7
CENELEC A, B, CD (60 Hz)	6
FCC, FCC-1, FCC-2 (50 Hz)	15
FCC, FCC-1, FCC-2 (60 Hz)	13
NOTE – 50 Hz and 60 Hz are the frequencies of the mains.	

The bits loaded onto the CES shall be generated using the PRBS generator defined in clause 8.4.5.2.1. The PRBS generator shall be initialized at the beginning of each CES with the same seed as for the preamble symbols S1 and S2. The first CES shall be equal to S2, while the second CES shall be an inverted copy of the first CES, i.e., –S2 = S1.

8.4.7 PMD control parameters

Table 8-27 summarizes the valid values of control parameters of an OFDM modulator described in clause 8.4.4. This list is a superset of all parameters used for different bandplans; a list of valid values of modulation parameters and their valid combinations for each bandplan is presented in clause 8.5.

Table 8-27 – OFDM control parameters

Notation	Parameter	Valid values or range
N	Number of subcarriers	2^k, $k = 7, 8$
F_{SC}	Subcarrier spacing [kHz]	$15.625/n$, $n = 5, 10$
$N_{GI\text{-}CES}$	Guard interval of the CES [samples]	0
$N_{GI\text{-}HD}$	Guard interval of the PFH [samples]	0
$N_{GI\text{-}PL}$	Guard interval of the payload [samples]	$(12/128)\times N$, $(24/128)\times N$
β	Window size [samples]	Any even integer between 0 and $N/16$
F_{US}	Up-shift frequency, [kHz]	$N/2\times F_{SC}$
NOTE – Guard interval and window size are expressed in samples at Nyquist rate.		

Secondary parameters of the OFDM modulator are presented in Table 8-28.

Table 8-28 – Secondary parameters of the modulator

Notation	Parameter	Definition
BW	Total bandwidth [Hz]	$BW = N \times F_{SC}$
NW	Total number of samples in an OFDM symbol	$N_W = N + N_{CP}$
T_{OFDM}	Symbol period [s]	$T_{OFDM} = \frac{N + N_{CP} - \beta}{N \times F_{SC}}$
N_{GI}	Guard interval	$N_{GI} = N_{CP} - \beta$
f_s	Transmit clock	$f_s = N \times F_{SC}$

8.5 Frequency band specification

See clause A.1 of [ITU-T G.9901].

8.5.1 CENELEC bandplans

For the OFDM control parameters in CENELEC bandplans, see clause A.1.1 of [ITU-T G.9901].

8.5.1.1 CENELEC-A bandplan

For the OFDM control parameters in the CENELEC A bandplan, see clause A.1.1.1 of [ITU-T G.9901].

8.5.1.2 CENELEC-B bandplan

For the OFDM control parameters in the CENELEC B bandplan, see clause A.1.1.2 of [ITU-T G.9901].

8.5.1.3 CENELEC-CD bandplan

For the OFDM control parameters in the CENELEC CD bandplan, see clause A.1.1.3 of [ITU-T G.9901].

8.5.2 FCC bandplans

For the OFDM control parameters in the FCC bandplans, see clause A.1.2 of [ITU-T G.9901].

8.5.2.1 Full FCC bandplan

For the OFDM control parameters in the full FCC bandplan, see clause A.1.2.1 of [ITU-T G.9901].

8.5.2.2 FCC-1 bandplan

For the OFDM control parameters in the FCC-1 bandplan, see clause A.1.2.2 of [ITU-T G.9901].

8.5.2.3 FCC-2 bandplan

For the OFDM control parameters in the FCC-2 bandplan, see clause A.1.2.3 of [ITU-T G.9901].

8.6 Transmit PSD mask

See clause A.2 of [ITU-T G.9901].

8.6.1 Frequency notching

See clause A.2.1 of [ITU-T G.9901].

8.7 Electrical specification

8.7.1 System clock frequency tolerance requirements

The node system clock frequency tolerance shall not exceed ±50 ppm.

The subcarrier frequencies and symbol timing shall be derived from this same system clock oscillator and thus shall have the same tolerance.

8.7.2 Transmit signal limits

See clause A.3.1 of [ITU-T G.9901].

8.7.2.1 CENELEC bandplans

See clause A.3.1.1 of [ITU-T G.9901].

8.7.2.2 FCC bandplans

See clause A.3.1.2 of [ITU-T G.9901].

8.7.2.3 Notched frequency bands

See clause A.3.1.3 of [ITU-T G.9901].

8.7.2.4 FCC standard termination network

See clause A.3.1.4 of [ITU-T G.9901].

8.7.3 Error vector magnitude limits

The deviation of the actual transmit signal from the corresponding constellation point shall be estimated by the value of the error vector magnitude (EVM) calculated as:

$$EVM = 10\log\left(\frac{error\_vector\_RMS}{reference\_signal}\right)^2$$

The interpretation of EVM components for a constellation point is illustrated in Figure 8-26.

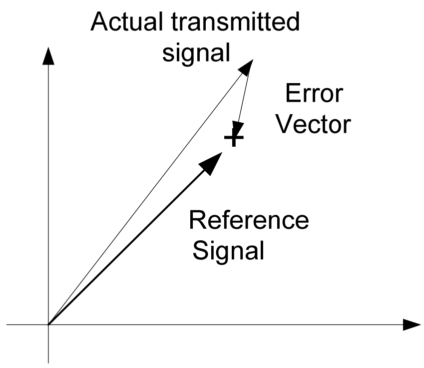


Figure 8-26 – Interpretation of EVM

The EVM shall be determined for the first 12 payload symbols of the transmitted frame using the following procedure:

1. Compute the rms error between the actually transmitted and the ideal constellation points for each symbol as the sum of the squared Euclidean distances between the two mentioned constellation points over all the subcarriers in the symbol (the PPM drift between the transmitter and sampling device should be estimated and corrected):

$$\text{error\_rmsi} = \sum_{c=0}^{K} abs\{A_{ic} \times \exp[j\Phi_{ic}] - B_{ic} \times \exp[j\Theta_{ic}]\}^2$$

where:

K	is the number of ASC in the symbol, numbered from c = 0,1, …K
A_{ic} and Φ_{ic}	are the multitude and phase of the actually transmitted constellation point
B_{ic} and Θ_{ic}	are the multitude and phase of the ideal constellation point.

2. Compute the total rms error as the sum of the rms errors of 12 individual payload symbols numbered from 0 to 11:

$$\text{total\_error\_rms} = \sum_{i=0}^{11} error\_rms_i$$

3. Compute the rms of each transmitted symbol as:

$$\text{Tx\_rmsi} = \sum_{c=0}^{K} A_{ic}^{\ 2}$$

and the total rms for 12 transmitted symbols as:

$$\text{total\_Tx\_rms} = \sum_{i=0}^{11} Tx\_rms_i$$

4. Compute EVM, as a ratio between the total error rms and total_Tx_rms, expressed in dB:

$$\text{EVM} = 10 \times \log(\text{total\_error\_rms}/\text{total\_Tx\_rms}).$$

The value of the EVM shall not exceed the values in Table 8-29.

Table 8-29 – Maximum allowed EVM values

Modulation	EVM, dB (Note)
1 and 2 bits	−15
3 and 4 bits	−19
NOTE – These EVM requirements shall be met for all applied transmit power levels.	

The EVM values specified in Table 8-29 shall be achieved when the device is loaded on standard termination impedance as defined in clauses A.3.1.1 and A.3.1.4 of [ITU-T G.9901] for CENELEC and FCC bandplans, respectively.

For modulation with 3 and 4 bits, the transmit power levels under which these requirements are met may be lower than those for 1 and 2 bit modulation.

8.8 PHY data, management and control primitives

This section describes in detail the PHY-related reference points defined in clause 5.2.2 (PMI_DATA, PMI_MGMT, and PHY_MGMT).

8.8.1 PMI-interface data primitives

The following data primitives at the PMI_DATA reference point are defined below:

Table 8-30 – PHY data primitives

Category	Primitive	Direction	Description
PMI_DATA	PMI_DATA.REQ	DLL → PHY	DLL requests the PHY to transmit an MPDU
	PMI_DATA.CNF	PHY → DLL	PHY reports to the DLL the status of the MPDU transmission (transmission complete, not complete, failed)
	PMI_DATA.IND	PHY → DLL	PHY passes to the DLL received MPDU data

8.8.1.1 PMI_DATA.REQ

This primitive is sent by the DLL to request transmission of the MPDU. The attributes of the primitive are defined in Table 8-31.

Table 8-31 – The attributes of the PMI_DATA.REQ primitive

Name	Type	Valid range	Description
MPDU length	Integer	0x00-0x6A9	The number of bytes contained in the MPDU to be transmitted by the PHY
MPDU	Array of bytes	Any	An array of bytes forming the MPDU to be transmitted by the PHY
Number of information codewords	Integer	1-32	The number of RS information blocks in the MPDU, *m* (Note)
NOTE – The size of the RS information blocks shall be as defined in clause 8.3.2.			

The PHY should start the transmission no later than 0.1× TTS after the PMI_DATA.REQ is issued by the MAC. The TTS is defined in clause 9.1.4.

8.8.1.2 PMI_DATA.CNF

This primitive reports the status of the MPDU transmission to a peer PHY. The attributes of this primitive are defined in Table 8-32.

Table 8-32 – The attributes of the PMI_DATA.CNF primitive

Name	Type	Valid range	Description
MPDU Tx status	Integer	0-3	The status of the MPDU requested for transmission: 0 – Transmitted successfully (PHY is ready to accept the next frame for transmission) 1 – Not transmitted (busy transmitting a frame, Note 1) 2 – Transmission failed (PHY is receiving a frame or getting ready to receive a frame, Note) 3 – Transmission failed (MPDU of invalid size)
NOTE – If the DLL sends a PMI_DATA.REQ when PMI_DATA.CNF = 1, 2, the PHY may ignore the PMI_DATA.REQ primitive and discard the MPDU requested for transmission.			

8.8.1.3 PMI_DATA.IND

This primitive indicates the transfer of a received MPDU from the PHY to the DLL. The attributes of the primitive are defined in Table 8-33.

Table 8-33 – The attributes for the PMI_DATA_MPDU.IND primitive

Name	Type	Valid range	Description
MPDU length	Integer	0x00-0x6A9	The number of bytes contained in the MPDU received by the PHY
MPDU	Array of bytes	N/A	An array of bytes forming the MPDU received by the PHY
MPDU error	Bit map	32-bit	This primitive indicates errors that were detected in the m RS information blocks of the received MPDU: 0 – No error detected in the codeword by the PHY 1 – PHY detected an error in the *k*-th RS information block of the received MPDU The first bit of the bit map shall correspond to the first RS information block, and the *m*-th bit of the bit map shall correspond to the last RS information block in the received MPDU.

8.8.2 PMI-interface and PHY management and control primitives

The control and management primitives at the PMI_MGMT reference point and PHY_MGMT reference point are defined in Table 8-34:

Table 8-34 – PHY management and control primitives

Category	Primitive	Description
PMI_MGMT	PMI_MGMT.REQ	DLL request to the PHY to apply particular parameters or perform particular functions
	PMI_MGMT.CNF	PHY confirms parameters and functions requested by the DLL
	PMI_MGMT.IND	PHY indicates to the DLL its status, status of the medium, and particular parameters of the received frame
	PMI_MGMT.RES	DLL acknowledges receipt of the PHY status, status of the medium, parameters of the received frame
PHY_MGMT.REQ	PCS_MGMT.REQ	PHY management entity requests to apply particular parameters of PCS, PMA, PMD for the transmit frames
	PMA_MGMT.REQ	
	PMD_MGMT.REQ	
PHY_MGMT.CNF	PCS_MGMT.CNF	PHY sublayers (PCS, PMA, PMD) confirm parameters applied for the transmit frame
	PMA_MGMT.CNF	
	PMD_MGMT.CNF	
PHY_MGMT.IND	PCS_MGMT.IND	PHY sublayers (PCS, PMA, PMD) report to the PHY management entity particular parameters of the received frame and acquired channel characteristics
	PMA_MGMT.IND	
	PMD_MGMT.IND	
PHY_MGMT.RES	PCS_MGMT.RES	DLL management acknowledges parameters of the received frame and channel characteristics reported by the PHY sublayers (PSC, PMA, PMD)
	PMA_MGMT.RES	
	PMD_MGMT.RES	

8.8.2.1 PMI_MGMT primitives

8.8.2.1.1 PMI_MGMT.REQ

This primitive requests the PHY to turn into a particular status (enable or disable the receiver), apply particular parameters, and perform particular functions asserted by the DLL. The attributes of the primitive are defined in Table 8-35.

Table 8-35 – The attributes of the PMI_MGMT.REQ primitive

Name	Type	Valid range	Description
RxEnbl	Integer	0, 1	Requests to turn on/off the receiver: 0 – Receiver is enabled 1 – Receiver is disabled (See Note)
Request for physical carrier sense	Integer	0-3	Requests the PHY for the physical carrier sense status: 0 – No request 1 – Request for ITU-T G.9902 carrier sense 2 – Request for non-ITU-T G.9902 carrier sense (see "Preamble-based coexistence mechanism" in clause 5.1.2.3)

Table 8-35 – The attributes of the PMI_MGMT.REQ primitive

Name	Type	Valid range	Description
			3 – Request for both ITU-T G.9902 and non-ITU-T G.9902 carrier sense
ACK request	Integer	0-3	Request for an ACK for the transmitter frame: 0 – No request 1 – Regular ACK requested 2 – Extended ACK requested 3 – Reserved by ITU-T
ACK data type	Integer	0-3	The type of ACK request for the transmitted frame (see clause 9.3.3.1.1): 0 – Acknowledgement to MS-MPDU 1 – Acknowledgement to SS-MPDU 2 – Extended acknowledgement 3 – Reserved by ITU-T
TP-PR	Array of bits	clause 9.3.3.1.1.1	The content of the TP partial report to be transmitted by the node using the format defined in clause 9.3.3.1.1.1
ACK data	Array of bits	clause 9.3.3.1.1	A set of bits forming the ACK related parameters to be transmitted by the PHY in the Imm-ACK frame (see clause 9.3.3.1.1)
PHY parameters	See 8.8.2.2	See clause 8.8.2.2	The attributes of the PHY_MGMT.REQ primitive asserted by the DLL management entity and defined in clause 8.8.2.2.1 (PCS), clause 8.8.2.3.1 (PMA), and clause 8.8.2.4.1 (PMD)
NOTE – The receiver shall be off when a node transmits a frame. The node shall not transmit when this primitive is set to 0.			

8.8.2.1.2 PMI_MGMT.CNF

This primitive confirms the status, parameters, and functions of the PHY in response to PMI_MGMT.REQ. The attributes of the primitive are defined in Table 8-36.

Table 8-36 – The attributes for the PMI_MGMT.CNF primitive

Name	Type	Valid range	Description
Receiver status	Integer	0-2	Confirms the status of the receiver 0 – Receiver is enabled 1 – Receiver is disabled 2 – Receiver is busy NOTE – The "busy" status indicates that the receiver is in the middle of receiving a frame and cannot perform the request to be disabled.
ACK TX status	Integer	0, 1	0 – Transmitted (PHY is ready to accept the next frame for transmission) 1 – Not transmitted (busy transmitting the ACK frame)

Table 8-36 – The attributes for the PMI_MGMT.CNF primitive

Name	Type	Valid range	Description
PHY parameters status	Array of integers	0, 1	The attributes of PHY_MGMT.CNF primitive, indicates whether the PHY parameters asserted by the DLL management entity and defined in clause 8.8.2.2.2 (PCS), clause 8.8.2.3.2 (PMA), and clause 8.8.2.4.2 (PMD) was accepted or denied 0 – Success 1 – Request is denied

8.8.2.1.3 PMI_MGMT.IND

This primitive indicates to the DLL management entity the status of the PHY and the medium, and the parameters of the received frame. The attributes of the primitive are defined in Table 8-37.

Table 8-37 – The attributes of the PMI_MGMT.IND primitive

Name	Type	Valid range	Description
Physical carrier sense	Integer	0-3	See physical carrier sense attribute of PMD_MGMT.IND primitive, Table 8-46
ACK request	Integer	0-3	See ACK request attribute of the PCS_MGMT.IND primitive, Table 8-40
ACK data type	Integer	0-3	See ACK data type attribute of the PCS_MGMT.IND primitive, Table 8-40
ACK data	Array of bits	See clause 9.3.3.1.1	See ACK data attribute of the PCS_MGMT.IND primitive, Table 8-40
TP-PR	Array of bits	See clause 9.3.3.1.1.1	See TP-PR attribute of the PCS_MGMT.IND primitive, Table 8-40
PHY parameters	See clause 9.8.2.2	See clause 9.8.2.2	The attributes of PHY_MGMT.IND primitive delivered by the received frame to be passed to the DLL management entity and defined in clause 8.8.2.2.3 (PCS), clause 8.8.2.3.3 (PMA), and clause 8.8.2.4.3 (PMD)

8.8.2.1.4 PMI_MGMT.RES

This primitive is for further study.

8.8.2.2 PCS_MGMT primitives

8.8.2.2.1 PCS_MGMT.REQ

This primitive requests the PCS to use particular parameters for frame transmission. The attributes of the primitive are defined in Table 8-38.

Table 8-38 – The attributes of the PCS_MGMT.REQ primitive

Name	Type	Valid range	Description
Type of frame	Integer	1-4	Type of the transmitted PHY frame
PFH data	See clauses 8.2.1and 9.3.3.1.1	See clauses 8.2.1 and 9.3.3.1.1	The PFH parameters of the transmitted frame are defined: – clause 8.8.1 – Imm-ACK frame related – clause 8.8.3 – PMA-related – clause 8.8.4 – PMD related

8.8.2.2.2 PCS_MGMT.CNF

This primitive confirms the particular parameters used by the PCS for frame transmission. The attributes of the primitive are as defined in Table 8-39.

If the PHY is unable to comply with a particular attribute in the PCS_MGMT.REQ, it shall set this primitive to one, which means that the request is denied (and the frame is not to be transmitted). Otherwise the value of the PCS_MGMT.CNF primitive shall be set to zero.

Table 8-39 – The attributes of the PCS_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0, 1	0 – Success 1 – Request is denied

8.8.2.2.3 PCS_MGMT.IND

This primitive provides the PHY management with particular parameters of the received frame derived from the received PFH. The attributes of the primitive are defined in Table 8-40.

Table 8-40 – The attributes of the PCS_MGMT.IND primitive

Name	Type	Valid range	Description
Virtual carrier sense	Integer	0-1024	Indicates the number of symbols in the payload of the frame sequence during which the medium will be busy (valid for ITU-T G.9902 frame types 2 and 4 only)
Type of the frame (Note)	Integer	1-4	Type of received PHY frame
RX PFH status	Integer	0-2	Status of PFH of the received frame: 0 – Correct 1 – HCS error 2 – Invalid content
MPDU size	Integer	0-255	Number of bytes in the MPDU of the received frame
Payload modulation	Integer	2-4	Number of bits per subcarrier used for payload modulation in the received frame
Payload repetitions	Integer	1-12	Number of repetitions in the payload of the received frame

Table 8-40 – The attributes of the PCS_MGMT.IND primitive

Name	Type	Valid range	Description
Payload interleaving mode	Integer	0, 1	Payload interleaving mode of the received frame: 0 – IoAC 1 – IoF
RS codeword size	Integer	0, 1	Maximum number of bytes in RS codeword in the payload of the received frame 0 – 239 1 – 128
Inner code rate	Integer	0, 1	Indicates the code rate of the convolutional encoder: 0 – 1/2 1 – 2/3
Tone mask	Array of bits	1_{16}-FF_{16} (CENELEC, FCC-1) 1_{16}-$FFFFFFFFFF_{16}$ (FCC, FCC-2)	Indicates the tone mask used to transmit the payload of the received frame
ACK request	Integer	0, 3	Indicates whether acknowledgement for the received frame is required: 0 – ACK not required 1 – A regular Imm-ACK required 2 – An extended Imm-ACK required 3 – Reserved by ITU-T
ACK data type	Integer	0-3	The type of the ACK data in received Imm-ACK frame (see clause 9.3.3.1.1): 0 – Acknowledgement to MS-MPDU 1 – Acknowledgement to SS-MPDU 2 – Extended ACK 3 – Reserved by ITU-T
ACK data	Array of bits	Clause 9.3.3.1.1	The ACK data delivered by the received Imm-ACK frame (see clause 9.3.3.1.1)
TP-PR	Array of bits	Clause 9.3.3.1.1.1	The TP partial report delivered by the received Imm-ACK frame (as defined in clause 9.3.3.1.1.1)
LQI	1-bit integer	0, 1	The LQI value delivered by the received Imm-ACK frame (clause 9.3.3.1.1)

Table 8-40 – The attributes of the PCS_MGMT.IND primitive

Name	Type	Valid range	Description
BAT type used	Integer	0-15	Indicates the BAT that was used in the received frame: 0 – BAT type 0 1 – BAT type 1 2 – BAT type 2 3 – BAT type 3 4 – BAT type 4 5 – BAT type 5 6 – BAT type 6 7 – BAT type 7 Other values are reserved by ITU-T
NOTE – The primitives that are irrelevant for the indicated frame type shall be set to a default value of 0.			

8.8.2.2.4 PCS_MGMT.RES

This primitive is for further study

8.8.2.3 PMA_MGMT primitives

8.8.2.3.1 PMA_MGMT.REQ

This primitive requests the PMA to use particular parameters for frame transmission. The attributes of the primitive are defined in Table 8-41.

Table 8-41 – The attributes of the PMA_MGMT.REQ primitive

Name	Type	Valid range	Description
Payload repetitions	Integer	1-12	Number of repetitions in the payload of the transmit frame; valid values are 1, 2, 4, 6, 12
Payload interleaving mode	Integer	0, 1	Payload interleaving mode of the transmit frame: 0 – IoAC 1 – IoF
Number of information codewords	Integer	1-32	The number of RS information blocks in the MPDU, *m* (Note)
Number of PFH symbols	Integer	0, 1	0 – Number of symbols used by the PFH shall comply with 'Normal' mode 1 – Number of symbols used by the PFH shall comply with 'Robust' mode
Inner code rate	Integer	0, 1	Indicates the code rate of the convolutional encoder: 0 – 1/2 1 – 2/3
NOTE – The size of the RS codeword shall be as defined in clause 8.3.2.			

8.8.2.3.2 PMA_MGMT.CNF

This primitive confirms the particular parameters used by the PMA for frame transmission. The attributes of the primitive are as defined in Table 8-42.

If the PMA is unable to comply with a particular attribute in the PMA_MGMT.REQ, it shall set this primitive to one, which means that the request is denied (and the frame is not to be transmitted). Otherwise the value of the PMA_MGMT.CNF primitive shall be set to zero.

Table 8-42 – The attributes of the PMA_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0, 1	0 – Success 1 – Request is denied

8.8.2.3.3 PMA_MGMT.IND

This primitive indicates to the PHY management the particular parameters of the received frame. The attributes of the primitive are defined in Table 8-43.

Table 8-43 – The attributes of the PMA_MGMT.IND primitive

Name	Type	Valid range	Description
RS codeword error	Bit map	32-bit	This primitive indicates errors that were detected in the m RS information blocks of the received MPDU: 0 – No error detected in the codeword by the PHY 1 – PHY detected an error in the *k*-th RS information block of the received MPDU The first bit of the bit map shall correspond to the first RS information block, and the *m*-th bit of the bit map shall correspond to the last RS information block in the received MPDU

8.8.2.3.4 PMA _MGMT.RES

This primitive is for further study.

8.8.2.4 PMD_MGMT primitives

8.8.2.4.1 PMD_MGMT.REQ

This primitive requests the PMD to use the particular parameters for frame transmission. The attributes of the primitive are defined in Table 8-44.

Table 8-44 – The attributes of the PMD_MGMT.REQ primitive

Name	Type	Valid range	Description
Bandplan	Integer	0-16	The bandplan to be used for transmission: 0 – CENELEC A 1 – CENELEC B 2 – CENELEC CD 4 – FCC 5 – FCC-1 6 – FCC-2 Other values are reserved by ITU-T

Table 8-44 – The attributes of the PMD_MGMT.REQ primitive

Name	Type	Valid range	Description
TX power	Integer	0-255	The power setting that the PHY has to use for the transmit frame The value represents the required transmit power in dB microvolt
Payload modulation	Integer	2-4	The number of bits per subcarrier to be used by the PHY for payload modulation in the transmit frame
Tone mask	Array	0-1	The tone mask that the PHY has to use for transmission of the frame: 0 – Indicates subcarriers that are not loaded bits (RMSC, ISC, and PSC) 1 – Indicates subcarriers that are loaded bits (ASC)
Number of preamble symbols	Integer	0, 1	The value of N_1 symbols that shall be used for the preamble: 0 – 8 symbols 1 – 8 + $ceiling(T_0/T_{OFDM})$ symbols, where T_0 = 5 ms for 50 Hz mains and T_0 = 4.167 ms for 60 Hz mains
TX power of inactive subcarriers	Integer	0, 1	The transmit power setting for inactive subcarriers (ISC set): 0 – Zero power on all inactive subcarriers 1 – Same power on all active and inactive subcarriers
NOTE – The primitives that are irrelevant for the indicated frame type shall be sent to a default value of 0.			

8.8.2.4.2 PMD_MGMT.CNF

This primitive confirms the particular parameters used by the PMD for frame transmission. The attributes of the primitive are as defined in Table 8-45.

If the PHY is unable to comply with a particular attribute in the PMD_MGMT.REQ, it shall set this primitive to one, which means that the request is denied (and the frame is not to be transmitted). Otherwise the value of the PMD_MGMT.CNF primitive shall be set to zero.

Table 8-45 – The attributes of the PMD_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0, 1	0 – Success 1 – Request is denied

8.8.2.4.3 PMD_MGMT.IND

This primitive provides to the PHY management particular parameters of the received frame. The attributes of the primitive are defined in Table 8-46.

Table 8-46 – The attributes of the PMD_MGMT.IND primitive

Name	Type	Valid range	Description
Physical carrier sense	Integer	0, 1	Indicates the status of the medium: (physical carrier sense based on preamble detection) 0 – IDLE 1 – BUSY due to ITU-T G.9902 transmission 2 – BUSY due to non-ITU-T G.9902 transmission (Note 1) 3 – BUSY due to both ITU-T G.9902 and non-ITU-T G.9902 transmission (Note)
Reception quality	Integer	For further study	A parameter determined by the vendor that characterizes the quality of the link (e.g., to generate channel estimation responses and LQI)
NOTE – Values of 2 and 3 are only valid if a preamble-based coexistence mechanism is enabled (see clause 5.1.2.3); otherwise the valid values of the primitive are 0 and 1 only.			

The physical carrier sense parameter should be changed to BUSY no later than TTS×0.8 after the actual transmission is started on the line (first sample of the first symbol in the preamble is transmitted). The TTS is defined in clause 9.1.4.

8.8.2.4.4 PMD_MGMT.RES

This primitive is for further study.

9 Data link layer (DLL) specification

9.1 Functional model of DLL and frame formats

The functional model of the DLL is presented in Figure 9-1. The A-interface and PMI are respectively, two demarcation reference points between the AE and DLL and between the DLL and PHY. Internal reference points x1 and x2 respectively, show logical separation between the APC and LLC and between the LLC and MAC.

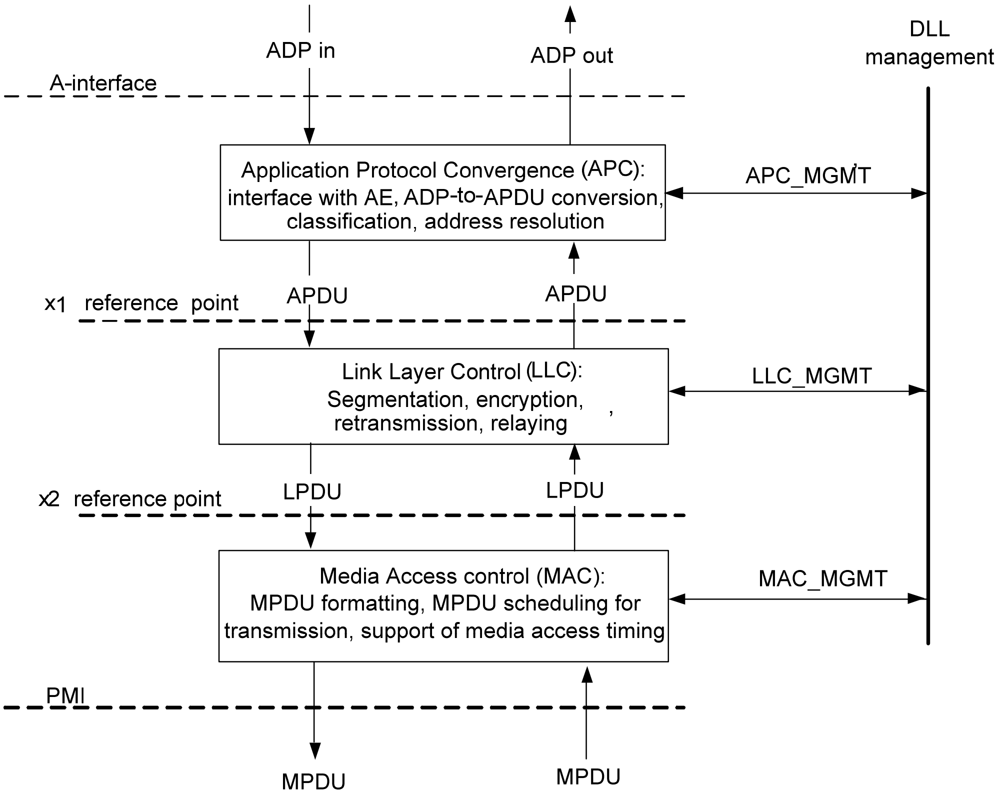


Figure 9-1 – Functional model of the DLL

In the transmit direction, application data primitives (ADP) enter the DLL from the AE via the A-interface. Every incoming ADP set meets the format defined by the particular application protocol; for an AE of IPv6 a particular type of ADP set is defined in clause B.1 (IPv6 APC). Each incoming ADP set is converted by the APC into application protocol data units (APDUs), which includes all parts of the ADP intended for communication to the destination node(s). The APC is responsible for forwarding APDUs to peers APCs.

The APC also identifies ADP classification primitives (e.g., class of service, priority tags, etc.) to support QoS requirements assigned for the service delivered by the ADP and shall maintain one or more priority queues associated with the APDUs it transmits; it assigns each APDU to the corresponding queue. The number of queues may depend on the application protocol and the profile of the device.

NOTE – The bridging function between the clients associated with the ITU-T G.9902 node (if more than one) is considered to be a part of the AE and is beyond the scope of this document.

The APC transfers APDUs to the LLC via the x1 reference point which is application independent. The LLC also receives from the DLL management entity, sets of management data primitives for LLC control frames which are mapped into link control data units (LCDU). The LLC is responsible for establishing the exchange of LCDUs (management frames) between peer LLCs.

In the LLC, the incoming APDU and LCDU are mapped into LLC frames and may be encrypted using assigned encryption keys (see clause 9.6). Long LLC frames are segmented as described in clause 9.1.2.2. Segments are transformed into LLC protocol data units (LPDU) and then passed to the MAC via the x2 reference point. The LLC is also responsible for the relay operation (if enabled) and for the retransmission of improperly received segments (if required).

The MAC is responsible for concatenating LPDUs into MAC protocol data units (MPDUs) and then convey these MPDUs to the PHY via the PMI, in the order determined by their MA priorities and according to the medium access rules established in the domain (e.g., see clause 9.2).

In the receive direction, MPDUs from the PHY enter via the PMI. The MAC disassembles the received MPDUs into LPDUs which are passed over the x2 reference point to the LLC. The LLC recovers the original APDUs and LCDUs from the received LPDUs, decrypts them if required, and conveys them to the APC and the LLC management entity, respectively. In the APC, the ADPs are recovered from the received APDUs and conveyed to the AE.

The LLC is responsible for the detection of erroneous LPDUs and the generation of acknowledgement. It discards the erroneous LPDUs and if the source node has requested the retransmission of erroneous LPDUs, the LLC generates an ACK response to trigger retransmission.

NOTE – No assumptions should be made on the partitioning of APC, LLC, and MAC in particular implementations; x1 and x2 are reference points and serve exclusively for the convenience of system definition.

9.1.1 Application protocol convergence sublayer (APC)

The functional model of the APC is presented in Figure 9-2. It is intended to describe in more detail the APC functional block presented in Figure 9-1.

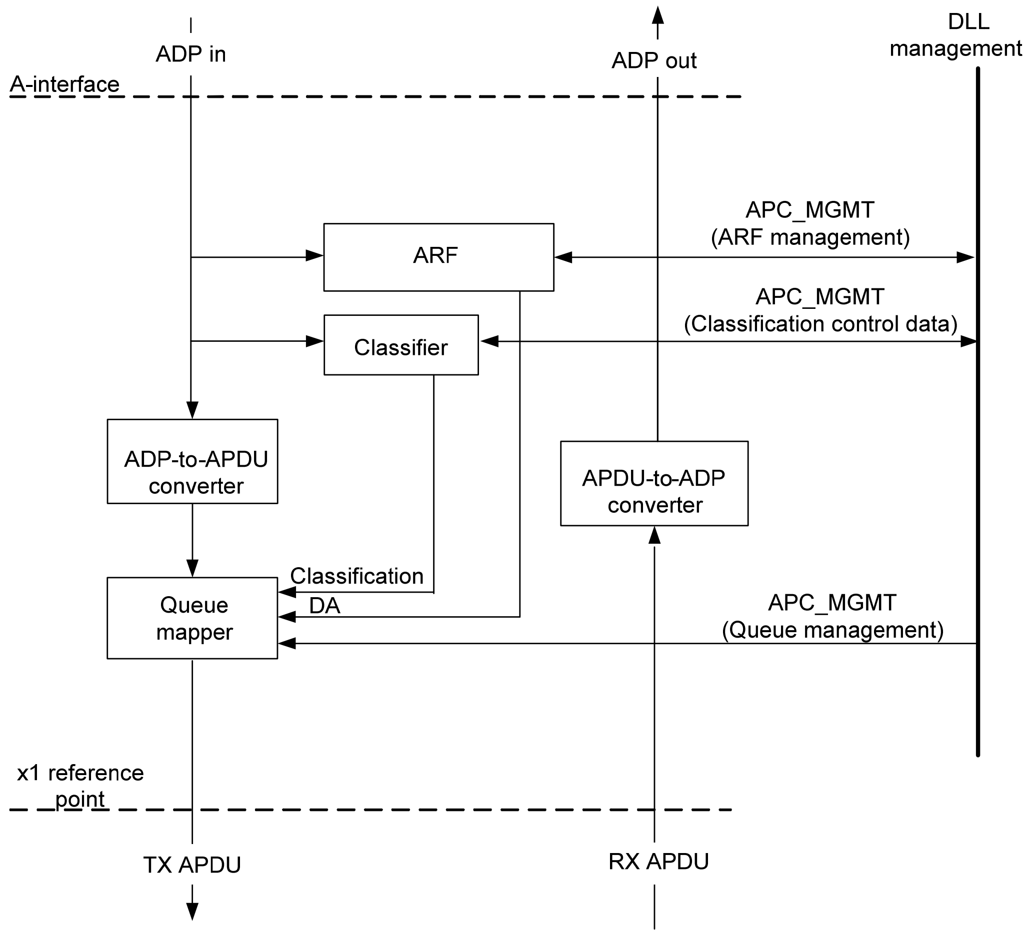


Figure 9-2 – Functional model of the APC

In the transmit direction, the incoming ADP data unit is converted into an APDU as defined in Annex B. The queue mapper maps APDUs into queues depending on their destination address (DA), class of service (priority), and control parameters provided by the DLL management entity, as described in Annex B. After mapping, each APDU, tagged with its priority, is sent to the LLC via the x1 reference point. The size of the ADP (in both transmit and receive directions) can be up to 1522 bytes.

In the receive direction, the APDUs incoming via the x1 reference point are converted back into the ADP set of the corresponding application protocol.

The classification information embedded in the ADP is extracted from the incoming data units and may be used to map the APDU into an appropriate priority queue. Relevant classification parameters depend on the application protocol and are defined in Annex B.

The address resolution function (ARF) associates the destination address of the incoming ADP with the physical address of the node that this ADP has to be sent to. The ARF may store addresses of the clients associated with the node collected from the incoming ADP and addresses of the clients associated with other nodes in the network (advertised by these nodes). The address resolution procedure performed by ARF depends on the application protocol and is defined in Annex B.

The APC sublayer allows one or more application protocols to be served. The functional model in Figure 9-2 shows an APC serving a particular AE protocol. If more than one application protocol is needed, multiple APCs shall be used, as shown in Figure 9-3.

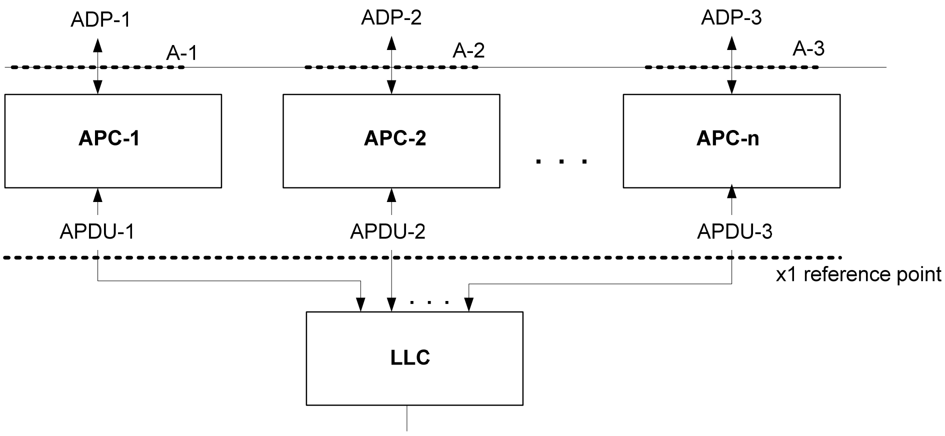


Figure 9-3 – Block diagram of an APC sublayer serving multiple application protocols

Each APC in Figure 9-3 (this also applies to those in Figure 9-2), serves a particular AE protocol. All APDUs from all APCs are passed to the LLC via the x1 reference point, each with a tag indicating its application protocol. The tag is further communicated to the receiving node in the LLC frame type (LLCFT field) of the LLC frame header (LFH) (see clause 9.1.2.1.2); based on this tag, the received APDU is passed to the corresponding APC. The format of each APDU is as per its application protocol, and shall be as defined in Annex B.

Simultaneous support of multiple application protocols is optional. The maximum number of APCs (not more than 8) and arbitration of the order in which APDUs of different APCs are passed to the LLC (internal priority of the APC) are for further study.

9.1.2 Logical link control sublayer (LLC)

The functional model of the LLC is presented in Figure 9-4. It is intended to describe in more detail the LLC functional block presented in Figure 9-1.

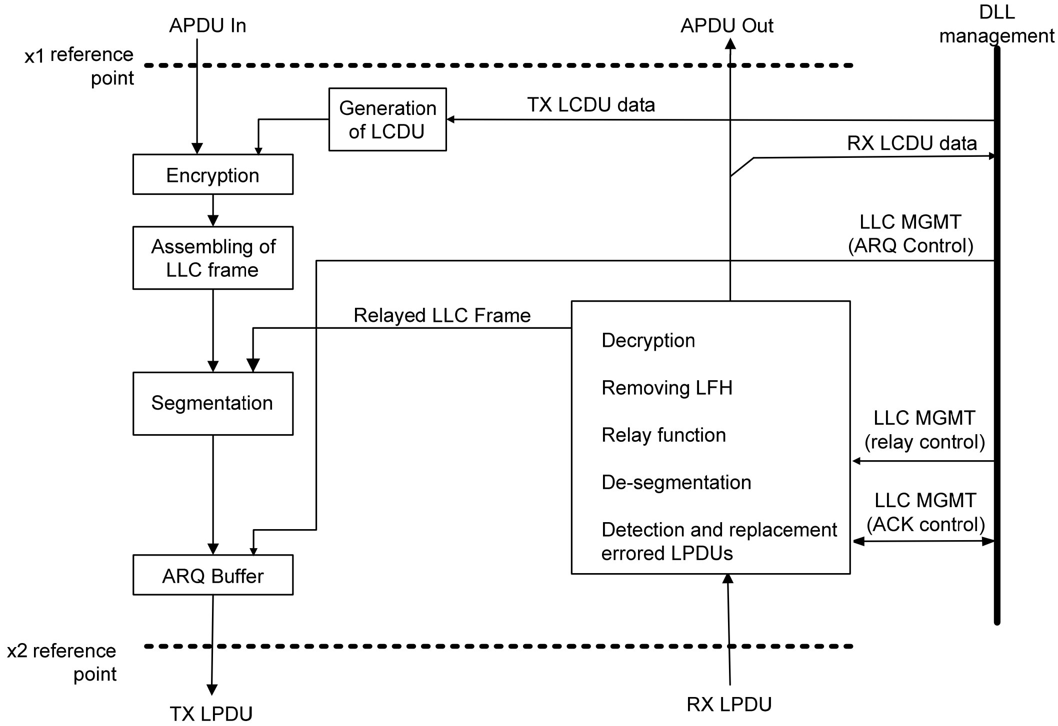


Figure 9-4 – Functional model of LLC

In the transmit direction, an LLC frame is formed from each APDU entering via the x1 reference point and from each LCDU entering via the LLC_MGMT reference point. The APDU has a format depending on the application protocol, as defined in Annex B. The LCDU carries management data and has a format defined in 9.1.2.1.3. The LLC frame may be encrypted using the encryption rules defined in clause 9.6.1.

Furthermore, the LLC frame is divided into segments of equal size, as defined in clause 9.1.2.2; the size of the segment is controlled by the LLC. Each segment is prepended by a header and appended with a CRC, thus forming an LPDU. LPDUs are passed to the MAC via the x2 reference point and also stored in the ARQ buffer, if the transmitted LLC frame has to be acknowledged.

LPDUs that need to be retransmitted are extracted from the ARQ buffer and passed to the MAC to be assembled into the outgoing MPDU. To assist retransmission, the receive part of the LLC generates ACKs which are passed to the MAC to be transmitted using Imm-ACK frames, as defined in clause 9.3.3.1.

In the receive direction, the LPDUs disassembled from the incoming MPDU in the MAC enter the LLC via the x2 reference point. The LLC verifies all received LPDUs, generates an acknowledgement, if so instructed by the source of the received frame, and discards erroneous LPDUs. Additionally, after all the LPDUs associated with the sent LLC frame are cleared, the LLC recovers the LLC frames from the received LPDUs. The recovered LLC frames are decrypted (if encryption is enabled) and their payloads are passed to the APC via an x1 reference point, if they were carrying an APDU or to the DLL management via the LLC_MGMT reference point, if they were carrying an LCDU.

The relay function, if enabled, extracts LLC frames that are subject to relaying using information in the LFH (see clause 9.1.2.1.2) and passes them back to the MAC for transmission to the next destination. The relayed LLC frames are processed as regular LLC frames, see Figure 9-4. The DLL management entity controls the priority settings for the relayed LLC frames as defined in clause 9.1.2.4. Relayed LLC frames shall not be decrypted.

9.1.2.1 Assembling of LLC frames

The LLC frame is formed from either an APDU or an LCDU, with a format as described in Figure 9-5.

9.1.2.1.1 LLC frame format

If encryption or other security measures (see Table 9-42) is required, the incoming APDU or LCDU shall be encrypted using the CCMP, as described in clause 9.6.1. A CCMP header and a message integrity check (MIC) shall be added as described in Figure 9-5 (case – CCM encrypted). The format and content of the CCMP header and the MIC shall be as specified in clause 9.6.1.2.3.

If no security measures are required, the CCMP header and the MIC shall not be added, as described in Figure 9-5 (Case – CCM unencrypted). The presence of a CCMP header and a MIC shall be indicated in the LLC frame header (LFH), as defined in clause 9.1.2.1.2.

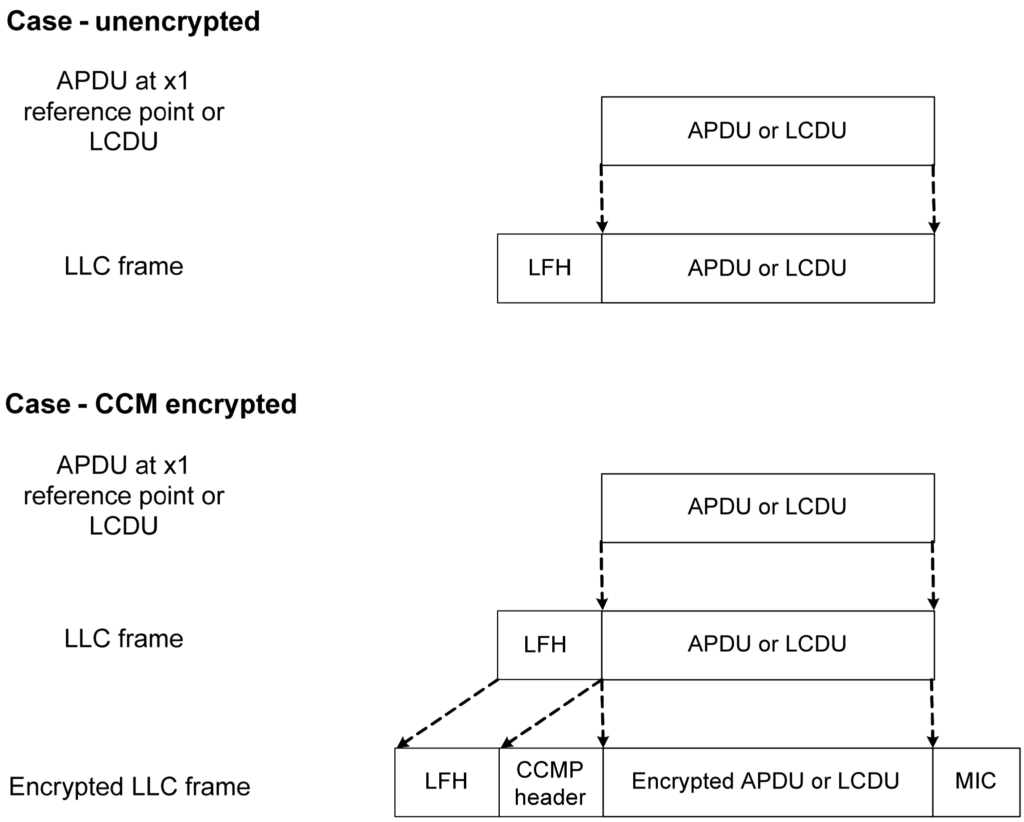


Figure 9-5 – LLC frame format (unencrypted and encrypted)

9.1.2.1.2 LLC frame header (LFH)

The LLC frame header (LFH) shall be composed of the fields described in Table 9-1. The LFH is variable in length. Octet 0 shall be passed to the MAC first.

Table 9-1 – LFH fields format

Field	Octet	Bits	Definition
LLCFT	0	[3:0]	LLC frame type
CCMPI	0	[4]	CCMP header presence indication
MHI	0	[5]	Mesh header presence indication
Reserved by ITU-T	0	[7:6]	Reserved by ITU-T
MESH	Variable	[7:0]	Mesh header (see Table 9-4)
NOTE – All bits reserved by ITU-T shall be set to the default value of zero by the transmitter and ignored by the receiver.			

9.1.2.1.2.1 LLC frame type (LLCFT)

The LLCFT field is a 4-bit field that indicates the type of APDU that makes up the LLC frame. Table 9-2 lists the valid LLC frame types:

Table 9-2 – LLC frame types

LLC frame type	Value
Reserved	0
Management frame (LCDU)	1
Data frame (APDU) – See Table 9-3	2-15

Table 9-3 – Data frame (APDU) types

APDU type	Value	Description
IPv6 APDU (non-compressed)	2	IPv6 [IETF RFC 2460]
Compressed IPv6 APDU, unicast	3	IPv6, default IPv6 header compression as per Annex C
Compressed IPv6 APDU, multicast	4	
Compressed IPv6 APDU (6LoWPAN)	5	IPv6, compressed as per clause 10 of [IETF RFC 4944], Annex C
	6	IPv6, compressed as per [IETF RFC 6282], Annex C
IPv4 APDU (non-compressed)	9	IPv4 [IETF RFC 791]
Ethernet APDU	12	Ethernet
Vendor APDU	15	L3 protocol (at the discretion of the vendor) and the APC (Note 2)
NOTE 1 – Other values are reserved by ITU-T. NOTE 2 – In the case of a vendor APDU, interoperability can be established for proprietary network solutions only.		

9.1.2.1.2.2 CCMP header presence indicator (CCMPI)

CCMPI is a 1-bit field that is used to indicate whether security is applied to the LLC frame or not. If set to one, the LLC frame shall be secured using one of the settings defined in Table 9-42 and the CCMP header shall follow the LFH and MIC shall be added. If set to zero, the LLC frame shall not be secured and shall not include the CCPM header and MIC.

9.1.2.1.2.3 Mesh header presence indicator (MHI)

The MHI is a 1-bit field that is used to indicate whether the mesh header field is present or not. The MHI shall be set to 0 if the mesh header is not present and set to 1 if it is present.

9.1.2.1.2.4 Mesh header (MESH)

The MESH field supports the L2-relaying operation (see clause 9.3.2.2) and extended addressing options.

The MESH field shall be used when at least one the following conditions apply:

1. The extended addressing mode (see clause 9.3.1) is required to identify the source node or the destination node or both.
2. The LLC frame is transmitted using L2-relaying.

In case neither of the conditions above applies, this field shall not be sent (MHI set to 0) and shall be ignored by the receiver.

The MESH field shall be formatted as defined in Table 9-4; the length of the field depends of whether the short (16-bit) or extended (64-bit) addressing mode is used and whether the source routing (SR) extension field is added to the mesh header. The used addressing mode is indicated in the ADM field. The content of the MESH field (except for the HopsLft field) shall not be changed when an LLC frame is relayed by another node.

Table 9-4 – Description of the MESH field

Field	Octet	Bits	Description
RELI	0	[0]	Indicates whether L2-relaying extension headers are present
ADM	0	[3:1]	Bit [2] indicates the SA format (V-bit); Bit [3] indicates the DA format (F-bit);
LPRI	0	[5:4]	LLC frame priority
SRI	0	[6]	Source routing extension presence indication
Reserved by ITU-T	0	[7]	Reserved by ITU-T
SA	3-4 or 3-10	[15:0] or [63:0]	Originator address (source)
DA	5-6 or 5-12 or 11-12 or 11-18	[15:0] or [63:0]	Final address (destination)
HopsLft	1	[7:0]	Maximum number of hops allowed
SN	2	[7:0]	Sequence number (flooding control)
SR	Variable	[15:0]	Source routing header
NOTE – All bits reserved by ITU-T shall be set to the default value of zero by the transmitter and ignored by the receiver.			

9.1.2.1.2.4.1 Relaying presence indicator (RELI)

This field shall be set to 1 to indicate the presence of additional L2-relaying headers, and shall be set to 0 otherwise.

If the RELI field is set to 1, the HopsLft and SN fields shall follow the DA field, and the LPRI and SRI fields shall be transmitted and shall be processed by the receiver. If, in addition, the SRI field is also set to 1, the SR header shall follow the SN field.

If the RELI field is set to 0, the LPRI and SRI shall be set to 0 by the transmitter and ignored by the receiver.

9.1.2.1.2.4.2 Addressing mode (ADM)

This field defines the addressing mode of the SA and DA as follows:

- SA: a short 16-bit format if bit [2] is set to 1 and EUI-64 format if bit [2] is set to 0;
- DA: a short 16-bit format if bit [3] is set to 1 and EUI-64 format if bit [3] is set to 0.

For intra-domain communications, for both the APDU and LCDU, bit [2] and bit [3] shall both be set to 1. When the 16-bit format for the source or destination node is not available, bit [2] or bit [3] shall be set to 0, respectively. For inter-domain communications, if allowed, bits [2] and [3] shall be both set to 0.

9.1.2.1.2.4.3 LLC frame priority (LPRI)

The LPRI field is a 2-bit field with valid values from 0 to 3 used to indicate the priority of L2-relayed LLC frames.

For LLC frames that are not L2-relayed, this field shall be set to 0 by the transmitter and ignored by the receiver.

For L2-relayed LLC frames carrying APDUs, this field shall be set to the priority assigned by the classifier (see clause 9.2.1) of the node that originated the LLC frame; otherwise, it shall be set to 0. For L2-relayed LLC frames carrying LCDUs, this field shall be set to 2.

9.1.2.1.2.4.4 Source routing header presence indicator (SRI)

This bit shall be set to 1 to indicate the presence of the SR field, and shall be set to 0 otherwise. If the RELI field is set to 0, this field shall be set to 0 by the transmitter and ignored by the receiver.

9.1.2.1.2.4.5 Originator address (SA)

Address of the originating node expressed in a 16-bit format if bit [2] of the ADM field is set to 1 or in EUI-64 format if bit [2] of the ADM field is set to 0.

9.1.2.1.2.4.6 Final destination address (DA)

Address of the final destination node expressed in a 16-bit format if bit [3] of the ADM field is set to 1 or in EUI-64 format if bit [3] of the ADM field is set to 0.

9.1.2.1.2.4.7 Number of hops left (HopsLft)

The HopsLft is an 8-bit field that shall indicate the number of times the LLC frame is allowed to be relayed expressed as an unsigned integer. If a node receives an LLC frame to be relayed with a HopsLft field not equal to zero, it shall relay the LLC frame and decrease the HopsLft by one in the relayed LLC frame. If a node receives an LLC frame with a HopsLft field equal to zero, such an LLC frame shall not be relayed.

The initial value of the HopsLft field shall be set by the node originating the frame. For the case where the source routing extension is not present (SRI = 0), the initial value of the HopsLft field shall be equal or higher than the number of times that the LLC frame is expected to be relayed before reaching its destination.

If the source routing extension is present (SRI=1), the initial value of the HopsLft field shall be set to the exact number of relays on the path from the source node to the destination node which equals the number of entries in the SR field (see clause 9.1.2.1.2.4.9).

9.1.2.1.2.4.8 Sequence number (SN)

The SN field is to assist L2 broadcast relaying of the frame inside the domain using the procedure defined in clause 9.3.2.2.2.

The field shall be formatted as an 8-bit unsigned integer that defines the sequential number of the LLC frame in the range from 0 to 255. The value shall wrap around, i.e., the SN=0 shall be set for the LLC frame which is next to one with SN = 255.

9.1.2.1.2.4.9 Source routing extension header (SR)

The SR field supports L2-relaying by using source routing (see clause 9.3.2.2.1); this field shall only be present if the SRI field is set to 1.

The SR field shall be formatted as defined in Table 9-5; the length of the field is variable.

Table 9-5 – Description of the SR field

Field	Octet	Bits	Description
RNAL_n	0-1	[15:0]	Short address of the last relay node (relay n, closest to the final destination node)
RNAL_n-1	2-3	[15:0]	Short address of the relay node preceding the last relay node (relay *n*-1)
. . .	. . .	. . .	. . .
RNAL_1	2(n-1) – (2(n-1)+1)	[15:0]	Short address of the first relay node (relay 1, closest to the source node)

The SR field shall consist of a list of short addresses of relay nodes participating in the path from the source node to the destination node, as described in Table 9-5.

The last relay node of the path (closest to the destination) shall be set as the first entry in the list and the first relay node of the path (closest to the source) shall be set as the last entry in the list.

For any given node, the next relay node address shall be indicated in the entry of the list that precedes the entry containing the address of the given node; the number of this entry equals to the value of HopsLft set by the given node.

9.1.2.1.3 LCDU frame format

The LCDU frame format, including size of the fields, shall be as presented in Figure 9-6.

	LSB MSB
≤ 1000 octets	LCDU payload
4 octets	FCS

Figure 9-6 – LCDU format

The LCDU payload shall be used for carrying management messages only, with the format defined in clause 9.5.1.

The LLC frame carrying an LCDU shall have a format as defined in clause 9.1.2.1.1 and LLCFT = 1 in the LPH. The DA and SA of the LCDU are a 16/64-bit originating node ID and a 16/64-bit final destination node ID indicated in the MESH field of the LFH, if MESH field is present, or a 16-bit SNID and DNID indicated in the MPH, if the MESH field is not present. The LCDUs communicated between nodes of the same domain shall use a 16-bit node ID both for the originating and the final addresses.

For FCS computation, the DA field followed by the SA field shall be prepended to the LCDU such that the LSB of the DA is transmitted first. The FCS shall be then computed over all fields, from the first bit of the DA field to the last bit of the LCDU payload using the standard IEEE 802.3 Ethernet 32-bit FCS computation algorithm [IEEE 802.3]. The FCS field shall not be included when MIC is used in the LLC frame encapsulating the LCDU.

NOTE – The MIC protects both the LPDU payload and its source/destination address that is a part of the "associated data" of the encrypted message, see clause 9.6.1.1.3.

An LCDU shall be transmitted starting from the first octet of the LCDU payload, LSB first.

The relaying of LLC frames carrying LPDUs shall follow the same rules as the relaying of LLC frames carrying APDUs, as described in clauses 9.1.2.3 and 9.3.2.2. If L3 relaying is set in the domain (L2 relaying is disabled), LCDUs that may require L2 relaying shall not be used and their functions, if required, shall be provided by corresponding L3 management messages or in-band management messages.

9.1.2.1.4 APDU frame format

APDU frame format depends on the used application protocol and may be different for different types of APC. APDU formats per APC are presented in Annex B.

9.1.2.2 Segmentation

9.1.2.2.1 Generation of LPDUs

The process of generating LPDUs from LLC frames is presented in Figure 9-7.

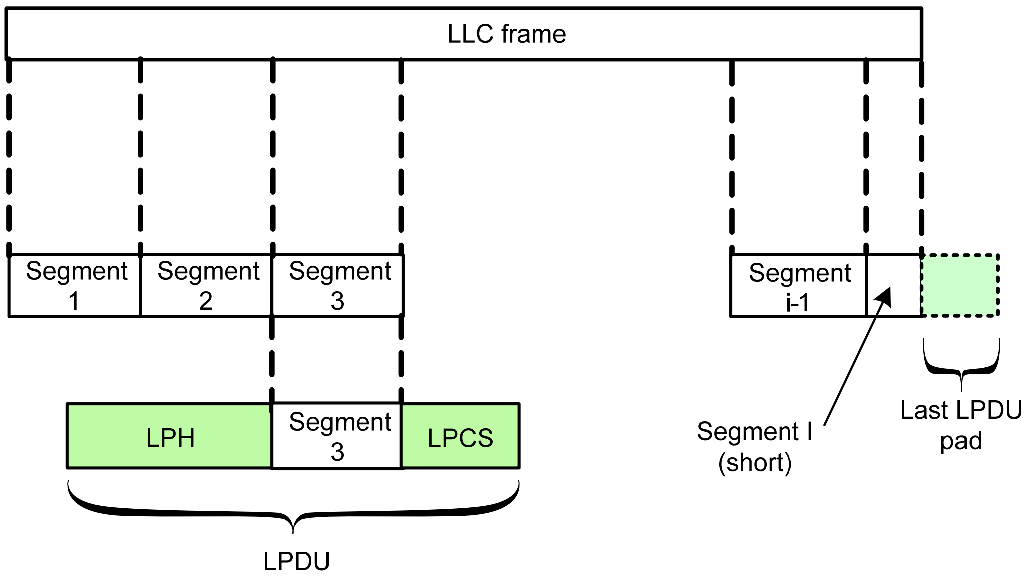


Figure 9-7 – Generation of LPDUs from LLC frames

An LLC frame shall be formed by prepending an LFH to an APDU or to an LCDU, and can either be unencrypted or encrypted, as described in clause 9.1.2.1. If the size of the LLC frame exceeds the assigned size of a single LLC segment, which is S_LLC bytes, the LLC frame shall be segmented as described in Figure 9-7:

- the first bit of the first segment shall be aligned with the first bit of the LLC frame;
- all the segments of the same LLC frame shall be of the same length, except the last one which may be shorter;
- the length of the segment shall be equal to or less than S_LLC bytes;
- the number of segments for an LLC frame shall not exceed 64.

The value of S_LLC shall be as defined in clause 9.4.

An LPDU is formed by prepending an LPDU header (LPH) to the segment and by appending an LPDU check sequence (LPCS), as shown in Figure 9-7. The LPH contains the information necessary to re-assemble the LLC frame from the received segments and the LPCS allows detection of corrupted LPDUs. The LPCS shall be calculated as described in clause 9.1.2.2.3. The last LPDU, if shorter than others, shall be padded accordingly. The content of the padding is determined by the vendor; the size of the pad is indicated in the MPH and shall not exceed 63 bytes. The format of the LPH is defined in clause 9.1.2.2.2. If the LLC frame is shorter than S_LLC bytes, this frame forms one segment and one LPDU respectively, which is also considered as the "last LPDU" carrying the segments of the LLC frame.

NOTES – The size of the segment (where NSEG ≤ S_LLC) for the given LLC frame of NLLC bytes long can be computed as:

1. Minimum overhead:

   $N_{SEG} = ceiling(N_{LLC}/NOS)$, where *NOS* is the desired number of segments.

   The last segment may be incomplete, and the corresponding LPDU will be padded by

   $P = N_{SEG} \times NOS - N_{LLC}$.

2. For multi-segment MPDU transmission, it is often beneficial if the size of the LPDU is the same or close to the size of the FEC codeword to be assigned by the PHY. This can be achieved by setting the size of the LPDU as:

   $N_{LPDU} = N_{LLC}/ [ceiling(N_{LLC}/(K-N_{LPH}-N_{LPCS}))]$,

   where K is the value of the maximum FEC codeword size, as indicated in the PFH (valid values of K are 128 and 239), N_{LPH} is the size of the LPH field, and N_{LPCS} is the size of the LPCS field.

3. No MPDU padding for MPDUs carrying a single LPDU:

   a. Compute $N_1 = ceiling(N_{LLC}/NOS)$;

   b. Compute the smallest value of a that brings $N_1 + N_{LPH} + N_{LPCS} + N_{MPH} + a$ to be a valid MPDU size (where N_{LPCS} equals the LPCS field size for MS-MPDU, and equals 0 for SS-MPDU, as defined in clause 9.1.3.1);

   c. Compute $N_{SEG} = N_1 + a$.

   d. The pad of the last LPDU, *P*, can be computed as for case 1 above.

The LPDUs of the same LLC frame shall be passed to the MAC via an x2-reference point in the numerical order of the segments they carry, starting from the LPDU carrying the first segment.

9.1.2.2.2 LPDU header (LPH) format

Table 9-6 shows the format of the LPH. Octet 0 shall be passed to the MAC first.

Table 9-6 – LPDU header format

Field	Octet	# of Bits	Definition
SSN	0	6	Segment sequence number
Reserved	0	2	Reserved by ITU-T (Note)
NOTE – Bits reserved by ITU-T shall be set to 0 by the transmitter, and shall be ignored by the receiver.			

9.1.2.2.2.1 Segment sequence number (SSN)

This 6-bit field identifies the relative order of the segment within the stream of segments corresponding to the transmitted LLC frame. The value shall be formatted as unsigned integers in the range of 0 to 63.

The SSN shall be initialized to 0 for the first segment of the LLC frame and shall be increased by 1 for each subsequent segment that is associated with this LLC frame.

9.1.2.2.3 LPDU check sequence (LPCS)

The LPCS is a 32-bit cyclic redundancy check (CRC) and shall be computed over all the fields of the LPDU in the order that they are transmitted, starting with the LSB of the SSN field of the LPDU header (clause 9.1.2.2.21) and ending with the MSB of the last octet of the LPDU segment.

The LPCS shall be computed using the following generator polynomial of degree 32:

$$G(x) = x^{32}+x^{28}+x^{27}+ x^{26}+x^{25}+x^{23}+x^{22}+x^{20}+ x^{19} +x^{18}+x^{14}+ x^{13}+x^{11}+ x^{10}+ x^{9}+ x^{8}+ x^{6}+1$$

1. The n bits of the LPDU subject to LPCS are considered to be the coefficients of a polynomial of degree n-1, such that the LSB of the first octet of the LPDU header is the coefficient of the x^{n-1} term, and the MSB of the last octet of the LPDU segment is the coefficient of the x^0 term. This polynomial is referred to as $N(x)$.
2. Replace the 16 highest-order coefficients of $N(x)$ with their ones'-complement values.
3. Multiply the result of step 2 by x^{32}. This result is referred to as $N_1(x)$.
4. Compute the LPCS as the ones-complement of the remainder of $N_1(x)$ divided by $G(x)$.

The bits of the LPCS shall be transmitted in sequential order, starting from the coefficient of the highest order term (x^{31}), referred as the MSB, and continuing to the x^0 term.

9.1.2.3 Retransmission of LPDUs

LPDUs assigned for retransmission (see clause 9.1.2) shall be either assembled into an outgoing MPDU carrying other LPDUs associated with the same LLC frame (to be transmitted for the first time or retransmitted) or sent in a separate MPDU. This separate MPDU shall be assigned MA priority as defined in clause 9.2.1.

LPDUs shall be retransmitted with no changes in their segments and in their LPH.

9.1.2.4 Transmission of LCC frames

The LLC frames shall be assembled and further processed in the order that the APDUs or LCDUs carried by these LLC frames arrive, except for the APDUs of priority 3. If an APDU and an LCDU arrive simultaneously, the one that has higher priority shall be assembled and processed first and, in the case of their having the same priority, the LCDU shall be assembled and processed first.

In the case where an incoming APDU is of priority 3, even if it arrives during the processing of another LLC frame of lower priority, it shall be processed first, i.e., the LPDUs sourced from an LLC frame carrying an APDU of priority 3 shall be submitted to an x2-reference point which has precedence over the remaining LPDUs comprising the LLC frame in transmission, see clause 9.3.3.3.1. With the exception of an LLC frame carrying APDUs of priority 3, no LPDU of an LLC frame is allowed to be transmitted until all LPDUs of previously transmitted LLC frames are cleared (i.e., transmitted entirely, if no acknowledgement is required, or acknowledged as received error free or timed out, if acknowledgement is required).

The LLC frames transmitted by any particular device shall be enumerated, thus allowing the receiver to distinguish between the different LLC frames in transmission; the sequence number of the LLC frame is indicated in the LFSN field of the MPH (see clause 9.1.3.1.1.5). In case the transmission of an LLC frame is interrupted by a frame of priority 3, the receiver detects the interruption by the value of the priority and the new sequence number; also the receiver may resume reception of interrupted frames as described in clause 9.3.3.3.2.

A node shall not relay an LLC frame if not all of its comprising segments are received correctly; relaying segments that belong to an incompletely received LLC frame is not allowed. The received LLC frame shall be relayed using the usual LLC frame transmission path, as presented in Figure 9-4.

The next hop destination address (DDID, DNID) of the relayed frame shall be determined as defined in clause 9.3.2.2.

The priority of a relayed LLC frame shall be as indicated in the LPRI field of the LFH. The corresponding MA priority of the MPDUs carrying a relayed frame shall be assigned based on the priority of the relayed LLC frame by using the rules defined in clause 9.4.3.2. The LLC frames with the same value of LPRI shall be relayed in the order that they have been received.

The content of a relayed LLC frame shall not be modified, except the HopsLft field of the LFH, which has to be decreased by 1 after each hop has passed, as defined in clause 9.1.2.1.2. The LLC frames to be relayed shall not be decrypted.

The relayed LLC frame shall use the same settings for acknowledgement (parameters ACK_REQ in the PFH and ACK_TYPE in the MPH, see clause 9.3.3.1) as used in the received LLC frame.

9.1.3 Medium access control sublayer (MAC)

The functional model of the MAC is presented in Figure 9-8. It is intended to describe in more detail the MAC functional block presented in Figure 9-1.

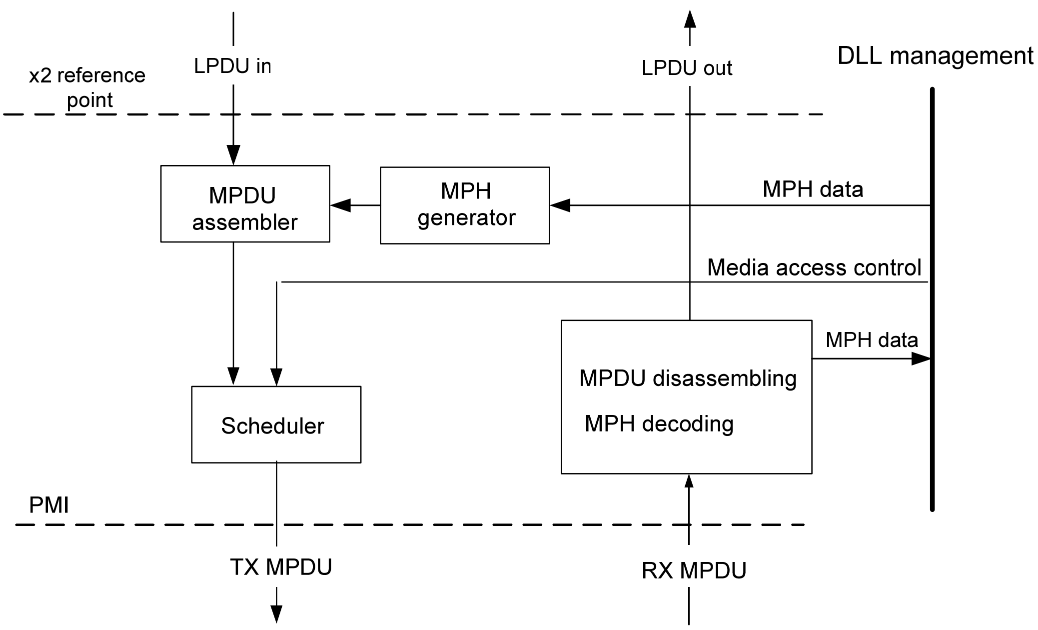


Figure 9-8 – Functional model of MAC

In the transmit direction, MPDUs are assembled from LPDUs that are passed over the x2 reference point and the MPH that is generated by the MAC. The MPDU assembler also adds the pad to meet the closest valid size of MPDU, as defined in clause 8.2.3.2.1.

The assembled MPDUs are scheduled for transmission according to the medium access procedure described in clause 9.2, using one or more established priorities queues. The medium access control primitives include the carrier sense (CRS) that indicates whether a medium is busy or not. After being scheduled for transmission, the MPDU is passed to the PHY via the PMI (see PMI_DATA.REQ primitives definition in clause 8.8).

The MAC is also responsible for scheduling an Imm-ACK frame transmission by the PHY if acknowledgement is required. The data of the Imm-ACK frame (see clause 9.3.3.1.1) is passed to the PHY via the PMI interface (see PMI_MGMT.REQ primitive).

In the receive direction, the MPDU enters via the PMI (PMI_DATA.IND primitive). The MPH is decoded, the MPDU is disassembled, the relevant MPDU parameters communicated in the MPH are passed to the DLL management via the MAC_MGMT reference point, and the recovered LPDUs are passed to the LLC via the x2 reference point. The received acknowledgement data is passed via the PMI interface (see PMI_MGMT.IND primitive). Duplicates of the received MPDUs shall be dropped based on the LFSN number and the sequential order of the MPDU in the LLC frame.

9.1.3.1 Assembling of MPDUs

The LPDUs shall be assembled into an MPDU in the order they cross the x2 reference point. The LLC shall support the following MPDU assembling modes:

- Single segment MPDU mode (SS-MPDU)
- Multi segment MPDU mode (MS-MPDU)
- The MPDU assembling mode shall be identified by the MPDU_TYPE field of the MPH (see clause 9.1.3.1.1.1).

  NOTE – The term "MPDU" is general throughout the Recommendation and applies to both SS-MPDU and MS-MPDU.

The generation of LPDUs process is described in clause 9.1.2.2.1. The assembling process is described in Figure 9-9.

An SS-MPDU shall include a single LPDU (excluding its LPCS field). An MS-MPDU shall include at least one LPDU, but no more than 16 LPDUs, starting with an LPDU that carries any segment of the LLC frame; this LPDU is numbered as LPDU#1 in Figure 9-9. All LPDUs in the MS-MPDU shall be sourced from the same LLC frame. The order of LPDUs in the MS-MPDU is determined by the vendor.

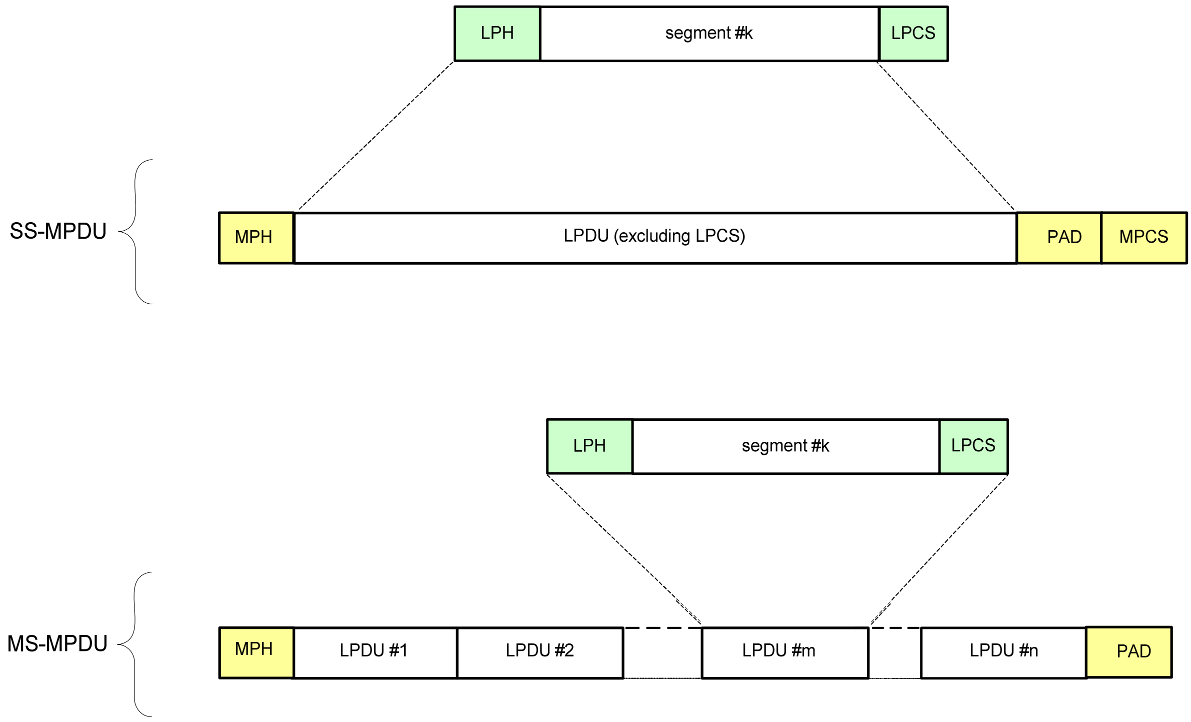


Figure 9-9 – Assembling of an SS-MPDU (top) and an MS-MPDU (bottom) from LPDUs

The size of the LPDU and the number of LPDUs in MS-MPDU shall be such that the time of MPDU transmission (time-on-the-wire) does not exceed the maximum allowed time on the medium, MaxMediumTime (see clause 9.1.4).

The position of the segment carried by the LPDU in the LLC frame is indicated in the LPH. The MPH shall be prepended to the first LPDU; MPH bit 0 of octet 0 shall be transmitted first. Octet 0 of the LPH of LPDU#1 of the MPDU shall be passed to the PHY first (see Figure 9-9). The MPH is defined in clause 9.1.3.1.1.

The pad field shall be appended to the last LPDU; the content of the pad is determined by the vendor; the valid size of the pad is up to 15 bytes; the receiver derives the size of the pad using the size of the LPDU indicated in the MPH.

The MPCS shall be attached after the PAD field in SS-MPDU and shall be calculated as specified in clause 9.1.3.1.4.

9.1.3.1.1 MPH format

The format of the MPH shall be as defined in Table 9-7. Octet 0 shall be passed to the PHY first.

Table 9-7 – MPH format

Field	Octet	Bits	Definition
MPDU type	0	[2:0]	000 – SS-MPDU 001 – MS-MPDU 010-111 – Reserved by ITU-T
ACK_TYPE	0	[3]	Shall be set to 0 if ACK per frame is required and set to 1 if ACK per LPDU is required (see clause 9.3.3.1)
NOS	0-1	[9:4]	Number of segments comprising the LLC frame
LSPL	1	[15:10]	Last segment pad length, in bytes
LFSN	2	[7:0]	LLC frame sequence number
SNID	3-4	[15:0]	A 16-bit NODE_ID of the node that transmits the frame
DNID	5-6	[15:0]	A 16-bit NODE_ID of the node that shall receive the frame
SDID	7-8	[15:0]	A 16-bit DOMAIN_ID of the node that transmits the frame
DDID	9-10	[15:0]	A 16-bit DOMAIN_ID of the node that shall receive the frame
NEST	11	[7:0]	Indicates the value of $N_{est}(x)$ used for transmission, see clause 9.2.2.3
LPDU-L	12-13	[10:0]	Length of the LPDU
TP-REQ	13	[12:11]	Request for transmission parameters recommendation
PRI	13	[14:13]	MA priority of the MPDU
Reserved	13	[15]	Reserved by ITU-T
Reserved	14	[7:0]	Reserved by ITU-T
MHCS	15	[7:0]	HCS (CRC-8)
NOTE – Bits reserved by ITU-T shall be set to 0 by the transmitter, and shall be ignored by the receiver.			

9.1.3.1.1.1 MPDU type

This 3-bit field identifies the type of the MPDU. The value shall be formatted as an unsigned integer. When set to 000, SS-MPDU mode is used, when set to 001, MS-MPDU mode is used. Other values are reserved by ITU-T.

9.1.3.1.1.2 ACK_TYPE

This 1-bit field identifies the type of acknowledgement as defined in clause 9.3.3.1.

9.1.3.1.1.3 Number of segments (NOS)

This 6-bit field indicates the total number of segments comprising the LLC frame minus 1, represented as an unsigned integer in the range from 1 to 64. The value of NOS shall be computed as:

NOS = *ceiling*(N_{LLC} / N_{SEG}), where N_{LLC}, N_{SEG} are defined in clause 9.1.2.2.1.

9.1.3.1.1.4 Last segment pad length (LSPL)

This 6-bit field indicates the length of the pad, in bytes, used in the last segment of the LLC frame, represented as an unsigned integer in the range from 0 (no padding) to 63.

9.1.3.1.1.5 LLC frame sequence number (LFSN)

This 8-bit field identifies to which LLC frame LPDUs comprising the MPDU belong. The value shall be formatted as an unsigned integer in the range from 0 to 255. The value of the LFSN shall be increased by 1 (with wraparound) for each subsequent transmitted LLC frame (either originated by the node or relayed).

9.1.3.1.1.6 SNID, DNID

A 16-bit value of the source node NODE_ID and destination node NODE_ID represented as an unsigned integer with constraints determined by the ITU-T G.9902 addressing scheme defined in clause 9.3.1.

9.1.3.1.1.7 SDID, DDID

A 16-bit value of the source node DOMAIN_ID and destination node DOMAIN_ID represented as an unsigned integer with constraints determined by the ITU-T G.9902 addressing scheme defined in clause 9.3.1.

9.1.3.1.1.8 NEST

The NEST field indicates the value of $N_{est}(x)$ that the transmitter has calculated as defined in clause 9.2.2.3 divided by 4 and rounded to the closest integer.

9.1.3.1.1.9 Length of the LPDUs (LPDU_L)

This 11-bit field indicates the length of the LPDUs carried by the MPDU in bytes, expressed as an unsigned integer. The valid range is from 5 to 1680 bytes. The length of the LPDU shall always exceed the size of the pad (if non-zero) applied at the end of the MPDU (see Figure 9-9).

9.1.3.1.1.10 Transmission parameters request (TP-REQ)

This 2-bit field indicates that the receiver of the frame shall provide to the requesting node a TP report with the content depending on the content of the TP-REQ field as follows:

– 00 – no TP report required;
– 01 – a full TP report using TP.ind message (see clause 9.5.4) is requested;
– 10 – a full TP report using the extended Imm-ACK (see clause 9.3.3.1) is requested;
– 11 – a partial TP report is requested.

This field shall be set to 00 for all broadcast and multicast transmissions (if DNID = 0xFFFF).

The content of the full TP report and partial TP report is defined in clause 9.5.4 and in clause 9.3.3.1.1, respectively. If the transmitter requesting a TP report (full or partial) has not received it or received a report that is incomplete, it may repeat the same type of request in the following frame.

NOTE – A receiver that gets a TP request of a particular type but that is not capable of providing a response in the Imm-ACK frame associated with the frame requesting the TP report, is expected to be ready for the same report in the Imm-ACK associated with the next frame from the same source.

9.1.3.1.1.11 Priority (PRI)

This 2-bit field indicates the MA priority used for the current transmission, represented as an unsigned integer.

9.1.3.1.1.12 MPH check sequence (MHCS)

The MHCS is an 8-bit cyclic redundancy check (CRC) and shall be computed over all the fields of the MPH in the order they are transmitted, starting with the LSB of the MPH byte 0 (clause 9.1.3.1.1) and ending with the MSB of the last byte of the MPH.

The MHCS shall be computed using the following generator polynomial of degree 8:

$$G(x) = x^8+x^5+x^4+ x^3+x^2+x^1+1$$

The MHCS shall be constructed as follows:

1. The n bits of the MPH subject to the MHCS are considered to be the coefficients of a polynomial of degree n-1, such that the LSB of the MPH byte 0 is the coefficient of the x^{n-1} term, and the MSB of the MPH byte 12 is the coefficient of the x^0 term. This polynomial is referred to as $N(x)$.
2. Replace the 8 highest-order coefficients of $N(x)$ with their ones'-complement values.
3. Multiply the result of step 2 by x^8. This result is referred to as $N_1(x)$.
4. Compute the MHCS as the ones-complement of the remainder of $N_1(x)$ divided by G(x).

The bits of the MHCS shall be transmitted in sequential order, starting from the coefficient of the highest order term (x^7), referred as the MSB, and continuing to the x^0 term.

9.1.3.1.2 MPDU check sequence (MPCS)

The MPCS is a 32-bit cyclic redundancy check (CRC) and shall be computed over all the fields of the SS-MPDU in the order they are transmitted, starting with the LSB of the MPH and ending with the MSB of the last octet of the PAD field.

The MPCS shall be computed using the following generator polynomial of degree 32:

$$G(x) = x^{32}+x^{28}+x^{27}+ x^{26}+x^{25}+x^{23}+x^{22}+x^{20}+ x^{19} +x^{18}+x^{14}+ x^{13}+x^{11}+ x^{10}+ x^9+ x^8+ x^6+1$$

The MPCS shall be constructed as follows:

1. The n bits of the SS-MPDU subject to the MPCS are considered to be the coefficients of a polynomial of degree n-1, such that the LSB of the first octet of the MPH is the coefficient of the x^{n-1} term, and the MSB of the last octet of the PAD field is the coefficient of the x^0 term. This polynomial is referred to as $N(x)$.
2. Replace the 16 highest-order coefficients of $N(x)$ with their ones'-complement values.
3. Multiply the result of step 2 by x^{32}. This result is referred to as $N_1(x)$.
4. Compute the MPCS as the ones-complement of the remainder of $N_1(x)$ divided by $G(x)$.

The bits of the MPCS shall be transmitted in sequential order, starting from the coefficient of the highest order term (x^{31}), referred as the MSB, and continuing to the x^0 term.

9.1.3.2 Scheduler

The scheduler assigns MPDUs for transmission over the medium; it determines which MPDU to be passed for transmission to the PHY via the PMI and the time this MPDU shall be passed to the PHY based on the medium access rules defined in clause 9.2. The scheduler also maintains MA priority queues necessary to support the medium's access. MPDUs are assigned to the particular MA priority queue based on their priority and some additional criteria, as defined in clause 9.2.1.

9.1.4 DLL parameters

Bandplan dependent parameters are specified in Table 9-8.

Table 9-8 – DLL bandplan dependent parameters

Parameter	CENELEC	FCC
T_{IFG_MIN}	2.1 ms	2.7 ms
T_{AIFG}	2.1-2.6 ms	2.7-3.2 ms

DLL parameters independent of the band plan are specified in Table 9-9.

Table 9-9 – DLL parameters independent of the band plan

Parameter	Value
TCP-MAX	250 ms
TTX_ON	10 μs
TTS	Same duration as 2 preamble symbols
MaxMediumTime	250 ms
MaxRetransmissions	Depends on the number of LPDUs (NLPDU) comprising the LLC frame as specified in Table 9-10
MaxAckRxTime	The value is determined by the A_MGMT.REQ management primitive, see clause 9.4.3.4.1
T_{CBIFG}	250 ms

Table 9-10 – Value of MaxRetransmissions for different numbers of LPDUs comprising an LLC frame

Number of LPDUs in an LLC frame	Number of retransmissions per LLC frame
Up to 5	$N_{LPDU} \times 3$
6-10	$15 + N_{LPDU} \times 2$
11-20	$25 + N_{LPDU} - 10$
21-31	$35 + \text{floor}[\ (N_{LPDU} - 20) \times 1/2\ ]$
32-46	$41 + \text{floor}[\ (N_{LPDU} - 32) \times 1/3\ ]$
47-64	$46+\text{floor}[\ (N_{LPDU} - 47) \times 1/4\ ]$

9.2 Medium access procedures

An ITU-T G.9902 node of a standard profile shall support prioritized contention-based medium access with four MA priorities denoted from 0 (lowest priority) to 3 (highest priority); nodes of low-complexity profile shall support two MA priorities, 0 and 2. The number of MA priorities supported by other profiles is for further study.

The prioritized contention-based medium access rules are defined in the following sub-sections. Other types of medium access are for further study.

9.2.1 Assignment of MA priorities

The classifier, in APC, assigns a certain priority to each APDU based on the QoS-related information carried by the primitives of the corresponding ADP or based on the relevant management primitives (see Annex I). This assigned APDU priority determines the priority of the LLC frame carrying the APDU and can be 0, 1 and 2. The priority value of 3 is assigned for pre-defined emergency data transfers only. All LLC frames carrying LCDUs (internal management communications) shall be assigned to priority 2.

The assignment of APDU priority depends on the application protocol but shall not exceed the number of MA priority queues supported by the node, which is 4 for a standard profile and may be less for other profiles for other profiles (see clause 7).

Nodes of the standard profile shall support:

– three MA priority queues for application data (priority 0, 1, and 2) and management data (priority 2);
– an MA priority queue for emergency signals (priority 3), that shall also be used for asynchronous beacons.

Nodes of other profiles shall support at least priority 0 for application data and priority 2 for management data (see clause 7); other supported priorities are for further study.

The MA priority of an MPDU shall be initially assigned the same priority as the LLC frame whose segment(s) it carries. If segments of an LLC frame are carried by multiple MPDUs, after transmission of the first MPDU carrying the LLC frame segments, the MA priority of all the remaining MPDUs carrying segments of the same LLC frame (including those carrying LPDUs assigned for retransmission) may be elevated from its initial value up to 2. All MPDUs sourced from an APDU with a priority that is equal to or greater than 2 shall use the initial priority.

9.2.2 Prioritized contention-based medium access

The prioritized contention-based medium access is defined in terms of contention periods (CP). The contention process starts at the beginning of the CP. The CP ends T_{IFG_MIN} after a node that won the contention completes the transmission of the frame sequence, which includes the transmitted frame and the ACK frame, if required. A new CP starts immediately after the end of the previous CP.

The CP shall consist of 4 priority resolution periods that may overlap. Each such period is associated with a contention window (CW). The CW consists of a variable number of fixed duration time slots (TS). The number of slots assigned to each CW shall vary; the minimum number of TS is 1. The size of the CW for each priority shall be set as specified in clause 9.2.2.3.

The CP shall start with the CW associated with the highest priority. A CW associated with a certain priority shall not start prior to a CW associated with a higher priority. The setting of a CW size associated with a given priority shall be as specified in clause 9.2.2.3.

An MPDU assigned with a specific MA priority is allowed to be transmitted during the CW associated with the same or lower priority according to the back-off procedure specified in clause 9.2.2.2.

The medium access process includes the following steps:

1. Nodes shall contend for the transmission of an MPDU based on the MA priority of the MPDU assigned by the scheduler.
2. Nodes shall contend for the transmission of an MPDU during the CW that is associated with the MA priority assigned to this MPDU or during a CW associated with a lower MA priority.
3. Nodes that won the contention may transmit their MPDU.
4. Nodes that lost the contention shall use the back-off rules specified in clause 9.2.2.2.
5. The next CP shall start T_{IFG_MIN} period after the transmission of the frame sequence is complete, i.e., after the ACK frame is sent (in the case where ACK was requested by the transmitter) or after the frame was transmitted (in the case where no ACK was requested by the transmitter).

The procedure of transmission during the CP is illustrated in Figure 9-10. The figure presents the positions and sizes of CWs for a particular CP in which a node that won the contention for CW_0 sends a low priority MPDU followed by an Imm-ACK frame.

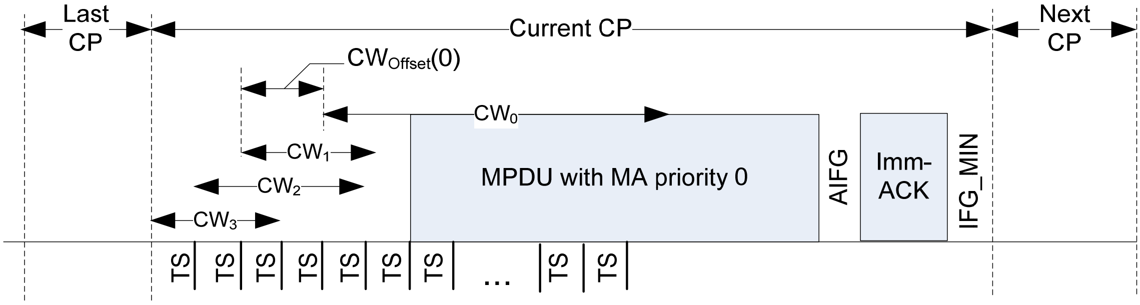


Figure 9-10 – Illustration of a typical transmission during the CP

9.2.2.1 CW offset setting

The CW offset is defined as a number of TS between the start of the CW and the start of the CW associated with the next higher priority. The values of CW offsets associated with a specific priority (relative to the CW associated with the next higher priority) is determined by the domain master and communicated to the nodes of the domain via the beacon (see clause 9.4.2.1) and set using the MAC_MGMT.REQ primitive CW_{Offset} (see clause 9.4.3.3.1). The default values of CW_{Offset} are presented in Table 9-11. The CW_{Offset} for priority 0 is shown in Figure 9-10.

9.2.2.2 CP back-off procedure

To support the back-off procedure described in this section, each node shall maintain at least the following back-off parameters:

- a back-off counter (BC) per transmission attempt
- a CW size for each MA priority.

The BC determines the number of TS the node shall wait for before it may begin the transmission if the medium is idle. The duration of the TS is denoted as TTS and shall be as specified in clause 9.1.4.

The CW size determines the range from which the back-off value shall be picked before transmission; this is the function of the MA priority and further denoted as CW(x), where x stands for the MA priority. The size of CW(x) is expressed in the number of TS.

A node that survives the priority resolution period preceding the start of the CW associated with the MA priority of the MPDU it intends to transmit (see clause 9.2.1), shall compete in the CW using its back-off parameters for that MA priority and shall act according to the following procedure before starting a transmission in a CP:

1. Pick a random value of the BC in the range 0 to CW(x)-1.
2. If the chosen value of BC is zero, the node shall start transmitting its frame within a time window of T_{TX_ON} (see clause 9.1.4) after the start of the first TS of the CW(x).
3. If the chosen value of BC is not zero, the node shall decrease its BC upon completion of each TS in which it detects no PHY preamble (the medium is idle).
4. If upon completion of a certain TS, the value of BC is zero, the node shall start transmitting its frame within a time window of T_{TX_ON} (see clause 9.1.4) after the end of this TS. The BC shall be disabled till the next CP.
5. If a node detects a PHY preamble during a TS and its BC at this TS is not zero, it shall not transmit in this CP and shall defer transmission to the next CP.
6. If a node gets an MPDU to transmit and no previous transmission was detected for more than $T_{CP\ MAX}$ (see clause 9.1.4) from the start of the CP, the node may transmit the MPDU immediately.

7. In the next CP, nodes shall repeat steps 1-6 above.

If a node that contends in a CP gets an MPDU to transmit after the start of a particular CW, it may still contend for this MPDU in this CW using the back-off procedure defined in this section, but only if the MA priority of the MPDU is equal to or higher than the priority associated with the CW. The node shall pick a random value of the BC in the CW in the same way as nodes that had the MPDU ready to transmit prior to the start of the CW. Alternatively, the node may defer transmission to the next CP.

9.2.2.3 CW update procedure

The update procedure shall be performed by a node in the following cases:

– after every transmitted frame, for the MA priority of this frame;

– after every received frame with no error in the MPH, for the MA priority of this received frame (indicated in the MPH);

– after every $T_{CWUpdate}$ time period in which a node does not transmit and does not detect transmissions (i.e., no preamble was detected) from other nodes, for all those MA priorities, *x*, for which the CW(*x*) is greater than $CW_{InActMin}(x)$ associated with the MA priority *x*; the value of the parameter $CW_{InActMin}(x)$ is expressed in TS and its default value is indicated in Table 9-9. If $T_{CWUpdate}$ is set to 2550 ms the update procedure due to the $T_{CWUpdate}$ time period shall not be performed.

Upon initialization, the node shall set its CW value for each priority to an initial value, $CW_{Init}(x)$, expressed in TS.

The CW update procedure is iterative. The parameters of the update procedure for a CW of the *n*-th iteration, $CW(x,n)$,associated with the MA priority *x* are $CW_{tx}(x,n)$ and $CW_{min\_tx}(x,n)$ that represent the actual value of the CW and the value of CW_{min} expressed in TS used in the last transmitted frame before the CW update, where CW_{min} is the number of TS from the beginning of the CW to the instant of actual frame transmission.

For each transmission, the transmitter shall calculate the value of $N_{est}(x,n)$ as follows:

$$N_{est}(x) = \frac{CW_{tx}(x) - CW_{min\_tx}(x)}{CW_{min\_tx}(x)}$$

All nodes shall update the CW used for the last transmission of an MA priority, *x*, by using the following procedure:

1. Calculate parameter $\upsilon_{est}(x,n)$ at the *n*-th iteration as follows:

   - $\upsilon_{est}(x,n)=1$ if the value of the NEST field on the MPH (in the transmitted or in the received frame respectively) is set to zero and $\upsilon_{est}(x,n)=1/(4\times\text{NEST})$ if the value of NEST field is non-zero;
   - $\upsilon_{est}(x,n)=1$ after every $T_{CWUpdate}$ time period in which a node does not transmit and does not detect transmissions.

2. Calculate parameter $\alpha(n)$ at the *n*-th iteration as follows:

$$\alpha = \begin{pmatrix} \dfrac{\gamma\times\delta}{1+\gamma\times\delta}, & \text{if } \gamma < \gamma_{max} \\ 1, & \text{if } \gamma = \gamma_{max} \end{pmatrix}$$

where $\delta = \upsilon_{est}(x,n)/\upsilon_{avg}(x,n-1)$, $\upsilon_{avg}(x,n-1)$ is the weighted average value of $\upsilon_{est}(x)$ over the past n-1 iterations, and $\upsilon_{avg}(x,0) = \frac{A}{CW_{Init}(x) - B}$

3. Calculate $\upsilon_{avg}(x,n)$ at the n-th iteration as follows:

$$\upsilon_{avg}(x) = \alpha \times \upsilon_{avg}(x) + (1-\alpha) \times \upsilon_{est}(x)$$

4. Update $CW(x)$ as follows:

$$CW(x) = \frac{A}{\upsilon_{avg}(x)} + B$$

The default values of the back-off update parameters shall be as specified in Table 9-11:

Table 9-11 – ITU_T G.9902 default back-off parameters

Parameter	Value
CW_{Init} (0)	160
CW_{Init} (1)	160
CW_{Init} (2)	8
CW_{Init} (3)	8
γ	9
A	2.7
B	0
$CW_{InActMin}$ (0)	8
$CW_{InActMin}$ (1)	8
$CW_{InActMin}$ (2)	160
$CW_{InActMin}$ (3)	160
TCW_{Update}	10 ms
CW_{offset} (0)	3 TS
CW_{offset} (1)	2 TS
CW_{offset} (2)	2 TS

9.2.2.4 Adjustment of back-off parameters

Back-off parameters can be adjusted from their default values presented in Table 9-11 by the domain master or special nodes assigned by the domain master (proxies). The domain master (or its selected proxies) may assert the values of default back-off parameters, only upon request from the upper layer management indicated by the A_MGMT.REQ primitive MAMode (see clause 9.4.3.4.1). The asserted values of the default back-off parameters are communicated to all nodes of the domain using special management messages. The details are for further study.

9.2.2.5 The starting and closing of a CP

Nodes shall start a new CP upon closure of the current CP, as defined in clause 9.2.1 or after a certain period of inactivity on the line. Nodes shall infer closure of a CP that was used for transmission according to one of the following methods:

– duration-based

– ACK reception-based

– timeout-based.

The starting of a CP after inferring persistent line inactivity is defined in clause 9.2.2.5.4.

9.2.2.5.1 Duration-based CP closure

Nodes that did not contend or backed off in a CP shall detect the frame transmitted during the CP. A node that received the preamble and the PFH of a transmitted frame without an HCS error (see clause 8.2.3), shall close the CP that was used for frame transmission T_{IFG_MIN} after the frame sequence ends, based on the duration that can be inferred from the related PFH fields of the transmitted frame in the frame sequence. If a request for ACK was indicated in the PFH, T_{IFG_MIN} shall be counted from the end of the ACK message transmission (assuming maximum value of T_{AIFG}); if no request for ACK was indicated in the PFH, T_{IFG_MIN} shall be counted from the end of the transmitted message.

The values of T_{IFG_MIN} and the valid minimum and maximum values for T_{AIFG} are defined in clause 9.1.4, the duration of the ACK shall be computed based on the Imm-ACK frame format, as defined in clause 9.3.3.

Figure 9-11 describes a duration-based CP closure (with and without ACK). The CP closes after the entire frame sequence duration.

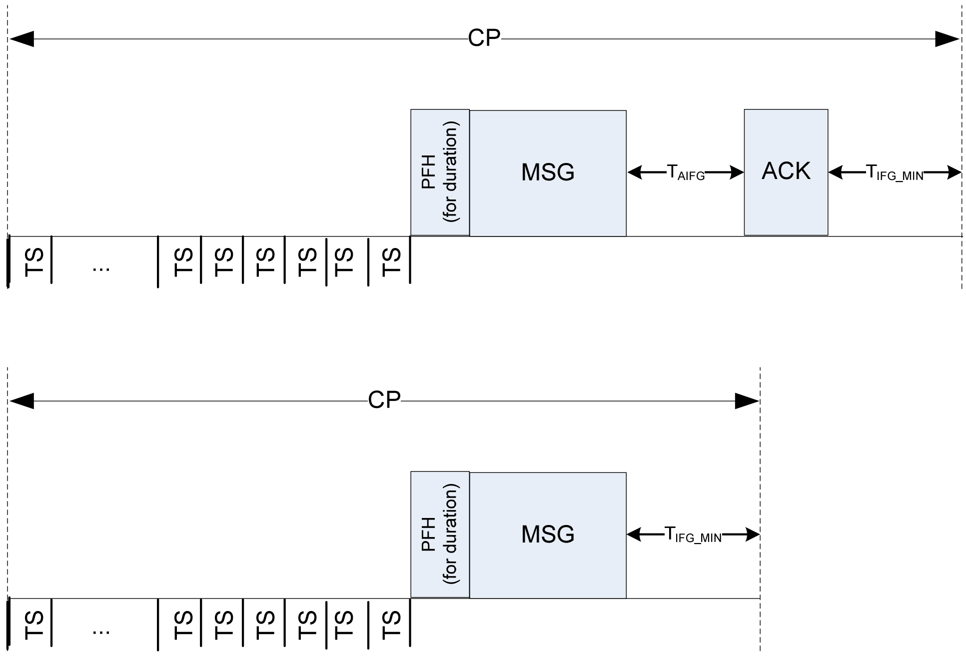


Figure 9-11 – Example of a duration-based CP closure (with and without ACK)

9.2.2.5.2 Imm-ACK reception-based CP closure

Nodes that did not contend or backed off in a CP, and received an Imm-ACK frame, shall close a CP that was used for transmission at T_{IFG_MIN} + max(T_{AIFG}) – min(T_{AIFG}) after the end of the Imm-ACK frame. If the received Imm-ACK frame was in response to a frame that originally inferred this receiver to close the CP based on duration, the receiver shall close the CP using the duration based CP.

Figure 9-12 describes an ACK reception-based CP closure.

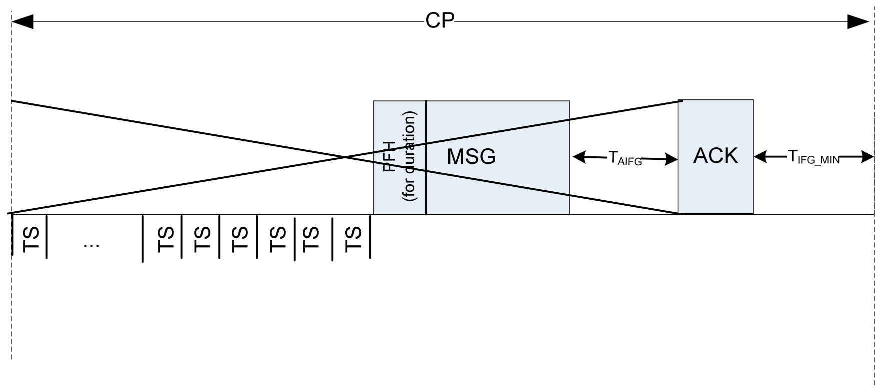


Figure 9-12 – Example of an Imm-ACK reception-based CP closure

A node that transmitted a frame containing an Imm-ACK request, but did not receive an Imm-ACK during max(T_{AIFG}) period after the end of the frame, shall assume a collision of the transmitted frame and shall start a waiting period T_{CBIFG} counted from the end of the last symbol of the PFH of their transmitted frame. If during the T_{CBIFG} period no preamble of any subsequent frame was detected, the node shall consider the CP closed after T_{CBIFG}. The value of T_{CBIFG} is defined in clause 9.1.4.

Figure 9-13 illustrates this scenario.

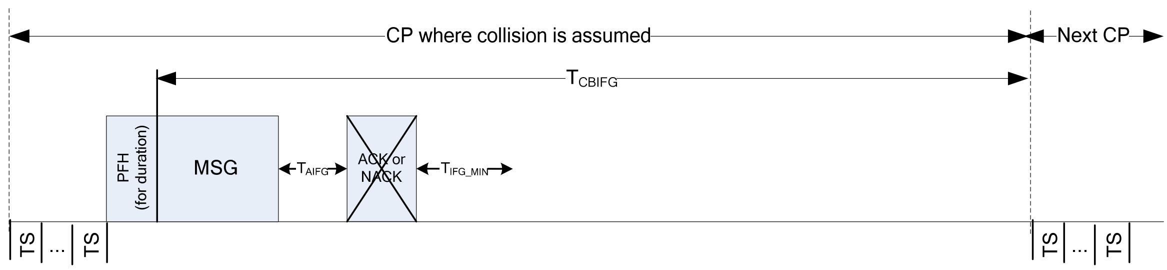


Figure 9-13 – Example of a CP closure in case no Imm-ACK is received

If during the T_{CBIFG} period a preamble of any subsequent frame was detected, the node shall re-set the CP closure based on that received frame, using the methods described in clause 9.2.2.5.1, clause 9.2.2.5.2 or clause 9.2.2.5.3, depending on reception conditions.

9.2.2.5.3 Timeout-based closure

Nodes that did not contend or backed off in a CP shall attempt to receive the frame transmitted during the CP. A node that received a frame with the HCS of the PFH detected with error, and no Imm-ACK frame or preamble or PFH of any subsequent frame or any other frame related to the CP was detected for at least T_{CBIFG} from the end of the last symbol of the PFH, shall consider the CP closed after T_{CBIFG} (clause 9.1.4).

Figure 9-14 illustrates this scenario.

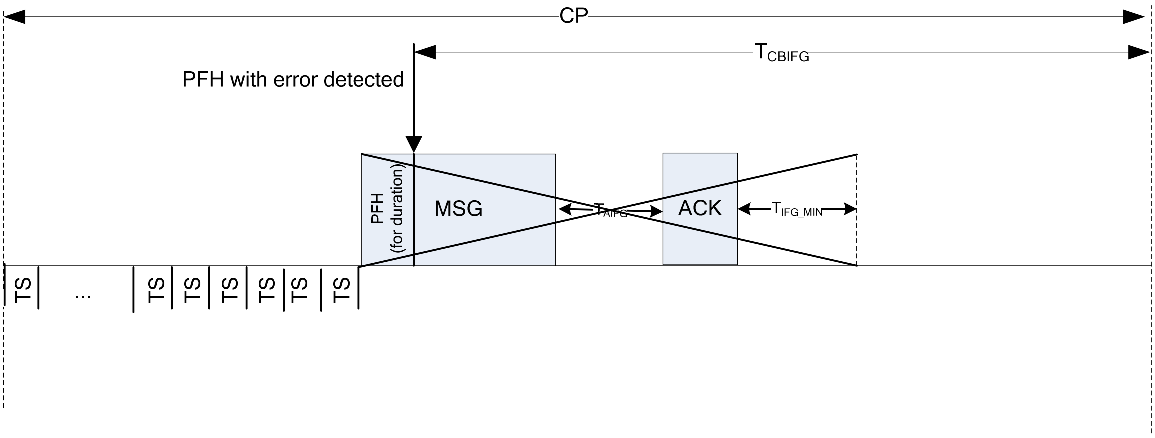


Figure 9-14 – Example of a timeout-based CP closure

9.2.2.5.4 The starting of a CP upon persistent line inactivity

If a node has a new frame ready for transmission after no activity on the line was detected for longer than TCP-MAX period, the node shall consider the previous CP as closed and thus it may immediately transmit the frame. Other nodes shall close the CP based on the reception conditions of this transmitted frame, as described in clauses 9.2.2.5.1, 9.2.2.5.2 and 9.2.2.5.3. Nodes shall not close the CP if no preamble of a frame was detected.

Figure 9-15 describes this scenario.

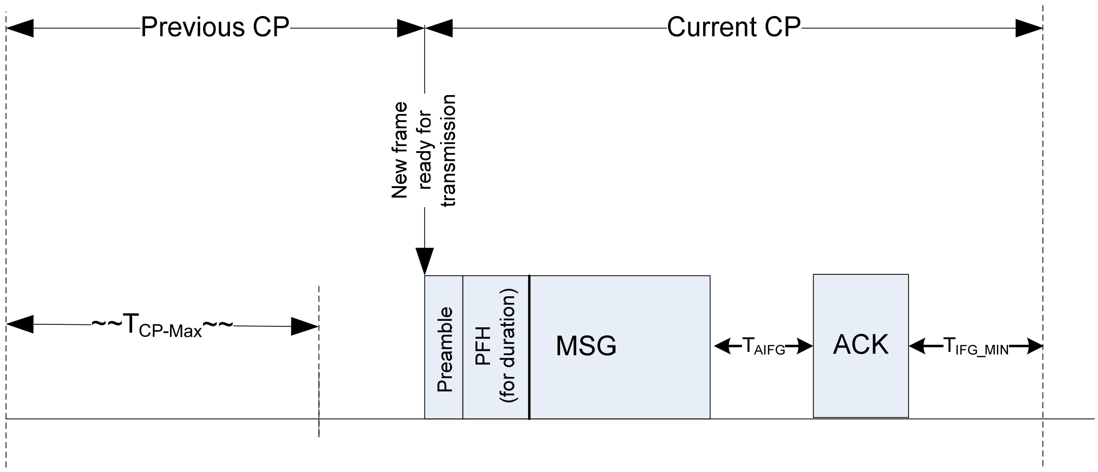


Figure 9-15 – Starting a CP after persistent line inactivity

9.3 Domain operation

9.3.1 Addressing scheme

9.3.1.1 Node identification

For the purpose of identification in the network:

1. Each node shall be assigned by the manufacturer with a 64-bit unique EUI-64 address (global address) using guidelines defined in [EUI-64].

2. Each node, after its registration into a particular domain (see clause 9.5.2), shall be assigned with a short address that includes:
   – a 16-bit DOMAIN_ID that is a unique identifier of the domain the node has registered with;
   – a 16-bit NODE_ID that is a unique node identifier inside of the domain the node has registered with.

9.3.1.2 Addressing modes

The following addressing modes are defined:

- short addressing mode
- extended addressing mode.

The defined addressing modes determine the node identification method to be used when transmitting or receiving a packet:

- Extended addressing mode:
  - by its EUI-64 address.
- Short addressing mode:
  - by its NODE_ID and DOMAIN_ID
  - by its MULTICAST_ID(s) and DOMAIN_ID
  - by the BROADCAST_ID and DOMAIN_ID.

The short addressing mode shall be used in all cases when the short address (NODE_ID, DOMAIN_ID) is available. The extended addressing mode shall be used only when short addresses are not available (e.g., during node registration and network initialization, as defined in clause 9.5.2). Valid values of node and domain identifiers in short addressing mode are presented in Table 9-12.

9.3.1.2.1 DOMAIN_ID

DOMAIN_ID is an identifier for a particular domain and is assigned at the creation of the domain. A DOMAIN_ID can be assigned using a manual mode of configuration or self-configuration mode. The selection of DOMAIN_ID and DOMAIN_ID conflict detection and resolution shall be performed as detailed in clause 9.5.1.

A BROADCAST_DOMAIN_ID is reserved for broadcast to all domains.

9.3.1.2.2 NODE_ID

The NODE_ID is an identifier of a particular node in a domain. In conjunction with a DOMAIN_ID, a NODE_ID identifies the source and destination nodes in the network. Valid values of NODE_ID are summarized in Table 9-12.

A unique (within the domain) NODE_ID value is assigned to a node as part of the registration to a domain, as described in clause 9.5.2, creating a unique DOMAIN_ID and NODE_ID combination. The assigned NODE_ID is valid until the node is resigned from the domain. After the node has resigned from the domain, the same value of the NODE_ID can be assigned to a new registered node in that domain.

The reserved NODE_IDs 0xFFFE and 0xFFFF are intended to identify a source node that has not yet been assigned a NODE_ID. The use of reserved values of the NODE_ID is defined in clause 9.5.2.

9.3.1.2.3 MULTICAST_ID

The range of MULTICAST_IDs is from 0x8000 to 0x9FFF.

Details of multicast addressing are for further study.

9.3.1.2.4 BROADCAST_ID

BROADCAST_ID is the same for all nodes in all domains and is intended for broadcast inside the domain. When used in conjunction with a BROACAST_DOMAIN_ID, as defined in Table 9-12, it allows the sending of a message to all nodes of all domains in the network.

9.3.1.2.5 Reserved IDs

Node identifiers in the range between 0xA000 and 0xFFFD are reserved by ITU-T.

Table 9-12 – Definition of node identifiers for the short addressing mode

	Parameter	Valid values	Description
Node identification	Reserved IDs	0xA000-0xFFFD	Reserved by ITU-T
	NODE_ID	0x0000-0x7FFF	The NODE_ID is used by registered nodes for communicating with other nodes within their domain or in other domains.
		0xFFFE, 0xFFFF	Reserved (see clause 9.3.1.2.2)
	BROADCAST_ID	0xFFFF	This ID is for broadcasts within identified domain when used as the destination node ID. Other use is for further study.
	MULTICAST_ID	0x8000-0x9000	These IDs are reserved for multicast within identified domain when used as the destination node ID. Other use is for further study.
Domain Identification	DOMAIN_ID	0x0000-0xFFFF	Domain IDs for manual configuration or self-configuration modes
	BROADCAST_DOMAIN_ID	0xFFFF	Domain ID reserved for broadcast to all domains

9.3.2 LLC frame transmission methods

The transmission methods are defined for both intra-domain connections (between nodes of the same domain) and inter-domain connections (between nodes of different domains).

9.3.2.1 Intra-domain single-hop transmission

9.3.2.1.1 Unicast transmission

An intra-domain unicast transmission enables communication between a single source node and a single destination node that are both associated with the same domain and have a direct connection at the physical layer. An intra-domain unicast transmission may be for data or management. Unicast transmission is allowed for both SS-MPDU and MS MPDU frames.

An intra-domain unicast transmission is uniquely identified by the tuple (source/destination DOMAIN_ID, source NODE_ID, destination NODE_ID). Intra-domain unicast transmissions may require acknowledgement, according to the rules specified in clause 9.3.3.

To unicast an MPDU using the short addressing mode, the transmitter shall set the following values of addressing fields in the MPH:

- The SDID and DDID fields – to the DOMAIN_ID of the domain the transmitter is associated with.
- The SNID field – to the NODE_ID of the source node.
- The DNID field – to the NODE_ID of the destination node.
- The MESH field shall not be transmitted (MHI field of the LFH shall be set to 0).

A receiver with a DOMAIN_ID that matches the DDID field indicated in the MPH and with a NODE_ID that matches the DNID indicated in the MPH shall recover LPDUs contained in the received MPDU, reassemble the LLC frame and forward it to the upper layer. If either the DDID field or DNID field indicated in the MPH do not meet, respectively, the DOMAIN_ID and the NODE_ID of the receiver, the receiver shall drop the MPDU.

If the one-hop unicast transmission is performed using the extended addressing mode, the source and destination addresses (SA, DA) shall be defined in the MESH field of the LFH; the corresponding address fields of the MPH (SNID, DNID) shall be set to the values of the corresponding NODE_ID, if available, or to 0xFFFF if either or both values of NODE_ID are not available. In the case where DNID = 0xFFFF, the MPDU shall be broadcasted, as described in clause 9.3.2.1.2. The RELI field of the LFH shall be set to 0.

NOTE – It is always beneficial to use the short addressing mode, therefore the extended addressing mode should be used only if the short address of either the destination or the source or both are not available.

The receiver shall drop all LLC frames that contain a MESH field indicating single-hop transmission (RELI field of the LFH is set to 0) with a DA that is different from the EUI of the receiver.

Acknowledgements by the receiver and retransmissions by the transmitter, if required, shall be performed as specified in clause 9.3.3.

9.3.2.1.2 Broadcast transmission

An intra-domain one-hop broadcast transmission enables communication between a single source node and all other nodes in the same domain that have direct connection at the physical layer. An intra-domain broadcast transmission may be for data or management. Broadcast transmission is allowed only for SS-MPDU frames. Transmission parameters are at the discretion of the vendor, but shall be limited to the mandatory values supported by nodes of all profiles.

A broadcast transmission is uniquely identified by the tuple (source/destination DOMAIN_ID, source NODE_ID).

Acknowledgements shall not be requested for intra-domain broadcast transmissions.

To broadcast an MPDU, the transmitter shall set the following values of addressing fields in the MPH:

- The SDID and DDID fields – to the DOMAIN_ID of the domain the transmitter is associated with.
- The SNID field – to the NODE_ID of the source node.
- The DNID field – to the BROADCAST_ID=0xFFFF.

A receiver with a DOMAIN_ID that matches the DDID field indicated in the MPH shall recover LPDUs contained in the received MPDU, reassemble the LLC frame and forward it to the upper layer. Other receivers shall drop the MPDU.

9.3.2.1.3 Multicast transmission

An intra-domain multicast single-hop transmission enables a node to send LLC frames to a selected set of destination nodes in the domain.

The protocol for single-hop multicast transmission is for further study.

9.3.2.2 Intra-domain multi-hop transmission

This Recommendation allows performing mesh networking by using either L2 routing (by relaying the LLC frames, as defined in clause 9.1.2) or L3 routing (by relaying the protocol data units above the A-interface). Prior to initialization, each domain shall be set into a particular routing mode that either allows intra-domain L2 routing (L2 relaying mode) or that disallows it (no L2 relaying is allowed for any node of the domain). The mode is determined by the A_MGMT.REQ primitive RelayMode (see clause 9.4.3.4.1).

The domain master shall inform all nodes registering into the domain whether L2 relaying is enabled or not. If L2 relaying is enabled, all relayed communications in the domain shall be performed as defined in this section. If L2 relaying is disabled, only single-hop communications in the domain (as per clause 9.3.2.1.1) are allowed, and nodes shall not relay any of the received frames, except beacon frames, if assigned by the domain master (see clause 9.3.4).

For a node, the L2-relaying capability is optional. Nodes shall indicate to the domain master their L2-relaying capabilities as they register into the domain. If L2-relaying within the domain is enabled, nodes that are relay-capable may be assigned as domain relays for unicast, multicast, or broadcast using the procedure described in clause 9.5.6.

The L2 relaying functionality of nodes shall be as follows:

- All nodes (including those that are not relay-capable) shall be capable of sourcing the relayed transmission (create a mesh header) and to receive LLC frames that were relayed to it as a final destination (process a frame that includes a mesh header).
- When L2 relaying is enabled, all relay-capable nodes assigned as domain relays shall perform full relaying functionality (i.e., as an LLC frame originator, a relaying node, and a receiver of relayed LLC frames) for unicast, multicast, or broadcast using the procedures described in clauses 9.3.2.2.1, 9.3.2.2.2 and 9.3.2.2.3 respectively.
- All relay-capable nodes that are not assigned as domain relays shall discard all frames for which this node is not a final destination.
- When L2 relaying is disabled, all nodes in the domain shall not generate LLC frames to be relayed and discard all received LLC frames for which that node is not the final destination.

  NOTE – A relay-capable node that is not assigned as a domain relay, may still be assigned as a beacon relay (BPR, see clause 9.3.4).

9.3.2.2.1 Unicast multi-hop transmission

Intra-domain unicast multi-hop transmission allows an originating node to unicast LLC frames to a final destination node via one or more intermediate L2-relays. The destination node and all relaying nodes shall be from the same domain as the originating node.

The next hop destination address and previous hop source address applied by originating nodes, relay nodes and destination nodes shall comply with the relaying database (routing tables or routing information in the SR header of the transmitted LLC frame). The rules for generating and updating the relaying database shall be as described in clause 9.5.6.

9.3.2.2.1.1 Operation of the originating node

For LLC frames that require relaying, the originating node shall perform as follows:

- Set both the V-bit and F-bit of the ADM fields of the mesh header to 1 (to indicate relaying based on short addresses).
- Set the HopsLft field of the mesh header to the maximum number of hops that the LLC frame is allowed to be relayed. If the LLC frame reaches the final destination node before the field reaches 0, the LLC frame shall still be processed by the final destination node.
- Set the originator address field of the mesh header to the node's NODE_ID.
- Set the final address field of the mesh header to the NODE_ID of the final destination node.
- Use the intra domain single-hop unicast transmission method described in clause 9.3.2.1 to transmit the LLC frame to the first relay node based on the information maintained in the relaying database (routing table or SR header) to reach the node with the NODE_ID matching the final address field indicated by the mesh header.

9.3.2.2.1.2 Operation of a relay node

If a node assigned as a domain relay receives an LLC frame containing a mesh header, and the receiving node is not the final destination node (i.e., the address of the receiving node does not match the value indicated on the final address field DA of the mesh header), and the DNID of the node the LLC frame was sent to and the SNID of the node the LLC frame was received from meets the relaying database, and the values of the SA and DA indicated in the mesh header meet the relaying database, the receiving node shall perform the following relaying procedure and, if either of the conditions above are not met, the received frame shall be discarded:

- If the HopsLft field in the received LLC frame is greater than 0:
  - The value of the HopsLft field of the relayed frame shall be the value indicated on the HopsLft field of the received frame decreased by 1.
  - Use the intra-domain single-hop unicast transmission method described in clause 9.3.2.1 to transmit the LLC frame to the next node based on the information maintained in the relaying database or in the SR header to reach the node with the NODE_ID matching the final address field indicated on the mesh header.
  - Use of LLC frame relaying rules shall be as defined in clause 9.1.2.4.
- If the HopsLft field in the received LLC frame equals 0, the received LLC frame shall be dropped.
- If the NODE_ID indicated on the DA field in the mesh header of the received LLC frame does not meet the current relaying database, this received LLC frame shall be dropped.
- If the NODE_ID of the node from which the frame was received (i.e., the SNID field of the MPH) or the value of the SA indicated in the mesh header does not meet the relaying database, the received LLC frame shall be dropped.

9.3.2.2.1.3 Operation of the destination node

If a node receives an LLC frame containing a mesh header, and the receiving node is the final destination node (i.e., the NODE_ID of the receiving node matches the value indicated on the final address field of the mesh header), and the NODE_ID of the source the frame was received (i.e., the SNID field of the MPH in the received frame) meets the relaying database, the receiving node shall accept the received LLC frame and pass it to the AE via the A-interface. If either of these conditions are not met, the received frame shall be dropped.

9.3.2.2.2 Broadcast multi-hop transmission

Intra-domain broadcast multi-hop transmission enables an originating node to broadcast LLC frames to all nodes of the domain via one or more intermediate L2-relays that belong to the same domain.

The next hop destination address and previous hop source address applied by the originating nodes, relay nodes and destination nodes shall comply with the broadcast relaying database (routing tables or SR header). For some hops the DNID may get a value of BROADCAST_ID = 0xFFFF. The rules for generation and updating the relaying database shall be as described in clause 9.5.6.

NOTE – This broadcast mechanism can be converted to flooding if the relaying database is set accordingly.

9.3.2.2.2.1 Operation of the originating node

For LLC frames that require relaying, the originating node shall perform as follows:

- Set both the V-bit and F-bit of the ADM fields of the mesh header to 1 (to indicate relaying based on short addresses).
- Set the HopsLft field of the mesh header to the maximum number of hops that the LLC frame is allowed to be relayed. If the LLC frame reaches the final destination node before the field reaches 0, the LLC frame shall still be processed by the final destination node. Set the originator address field of the mesh header to the node's NODE_ID.
- Set the originator address field of the mesh header to the node's NODE_ID.
- Set the final address field of the mesh header to 0xFFFF.
- Use the intra domain single-hop unicast transmission method to transmit the LLC frame to the first relay node or single-hop broadcast transmission method, depending on the information maintained in the broadcast relaying database to reach all the nodes in the domain. The single-hop transmission shall be performed as described in clause 9.3.2.1.

9.3.2.2.2.2 Operation of a relay node

If a node assigned as a broadcast domain relay receives an LLC frame containing a mesh header with a BROADCAST_ID, and the NODE_ID of the source node the LLC frame was sent from (i.e., the SNID field of the MPH in the received frame) meets the broadcast relaying database, and the value of SA indicated in the mesh header meets the relaying database the receiving node shall:

– accept the received LLC frame (if this frame has not already been received once), recover the ADP, and pass it to the AE via the A-interface;

– perform the broadcast relaying procedure described in this section.

If any of the conditions above are not met, the received frame shall not be relayed.

Broadcast relaying procedure:

- If the HopsLft field in the received LLC frame is greater than 0:
    - The value of the HopsLft field of the relayed frame shall be the value indicated on the HopsLft field of the received frame decreased by 1.
    - Use the intra-domain single-hop unicast transmission method to transmit the LLC frame to the next node or single-hop broadcast transmission, based on the information maintained in the broadcast relaying database. Other relaying rules shall be as defined in clause 9.1.2.4.
- If the HopsLft field in the received LLC frame equals 0, the received LLC frame shall be dropped.

- If the NODE_ID of the node from which the broadcast relay received the frame (i.e., the SNID field of the MPH in the received frame) does not meet the broadcast relaying database or the value of the SA indicated in the mesh header does not meet the relaying database, the received LLC frame shall be dropped.

9.3.2.2.2.3 Operation of the destination node

If a node receives an LLC frame containing a mesh header with the final address equal to 0xFFFF, and this node is not assigned a broadcast relay, and the NODE_ID of the source the frame was received from (i.e., the SNID field of the MPH in the received frame) meets the broadcast relaying database, and the value of SA indicated in the mesh header meet the relaying database, the receiving node shall accept the received LLC frame (if this frame has not already been received once), recover the ADP, and pass it to the AE via the A-interface. If either of the last two conditions are not met, the received frame shall be dropped.

9.3.2.2.3 Multicast multi-hop transmission

Intra-domain multicast multi-hop transmission enables an originating node to send LLC frames to a selected set of destination nodes in the domain via one or more intermediate L2-relaying nodes that all belong to the same domain.

The protocol for multi-hop multicast transmission is for further study.

9.3.2.3 Inter-domain single-hop transmission

Transmission methods for single-hop inter-domain communications are for further study.

9.3.2.4 Inter-domain multi-hop transmission

Transmission methods for multi-hop inter-domain communications are for further study.

9.3.3 Retransmission and acknowledgement

The retransmission and acknowledgment protocol comprises the following steps:

- the transmitter sends a frame with a request to acknowledge the MPDU and indicates the type of the acknowledgement in the ACK_REQ field of the PFH and the ACK_TYPE field of the MPH;
- the receiver acknowledges the MPDU by transmitting an Imm-ACK frame with the format defined in 9.3.3.1.1, using the type of acknowledgement requested by the transmitter and using rules described in clause 9.3.3.1, clause 9.3.3.3. No Imm-ACK frame shall be sent if acknowledgement is not requested by the transmitter;
- the transmitter retransmits the MPDU or its tributary LPDUs indicated in the received Imm-ACK frame using the protocol described in clause 9.3.3.3.

9.3.3.1 Acknowledgment of unicast frames

To request acknowledgement on the MPDU carried by a unicast PHY frame, the transmitter shall indicate the request in the ACK_REQ field of the PFH, as defined in clause 8.2.3.2, and in the ACK_TYPE field of the MPH, as defined in clause 9.1.3.1.1. All the ACK types presented in Table 9-13 shall be supported:

Table 9-13 – Supported ACK request types

ACK_REQ (PFH)	ACK_TYPE	Type of request	Receiver action (Note)
0	N/A	No acknowledgement required	No Imm-ACK frame shall be sent.
1	0	Requires an ACK/NACK for the whole MPDU only	An Imm-ACK frame shall be sent indicating whether or not the MPDU was received error-free. The bit "ACK/NACK" in the PFH of the Imm-ACK frame shall be set to 1 if the MPDU was received error-free and 0 otherwise. The value of 0 shall be interpreted as "NACK".
1	1	Requires an ACK for the MPDU with an indication of each LPDU of the MPDU that was received in error	An Imm-ACK frame shall be sent indicating in the ACKI field of the PFH of the Imm-ACK frame all the LPDUs of the MPDU that were received in error. The coding of the ACKI field shall be as defined in clause 9.3.3.1.1
NOTE – Receiver actions are described in more detail in clause 9.3.3.3.2.			

If acknowledgement is required for a unicast frame, the acknowledging node (receiver) shall respond to the receipt of a frame with an Imm-ACK frame that is a type 3 frame and consists of a preamble and a PFH, but no payload, as specified in clause 8.2.1. The format of the Imm-ACK frame header shall be as defined in clause 9.3.3.1.1. The destination of the Imm-ACK frame shall be the same as the source address of the node that requested the acknowledgement. The receiver shall not acknowledge incoming frames if not requested by the transmitter.

The Imm-ACK frame shall be transmitted T_{AIFG} seconds after the end of the received PHY frame that has requested the Imm-ACK (see Figure 9-16). If the MAC frame is received with errors as determined by the FCS, the receiver may send a NACK to the originator only if an acknowledgment is requested and the destination address of the frame matches the receiver's device address. However, if the receiver can determine that the error is caused by collision, it may avoid sending a NACK (no response) to invoke a collision state on transmitting station. The transmitting station shall infer a collision from the absence of any response to a transmission when a response is expected. In this case the transmitting station shall attempt a retransmission after a T_{CBIFG} interval.

The node that shall transmit the Imm-ACK is not allowed to receive frames after the end of the received PHY frame until the Imm-ACK is sent.

All nodes in the domain shall refrain from transmission when the Imm-ACK is expected and within T_{IFG_MIN} after the transmission of the Imm-ACK frame transmission is complete. The refrain time shall be computed based on the PHY frame type and the status of the ACK_REQ bit in the PFH. If the sender indicates the PHY frame transmission interval using the FL field of the PFH (PHY frame type 2 or 4, see clause 8.2.3.2), this interval shall include the maximum value of T_{AIFG} and Imm-ACK transmission time, if the Imm-ACK is requested by the sender. The sender of a frame shall ensure that the overall duration of the frame with the following Imm-ACK frame is in the range covered by the FL.

NOTE – The range of the FL is capable of accommodating the overall frame duration up to 360 ms, which is usually longer than the allowed time-on-wire defined by the parameter MaxMediumTime (see clause 9.1.4).

The timing of the Imm-ACK frame transmission is described in Figure 9-16. The values of T_{AIFG} and T_{IFG_MIN} shall be as defined in clause 9.1.4.

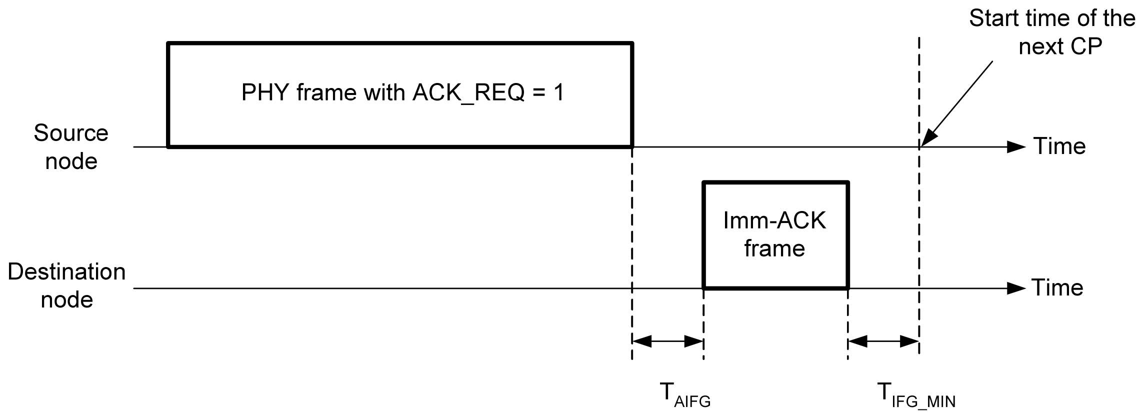


Figure 9-16 – Timing of the Imm-ACK frame

An MS-MPDU capable and SS-MPDU capable receiver shall respond using an Imm-ACK frame of the corresponding type (see clause 9.3.3.1.1).

An MS-MPDU receiver that has no sufficient resources to provide a selective ACK (in the case where the ACK_TYPE = 1 in the received frame) may send an ACK for the whole MPDU (as per request ACK_TYPE = 0) that informs the source that its request for selective ACK cannot be accomplished. Each node shall always be capable of providing ACK and NACK for the whole MPDU (as required per ACK_TYPE = 0).

If a receiver that does not support decoding of MS-MPDU frames and receives an MS-MPDU frame with no errors in the MPH, it shall respond by sending an Imm-ACK as for the SS-MPDU (Table 9-15) with an unable-to-comply (UTC) bit set, indicating that it does not support the MS-MPDU.

If a node receives a frame that carries an MPDU that has already been received correctly or carries an MPDU containing LPDUs that had already been received correctly in the previously transmitted frames, the receiver shall be capable of filtering out and dropping this duplicate MPDU or all of these duplicate LPDUs. If these duplicates are received in error, the receiver shall acknowledge them as received correctly (to avoid unnecessary retransmissions).

9.3.3.1.1 Format of the Imm-ACK frame

The Imm-ACK frame shall have a format of a PHY frame type 3 with the fields defined in Table 8-12. The reserved field of the PHY frame shall be used for the acknowledgement data. Four sub-types of type 3 frames are defined, depending on the 2-bit frame type field (bits [3:2] of the PFH):

– 00 – Imm-ACK of a MS-MPDU capable node
– 01 – Imm-ACK of a SS-MPDU capable node
– 10 – Extended Imm-ACK frame
– 11 – Reserved

The Imm-ACK frame generated by an MS-MPDU capable node shall be as indicated in Table 9-14.

Table 9-14 – Imm-ACK data field in PFH (frame type 3)

Field	Bits	Description
Frame type	[3:2]	00
Reserved	[5:4]	Reserved by ITU-T (for flow control, see clause 9.5.7)
DNID	[11:6]	The 6 LSB of the NODE_ID of the node that requested the acknowledgement (Note 2)
LFSN	[19-12]	Sequence number of the LLC frame carried by the MPDU that is acknowledged by this Imm-ACK frame. Shall use the same format as defined in the MPH of the MPDU that is acknowledged
ACK TYPE	[20]	Type of ACK data: 0 indicates per-MPDU acknowledgement, 1 indicates selective acknowledgement. In case a node is requested to provide selective acknowledgement but not capable, this bit shall be set to 0.
ACK/NACK	[21]	If ACK TYPE = 0, indicates per-MPDU acknowledgement (see Table 9-13 for the definition of the acknowledgement types) and shall be set to 1 for ACK (all LPDUs and MPCS of the MPDU are received with no errors) and to 0 for NACK (error in at least one of the LPDUs). If ACK TYPE = 1, indicates selective acknowledgement on the first LPDU of the received MPDU, and shall be set to 1 if the LPDU is received correctly and 0 otherwise.
ACKI/TP-PR	[28:22]	ACKI operation Bit map of errored LPDUs (1= LPDU received correctly, 0 = LPDU received in error). The first bit of the bit map represents the second LPDU of the received MPDU (the first LPDU is the one that follows the MPH and its status is represented by the ACK/NACK field). Valid only if ACK TYPE = 1 In case ACK TYPE = 0 and the received value of TP-REQ ≠ 11, bits [28:22] shall be set to 0 and ignored by the receiver of the Imm-ACK frame TP-PR operation: For CENELEC and FCC1 bandplans only, if the received value TP-REQ = 11, this field shall carry the TP-PR field value (regardless of the ACK-TYPE assignment), using the format defined in Table 9-17 (Note 3).
LQI	[29]	Link quality indicator. Shall be set to 0 if the receiver does not recommend changes in transmission parameters and set to 1 otherwise. If ACK and all ACKI bits are set to 1, the LQI = 1 represents that the received SNR is substantially higher than the one required for used transmission parameters (increase of bit rate is possible). If ACK or one of ACKI bits is set to 0, the LQI = 1 represents that the received SNR is substantially lower than the one required for used transmission parameters (more redundancy shall be used to keep the link reliable). The evaluation of SNR for LQI setting is determined by the vendor.

Table 9-14 – Imm-ACK data field in PFH (frame type 3)

Field	Bits	Description
TP-PR	[36:30]	Partial TP recommendation of the receiver, represented in a format defined in Table 9-17. Applicable for FCC and FCC2 bandplans only
Reserved	[61:37]	Reserved by ITU-T (applicable to FCC and FCC2 bandplans only)
NOTE 1 – All bits reserved by ITU-T shall be set to 0 by the transmitter and ignored by the receiver. NOTE 2 – The destination node is identified by the 6 LSB bits of the NODE_ID and the LFSN of the LLC frame to which the segments carried by the acknowledged LPDUs belong. NOTE 3 – For CENELEC and FCC1 bandplans, if the received value of TP-REQ=11, the ACK/NACK indicates the status of the whole MPDU (the transmitter shall set the ACK_TYPE bit of the MPH to 0).		

The format of an Imm-ACK frame generated by a node that is only capable of receiving SS-MPDUs (a node of low complexity profile) shall be as defined in Table 9-15.

Table 9-15 – Imm-ACK data field in PFH (frame type 3)

Field	Bits	Description
Frame type	[3:2]	01
Reserved	[5:4]	Reserved by ITU-T (for flow control, see clause 9.5.7)
DNID	[11:6]	The 6 LSB of the NODE_ID of the node that requested the acknowledgement (Note 2).
LFSN	[19-12]	Sequence number of the LLC frame carried by the MPDU that is acknowledged by this Imm-ACK frame. Shall use the same format as defined in the MPH of the MPDU that is acknowledged.
UTC	[20]	Unable to comply – shall be set to 1 if the receiver is incapable of processing the incoming MS-MPDU frame addressed to it and 0 otherwise.
ACK/NACK	[21]	Shall be set to 1 for ACK (the MPDU received with no errors) and to 0 for NACK (MPDU received in error). Valid only if UTC bit is 0.
TP-PR	[25:22]	Partial TP recommendation of the receiver, represented in a format defined in Table 9-17 (applicable for all bandplans)
Reserved	[28:26]	Reserved by ITU-T
LQI	[29]	Link quality indicator. Shall be set to 0 if the receiver does not recommend changes in transmission parameters and set to 1 otherwise. If ACK/NACK bit is set to 1, the LQI = 1 represents that the received SNR is substantially higher than the one required for used transmission parameters (increase of bit rate is possible). If ACK/NACK bit is set to 0, the LQI = 1 represents that the received SNR is substantially lower than the one required for used transmission parameters (more redundancy shall be used to keep the link reliable). The evaluation of SNR for LQI setting is determined by the vendor.
Reserved	[61:30]	Reserved by ITU-T (only valid for FCC and FCC2 bandplans)
NOTE 1 – All bits reserved by ITU-T shall be set to 0 by the transmitter and ignored by the receiver. NOTE 2 – The destination node is identifies by 6 LSB of the NODE_ID and the LFSN of the LLC frame to which the segments carried by the acknowledged LPDUs belong.		

An extended Imm-ACK frame contains regular ACK fields and extension fields attached after the regular ACK fields. If a node is incapable of providing the requested data for one or more extension fields, it shall respond with an unable-to-comply-extension (UTCE) bit set in the extension field of the extended Imm-ACK. The ACK type extension field indicates whether the extended Imm-ACK is an SS-MPDU or MS-MPDU type. Format and content of other extension fields is for further study.

Table 9-16 – Extended Imm-ACK – extension data fields

Field	Bits	Description
ACK type	[0]	0 – for MS-MPDU 1 – for SS-MPDU
UTCE	[1]	Set to 1 if unable to support the extension field, and 0 otherwise.
Reserved	Reserved	For further study
NOTE – All bits reserved by ITU-T shall be set to 0 by the transmitter and ignored by the receiver.		

9.3.3.1.1.1 TP-PR field

The TP-PR field indicates the partial TP response, which is the receiver recommendation on a particular set of TP. It shall be filled by the receiver in response to the request of the transmitter for a partial TP report (TP-REQ = 11, see clause 9.1.3.1.1). If the receiver is incapable of providing a recommendation for one or more parameters in the Imm-ACK associated with the frame where the request was sent, it shall set the corresponding TP-PR fields to a "No change" indication (see Table 9-17) and prepare to provide the report in the Imm-ACK associated with the next frame received from the same source. If no request for a partial TP report from the transmitter was sent, the receiver shall put "No change" into all TP-RP fields, if applicable.

For CENELEC and FCC1 bandplans, the transmitter that requests a partial TP report for an MS-MPDU shall set ACK_TYPE = 0 (shall not request a selective ACK). For FCC and FCC2 bandplans, the transmitter of MS-MPDU may request both a selective ACK (ACK_TYPE = 1) and partial TP report.

The coding of the TP-PR fields shall be as defined in Table 9-17.

Table 9-17 – Format of the TP-PR field

Field	Bits			Note
	MS-MPDU FCC, FCC2	MS-MPDU CENELEC, FCC1	SS-MPDU All bandplans	
RS codeword size (RSCW)	[30]	[22]	[22]	Same as RSCW field described in clause 8.2.3.2.4
CC Rate (CCR)	[31]	[23]	N/A	Same as CCR field described in clause 8.2.3.2.5
Change the number of repetitions (REP)	[33:32]	[25:24]	[24:23]	00 – No change 01 – One step down 10 – One step up 11 – Reserved by ITU-T
Interleaving mode (INTM)	[34]	[26]	[25]	Same as INTM field described in clause 8.2.3.2.7

Table 9-17 – Format of the TP-PR field

Field	Bits			Note
	MS-MPDU FCC, FCC2	MS-MPDU CENELEC, FCC1	SS-MPDU All bandplans	
Modulation (MOD)	[36:35]	[28:27]	N/A	Same as MOD field described in clause 8.2.3.2.8
NOTE – The coding of all TP-PR fields, except REP, shall be interpreted as: – if the receiver has got no recommendation (cannot estimate the parameter) it shall recommend the same value as it was assigned by the transmitter ("No change" indication); – if the receiver has got a recommendation to change the value, it shall put in the recommended value.				

9.3.3.2 Acknowledgement for multicast and broadcast frames

The source of multicast and broadcast frames shall not request acknowledgement (ACK_REQ bit of the PFH shall be set to 0).

9.3.3.3 Retransmission protocol for unicast frames

9.3.3.3.1 Requirements for the transmitter

If acknowledgement is required for an LLC frame, the transmitter shall request acknowledgement for each MPDU carrying segment(s) of this LLC frame. For different MPDUs, the transmitter may request different types of ACK (per MPDU or per LPDU, if available). Upon receipt of an Imm-ACK frame indicating that a particular MPDU or its tributary LPDUs were received in error, the transmitter shall suspend transmission of other MPDUs of the same destination, except MPDUs of MA priority 3, and retransmit the indicated erroneous MPDU or its tributary LPDU using the rules described in clause 9.1.2.3. No MPDU of an LLC frame is allowed to be transmitted until all LPDUs of previously transmitted LLC frames are cleared (received error free) or timed out. If an MPDU of MA priority 3 is ready for transmission, it shall be transmitted at the first opportunity, interrupting the transmission of a data or management frame.

If a node that transmitted a frame receives in response an Imm-ACK frame indicating that the receiver does not support the MS-MPDU mode, the node shall retransmit the LPDUs of this MS-MPDU using the SS-MPDU mode.

If acknowledgements on all LPDUs of a particular LLC frame are not received by the transmitter (no Imm-ACK frame received or a persistent negative acknowledgement is received) after a total of MaxRetransmissions (see clause 9.1.4) attempts, the transmitter shall discard the LLC frame and notify upper layers that transmission of the LLC frame failed using the A_MGMT.IND primitive FrameTxFailure (see clause 9.4.3.4.3). The same notification shall be sent if the transmitter does not get all the LPDUs of an LLC frame acknowledged during the MaxAckRxTime (see clause 9.1.4) timeframe. The transmitter shall adjust this timeframe in case transmission of the LLC frame is interrupted by a priority 3 frame by freezing the MaxAckRxTime counter of the interrupted frame till receipt of the interrupting frame is complete.

NOTE – The upper layer may further decide to transmit the discarded LLC frame again. This functionality is beyond the scope of this Recommendation.

9.3.3.3.2 Requirements for the receiver

The receiver shall first identify whether the received frame with ACK_REQ = 1 is valid for acknowledgement by checking the LPH and MPH. If the MPH is received in error (MHCS does not fit or any of the MPH fields are invalid) or at least one of the address fields indicated in the MPH (DNID, SNID, DDID, SDID) does not fit, or a segment sequence number in the LPH is inconsistent, the value of the MDPU type is invalid, the receiver shall not send an Imm-ACK frame and shall drop the received frame. In the received MPDUs that are valid for acknowledgement, the receiver shall verify the segmentation method of the received MPDU (i.e., whether this is an MS-MPDU or an SS-MPDU).

The receiver shall apply the following criteria to identify errored MPDUs and LPDUs. For an MS-MPDU the receiver shall identify errored LPDUs by checking their LPCS; the MPDU is considered in error if at least one LPCS is received in error. For an SS-MPDU, the receiver shall identify errored MPDUs by checking the MPCS. If a receiver is only capable of processing the SS-MPDU, it shall respond on the received MS-MPDU by ACK frame type 01 with the UTC bit set to 1 or, if capable, of receiving an MS-MPDU, with an ACK/NACK as defined in Table 9-14.

If the received frame has an inconsistent sequence number and the PRI field of the MPH is set to 11b, the receiver shall consider this frame as an MA priority 3 interrupting frame.

NOTE – To manage the receive buffers, the receiver may perform a time-limited reception of each LLC frame based on MaxRetransmissions and MaxAckRxTime parameters of the transmitter (see clause 9.3.3.3.1, clause 9.1.4). In particular, if during MaxAckRxTime not all of the LLC frame segments are received, the receiver may discard all previously received segments of this LLC frame. The count of MaxAckRxTime in this case shall start from the time the first MPDU carrying segments of an LLC frame has been received.

9.3.4 Beacon operation

Beacons are special management frames used by the domain master to control and maintain the domain. The format of the beacon frame is defined in clause 9.3.4.4.

A domain shall be initialized in one of three beacon modes:

- synchronous beacon mode (see clause 9.3.4.1)
- asynchronous beacon mode (see clause 9.3.4.2)
- beaconless mode (see clause 9.3.4.3).

During the domain operation, the domain master can change the beacon mode based on the corresponding request from the upper layer management (management primitive BeaconMode at A_MGMT interface defined in clause 9.4.3.4.1). The procedure of changing the beacon mode is for further study.

The following rules related to the beacon modes shall be respected:

- The receiving and processing of received beacons frames shall be supported by all nodes.
- Support of the synchronous beacon mode is optional.
- In synchronous beacon mode, nodes that support only the asynchronous beacon mode shall respond to the asynchronous portion of the synchronous beacon (i.e., may ignore the BCN_SyncSchdle.req message – see clause 9.3.4.4).

The particular beacon mode used in the domain is indicated in each transmitted beacon frame (in the BM field of the BCN_DI.req message, Table 9-19). Each beacon frame also includes a serial number (in BCN_ID field of the BCN_DI.req message), in the range between 0 and 255. Whenever a new beacon frame is generated by the domain master, the BCN_ID field shall be increased by 1 and shall wrap back to 0x00 after reaching 0xFF.

The beacon frame shall always be sent using the robust communication mode (RCM); values of the transmission parameters are defined in clause 9.3.4.4.

Only the domain master is allowed to transmit beacons and it is a responsibility of the domain master to provide beacons for all nodes of the domain. To reach hidden nodes, the domain master may assign beacon re-generation proxies (BPR).

9.3.4.1 Synchronous beacon mode

In synchronous beacon mode, transmission is organized by beacon cycles of a fixed duration. Each cycle starts with a beacon frame (see clause 9.3.4.4) and is followed by time slots reserved for relayed beacons (R-beacons), followed by the time period assigned for frame transmission, Figure 9-17.

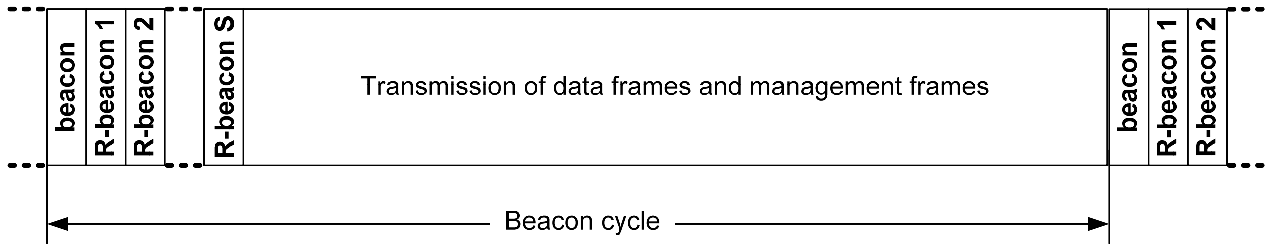


Figure 9-17 – Beacon cycle structure in the synchronous beacon mode

The domain master shall transmit the beacon in the beacon slot only. By sending the next beacon, the domain master determines the duration of the beacon cycle. The duration of the beacon cycle shall not exceed the duration that corresponds to the maximum value of the BCD field of the BCN_SyncSchdle.req message (see Table 9-21).

All assigned BPRs shall be capable of supporting the synchronous beacon mode. The domain master shall also assign a number of slots in the beacon cycle for R-beacon transmissions following the beacon slot (see Figure 9-17). The number of R-beacon slots in a beacon cycle shall be limited to $S \leq 8$. If more than S slots for R-beacons are required, the domain master may assign a particular set of BPRs for each beacon cycle, ensuring that each BPR transmits R-beacon at least once during the period BCNMaxT which is an LLC_MGMT primitive (see clause 9.4.3.2) determined by the domain master and communicated to the BPR nodes in the corresponding field of the BCN_DI.req message (see Table 9-20).

NOTE – One simple way of achieving BPR assignment can be through a round robin method: a particular BPR may be assigned to transmit R-beacon #N at every *m*-th beacon cycle starting from the beacon cycle #D.

To facilitate an order in R-beacon generation, every beacon:

- carries the serial number of the beacon cycle on the BCID field on the BCN_SyncSchdle.req message as described in Table 9-21. The serial number of the beacon is indicated in each of the corresponding R-beacons; this field shall be increased by 1 for each new beacon cycle, and shall wrap back to 0x00 after reaching 0xFF;
- indicates the set of BPRs assigned to transmit R-beacons in the particular beacon cycle on the BRP_N and BPRi fields on the BCN_SyncSchdle.req message (see Table 9-21).

The duration of R-beacon slots and other timing and medium access details concerning synchronous beacons are for further study.

All end-nodes of the domain shall not transmit during the beacon and any R-beacon time periods; BPRs are allowed to transmit R-beacons only in a particular R-beacon period assigned by the domain master and communicated in the corresponding BPRi field of the beacon.

9.3.4.2 Asynchronous beacon mode

In asynchronous beacon mode no particular beacon cycle boundaries are defined. Beacon frames are transmitted at the discretion of the domain master, and R-beacons are transmitted on discretion of the BPRs assigned by the domain master. In asynchronous beacon mode, the assigned BPR nodes may not support the synchronous beacon mode.

Both beacons and R-beacons shall be transmitted with MA priority 3.

A beacon shall be transmitted at least once during a predefined time period specified on the BCNMaxT field of the BCN_DI.req message (see Table 9-20). The timing and other rules of R-beacon transmission are for further study.

9.3.4.3 Beaconless mode

In beaconless mode beacon frames shall not be sent (all domain controls are provided as part of the corresponding L3 protocols, by L3 in-band management messages). More details on beaconless mode are for further study.

9.3.4.4 Beacon frame format

The beacon frame includes one or more beacon management messages (message category 001, as described in clause 9.4.2). No other management messages are allowed to be included into a beacon frame.

A beacon frame shall include the BCN_DI.req management message defined in Table 9-20, as the first management message of the beacon frame. More beacon management messages may follow. In synchronous beacon mode, the beacon frame shall also include the BCN_SyncSchdle.req message defined in Table 9-21.

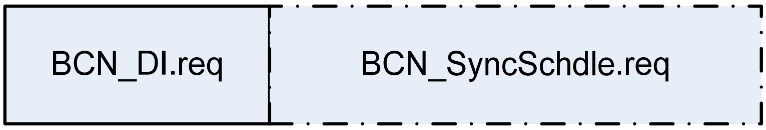


Figure 9-18 – Example of a beacon frame format including the BCN_DI.req message and one additional beacon message (BCN_SyncSchedle.req)

9.3.4.4.1 Transmission of beacon frames

The following setting and transmission parameters shall be set for the transmission of beacon frames.

The LFH fields of an LLC frame that carries the LCDU of a beacon frame shall be set as follows:

- LLCFT = 1 (LCDU)
- MHI = 1 (mesh header is present)
- CCMPI = 0 (not secure).

The mesh header fields shall be set as follows:

- RELI = 1 (relaying extension are present)
- ADM = 00 (short addressing mode for both source and destination node)
- LPRI = 3
- SRI = 0 (no source routing)
- SA = Node ID of the domain master
- DA = 0xFFFF (BROADCAST_ID)
- HopsLft = maximum number of hops to relay the beacon (equal to S for synchronous beacon).

The MPH fields of each MPDU comprising LPDUs of a beacon frame shall be set as follows:

- MPDU_TYPE = 000 (SS-MPDU)
- ACK_TYPE = 0
- SNID = NODE_ID of the transmitter (domain master or BPR)

- DNID = 0xFFFF (BROADCAST_ID)
- SDID = DDID = DOMAIN_ID.

The transmission parameters of the beacon frames shall be the following:

- code rate = 1/2
- number of repetitions: 4, 6, or 12
- robust header and robust preamble as per clauses 8.3.4 and 8.4.5.1, respectively.

Other transmission parameters are determined by the vendor, but shall be limited to the values supported by nodes of all profiles.

9.3.4.4.2 Transmission of R-beacon frames

In asynchronous beacon mode, the R-beacon frames shall use the same format, settings and transmission parameters as defined in clause 9.3.4.4.1. Exceptions and additional details related to the synchronous beacon mode are for further study.

9.4 Management plane

9.4.1 Management communications

Two types of management communications are defined: internal and in-band (external). Internal management messages are carried by LCDUs and are exchanged between the peer management entities of the nodes in the domain via the LLC_MGMT SAP and A_MGMT interface (see clauses 9.4.3.4.1 and 9.4.3.4.3). Management messages intended for in-band communications are carried by APDUs and exchanged between the peer management entities of the nodes via the A_DATA interface (see Annex I).

Both types of management communications shall use the same management message format presented in Figure 9-19. Figure 9-19 also shows the mapping of a management message into an LCDU and APDU: the APDU and LCDU payload may contain one or more concatenated management messages (MM). The FCS shall be 32-bit long and computed as defined in clause 9.1.2.1.3, and shall not be included when MIC is used in the LLC frame encapsulating the LCDU or the APDU. All MMs shall be encapsulated so that their first byte (octet 0 of the MMH) is transmitted first and the first byte of MM-1 is the first byte of the LCDU or the APDU payload, respectively. Additionally, encapsulation of the APDU or LCDU carrying a management message into an LLC frame shall be as shown in Figure 9-5.

NOTE – The format of APDU depends on the used L3 protocol and is defined in Annex B. For the case of IPv6, see Figure B.1.

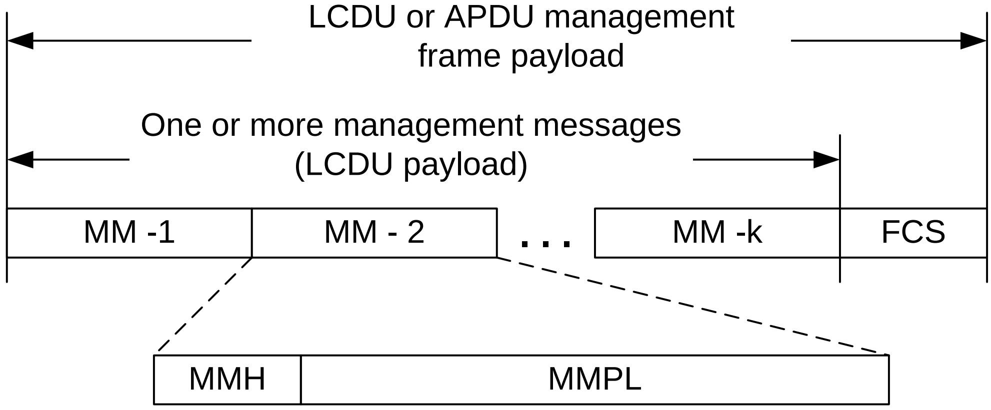


Figure 9-19 – Format of a management message and its mapping into LCDU and APDU

9.4.1.1 Management message format

All management messages shall be formatted as shown in Figure 9-19, and shall comprise a management message header (MMH) and a management message parameter list (MMPL).

The MMH defines the type, the length, the frame sequence number, and parameters related to message segmentation. The MMH also indicates whether the message is the last one mapped into an LCDU/APDU or it is followed by another management message. The type of the message is identified by an OPCODE associated with a particular management function, as presented in Table 9-19. The MMPL includes a list of parameters associated with a particular management function associated with the management message. The format of the MMH shall be as shown in Table 9-18.

The format of any MMPL may be revised in future versions of this Recommendation by appending additional fields. Furthermore, fields may be defined using bits that are currently indicated as reserved for ITU-T. Nodes shall indicate the version of the Recommendation that they support during the registration (see clause 9.5.2) and in their topology reports. Nodes shall be able to parse any MMPL (the length of the MMPL is specified in the MMH) by ignoring the fields associated with later versions of the Recommendation (which legacy nodes cannot interpret).

The maximum size of the management message carried by an APDU is determined by the particular AE protocol and thus always complies with the maximum APDU size. The maximum size of the management data carried by the LCDU is limited by the size of the LCDU to 992 bytes (see clause 9.1.2.1.3 and Table 9-15). If the amount of management data to be sent exceeds 992 bytes, segmentation shall be used, as defined in Table 9-15, providing a MMPL size that is less than 992 bytes.

NOTE – If channel conditions require the use of shorter frames, this is provided by the segmentation mechanism of the LLC frame, see clause 9.1.2.2.

Table 9-18 – Format of MMH

	Content	Octet	Bits	Description
MMH	MMPL length (LG)	0-2	[11:0]	Length of the MMPL segment in octets, encoded as a 12-bit unsigned integer. The length value shall not exceed 992 bytes.
	OPCODE		[23:12]	12-bit OPCODE, indicates message type (Note 1)
	Reserved	3	[7:0]	Reserved by ITU-T
	Number of segments	4	[3:0]	Number of segments minus 1, represented as an unsigned integer between 0 and F16. It shall be set to 0016 if the message is not segmented.
	Segment number		[7:4]	Segment number, represented as an unsigned integer between 016 and F16; set to 016 for the first segment and if message is not segmented (Note 2).
	Sequence number	5-6	[15:0]	Sequence number of the message in a format of 16-bit unsigned integer (Note 3)
	Next header	7	[0]	Shall be set to 0 if this message is the last one mapped in the APDU/LCDU, and set to 1 otherwise.
	Reserved	7	[7:1]	Reserved by ITU-T

Table 9-18 – Format of MMH

	Content	Octet	Bits	Description
MMPL	Message Parameters	8 to (LG+7)	[(8×LG-1):0]	Depends on the OPCODE, see Table 9-19
	Pad	> (LG+7)	[7:0]	Shall be filled by zeros (Note 4)

NOTE 1 – The OPCODES are defined in Table 9-19.

NOTE 2 – Segmentation shall be in the ascending order of octets, i.e., starting from octet #0. Segmentation shall not be applied segmented if the required MMPL is shorter than 992 bytes.

NOTE 3 – The sequence number identifies the relative time the message was sent: bigger sequence number corresponds to later transmission time. Actions associated with the sequence number of the received message depend on the OPCODE. All segments of the same message shall have the same sequence number. The value wraps around after reaching 216.

NOTE 4 – Zero bytes can be added at the end of the MMPL to reach the minimum size of the payload (if required) or for any other purpose.

9.4.1.2 Management message OPCODEs

OPCODEs shall be formatted as 12-bit unsigned integers, where the 8 MSBs indicate the category of the message and the 4 LSBs indicate the message ID in a category. Valid values of OPCODEs are presented in Table 9-19. Management messages are categorized by their associated protocol or procedure. If the number of messages for a particular protocol or procedure is more than 16, two or more 8-bit values can be assigned for the same category.

9.4.2 Management messages

A complete list of management messages used in the Recommendation is presented in Table 9-18. The MMPL content of each message is defined in the following sub-sections. Four types of messages are defined: Request (.req), Confirmation (.cnf), Indication (.ind), and Respond (.rsp), according to the messaging convention in clause 6.2.

Table 9-19 – OPCODEs of management messages

Category	Message name	OPCODE (hex)	Description	MMPL Reference
Beacon (00X)	BCN_DI.req	000	Domain information	Table 9-20
	BCN_SyncSchdle.req	001	Synchronous beacon scheduling parameters	Table 9-21
Channel Estimation (03X)	Transmission Parameters Request (TP.req)	031	Request for transmission parameters	N/A
	Transmission Parameters (TP.ind)	032	Transmission parameters recommendation	Table 9-20
Request for Information (04X)	RFI_BcnInfo.req	000	Request for beacon information	Table 9-22
	RFI_BcnInfo.ind	001	Beacon information indication	Table 9-23

Other messages and message categories are for further study.

9.4.2.1 Beacon messages

The MMPL of the beacon messages are described in Tables 9-17 to 9-18.

9.4.2.1.1 Domain information message

The BCN_DL.req message is intended to distribute domain information to the nodes of the domain. The MMPL field of the BCN_DL.req message is presented in Table 9-20.

Table 9-20 – Description of the MMPL of the BCN_DI.req message

Field	Octet	Bits	Description
BCN_ID	0	[7:0]	Beacon ID, formatted as a non-signed integer
PFHM	1	[0]	Domain-wise PFH mode: 0 – Normal PFH shall be used for transmission in the domain 1 – Robust PFH shall be used for transmission in the domain
PRMBLM	1	[1]	Domain-wise preamble mode: 0 – Normal preamble shall be used for transmission in the domain 1 – Robust preamble shall be used for transmission in the domain
L2R	1	[2]	L2 relaying support: 0 – No L2 relaying allowed in the domain 1 – L2 relaying is allowed in the domain
BM	1	[4:3]	Beacon mode: 00 – Beaconless mode 01 – Synchronous mode 10 – Asynchronous mode 11 – Reserved by ITU-T
SL	1	[6:5]	Security level: 00 – None (encryption disabled, no MIC) 01 – Normal security (encryption enabled, 4 bytes MIC) 10 – High security (encryption enabled, 16 bytes MIC) 11 – Reserved by ITU-T
Reserved	1	[7]	Reserved by ITU-T
REC	2	[7:0]	Version of the recommendation supported by the domain master. Set to 0x00
DMID	3-4	[15:0]	Node ID of domain master
DMExtAddr	5-20	[7:0]	EUI-64 address of the domain master
BDMID	21-22	[15:0]	Node ID of the backup domain master (0xFFFF if not assigned)
DNI	23-24	[15:0]	Domain name identifier
BPL	25	[7:0]	Lowest tone index of the bandplan
BPU	26	[7:0]	Upmost tone index of the bandplan
BCNMaxT	27-30	[31:0]	Maximum time allowed in the domain without transmission of a beacon in resolution of 100 ms
RMSC_N	31	[7:0]	Number of RMSC index fields following this field
RMSC1	32	[7:0]	1st index of the RMSC set
…		…	…
RMSCn	31+n	[7:0]	n-th index of the RMSC set
NOTE – All fields reserved by ITU-T shall be set to 0 by the transmitter and ignored by the receiver.			

9.4.2.1.2 Domain synchronization schedule message

The BCN_SyncSchedule.req message is intended to distribute the node transmission schedule and shall be used only if the domain operates in synchronized beacon mode. The MMPL field of the BCN_SyncSchedule.req message is presented in Table 9-21.

Table 9-21 – Description of the MMPL of the BCN_SyncSchedule.req message

Field	Octets	Bits	Description
BCD	0-3	[31:0]	Duration of beacon cycle in resolution of 100ms
BCID	4	[7:0]	Synchronous beacon cycle ID
BPR_N	5	[7:0]	Number n of beacon regeneration proxy (BPR) slots in this beacon cycle (the n fields following this field define the NODE_ID of each BPR)
BPR1	6-7	[15:0]	Node ID of BPR1
…	…	…	…
BPRn	6+2xn – 7+2xn	[15:0]	Node ID of BPRn

9.4.2.2 Request for information (RFI) messages

9.4.2.2.1 Request for information message

The MMPL of an RFI_BcnInfo.req message is described in Table 9-22.

Table 9-22 – Description of the MMPL of the RFI_BcnInfo.req message

Field	Octets	Bits	Description
OPC-Req	0-1	[11:0]	Bit map of the opcodes of the category 00X (Beacon) management messages that are requested to be transmitted in response to the RFI_BcnInfo.Req message. The LSB of the bitmap represents opcode 000. If bit *i* is set to 1, the RFI_BcnInfo.Ind management message may include the management message with opcode 00i
Reserved	1	[15:12]	Reserved by ITU-T. Shall be set to 0 by the transmitter and shall be ignored by the receiver.

9.4.2.2.2 Response on beacon information message

The MMPL of an RFI_BcnInfo.ind message is described in Table 9-23.

Table 9-23 – Description of the MMPL of the RFI_BcnInfo.ind message

Field	Octets	Bits	Description (Reference)
OPC-Resp	0-1	[11:0]	Bit map of the category 00X (Beacon) opcodes of the management messages that are transmitted in response to the RFI_BcnInfo.Req message. The LSB of the bitmap represents opcode 000. If bit *i* is set to 1, the RFI_BcnInfo.Ind management message shall include the management message with opcode 00i. If bit *i* is set to 0, the RFI_BcnInfo.Ind management message shall not include the management message with opcode 00i. (Note)
Beacon message 1	Variable	Variable	Beacon management message with an opcode corresponding to the first LSB that is set to 1 in the OPC-Resp field
…			
Beacon message n	Variable	Variable	Beacon management message with an opcode corresponding to the first MSB that is set to 1 in the OPC-Resp field
NOTE – When bit *i* was set to 1 in the RFI_BcnInfo.Req and the same bit *i* is set to 0 in the RFI_BcnInfo.Ind sent in response, it indicates that the responding node is unable to comply with the request for information on opcode 00i.			

9.4.2.3 Channel estimation messages

9.4.2.3.1 Transmission parameter request message (TP.req)

The MMPL field of the TP.req message is empty.

9.4.2.3.2 Transmission parameter response message (TP.ind)

The MMPL field of the TP.ind message shall be as presented in Table 9-24.

Table 9-24 – MMPL field of TP.ind message format

Field	Bits		Description (Reference)
	CENELEC, FCC-1	FCC, FCC-2	
Tone mask (TM)	[7:0]	[39:0]	TM recommendation, where the TM field is described in clause 8.2.3.2.3
RS code word size (RSCW)	[8]	[40]	RSCW recommendation, where the RSCW field is described in clause 8.2.3.2.4
CC rate (CCR)	[9]	[41]	CCR recommendation, where the CCR field is described in clause 8.2.3.2.5
Repetitions (REP)	[12:10]	[44:42]	REP recommendation, where the REP field is described in clause 8.2.3.2.6
Interleaving mode (INTM)	[13]	[45]	INTM recommendation, where the INTM field is described in clause 8.2.3.2.7
Modulation (MOD)	[15:14]	[47:46]	MOD recommendation, where the MOD field is described in clause 8.2.3.2.8
Req-LFSN	[23:16]	[55:48]	The value of the LFSN field from the request to which the response corresponds

9.4.3 DLL management and control primitives

The DLL management and control primitives related to all sublayers of the DLL (APC_MGMT, LLC_MGMT, and MAC_MGMT) are defined in clause 5.2.2.3 and shown in Figure 8-1. This section describes these primitives in detail, as presented in Table 9-25.

Table 9-25 – DLL management and control primitives

Category	Primitive	Description
DLL_MGMT.REQ	APC_MGMT.REQ	Requests from the DLL management entity to the APC, LLC, and MAC to apply particular parameters for the transmit frame and prompts for parameters of the received frame
	LLC_MGMT.REQ	
	MAC_MGMT.REQ	
	A_MGMT.REQ	Request from the upper layer management entity to the DLL management entity
DLL_MGMT.CNF	APC_MGMT.CNF	The DLL sublayers (APC, LLC, MAC) confirm to the DLL management entity that the requested parameters are applied for the transmit frame.
	LLC_MGMT.CNF	
	MAC_MGMT.CNF	
	A_MGMT.CNF	The DLL management entity confirms to the upper layer management that the requested parameters are applied.
DLL_MGMT.IND	APC_MGMT.IND	The DLL sublayers (PCS, PMA, PMD) report to the DLL management entity the particular parameters acquired from the transmitted and received frames.
	LLC_MGMT.IND	
	MAC_MGMT.IND	
	A_MGMT.IND	The DLL management entity reports to the upper layer management the particular parameters acquired by the DLL.
DLL_MGMT.RES	APC_MGMT.RES	The DLL management entity acknowledges the parameters it receives from the DLL sublayers.
	LLC_MGMT.RES	
	MAC_MGMT.RES	
	A_MGMT.RES	The upper layer management acknowledges the parameters it receives from the DLL management entity.

9.4.3.1 APC_MGMT primitives

The APC_MGMT primitives are defined in clause B.1.6.2.

9.4.3.2 LLC_MGMT primitives

9.4.3.2.1 LLC_MGMT.REQ

This primitive requests the LLC to use particular parameters for frame processing. The attributes of the primitive are defined in Table 9-26.

Table 9-26 – The attributes of the LLC_MGMT.REQ primitive

Name	Type	Valid range	Description
TX-FrameType	4-bit integer	0_{16}-F_{16}	Indicates the type of the LLC frame to be transmitted (APC type, compressed/uncompressed, sourced from the APC, see clause 9.1.2.1.2.1)
CCMPI	1-bit integer	0, 1	Indicates whether the frame shall be transmitted secured (1) or not (0), see clause 9.1.2.1.2.2 and Table 9-42
RelayMode	1-bit integer	0, 1	Indicates whether the LLC relay function shall be enabled (1) or disabled (0)
LPRI	2-bit integer	00_2-11_2	Indicates the priority of the transmit LLC frame (sourced from the APC)
DA	16-bit or 64-bit integer	See clause B.1.6.2	The final destination address of the transmit LLC frame (sourced from the APC).
S_LLC	16-bit integer	0001_{16}-$05F2_{16}$	The maximum length of an LLC frame segment in bytes (see clause 9.1.2.2.1).
Frame length	16-bit integer	0001_{16}-$05F2_{16}$	Indicate the APDU length or the LCDU length in bytes of the transmit LLC frame
LCDU payload	String of bytes	See clause 9.4.1.1	The content of the management messages to transmit within the LCDU, with the format as detailed in clause 9.4.1.1.
BCNMaxT	32-bit integer	000116-FFFF16	The value of the BCNMaxT field as indicated in MMPL of the BCN_DI.req message, expressed in resolution of 100 ms.

Additional primitives are for further study.

9.4.3.2.2 LLC_MGMT.CNF

This primitive confirms to the DLL management entity that the requested parameters are used by the LLC for frame processing. The attributes of the primitive are as defined in Table 9-27.

If the LLC is unable to comply with a particular attribute in the LLC_MGMT.REQ, it shall set this primitive to one, which means that the request is denied. Otherwise the value of the LLC_MGMT.CNF primitive shall be set to zero.

Table 9-27 – The attributes of the LLC_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0,1	0 – Success 1 – Request is denied

Additional primitives are for further study.

9.4.3.2.3 LLC_MGMT.IND

This primitive provides the DLL management entity with particular parameters of the frames processed by the LLC in the transmit and receive directions. The attributes of the primitive are defined in Table 9-28.

Table 9-28 – The attributes of the LLC_MGMT.IND primitive

Name	Type	Valid range	Description
RX-FrameType	4-bit integer	0_{16}-F_{16}	Indicates the type of the received LLC frame (APC type, compressed/uncompressed, sourced from the APC, see clause 9.1.2.1.2.1)
LPRI	2-bit integer	00_2-11_2	Indicates the priority of the received LLC frame (if needs to be relayed)
SA	16-bit or 64-bit integer	See clause B.1.6.2	The originating source address of the received frame
LCDU payload	String of bytes	See clause 9.4.1.1	The content of the received management messages within the LCDU, with the format as detailed in clause 9.4.1.1

Additional primitives are for further study.

9.4.3.2.4 LLC_MGMT.RES

This primitive is for further study

9.4.3.3 MAC_MGMT primitives

9.4.3.3.1 MAC_MGMT.REQ

This primitive requests the MAC to use particular parameters for frame processing. The attributes of the primitive are defined in Table 9-29.

Table 9-29 – The attributes of the MAC_MGMT.REQ primitive

Name	Type	Valid range	Description
PRI	2-bit integer	00_2-11_2	Indicates the MA priority of the transmit MPDU (based on the LPRI value sourced from the APC)
CW_{Offset} (0)	16-bit	0000_{16}-$03FF_{16}$	Indicates the CW offset of MA priority 0 expressed in TS (see clause 9.2.2.1)
CW_{Offset} (1)	16-bit	0000_{16}-$03FF_{16}$	Indicates the CW offset of MA priority 1 expressed in TS (see clause 9.2.2.1)
CW_{Offset} (2)	16-bit	0000_{16}-$03FF_{16}$	Indicates the CW offset of MA priority 2 expressed in TS (see clause 9.2.2.1)
CW_{Init} (0)	16-bit integer	0000_{16}-$03FF_{16}$	Indicates the initial value of CW (see clause 9.2.2.3) for MA priority 0 expressed in TS
CW_{Init} (1)	16-bit integer	0000_{16}-$03FF_{16}$	Indicates the initial value of CW (see clause 9.2.2.3) for MA priority 1 expressed in TS
CW_{Init} (2)	16-bit integer	0000_{16}-$03FF_{16}$	Indicates the initial value of CW (see clause 9.2.2.3) for MA priority 2 0 expressed in TS
CW_{Init} (3)	16-bit integer	0000_{16}-$03FF_{16}$	Indicates the initial value of CW (see clause 9.2.2.3) for MA priority 3 0 expressed in TS
γ	8-bit integer	00_{16}-FF_{16}	Indicates the value of γ (see clause 9.2.2.3) . Valid only if MA mode = 1, where $\gamma_{max} = FF_{16}$
A	8-bit integer	00_{16}-FF_{16}	Indicates the value of A (see clause 9.2.2.3) multiplied by 10 in steps of 0.1 (i.e., A can be set to 0-25.5). Valid only if MA mode = 1

Table 9-29 – The attributes of the MAC_MGMT.REQ primitive

Name	Type	Valid range	Description
B	8-bit integer	00_{16}-FF_{16}	Indicates the value of B (see clause 9.2.2.3). Valid only if MA mode = 1
$CW_{InActMin}$ (0)	16-bit integer	0000_{16}-0400_{16}	Indicates the value of $CW_{InActMin}$ for MA priority 0 used as described in clause 9.2.2.3 expressed in TS
$CW_{InActMin}$ (1)	16-bit integer	0000_{16}-0400_{16}	Indicates the value of $CW_{InActMin}$ for MA priority 1 used as described in clause 9.2.2.3 expressed in TS
$CW_{InActMin}$ (2)	16-bit integer	0000_{16}-0400_{16}	Indicates the value of $CW_{InActMin}$ for MA priority 2 used as described in clause 9.2.2.3 expressed in TS
$CW_{InActMin}$ (3)	16-bit integer	0000_{16}-0400_{16}	Indicates the value of $CW_{InActMin}$ for MA priority 3 used as described in clause 9.2.2.3 expressed in TS
TCWUpdate	8-bit integer	00_{16}-FF_{16}	Indicates the time period for the potential reduction of the CW for all priorities as described in clause 9.2.2.3, expressed in units of 10 ms
Destination NODE_ID	16-bit integer	See clause 9.3.1.2.2	Short address of the destination node ID (DNID)
Destination DOMAIN_ID	16-bit integer	See clause 9.3.1.2.1	Destination node DOMAIN_ID (DDID)
MPDU type	1-bit integer	0, 1	Indicates the type of the transmit MPDU: 0 – SS-MPDU 1 – MS-MPDU
ACK_TYPE	1-bit integer	0, 1	The ACK type required from the receiver of the transmitted MPDU encoded as detailed in clause 9.1.3.1.1.2
TP-REQ	2-bit integer	00_2-11_2	The type of TP report required from the receiver of the transmitted MPDU encoded as detailed in clause 9.1.3.1.1.10
Virtual carrier sense	Integer	0-1024	Indicates the number of symbols in the payload of the frame sequence during which the medium will be busy (Note 2)
Physical carrier sense	Integer	0-3	Indicates the status of the medium: (physical carrier sense based on preamble detection) 0 – IDLE 1 – BUSY by ITU-T G.9902 transmission 2 – BUSY by non-ITU-T G.9902 transmission 3 – BUSY due to both ITU-T G.9902 and non-ITU-T G.9902 transmission
NOTE 1 – Values of 2 and 3 are only valid if the preamble-based coexistence mechanism is enabled (see clause 5.1.2.3); otherwise the valid values of the primitive are 0 and 1 only. NOTE 2 – Virtual carrier sense is for ITU-T G.9902 frames only. In the case of frame types 2 and 4, this value is provided by the "Virtual Carrier Sense" attribute of the PMI_MGMT.IND primitive. In the case of a type 1 frame, the DLL management system shall derive the number of symbols from the "PHY parameters" attribute of the PMI_MGMT.IND defined in clause 8.8.2.3.3 (the relevant PCS_MGMT,IND attributes: "MPDU size, "Payload Modulation", "Payload Repetitions", etc.).			

Additional primitives are for further study.

9.4.3.3.2 MAC_MGMT.CNF

This primitive confirms to the DLL management entity that the requested parameters are used by the MAC for frame processing. The attributes of the primitive are as defined in Table 9-30.

If the MAC is unable to comply with a particular attribute in the MAC_MGMT.REQ, it shall set this primitive to one, which means that the request is denied. Otherwise the value of the MAC_MGMT.CNF primitive shall be set to zero.

Table 9-30 – The attributes of the MAC_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0,1	0 – Success 1 – Request is denied

Additional primitives are for further study.

9.4.3.3.3 MAC_MGMT.IND

This primitive provides the DLL management entity with particular parameters of the frames processed by the MAC in the transmit and receive directions. The attributes of the primitive are defined in Table 9-31.

Table 9-31 – The attributes of the MAC_MGMT.IND primitive

Name	Type	Valid range	Description
MPDU type	1-bit integer	0, 1	Type of MPDU in the received frame: 0 – SS-MPDU 1 – MS-MPDU
ACK_TYPE	1-bit integer	0,1	Indicates the ACK type required by the received frame: 0 – ACK per frame is required 1 – ACK per LPDU is required (see clause 9.3.3.1)
Source NODE_ID	16-bit integer	See clause 9.3.1.2.2	Short address of the source node ID (SNID)
Source DOMAIN_ID	16-bit integer	See clause 9.3.1.2.1	Source node DOMAIN_ID (SDID)
TP-REQ	2-bit integer	00_2-11_2	The type of TP report required for the receiver frame, encoded as defined in clause 9.1.3.1.1.10
NOS	6-bit integer	1-64	Number of segments comprising the LLC frame associated with the received MPDU
LSPL	1	[15:10]	Last segment pad length, in bytes
LFSN	2	[7:0]	LLC frame sequence number
PRI	13	[14:13]	MA priority of the MPDU
NEST	8-bit integer	00_{16}-FF_{16}	Indicates the received value of NEST parameter expressed in TS. See clause 9.1.3.1.1 for details

Additional primitives are for further study.

9.4.3.3.4 MAC_MGMT.RES

This primitive is for further study.

9.4.3.4 A_MGMT primitives

The A_MGMT primitives that are specific for a particular type of APC are defined in clause B.1.6.1.

9.4.3.4.1 A_MGMT.REQ

This primitive requests the DLL or PHY management to use particular parameters for associated functions. The attributes of the primitive are defined in Table 9-32.

Table 9-32 – The attributes of the A_MGMT.REQ primitive

Name	Type	Valid range	Description
NewDomainID	16-bit integer	See Table 9-7	The value of the domain ID asserted for the domain by the upper layers of the management system (for user configuration or domain ID resolution)
SecurityLevel	2-bit integer	00_2-11_2	Asserts the security level for domain operation, as defined in Table 9-36 (valid for domain master only): 00 – Non-secure 01 – Low 10 – Medium 11 – High
BeaconMode	2-bit integer	00_2-11_2	Asserts the beacon mode of the domain, as defined in clause 9.3.4 (valid for domain master only): 00 – Beaconless mode 01 – Asynchronous beacon mode 10 – Synchronous beacon mode 11 – Reserved
MaxAckRxTime	8-bit integer	$001A_{16}$-$14FF_{16}$	The value that determines the maximum time to transmit an LLC frame, as defined in clauses 9.3.3.3.1 and 9.1.4. Represented as an unsigned integer, in 10ms units, with the range 250ms-51000ms. The value $FFFF_{16}$ is a special value that indicates that the MaxAckRxTime limitation is disabled.
RelayMode	1-bit integer	0, 1	Indicates whether L2 relaying is enabled in the domain (valid for the domain master only)
MAMode	1-bit integer	0, 1	Indicates whether the centralized medium access mode (see clause 9.2) or distributed medium access mode shall be used (valid for domain master only)

Table 9-32 – The attributes of the A_MGMT.REQ primitive

Name	Type	Valid range	Description
TransmMode	2-bit integer	00_2-11_2	Indicates whether normal or robust transmission shall be used in the domain (valid for domain master only): 00 – Normal length for both the PFH and preamble 10 – Extended length for the PFH and normal for preamble 01 – Normal length for the PFH and extended for preamble 11 – Extended length for both the PFH and preamble
Enable CX mechanism	1-bit integer	0-1	Enables coexistence mechanism (see clause 9.2.4): 0 – Disabled 1 – Enabled
Management message	Byte string	1-992 bytes	Internal management message sourced by upper layers (for transmission using the LCDU)

Additional primitives are for further study.

9.4.3.4.2 A_MGMT.CNF

This primitive confirms the particular parameters used by the DLL or PHY. The attributes of the primitive are as defined in Table 9-33.

If the DLL or PHY is unable to comply with a particular attribute in the A_MGMT.REQ, it shall set this primitive to one, which means that the request is denied. Otherwise the value of the A_MGMT.CNF primitive shall be set to zero.

Table 9-33 – The attributes of the A_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0,1	0 – Success 1 – Request is denied

Additional primitives are for further study.

9.4.3.4.3 A_MGMT.IND

This primitive provides the AE management with management parameters received or derived by the DLL or PHY management and submitted to the A-interface. The attributes of the primitive are defined in Table 9-34.

Table 9-34 – The attributes of the A_MGMT.IND primitive

Name	Type	Valid range	Description
DomainID	16-bit integer	Table 9-7	The value of the domain ID selected by the DLL (self-configuration mode)
DomainIDConflict	1-bit flag	N/A	Indicates that same domain ID detected and change of the domain ID is required
NeighborDomainsList	*m* 16-bit integers	Table 9-7	Values of *m* discovered for neighbour domain IDs, each formatted as defined in Table 8-6
FrameTxFailure	1-bit flag	N/A	Shall be raised by the DLL management entity to indicate that the transmitted LLC frame was dropped due to persistent error or timeout. Details are described in clause 9.3.3.3.1
TxSuspended	Flag and a 16-bit integer	For the 16-bit integer, see Table 9-7	Shall be raised by the DLL management entity to indicate that transmission to the destinations address attached in the 16-bit field is not allowed.
ResourceLimit	1-bit flag	N/A	Shall be raised by the DLL management entity to indicate that the node lacks resources to continue receive and transmit frames.
SecurityLevel	2-bit integer	00_2-11_2	Indicates to the upper layers the security level asserted in the domain, as defined in Table 9-36 (valid for end-nodes only): 00 – Non-secure 01 – Low 10 – Medium 11 – High
BeaconMode	2-bit integer	00_2-11_2	Indicates to the upper layer the beacon mode of the domain, as defined in clause 9.3.4 (valid for end-nodes only): 00 – Beaconless mode 01 – Asynchronous beacon mode 10 – Synchronous beacon mode 11 – Reserved
MaxAckRxTime	8-bit integer	$001A_{16}$-$14FF_{16}$	Indicates to the upper layers the value of the maximum allowed time to transmit an LLC frame, as defined in clauses 9.3.3.3.1 and 9.1.4. Represented as an unsigned integer, in 10ms units, with the range 250 ms-51000 ms. The value FFFF16 is a special value that indicates that the MaxAckRxTime limitation is disabled. Valid for end-nodes only
MAMode	1-bit integer	0, 1	Indicates whether centralized medium access mode (1) or distributed medium access mode (0) shall be used (see clause 9.2.2.3). This parameter is valid for the domain master only.

Table 9-34 – The attributes of the A_MGMT.IND primitive

Name	Type	Valid range	Description
TransmMode	2-bit integer	00_2-11_2	Indicates whether a node shall use normal or robust transmission (valid for end-nodes only): 00 – Normal length for both the PFH and preamble 10 – Extended length for the PFH and normal for preamble 01 – Normal length for the PFH and extended for preamble 11 – Extended length for both the PFH and preamble
Management message	Byte string	1-992 bytes	Internal management message recovered from the received LCDU

Additional primitives are for further study.

9.4.3.4.4 A_MGMT.RES

This primitive is for further study.

9.5 Domain management protocols

9.5.1 Setup of a domain

Domain in considered to be established if, and only if, there is a node operating as a domain master with a particular DOMAIN_ID. Therefore, to establish a domain the following has to be done:

– one of the nodes takes the role of the domain master;

– the domain master selects a DOMAIN_ID for its domain;

– all other nodes intended to be part of the established domain take a role of end-nodes and register into the domain.

Only one node in the domain may take the role of a domain master at any one time, while multiple nodes may be capable of operating as a domain master (DM-capable nodes). A domain master selection protocol defined in clause 9.5.1.1 shall be used to pick a single domain master in the presence of multiple DM-capable nodes.

Each domain is uniquely identified by its domain name which is set by the user or generated autonomously by the domain master that initializes the domain. Setup of the domain name is beyond the scope of this Recommendation. The domain name indicator (DNI) transmitted in the beacon frame indicates the domain name. The format of the DNI is defined in clause 9.5.1.3. Once generated, the DNI shall not change when the role of the domain master is passed to another node.

NOTE – The DNI is similar to SSID in 802.11 networks.

The domain master shall set a DOMAIN_ID for its domain using the DOMAIN_ID selection protocol defined in clause 9.5.1.2; this protocol provides, for a new-established domain, a value of DOMAIN_ID that is different from those in neighbouring domains.

9.5.1.1 Domain master selection protocol

This is for further study.

9.5.1.1.1 Domain master selection at initialization

This is for further study.

9.5.1.1.2 Domain master recovery if no backup domain master is assigned

This is for further study.

9.5.1.2 Domain ID selection protocol

This is for further study.

9.5.1.2.1 Domain ID conflicts detection

This is for further study.

9.5.1.2.2 Domain ID conflicts resolution

This is for further study.

9.5.1.3 Parameters of domain setup procedures

This is for further study.

9.5.2 Admission to the domain

The procedure of admitting a new node to the domain (also called "registration") is for further study.

9.5.3 Backup of the domain master

Assignment of a node to back up the domain master and the domain master backup procedures are for further study.

9.5.4 Channel estimation

The channel estimation procedure allows the transmitter to request for a recommendation from the receiver on the values of at least the following transmission parameters:

– tone mask
– modulation
– repetition
– interleaver mode
– convolution coding rate
– maximum RS code word size.

NOTE – The format of the channel estimation messages presented in clause 9.4.2.3 allows the extending of the list of the parameters, if necessary.

The transmitter may, at its own discretion, request for a recommendation from the receiver using one of the following two methods:

– transmission parameter request message (TP.req), as specified in clause 9.4.2
– setting the TP-REQ indicator on the MPH, as specified in clause 9.1.3.1.1.

Upon receipt of a TP.req message, the receiver shall respond with a TP.ind message (see clause 9.4.2.3, containing its recommendation for transmission parameters (see Table 9-24).

The transmitter may, at its own discretion, set the TP-REQ indicator on the MPH of any regular data or management frame. No special setting for transmission parameters in frames containing the TP-REQ indicator is required.

Upon receipt of a frame with an MPH without error and with the TP.REQ indicator set to 01, the receiver shall respond with a TP.ind message containing its recommendation for transmission parameters in the format defined in Table 9-24.

Upon receipt of a frame with an MPH without error and with the TP.req indicator set to 10, the receiver, if capable, shall respond by using an extended Imm-ACK frame (see clause 9.3.3.1.1) containing its recommendation for transmission parameters as defined in Table 9-24.

Upon receipt of a frame with an MPH without error and with the TP.req indicator set to 11, the receiver shall respond with an Imm-ACK frame containing its recommendation for a partial set of transmission parameters. The format of partial response is defined in Table 9-16.

If the receiver is incapable of providing the report by sending an Extended Imm-ACK frame, it shall reply to the transmitter with an extended Imm-ACK frame indicating "unable to comply", as described in clause 9.3.3. A node that requested a TP-recommendation using an extended Imm-ACK, if necessary, may further send another type of TP request.

The TP.ind message shall always be transmitted using the RCM, with the following transmission parameters:

– code rate: 1/2

– number of repetitions: at least 4.

NOTE 1 – Upon receipt of the TP.ind message from the receiver, the transmitter may use the receiver's recommendation of transmission parameters for subsequent transmissions to that receiver.

NOTE 2 – Using the robust communication mode (RCM) for transmission of a TP.req message improves the performance of channel estimation.

9.5.5 Beacon related management procedures

Beacon related management procedures are for further study.

9.5.6 Relaying database management

Management procedures related to the relaying database are for further study.

9.5.7 Flow control

Flow control procedures are for further study.

9.6 Security

Security is provided by encryption and authentication of the relevant data and management frames communicated between the nodes of the domain. The specified encryption method is based on AES-128 and is described in clause 9.6.1.

Node authentication, generation and distribution of encryption keys between nodes, encryption key updates and node authentication updates are provided by a set of authentication and key management (AKM) procedures, described in clause 9.6.2. The AKM procedure can establish both group keys (i.e., a unique set of keys for a particular group of nodes) and pair-wise keys (i.e., a unique set of keys per every pair of communicating nodes).

9.6.1 Encryption

The specified encryption method is based on the advanced encryption standard (AES) according to [NIST PUB 197] and the counter mode with cipher block chaining message authentication code algorithm (CCM), according to [NIST SP 800-38C]. The encrypted LLC frame is communicated using the CCM protocol (CCMP) that includes a CCMP header, encrypted frame, and a message integrity code (MIC) for frame authentication. The CCMP header includes information necessary to facilitate decryption.

9.6.1.1 Description of CCMP

9.6.1.1.1 CCM encryption

The CCM encryption algorithm complies with the [NIST SP 800-38C].

Prerequisite:

- block-cipher algorithm AES-128, [NIST PUB 197]
- encryption key *K*: 128 bits long
- counter-generation function: produces 128-bit counter blocks (*Ctr*)
- length of the message integrity code (MIC), *Tlen*.

Input:

- nonce *N*: a bit-string of less than 128 bits long
- payload *P* of length *Plen* bytes: the part of the data unit (APDU or LCDU) to be both encrypted and protected by the MIC
- associated data A of length *Alen* bytes: the unencrypted part of the data unit and additional data to be protected by the MIC.

Output:

- ciphertext (encrypted payload) *C*
- MIC of the length *Tlen*.

Steps of the algorithm:

1. Apply the formatting function, as described in clause 9.6.1.1.3 to the input variables *N*, *A*, and *P* to produce the 128-bit blocks $B_0, B_1, \ldots, B_r$.
2. Set $Y_0 = \mathrm{CIPH}_K(B_0)$: apply the block-cipher algorithm with the key *K*.
3. For $i = 1$ to *r*, do $Y_i = \mathrm{CIPH}_K(B_i \oplus Y_i\text{-}1)$: chaining the blocks.
4. Set $T = \mathrm{MSB}_{Tlen}(Y_r)$: the *Tlen* most significant bits of the final round of this computation.

   NOTE – These first 4 steps constitute the cipher-block chaining that calculates the value of T to generate MIC. If the value of T computed for the received *frame* will still match the received value of MIC, it assures authenticity (integrity) of the received message with sufficiently high probability.
5. Generate the counter blocks *Ctr*0, *Ctr*1,…, Ctr_m, where $m = ceiling\ (Plen/128)$.
6. For $j = 0$ to m, do $S_j = \mathrm{CIPH}_K(Ctr_j)$: apply the block-cipher algorithm with the key *K*.
7. Set $S = S_1 \| S_2 \| \ldots \| S_m$: this defines the string of encrypted counter blocks. Note that S_0 is skipped.

Compute $C = (P \oplus \mathrm{MSB}_{Plen}(S)) \| (T \oplus \mathrm{MSB}_{Tlen}(S_0))$: the ciphertext is the string of counter blocks XOR'd with the payload data; the message-authentication code is produced by XOR'ing *T* with S_0.

NOTE – The second 4 steps constitute generation of the actual ciphertext of encrypted data concatenated with the MIC. The associated data, *A*, are not incorporated into the ciphertext *C*, but incorporated in the calculation of the MIC, and thus are protected against undetected alteration. Thus, relevant part of the data that are sent unencrypted, as described in clause 9.6.1.2.1, are included to be a part of the *A*-data.

A block diagram illustrating the CCM encryption and MIC generation algorithm described above for the case of $33 \le Alen \le 48$ bytes is presented in Figure 9-20.

The *B*-blocks from B_3 onwards contain payload bits (*P*) and blocks B_0, B_1, and B_2 contain associated data bits (*A*). The AES-blocks stand for AES-128 functions. Those are fed by 128-bit counter blocks ($Ctr_0 - Ctr_m$). The PAD compliments the last payload block to 128 bits. If *Alen* is longer than 48 bytes or shorter than 32 bytes, the number of blocks in Figure 9-20 containing associated data has to be, respectively increased or decreased. The output (ciphertext) includes 128-bit blocks $C_1 - C_{m-1}$; the last block is shorter by the size of the PAD.

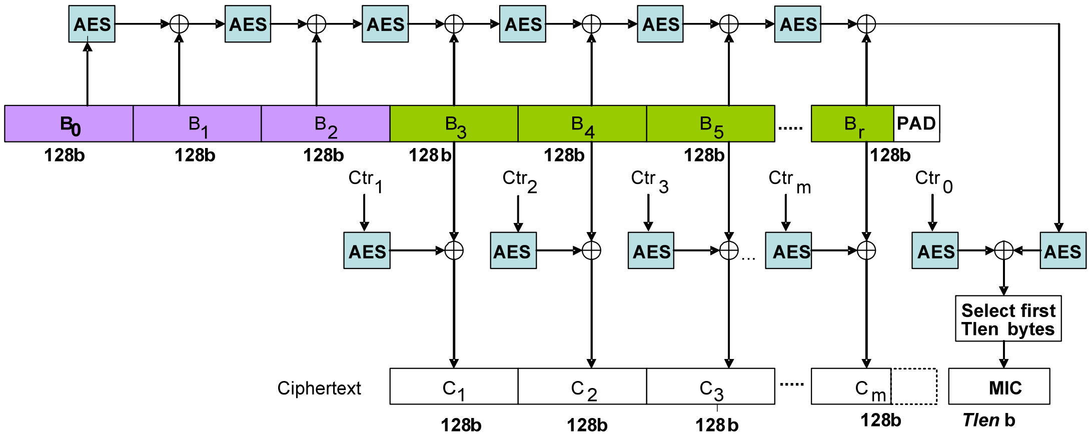


Figure 9-20 – Functional diagram of CCM encryption and message authentication with three blocks of associated data

9.6.1.1.2 Parameters

Valid values of the CCM encryption parameters are presented in Table 9-35.

Table 9-35 – CCM parameters

Parameter	Valid values
MIC size (Tlen), bytes	4, 8, 16
Payload size (Plen), bytes	$\leq (2^{11} - 1)$
Associated data size (Alen), bytes	Variable

NOTE – Selection of MIC size should be based on the guidance provided in [NIST SP 800-38C].

9.6.1.1.3 Input variables

The input variables to support CCM encryption are:

- counter blocks (Ctr_n)
- associated data blocks (B_0, $B_1 - B_M$)
- payload blocks (B_M+1 to B_r)
- encryption key.

The 16-byte counter blocks Ctr_0, Ctr_1…, Ctr_m shall have the format presented in Table 9-36. Each block shall comprise a 1-byte flag, a 13-byte nonce, and a 2-byte counter block number (in the range from 0 to *m*) expressed as a 16-bit binary integer. All bytes of the counter block shall be formatted MSB first: the first bit of the byte 0 is MSB (bit 7) and the last bit of the byte 15 is LSB (bit 0).

Table 9-36 – Format of the *Ctr* blocks

Byte number	0	1, 2 …, 13	14, 15
Contents	Flags (Note)	Nonce	Counter block number
NOTE – The content of the flags byte is: bits [7:6] – reserved by ITU-T for NIST, shall be set to 00 bits [5:3] – shall be set to 000 bits [2:0] – shall be set to 001.			

The 13-byte nonce shall be constructed as presented in Table 9-37. The MSB of byte 0 of the nonce in Table 9-37 shall be mapped to the MSB of byte 1 of the *Ctr* block. The value and format of the frame number (FN) shall be as specified in clause 9.6.1.2, Table 9-41. The LSB of the FN shall be mapped to the LSB of byte 12 of the nonce. The source address field of the nonce shall have a format defined in Table 9-38 which depends on the selected addressing option. All bytes of the nonce shall be formatted MSB first: the first bit of the byte 0 is MSB (bit 7) and the last bit of the byte 12 is LSB (bit 0).

Table 9-37 – Format of the nonce

Byte number	0	1 – 8	9-12
Contents	Flags (Note)	Source address	Frame number (FN)
NOTE – The content of the flags byte is: Bits [7:3] – the same bits of Byte 0 in the CCMP header Bits [1:0] – the LLC frame priority (see clause 9.2.1, clause B.1.3) Bits [2] – reserved by ITU-T. All reserved bits of the Flags byte shall be set to zero.			

Table 9-38 – Format of the nonce source address field

Bytes	MESH header present (MHI = 1)		No MESH header present (MHI = 0)
	V-bit = 0	V-bit = 1	
1-2	SA field of the LFH	SA field of the LFH	SNID field of MPH
3-4		SDID field of the MPH	SDID field of MPH
5-6		0x00	0x00
7-8		0x0000	0x0000

The value of the nonce (for the given key) shall never be the same for different encrypted payloads, and shall always be the same for identical encrypted payloads (e.g., when the APDU or LCDU is retransmitted or relayed). The encryption key shall be changed promptly to avoid repetition of the nonce.

The 16-bytes block B_0 shall have a format as presented in Table 9-39. The length of the encrypted payload in octets (*Plen*) shall be represented as a 16-bit unsigned integer with the LSB mapped to the LSB of byte 15 of B_0.

Table 9-39 – Format of block B0

Byte number	**0**	**1, 2…, 13**	**14-15**
Content	Flags (Note)	Nonce	Length of the payload (*Plen*)
NOTE – The content of the flags byte: Bit [7] – reserved by ITU-T for NIST, shall be set to zero Bit [6] – shall be set to one Bits [5:3] – shall indicate the length of the MIC encoded as: 001 – 4-byte MIC 011 – 8-byte MIC 111 – 16-byte MIC All other values are reserved by ITU-T Bits [2:0] – shall be set to 001.			

Blocks B_1-B_M shall all be 16-byte long, and shall have a format as presented in the Table 9-40. Byte 0 is the first byte of block B_1, byte (M×16-1) is the last byte of block B_M. The number of blocks, M, shall be: M = *ceiling*(*Alen*/16).

The receiver shall derive the value of the *Alen* from the corresponding settings in the LFH (see Table 9-1), the type of the encrypted data (APDU or LCDU) and the APDU format determined by the used application protocol (both parameters indicated in the LPH).

Table 9-40 – Format of blocks B1-BM

Block	**Bytes**	**Contents**
$B_1 – B_M$	0-1	Reserved by ITU-T (Note 1)
	2-V_1	LFH, except the field HopsLft, if present in LFH, shall be set to 0x00; byte 0 of LFH shall be mapped on byte 2 of B_1–B_M block.
	(V_1 + 1) – V_2	MPH fields from byte 0 to byte 10; byte 0 of the MPH shall be mapped to byte V_1+1 and bits [3:0] of byte 0 and all bits of byte 2 shall be set to 0.
	(V_2 + 1) – V_3	Unencrypted part of the APDU or LCDU as defined in clause 9.6.1.2.2. The byte 0 of the unencrypted part shall be mapped on byte (V_2+1).
	(V_3 + 1) – (M × 16 – 1)	Reserved by ITU (Note 1)
NOTE 1 – All reserved bits shall be set to zeros by the transmitter and ignored by the receiver. NOTE 2 – V_2 = V_1 + 15.		

All bytes of the block B0 and the associated data blocks B_1 – B_M shall be formatted MSB first: the first bit of the byte 0 is MSB (bit 7) and the last bit of the last byte of B_M block is LSB (bit 0).

Payload blocks (B_{M+1} to B_r) are 16-byte long and shall contain bytes of the APDU or LCDU to be encrypted (see clause 9.6.1.2.2, Encrypted part of APDU or LCDU). The APDU or LCDU bytes shall be mapped to payload blocks in sequential order, so that the first byte of the APDU or LCDU to be encrypted shall be mapped on byte 0 of B_M+1, the second byte of the payload is mapped on byte 1 of B_{M+1}, the 17-th byte shall be mapped on byte 0 of B_{M+2}, and so on. If the last byte of the payload does not fall on the 15-th byte of B_r, the payload shall be padded to fill the last block by appending zero bytes (0016). All bytes of the payload blocks shall be formatted MSB first: the first bit of byte 0 of block B_{M+1} is MSB (bit 7) and the last bit of the byte 15 of block Br is LSB (bit 0).

The first byte of the ciphertext, C_1, shall be the first byte of the encrypted part of the LLC frame (encrypted APDU or LCDU in Figure 9-5).

The encryption key is 128 bits long and shall be generated and assigned as described in clause 9.6.2.

9.6.1.2 CCM encryption protocol (CCMP)

9.6.1.2.1 Functional description

The functional model of the CCMP is presented in Figure 9-21. The incoming APDU or LCDU is encrypted by the CCM, as described in clauses 9.6.1.1 and 9.6.1.2.2. The LFH is sent unencrypted. The relevant parts of the LFH and MPH, and the unencrypted part of the APDU (or LCDU) are protected by the MIC as a part of "associated data", as defined in Table 9-40.

The key ID, the frame number (FN), and the length of the MIC associated with the encrypted LLC frame are conveyed to the receiver in the CCMP header to assist decryption; the CCMP header is sent unencrypted and described in clause 9.6.1.2.3, but is also protected by the MIC. Construction of the nonce (N) and the associated data is as described in clause 9.6.1.1. Construction of the CCMP header is defined in clause 9.6.1.2.3.

When encryption is not required, the CCMP header can be used to protect the packet order by discarding packets that arrive out of order (e.g., in case of relaying). The packet order protection (POP) settings are defined in Table 9-42.

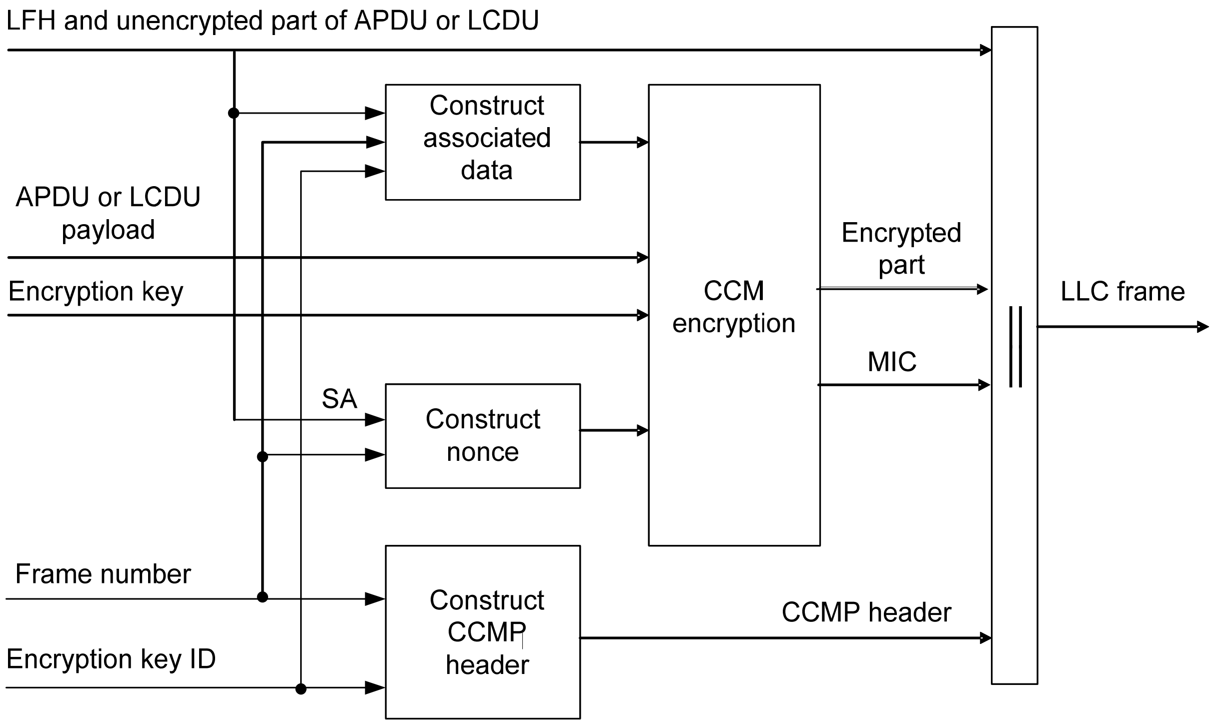


Figure 9-21 – Functional diagram of CCMP encryption

9.6.1.2.2 CCMP encryption format

The format of the encrypted LLC frame is presented in Figure 9-22 (see also Figure 9-5). The encrypted APDU or LCDU consists of four parts: CCMP header, unencrypted part, encrypted part (ciphertext, see clause 9.6.1.1.1), and MIC.

LFH	CCMP header	Unencrypted part of APDU or LCDU	Encrypted part of APDU or LCDU	MIC
Clause 9.1.2.1.2	Clause 9.6.1.2.3	This section	Clause 9.6.1.1.1	Clause 9.6.1.1.1

Figure 9-22 – Format of a CCMP-encrypted LLC frame

The format of the LFH shall be as described in clause 9.1.2.1.2. The format of the CCMP header shall be as described in clause 9.6.1.2.3. Generation of the ciphertext and the MIC shall be as described in clause 9.6.1.1.1.

The unencrypted part of the APDU depends on the type of APC and is described per APC in Annex B. No unencrypted parts shall be present in encrypted LCDU.

9.6.1.2.3 CCMP header

The CCMP header consists of 5 bytes and shall have a format as presented in Table 9-41. It conveys the encryption key identification number (key ID), the length of the MIC, and the security frame number (FN). These three parameters are necessary for decryption and authentication of the received frame.

The length of the MIC shall be selected in the range defined in Table 9-35, according to the procedure defined in clause 9.6.2.

Table 9-41 – CCMP header format

Field	Octet	Bits	Description
Security control	0	[2:0]	Length of the MIC encoded as: 001 – 4-byte MIC 011 – 8-byte MIC 111 – 16-byte MIC A value of 000 indicates POP node (no encryption, see Table 9-42, Note 1) All other values are reserved by ITU-T.
		[4:3]	The type of encryption key: 00 – Node-to-node (NN) key or domain membership key (DMK) 10 – Domain broadcast (DB) key 01 – Node authentication (NA) key 11 – Reserved by ITU-T
		[5]	Encryption key ID: 0 – First key 1 – Second key
		[7:6]	Reserved by ITU-T (Note 2)
Frame number	1-4	[31:0]	32-bit FN, formatted as an unsigned binary integer
NOTE 1 – If bits [2:0] of octet 0 are set to 0, all other bits of octet 0 shall be ignored by the receiver. NOTE 2 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

The format of the key type is a 2-bit unsigned binary integer, intended to identify the specific type of the key used to encrypt the frame. Two encryption keys of each type shall be established, with key IDs 0 and 1 respectively, during the AKM procedure, as described in clause 9.6.2. Keys assigned for communication with different peers may have the same key IDs. Use of encryption keys of different types and different IDs is described in clause 9.6.2.

NOTE – Two keys are necessary to facilitate the key update procedure without interruption of the connection: while the pool of valuable FN values is expired for one key, the transmitting nodes change the key to the second one, allowing for the first key to get updated. The details are described in clause 9.6.2.

The FN is a serial number of the encrypted frame and shall be represented as a 32-bit unsigned binary integer (LSB mapped on bit 0 of byte 4). The FN shall be set to its initial value when a new encryption key is established and increased by 1 with every encrypted frame passed using this key. In case of POP, FN shall be increased by 1 with every frame passed and wrap around to value 1. The FN shall never be repeated for the same value of the key: the key shall be changed prior to the FN reaching its maximum value. The default initial value is 1.

9.6.1.3 Requirements for the receiver

The receiver shall decrypt the received message and compute the MIC using the encryption key for which the type and ID is indicated in the CCMP header. If the computed value of the MIC does not match the value of the MIC in the received frame, the receiver shall discard the frame.

If the receiver detects that encryption rules in the received frame are violated, it shall discard the frame. Those violations are:

- The FN is 0 or is smaller than or equal to the FN values received in previous LLC frames of the same priority from the same originating node with no change of the encryption key.
- The length of the MIC does not fit the security level assigned in the domain.

NOTE – The LLC frame priority is equal to the value of the PRI field communicated in the MPH of the first MPDU carrying segments of the LLC frame (this PRI value is the minimal value of PRI over all MPDUs carrying segments of the LLC frame, and is equal to the value of the LPRI field of the MESH header, if present).

9.6.1.4 Configuration of security level

The valid security levels and corresponding parameters are defined in Table 9-42.

Table 9-42 – Valid security levels

Security level	Encryption	MIC, bytes
High	Yes	16
Medium	Yes	8
Low	Yes	4
POP	No	0

All nodes of the domain shall be set to the same level of security, i.e., encryption and the size of the MIC. The required level of security in the domain is determined by the user via the A_MGMT.REQ primitive SecurityLevel (see clause 9.4.3.4.1). Further, the asserted security level is conveyed by the domain master to every node in the domain as defined in clause 9.3.4, clause 9.4.2.1, clause 9.5.5). Nodes indicate the adopted security level to the upper layers using A_MGMT.IND primitive SecurityLevel (see 9.4.3.4.3). Nodes shall discard all frames for which the security level is below the one required in the domain.

9.6.2 Authentication and key management procedures

For further study.

Annex A

Extremely robust mode

(This annex forms an integral part of this Recommendation.)

This annex specifies procedures and extensions to the main body of this Recommendation which are required for operation with the extremely robust mode (ERM).

NOTE 1 – The implementation of this annex requires mandatory support for the main body of this Recommendation.

NOTE 2 – This annex may be used in companion with the main body of this Recommendation to support highly robust communications under harsh conditions.

NOTE 3 – Implementation of this annex allows operation at the SNR without any need for any special MAC management protocols for synchronizing time or frequency bands between the transmitter and the receiver.

NOTE 4 – One potential application scenario is a channel that includes MV/LV transformer(s), although this annex does not preclude other methods that provide a solution for the same or similar application scenarios.

NOTE 5 – This annex specifies a default set of parameters to support ERM. Management procedures required to change the ERM default parameters are for further study.

NOTE 6 – Additional management functions related to the ERM are for further study.

A.1 Use of PFH fields in ERM

The representation of the PFH fields in ERM shall be the same as the main body of this Recommendation, except for the following fields:

1. The value of the duration (FL) field (see clause 8.2.3.2.2) shall be in units of 10 times K_{Dur} OFDM symbols.
2. An extended set of valid values of the repetitions (REP) field (see clause 8.2.3.2.6) shall be used, as described in Table A.1.

Table A.1 – Extensions to the REP field

REP field value	R parameter of the FRE
101	32
110	64
111	128

A.2 ERM extensions to PMA functionality

1. An extended set of valid values of R defined in Table A.1 shall be supported by FRE. Other values are for further study.
2. The default number of symbols in the PFH used in ERM shall be as specified in Table A.2. Other values are for further study.

Table A.2 – Default number of symbols in encoded PFH

Bandplan	Number of symbols, N_{SH}
CENELEC A	200
CENELEC B	400
CENELEC CD	600
FCC	160
FCC-1	200
FCC-2	210

A.3 ERM extensions of PMD functionality

1. A predefined BAT type 7 shall be specified for uniform 1-bit loading on all subcarriers except the PMSC and PSC sets.
2. The tone mapping for PFH shall use a uniform loading of 1 bit per subcarrier on all subcarriers except the PMSC set and PSC set (BAT type 7).
3. The tone mapping for RCM payload transmission in ERM shall use a uniform bit loading of 1 bit per subcarrier (BAT type 5 or BAT type 7).

A.3.1 ERM preamble

In ERM a node shall use the ERM preamble defined in this section.

A.3.1.1 General preamble structure

Table A.3 describes the general preamble structure for ERM.

Table A.3 – Structure of the ERM preamble

Parameter	1st section	2nd section
Number of symbols (N_1)	N_1 (Note 1)	N_2 (Note 3)
Subcarrier spacing	F_{SC}	F_{SC}
Type of symbol (S_1)	S_1 (Note 2)	S_2 (Note 4)

NOTE 1 – The upper and lower limit to the value of N_1 depends on the bandplan. Default upper and lower limit values for N_1 are specified in Table A.4. Other valid upper and lower limit values of N_1 are for further study. The value of N_1 (from its lower limit and up to its upper limit) to be used is determined by the PMD_MGMT.REQ primitive.

NOTE 2 – The same symbol S_1 as specified in the main body of this Recommendation is used for ERM.

NOTE 3 – The default value of N_2 is 30. Other values of N_2 are for further study. The value of N_2 to be used is determined by the PMD_MGMT.REQ primitive.

NOTE 4 – The n-th OFDM S_2 symbol of the 2nd section, in frequency domain shall be: $e^{2\pi j \cdot \varphi_n} \cdot S_3$, where S_3 is a frequency domain QPSK modulated OFDM symbol. S_3 is generated in frequency domain as specified in clause A.3.1.2, and the default phase values of φ_n shall be $\{\varphi_n, n=1,2,...,30\}$=[0, 0, 2, 3, 2, 1, 26, 25, 2, 4, 13, 18, 20, 11, 6, 28, 24, 6, 26, 16, 6, 13, 0, 23, 8, 21, 31, 13, 27, 6] / 32. Other phase values of φ_n are for further study. The φ_n to be used is determined by the PMD_MGMT.REQ primitive.

Table A.4 – The default upper and lower limit values of N1 per bandplan

Bandplan	Default lower limit values of N_1	Default upper limit values of N_1
CENELEC A	100	200
CENELEC B	200	400
CENELEC CD	300	600
FCC	50	100
FCC-1	100	200
FCC-2	65	130

A.3.1.2 Frequency-domain preamble symbol generation

The signal S_3 is generated using the same procedure as the signal S_1 (see clause 8.4.5.2.1), where the difference is in the value of the PRBS seed as specified in Table A.5.

Table A.5 – Default seed value of S_3

Bandplan	Default seed value of S_3
CENELEC A	25_{16}
CENELEC B	19_{16}
CENELEC CD	05_{16}
FCC	$4A_{16}$
FCC-1	$6C_{16}$
FCC-2	63_{16}

Other values of the seed are for further study.

A.3.1.3 Time-domain preamble symbol generation

To form a section of a preamble, the output preamble symbol shall be repeated NI times.

The first and second sections of the preamble shall be windowed, overlapped and added as described below:

First section:

a. The first symbol of the first section is cyclically extended by prepending the last β/2 samples of the symbol S_1.

b. The last symbol of the first section is cyclically extended by appending the first β/2 samples of the symbol S_1.

c. The first and last β samples of the extended first section are windowed with a window function $w_\beta(n)$ and $w_\beta(\beta-n-1)$ respectively.

Second section:

a. Each $S_2(i)$ symbol of the second section is cyclically extended by prepending the last β/2 samples of the symbol $S_2(i)$ and further cyclically extended by appending the first β/2 samples of the symbol $S_2(i)$.

b. The first and last β samples of each $S_2(i)$ symbol of the second section are windowed with a window function $w_\beta(n)$ and $w_\beta(\beta-n-1)$ respectively.

Overlap and add:

a. The β windowed samples at the end of the first section and at the beginning of the second section are overlapped and added.

b. The β windowed samples at the end of the first N_2-1 $S_2(i)$ symbols of the second section are overlapped and added with the β windowed samples at the beginning of the next $S_2(i+1)$ symbol of the second section.

c. The β windowed samples at the end of the second section are overlapped and added with the β windowed samples at the beginning of the PFH as described in clause 8.4.4.4.

The window shaping function $w_β(n)$ shall comply with the rules specified in clause 8.4.4.4.

Assembling of the OFDM symbols in the ERM preamble is illustrated in Figure A.1.

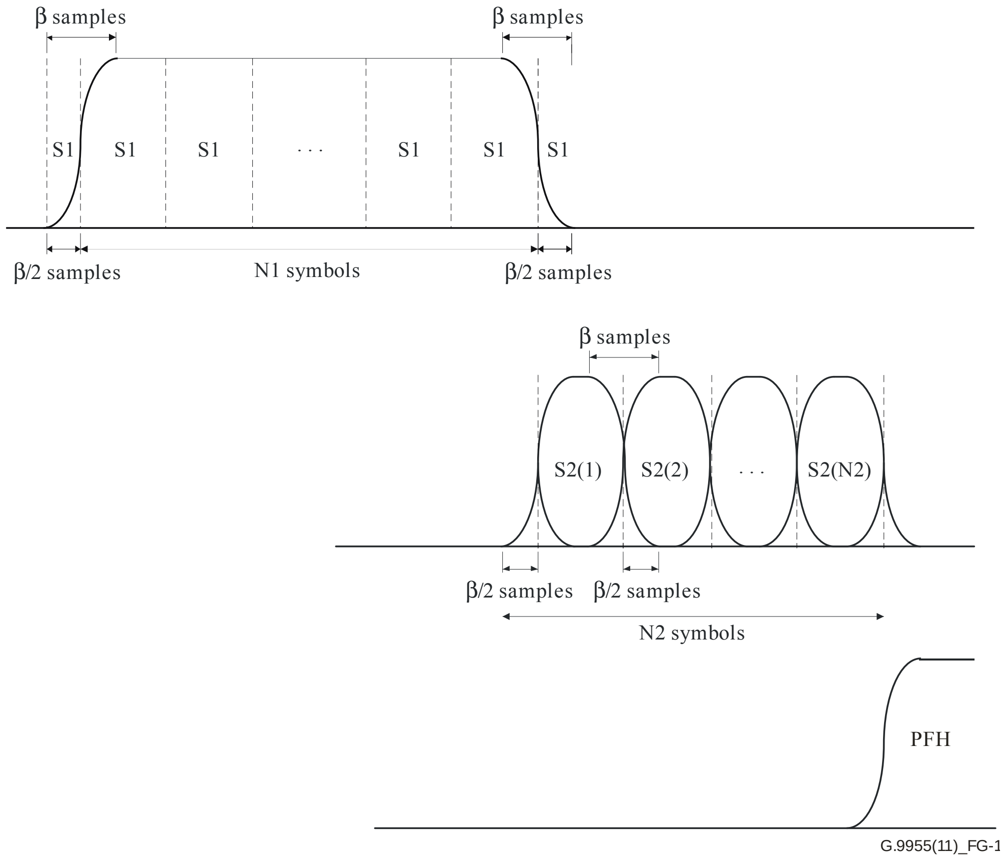


Figure A.1 – Time-domain preamble symbol generation for ERM

The total number N_{pr} of samples in the ERM preamble can be computed as:

$$N_{pr} = \beta + N_1 \times N + N_2 \times N = \beta + N \times (N_1 + N_2)$$

A.3.1.4 CES symbols

In ERM, no CES symbols shall be transmitted.

Annex B

Application protocol convergence sublayer

(This annex forms an integral part of this Recommendation.)

Application protocol convergence specific sublayer (APC) maps the primitives of the application protocol used by the AE into the native protocol of the data link layer. It is the responsibility of the APC to convert incoming data units of the particular application protocol used by the AE into APDUs.

By default, the APC shall support an interface to IPv6 while interfaces to other protocols can also be supported. The APC supporting IPv6 is denoted as IP6-APC. If more than one APC is associated with the particular AE, the bridging function between APCs associated with the AE shall be implemented by the AE and is beyond the scope of this standard.

Each of the APCs generates an APDU based on its particular application protocol that determines the content of the APDU, as described in this annex. At the receive side, the type of the APC that generated a particular APDU is indicated in the LLC frame header (LLCFT field of the LFH). The APC priority and other detailed description of multiple APC support is for further study.

The DLL management primitives related to the APC (APC_MGMT) may depend on the application protocol, as defined in this annex, while the set of management primitives related to the lower sublayers of the ITU-T G.9902 transceiver is unified.

The description of each APC in this annex is partitioned into data plane and management plane. The data plane part specifies conversion of the AE data units into APDUs and back. The management plane part specifies APC address resolution, classification, and other primitives and protocols related to supporting APC peer-to-peer management services.

B.1 IP6-APC

The IP6-APC is intended to operate with the IPv6 AE and supports transmission of IPv6 datagrams over the medium using ITU-T G.9902 as the underlying L1/L2 layers technology.

In the transmit direction, IP6-APC converts the standard set of IPv6 primitives of the incoming ADP at the A-interface into an APDU, which is further communicated through the ITU-T G.9902 domain(s) to the peer(s) IP6-APC.

At the receiver direction, the IP6-APC converts the APDU received from the LLC into a standard primitive set of an IPv6 datagram0. The IPv6-to-APDU frame conversion shall be as described in clause B.1.1. The IP6-APC shall support address configuration methods, as specified in clause B.1.2.

B.1.1 Frame conversion

The incoming set of primitives (AIF_DATA.REQ) and the outgoing set of primitives (AIF_DATA.IND) at the A-interface of IP6-APC represent an IPv6 datagram submitted by the AE to the APC (for transmission over the medium) and submitted by the APC to the AE (after it has been received from the medium), respectively. Each datagram is defined as a set of IPv6 primitives specified by [IETF RFC 2460] of IPv6_datagram.request and IPv6_datagram.indication, respectively – see Tables B1 and B.2.

Table B.1 – A-interface primitives

AIF_DATA.REQ (AE → IP6APC)	AIF_DATA.IND (IP6APC → AE)
IPv6_datagram.request (version traffic class flow label payload length next header hop limit source address destination address data_payload)	IPv6_datagram.indication (version traffic class flow label payload length next header hop limit source address destination address data_payload)

Table B.2 – A-interface primitives description (IETF RFC 2460)

Primitive name	Description
version	4-bit Internet Protocol version number = 6
traffic class	8-bit traffic class field
flow label	20-bit flow label
payload length	16-bit integer – length of the data_payload in octets
next header	8-bit selector – identifies the type of the header immediately following the IPv6 header
hop limit	8-bit unsigned integer – decreased by 1 by each node that forwards the datagram (at L3)
source address	The IPv6 datagram source address in a standard 16-octet format
destination address	The IPv6 datagram destination address in a standard 16-octet format (Note)
data_payload	The IPv6 datagram payload to carry from the source address to the destination address
NOTE – The IPv6 neighbour discovery protocol [IETF RFC 4861], provides a link layer final destination address in a format that meets the underlying data link layer, as defined in clause 8 of [IETF RFC 4944]. For ITU-T G.9902, short and extended format of link layer address are defined, see clause 9.3.1.	

The primitives described in Table B.1 for AIF_DATA.REQ shall be converted into the APDU format presented in Figure B.1, containing a header and a payload. The Table B.1 includes only primitives of the IPv6 header (as per clause 3 of [IETF RFC 2460]), while IPv6 header extensions, if present, are considered as a part of the data_payload primitive. The total length of the APDU shall not exceed 1522 octets.

Header		LSB MSB
	≤ 40 octets	IPv6 header of the incoming datagram, uncompressed or compressed using one of the valid compression methods described in Annex C (Note 1)
Payload	≤ 1482 octets	The *data_payload* primitive of the incoming IPv6 datagram (Note 2)
NOTE 1 – The IPv6 header format is defined in clause 3 of [IETF RFC 2460] and shall be used as a default (uncompressed header). The valid compression methods are defined in Annex C, which can be used in conjunction with this annex. The used compression method is indicated in the LLCFT field of LFH (see clause 8.3.1.2). NOTE 2 – For extension headers (e.g., for UDP header), if present, only the compression options associated with the applied IPv6 header compression method defined in Annex C shall be used.		

Figure B.1 – IP6-APC APDU format

Bits of APDU shall be transmitted towards the LLC (x1 reference point) starting from the first octet of the header. The LSB shall be transmitted first.

The order of outgoing APDUs at the x1 reference point associated with a particular destination and particular user priority shall be the same as for the order of incoming IPv6 datagrams of these same user priority and destination. No re-ordering inside the same user priority group for the same destination is allowed.

The receive path of the IP6-APC shall decompress the header and the payload of the APDU received from the x1 reference point, if compression was applied, as defined in Annex C, and reconstruct the original datagram primitives required for the AIF_DATA.IND (see Table B.1).

B.1.2 Address assignment and address resolution

B.1.2.1 Stateless address auto-configuration

An ITU-T G.9902 node is assigned with a short address (16-bit NODE_ID) upon its registration into the domain. In the case of the L2 registration procedure, defined in clause 8.6.3, this short address is communicated to the AE via the A-interface (A_MGMT.IND primitive, clause 8.4.3.4.3). Moreover, the AE shall configure its IPv6 interface identifier of the link local address (local scope) based on the received node address, its IPv6 interface identifier of the link local address (global scope) based on the node EUI-64 address according to clause 3.2.2 of [IETF RFC 6282]. In the case of the L3 registration procedure, the IPv6 link local address and the corresponding short address are generated by the applied IPv6-based protocol, which is beyond the scope of this Recommendation. These addresses shall be communicated to the DLL via the A-interface (A_MGMT.REQ primitive, clause B.1.6.1.1).

B.1.2.2 Stateful address assignment

With stateful assignment, the IPv6 link local address is assigned independently from the node short address (16-bit NODE_ID, 16-bit DOMAIN_ID) or extended address. Thus, there is no generic rule to derive a short address or extended address from IPv6 address and vice versa. The address resolution shall be performed as defined in B.1.2.3.

B.1.2.3 Address resolution function (ARF)

The IPv6 neighbour discovery protocol [IETF RFC 4861], provides a link layer final destination address of a node corresponding to the destination IPv6 address, thus providing the address resolution.

In the transmit direction, the short address and extended destination addresses are passed from the AE to the DLL via the AIF_DATA.REQ primitive defined in clause B.1.1.

The APC shall pass the resolved 16-bit or 64-bit DA (unicast, multicast, or broadcast) as a control primitive IP6-APC_MGMT.IND to the LLC via an x1 reference point (see clause B.1.6.2.3). At the destination node, short or extended addresses recovered from the received APDUs (DA, SA) are passed to the APC from the LLC via an x1 reference point as a control primitive IP6-APC_MGMT.REQ (see clause B.1.6.2.1).

In the case where the assignment of IPv6 address is by using stateless auto-configuration (see clause B.1.2.1), the IPv6 interface identifier of the link local address can be derived from the 16-bit NODE_ID of a node or from the EUI-64 address of a node as defined in clause 3.2.2 of [IETF RFC 6282].

B.1.3 Classification

IP6-APC may classify the outgoing APDUs based on the following AIF_DATA.REQ primitives:

- traffic class
- flow label
- destination address
- source address
- special management primitives (e.g., emergency indicators) associated with the AE.

As a result of classification, the outgoing APDU is associated with a particular user priority: 0, 1, 2, or 3. The presented criteria indicate possible classification options; the rules of how to classify APDUs, assign a priority, and associate it with a particular user priority, except for the restrictions presented below, depend on the used L3 protocol and are beyond the scope of this Recommendation. The restrictions are:

1. Priority 3 shall be assigned only to APDUs that carry emergency primitives. Definition of these emergency primitives is for further study.
2. The number of user priorities assigned by the APC shall not exceed the number of MA priority queues supported by the node (which depends on the profile).
3. The APDUs generated from ADPs with no or irrelevant QoS requirements shall be assigned priority 0.

Other criteria and restrictions for classification of the APDUs are beyond the scope of this Recommendation.

B.1.4 Flow control

Flow control at the A-interface is necessary to avoid packet loss for when the traffic generated by the source AE exceeds the throughput of the link between the source and the destination nodes. The flow control may be implemented by communicating an appropriate set of AIF_DATA.IND primitives from the APC to AE (e.g., causing a pause in the stream of incoming datagrams) or a set of AIF_DATA.CNF primitives, or by appropriate signalling at the management plane. The format of AIF_DATA.CNF and signalling used for flow control is determined by the vendor.

B.1.5 Encryption

In case the LLC frame carrying an APDU shall be encrypted, the header of the APDU shall be left unencrypted ("Unencrypted part of the APDU" in Figure 9-22), while the payload of the APDU shall be encrypted.

B.1.6 Management plane

B.1.6.1 A_MGMT primitives

The DLL management entity supports the operation of IP6-APC using a set of management primitives A_MGMT presented in Table B.3 (A-reference point).

Table B.3 – IP6-APC A_MGMT primitives

Primitive	Direction	Description
A_MGMT.REQ	AE → DLL	Management request from AE to DLL
A_MGMT.CNF	DLL → AE	Confirmation from DLL to AE on A_MGMT.REQ
A_MGMT.IND	DLL → AE	Management parameters from DLL to AE

B.1.6.1.1 A_MGMT.REQ

This primitive requests the DLL or PHY management to use particular parameters for associated functions. The attributes of the primitive are defined in Table B.4.

Table B.4 – The attributes of the A_MGMT.REQ primitive

Name	Type	Valid range	Description

Table left blank intentionally. Additional primitives are for further study.

B.1.6.1.2 A_MGMT.CNF

This primitive confirms the particular parameters used by the DLL or PHY. The attributes of the primitive are as defined in Table B.5.

If the DLL or PHY is unable to comply with a particular attribute in the A_MGMT.REQ, it shall set this primitive to one, which means that the request is denied. Otherwise the value of the A_MGMT.CNF primitive shall be set to zero.

Table B.5 – The attributes of the A_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0,1	0 – Success 1 – Request is denied

B.1.6.1.3 A_MGMT.IND

This primitive provides the AE management with management parameters received or derived by the DLL or PHY management and submitted to the A-interface. The attributes of the primitive are defined in Table B.6.

Table B.6 – The attributes of the A_MGMT.IND primitive

Name	Type	Valid range	Description
NODE_ID	16-bit integer	See Table 8-6	NODE_ID of the node obtained during the registration to the domain
Extended address	64-bit integer	See EUI-64™	EUI-64 address of the node (assigned by the manufacturer)
L3 routing primitives	N/A	N/A	A set of primitives to support IPv6 L3 routing. The nomenclature and format is for further study.
Performance primitives	N/A	N/A	A set of L1 and L2 performance primitives to support IPv6 L3 routing. The nomenclature and primitives are for further study.

Additional primitives are for further study.

B.1.6.2 IP6-APC_MGMT primitives

B.1.6.2.1 IP6-APC_MGMT.REQ

This primitive requests the APC to use particular parameters for associated functions. The attributes of the primitive are defined in Table B.7.

Table B.7 – The attributes of the IP6-APC_MGMT.REQ primitive

<table>
<tr><th>Name</th><th>Type</th><th>Valid range</th><th>Description</th></tr>
<tr><td rowspan="2">SA (Note 1)</td><td>16-bit integer</td><td>See Table 8-12</td><td>Short address (NODE_ID) of the node that is an originating source address of the received APDU</td></tr>
<tr><td>64-bit integer</td><td>See EUI-64™</td><td>Extended address (EUI-64) of the node that is an originating source address of the received APDU</td></tr>
<tr><td>SA-TYPE (Note 2)</td><td>1-bit integer</td><td>0, 1</td><td>0 – SA of the received APDU is represented in extended address format
1 – SA of the received APDU is represented in short address format</td></tr>
<tr><td>RX-FrameType</td><td>4-bit integer</td><td>0010-1111</td><td>Indicates the format of the received APDU (APC type, uncompressed or compressed), with coding as defined in Table 8-2.</td></tr>
<tr><td colspan="4">NOTE 1 – Only one of the representations shall be available.
NOTE 2 – If the LFH of the received LLC frame carrying the APDU includes a mesh header (see clause 8.1.2.1.2), the value and the format of the SA shall be the one indicated in the mesh header. Otherwise, the SA shall be a short address indicated in the SNID field of the MPH of the MPDUs conveyed by the segments of the received LLC frame (see clause 8.1.3.1).</td></tr>
</table>

Additional primitives are for further study.

NOTE – If the LLC frame is conveyed using more than one MPDU, the MPH of these MPDUs have identical SNID fields and identical DNID fields.

B.1.6.2.2 IP6-APC_MGMT.CNF

This primitive confirms the particular parameters used by the APC. The attributes of the primitive are as defined in Table B.8.

If the APC is unable to comply with a particular attribute in the IP6-APC_MGMT.REQ, it shall set this primitive to one, which means that the request is denied. Otherwise the value of the IP6-APC_MGMT.CNF primitive shall be set to zero.

Table B.8 – The attributes of the IP6-APC_MGMT.CNF primitive

Name	Type	Valid range	Description
Status	Integer	0, 1	0 – Success 1 – Request is denied

Additional primitives are for further study.

B.1.6.2.3 IP6-APC_MGMT.IND

This primitive provides the DLL management with parameters of the datagram received by the IP6-APC from the A-interface. The attributes of the primitive are defined in Table B.9.

Table B.9 – The attributes of the IP6-APC_MGMT.IND primitive

Name	Type	Valid range	Description
LPRI	2-bit integer	00_2-11_2	User priority of the APDU.
DA (Note 1)	16-bit integer	See Table 9-12	Short address (NODE_ID) of the node that is a final destination address of the transmitted APDU.
	64-bit integer	See [EUI-64]	Extended address (EUI-64) of the node that is a final destination address of the transmitted APDU.
DA-TYPE (Note 2)	1-bit integer	0, 1	0 – DA of the transmitted APDU to be represented in extended address format. 1 – DA of the transmitted APDU to be represented in short address format.
TX-FrameType	4-bit integer	0010_2-1111_2	Indicates the format APDU shall be transmitted (APC type, uncompressed or compressed) with coding as defined in Table 9-3.
APDU length	16-bit integer	0001_{16}-$05F2_{16}$	Indicate the APDU length in bytes.
NOTE 1 – Only one of the representations shall be available. NOTE 2 – If the LFH of the transmitted LLC frame carrying the APDU includes a mesh header (see clause 9.1.2.1.2), the value and the format of the DA shall be one indicated in the mesh header. Otherwise, the DA shall be a short address indicated in the DNID field of the MPH of the MPDUs conveyed the segments of the received LLC frame (see clause 9.1.3.1).			

Additional primitives are for further study.

B.1.6.2.4 IP6-APC_MGMT.RES

This primitive is for further study.

Annex C

IP6 header compression methods

(This annex forms an integral part of this Recommendation.)

This annex describes valid compression methods for IPv6 header and header extensions. It is intended to be used in conjunction with clause B.1.

NOTE 1 – Some of the IPv6 compression methods defined in this annex are not transparent to some of AIF_DATA primitives (e.g., traffic class, flow labels).

The default compression method shall use the APDU format that includes a compressed IPv6 header and non-compressed APDU payload. The format of the APDU compressed header shall be as presented in Table C.1.

Table C.1 – APDU header – description

IPv6 header field	Description, [IETF RFC 2460]	Compressed format
version	4-bit Internet Protocol version number = 6	0 bits
traffic class	8-bit traffic class field	0 bits
flow label	20-bit flow label	0 bits
payload length	16-bit integer – length of the data_payload in octets	0 bits
next header	8-bit selector – identifies the type of the header immediately following the IPv6 header.	Octet 0, original format
hop limit	8-bit unsigned integer – decreased by 1 by each node that forwards the datagram (at L3).	Octet 1, original format
source address	The IPv6 datagram source address in a standard 16-octet format.	0 bits (Note 1)
destination address	The IPv6 datagram destination address in a standard 16-octet format (Note 1)	Unicast: 0 bits Multicast: octets 2-17 original format (Note 2)
NOTE 1 – The IID (IPv6 interface identifier) shall be recovered by the receiver based on the IPv6 address association with L2 addresses. The IPv6 address prefix shall be set to the value of the receiver's address prefix. NOTE 2 – For unicast, the address shall be recovered as defined in Note 1. For multicast, the transmitter shall use the original format of the destination addresses.		

NOTE 2 – The value of payload length field of the IPv6 header is communicated using parameters of the MPH: LPDU-L, NOS, and LSPL. The receiver uses these parameters to compute the size of the received APDU as defined in clauses 9.1.2.2 and 9.1.3.1.

Besides default compression, the IPv6 datagram may be either sent uncompressed or, if applicable, use one of the alternative valid compression options described in Table C.2. Support of valid compression methods is optional, thus one can be used only in case this compression method is supported by the peer IP6-APC.

NOTE 3 – The default compression is only suitable to L3 networks where all nodes have the same prefix.

Table C.2 – Alternative valid compression methods

Compression method	LFH indication code	Description
HC1 and HC2 of IETF RFC 4944	4	Compression as per clause 10 of [IETF RFC 4944].
LOWPAN_IPHC and LOWPAN_NHC of [IETF RFC 6282]	5	Compression as per [IETF RFC 6282].
NOTE 1 – For extension headers, if present, only the compression options defined by the applied IPv6 header compression method shall be used. For instance, with the IPV6 compression method defined in clause 10.1 of [IETF RFC 4944], the UDP header, if present, may be compressed as defined in clause 10.2 of [IETF RFC 4944]. NOTE 2 – In [IETF RFC 6282] and clause 10 of [IETF RFC 4944], it is required to use the "Frame Length" field of the IEEE 802.15.4 to recover the length of the PPDU. For an equivalent computation in the case of ITU-T G.9902, to recover the length of the APDU it is required to use the relevant fields of MPH. In ITU-T G.9902 the 6LoWPAN fragmentation header is not used.		

Other compression methods are for further study.

Annex D

Examples and use cases of ITU-T G.9902 network topologies

(This annex forms an integral part of this Recommendation.)

D.1 Examples of UAN topologies and deployments scenarios

D.1.1 EM-UAN network deployment examples

The UAN domains addressed by this Recommendation may be established over low voltage (LV) and medium voltage (MV) legs of a power line distribution. The domains established over LV-legs are often associated with a transformer between MV and LV lines (MV-LV transformer). The head-end (domain master) is typically residing at the MV-LV transformer and end-nodes are at the CPs, connected to the LV line. One example of a service area associated with this type of UAN domain is presented in Figure D.1 and shows a sample of AMI/AMM installation within a neighbourhood. Each black square represents a residence meter, and each residence may potentially also include an EM-HAN.

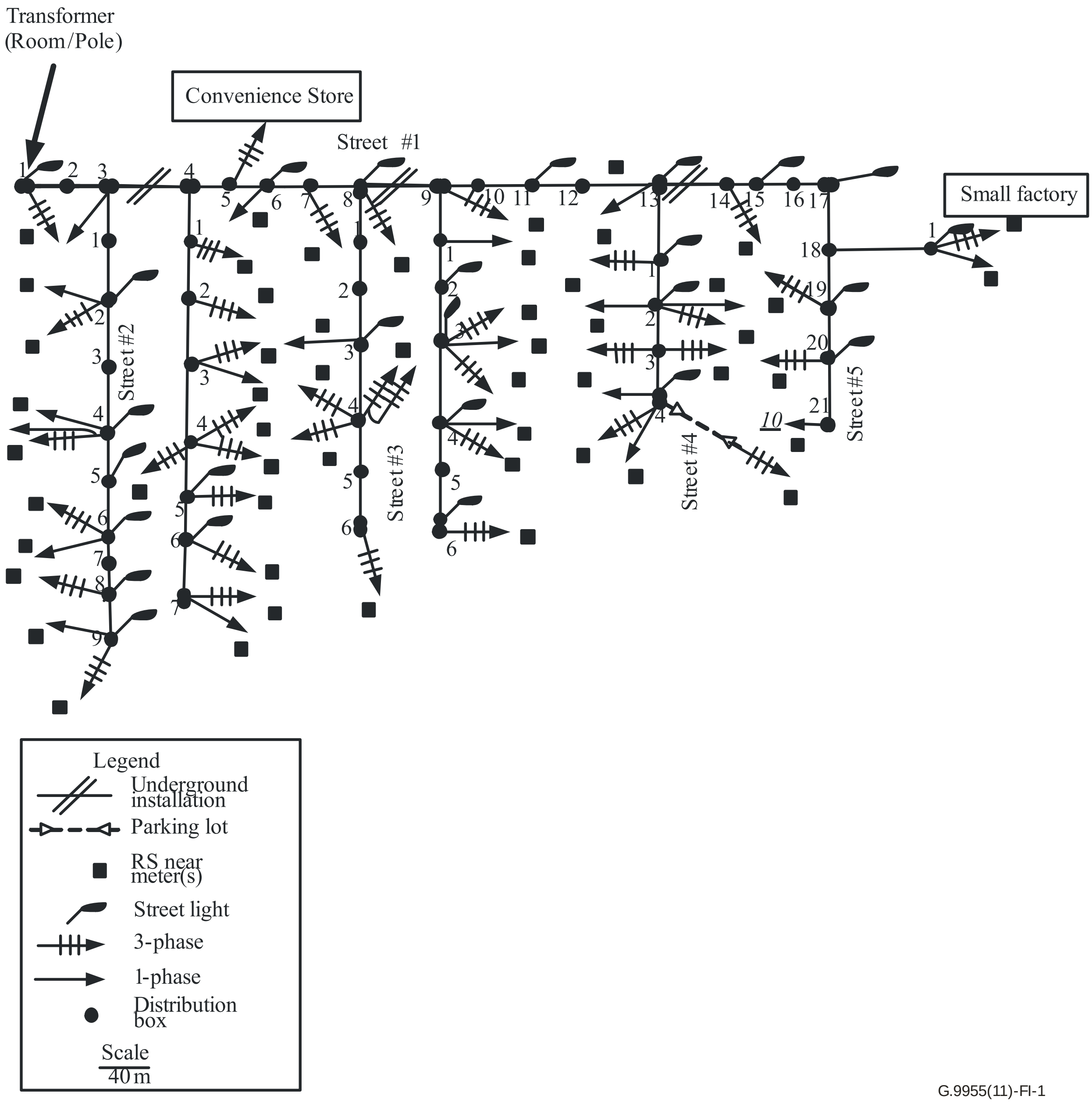
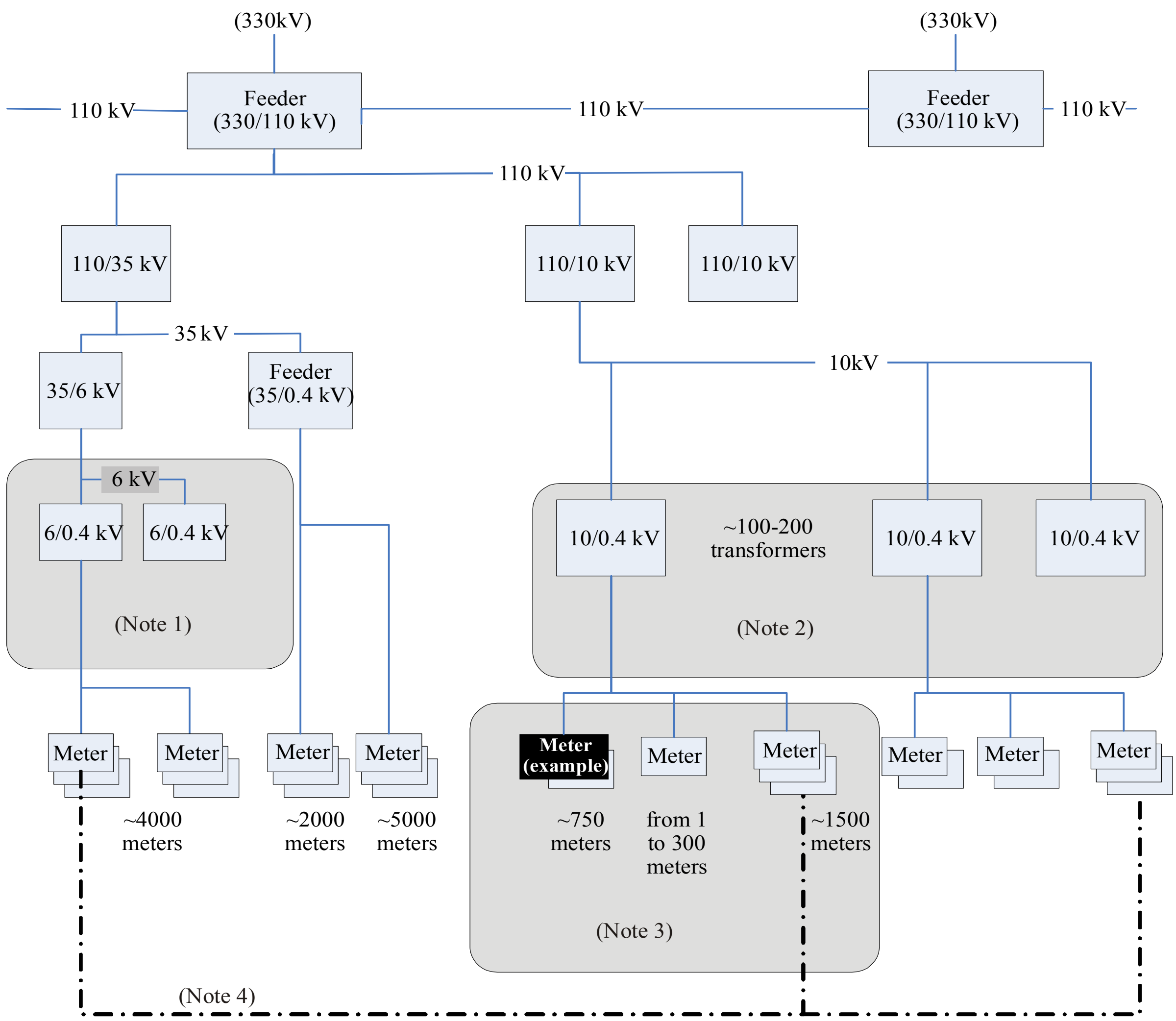


Figure D.1 – Example of neighbourhood AMI/AMM installation diagram

In other network topologies, communication between nodes of the same domain or between different domains may pass through MV-LV transformers (the 35/6 kV, 35/0.4 kV, 10/0.4 kV). Communications over high-voltage (HV) lines (110 kV) are not expected to be used due to safety reasons, although signals may pass even through MV-HV transformers, though with much lower probability. Overlapping between different parts of the UAN network is expected as shown in Figure D.2.



Note 1 – 80% to 100% of the communication between modems located at the LV-MV 6 kV transformer rooms/poles are under the same HV branch.

Note 2 – 80% to 100% of the communication between modems located at the LV-MV 10 kV transformer rooms/poles are under the same HV branch.

Note 3 – Communication between a selected meter with other meters in physical network is 30%-50% at different times.

Note 4 – 1% of communication between modems is separated by 110 kV feeders.

G.9955(11)_FI-2

Figure D.2 – Example of AMI/AMM installation

Another type of UAN domain can be established over MV lines, with its end-nodes residing at MV-LV transformers and bridging this MV domain to the corresponding LV domains. An example model of a UAN network that includes LV domains is presented in Figure D.3. The model includes UAN domains associated with LV lines (called branch domains, in Figure D.3) and UAN domains connecting branch domains to the utility head-end (called core domains in Figure D.3).

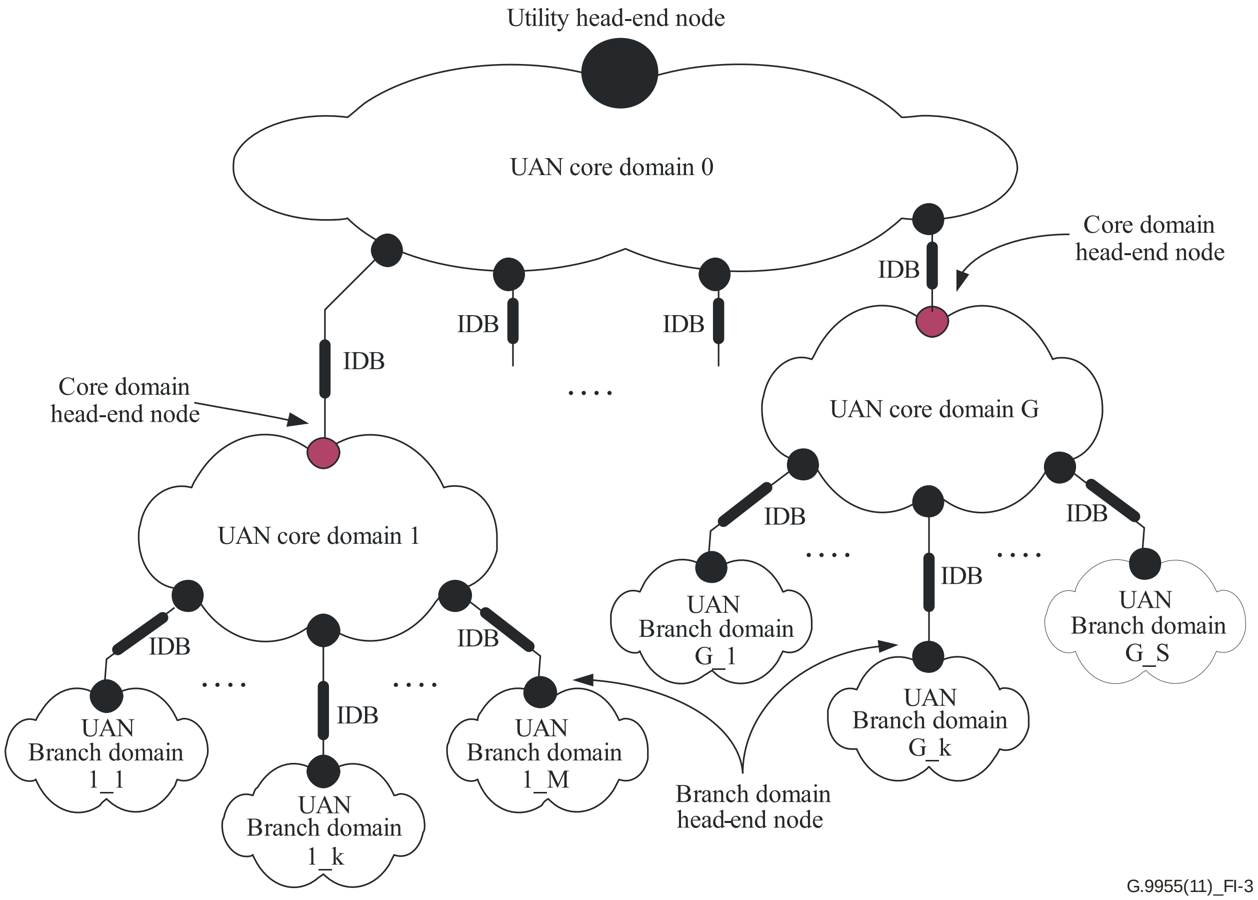


Figure D.3 – Example of UAN with multiple domains

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	Terminals and subjective and objective assessment methods
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, 2013