International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.9959
(02/2012)

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

Access networks – In premises networks

Short range narrow-band digital radiocommunication transceivers – PHY and MAC layer specifications

Recommendation ITU-T G.9959

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.9950–G.9999**

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

Recommendation ITU-T G.9959

Short range narrow-band digital radiocommunication transceivers – PHY and MAC layer specifications

Summary

Recommendation ITU-T G.9959 contains the MAC/PHY layer specification for short range narrow-band digital radiocommunication transceivers. This Recommendation is a joint piece of work between ITU-R and ITU-T, each contributing material from their respective remits. This Recommendation contains the non-radio (frequency) related aspects of the radiocommunication transceiver. The Recommendation specifies sub 1 GHz transceivers which shall be interoperable with transceivers complying with Annex A of this Recommendation.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T G.9959	2012-02-13	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 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/.

© ITU 2012

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

Recommendation ITU-T G.9959

Short range narrow-band digital radiocommunication transceivers – PHY and MAC layer specifications

1 Scope

This Recommendation contains the MAC/PHY layer specification for short range narrow-band digital radiocommunication transceivers. This Recommendation is a joint work of ITU-R and ITU-T, each contributing material from their respective remits. This Recommendation contains the non-radio (frequency) related aspects of the radiocommunication transceiver. The Recommendation specifies sub 1 GHz transceivers which shall be interoperable with transceivers complying with Annex A of this Recommendation.

2 References

None.

3 Definitions

This Recommendation defines the following terms:

3.1 alien domain: Any group of non-ITU-T G.9959 nodes connected to the same or a different medium (wired or wireless) operating in close proximity. These domains can be used as backbones to the ITU-T G.9959 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.

3.2 bridge to alien domain/network: An application device implementing an L2 or L3 bridging function to interconnect an ITU-T G.9959 domain to an alien domain (or alien network). Bridging to alien domains/networks is beyond the scope of this Recommendation.

3.3 broadcast: A type of communication where a node sends the same frame simultaneously to all other network nodes within a direct range. In a multi-hop domain, some nodes of the domain may not receive a broadcast frame.

3.4 clear channel assessment (CCA): Generated by the receiver, a CCA indicates that the medium is busy, i.e., a PHY frame, or sequence of PHY frames are currently transmitted on the medium by another node.

3.5 channel: A transmission path between nodes. One channel is considered to be one transmission path. Logically a channel is an instance of a communication medium being used for the purpose of passing data between two or more nodes.

3.6 data: Bits or bytes transported over the medium or via a reference point that individually convey 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 the preamble.

3.7 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.

3.8 **domain**: A part of an ITU-T G.9959 home network comprising the domain master and all those nodes that are registered with the same domain master. In the context of this Recommendation, use of the term 'domain' without a qualifier means 'ITU-T G.9959 domain', and use of the term 'alien domain' means 'non-ITU-T G.9959 domain'. Additional qualifiers (e.g., 'power-line' or RF) may be added to 'alien domain'.

3.9 **domain ID**: A unique identifier of a domain.

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

3.11 **home area network (HAN)**: A network at a customer's premises that interconnects customer-owned devices for energy management and communications with the utility.

3.12 **HomeID**: Information unit used as a domain ID in sub-1 GHz G.wnb networks.

3.13 **inter-domain bridge (IDB)**: A bridging function to interconnect nodes of two different domains.

3.14 **ISM band**: Frequency bands for industrial, scientific and medical use, allocated by the ITU.

3.15 **L3 bridge**: A bridging function which makes forwarding decisions based on layer 3 information. Typically, layer 3 implements IP routing.

3.16 **latency**: A measure of the delay from the instant that 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 the receiver protocol stack. Mean and maximum latency estimations are assumed to be calculated on the 99^{th} percentile of all latency measurements. If retransmission is set for a specific flow, the retransmission time is a part of latency for the protocol reference points above MAC.

3.17 **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.

3.18 **medium**: The air carrying the RF signals. Walls and other building components may affect the quality of the medium. Nodes communicating via the same medium on the physical layer may interfere with each other.

3.19 **multicast**: A type of communication when a node sends the same frame simultaneously to one or more other nodes in the network.

3.20 **net data rate**: The data rate available at the A-interface of the transceiver reference model.

3.21 **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.

3.22 **node**: Any network device that contains an ITU-T G.9959 transceiver. In the context of this Recommendation, use of the term 'node' without a qualifier means an 'ITU-T G.9959 node', and use of the term 'alien node' means a 'non-ITU-T G.9959 node'. Additional qualifiers (e.g., 'relay') may be added to either 'node' or 'alien node'.

3.23 **node ID**: A unique identifier allocated to a sub-1 GHz node operating in the network during its registration.

3.24 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.

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

3.26 priority: A value assigned to the specific frame(s) that determines the relative importance of transmitting frame(s) during the upcoming opportunity to use the medium.

3.27 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.

3.28 symbol frame: A frame composed of bits of a single modulation symbol period.

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

3.30 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.

3.31 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).

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

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ACK	Acknowledge
ADP	Application Data Primitives
AE	Application Entity
AIS	Application Interface Sublayer
AL	Always Listening
BER	Bit Error Rate
CC	Channel Configuration
CCA	Clear Channel Assessment
CER	Communication Error Rate
CP	Customer Premises
CRC	Cyclic Redundancy Check
CW	Carrier Wave
DM	Domain Master
Dst	Destination
EFD	End of Frame Delimiter
EIRP	Effective Isotropic Radiated Power
EMC	Electromagnetic Compatibility
ERP	Effective Radiated Power
FCS	Frame Check Sequence

FER	Frame Error Rate
FL	Frequently Listening
FLN	Frequently Listening Node
FSK	Frequency Shift Keying
GFSK	Gaussian Frequency Shift Keying
HAN	Home Area Network
EHR	End Header
HVAC	Heating, Ventilation, Air Conditioning
ID	Identification
IDB	Inter-Domain Bridge
INB	Inter-Network Bridge
ISI	Inter-Symbol Interference
ISM	Industrial, Scientific and Medical
IUT	Implementation Under Test
LBT	Listen Before Talk
LLC	Logical Link Control
LO	Local Oscillator
LSB	Least Significant Bit
MAC	Medium Access Control
MD	MAC Data
MD-SAP	MAC Data – Service Access Point
MDI	Medium-Dependent Interface
MFR	MAC Footer
MHR	MAC Header
MIB	Management Information Base
MLME	MAC Layer Management Entity
MLME-SAP	MAC Layer Management Entity – Service Access Point
MPDU	MAC Protocol Data Unit
MSB	Most Significant Bit
MSDU	MAC Service Data Unit
NHR	Network Header
NPDU	Network layer Protocol Data Unit
NRZ	Non Return to Zero
NSDU	Network Service Data Unit
OSI	Open System Interconnect
PCS	Physical Coding Sublayer
PD	PHY Data

PD-SAP	PHY Data – Service Access Point
PDU	Protocol Data Unit
PHR	PHY Header
PHY	Physical layer
PLME	Physical Layer Management Entity
PLME-SAP	Physical Layer Management Entity – Service Access Point
PMI	Physical Medium-Independent interface
PPDU	PHY Protocol Data Unit
PPM	Parts Per Million
PSDU	PHY Service Data Unit
R1	Type 1 of supported data Rate, i.e., 9.6 kbit/s
R2	Type 2 of supported data Rate, i.e., 40 kbit/s
R3	Type 3 of supported data Rate, i.e., 100 kbit/s
RF	Radio Frequency
RSSI	Receive Signal Strength Indication
RX	Receive/Receiver
SAP	Service Access Point
Src	Source
SDU	Service Data Unit
SFD	Start of Frame Delimiter
SHR	Start Header
SNR	Signal to Noise Ratio
TRX	Transceiver
TX	Transmit/Transmitter

5 Network architecture and reference models

5.1 Network architecture and topology

5.1.1 Basic principles of ITU-T G.9959 networking

The followings are the basic principles of the ITU-T G.9959 network architecture:

1. The network is divided into domains:
   - The division of physical nodes into domains is logical. Domains may fully or partially overlap as there is no physical separation.
   - The number of domains within the physical network may be up to *N* domains. The number of *N* domains is limited by the 32 bit domain ID. For the sub 1 GHz ITU-T G.9959 transceivers, the domain identification is indicated by the SrcHomeID field, see Table 8-2.
   - 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 IDBs.
- Besides ITU-T G.9959 domains, a network may include alien domains. Connection between ITU-T G.9959 domains and alien domains is via L3 bridges.
- Operation of different domains in the same physical media is handled by individual domain masters.

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). For the sub 1 GHz ITU-T G.9959 transceivers, the node identification is indicated by the SrcNodeID field, see Table 8-2.
   - 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 identified 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.9959 networks:
   - can communicate via inter-network bridges (INB). The INB function is an L3 bridging function associated with one or more dedicated nodes on network domains.

A generic network architecture of an ITU-T G.9959 network is presented in Figure 5-1.

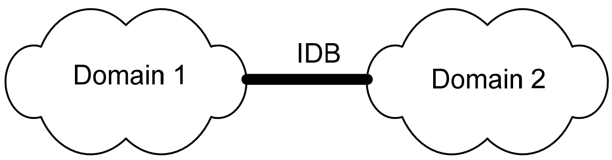

Figure 5-1 – Generic network architecture

The details of domain operation rules, types of communications inside a domain, and the functionalities of domain master and endpoint nodes are beyond the scope of this Recommendation. The ITU-T G.9959 network supports a mesh topology, that allows each node to communicate with any other node either directly, or via one or more relays, or via relays and IDBs. This allows support of any type of network topology on L3 and above, such as star, tree, multiple trees, and others.

Alien domains and bridges to alien domains are beyond the scope of this Recommendation.

The scope of this Recommendation is limited to the PHY/MAC layer of ITU-T G.9959 transceivers capable of operating either with extended capabilities (e.g., domain master, relay node, or combinations thereof) or without extended capabilities, as endpoint nodes.

5.1.2 Home control network architecture and topology

An example architectural model of a home control network is presented in Figure 5-2. It contains the local area network (LAN), L3 bridges and the home control home area network (HC-HAN) at customer premises (CP). Each HC-HAN is one domain.

The domains of HAN include customer-owned and may include some service provider-owned devices related to, e.g., energy management (e.g., home appliances, PSTs, EVSEs) that reside at the CP. In this example, each HAN is connected to a LAN via an L3 bridge.

NOTE – This architectural model is exclusively for reference purposes and does not limit the use of ITU-T G.9959 transceivers for other network configurations.

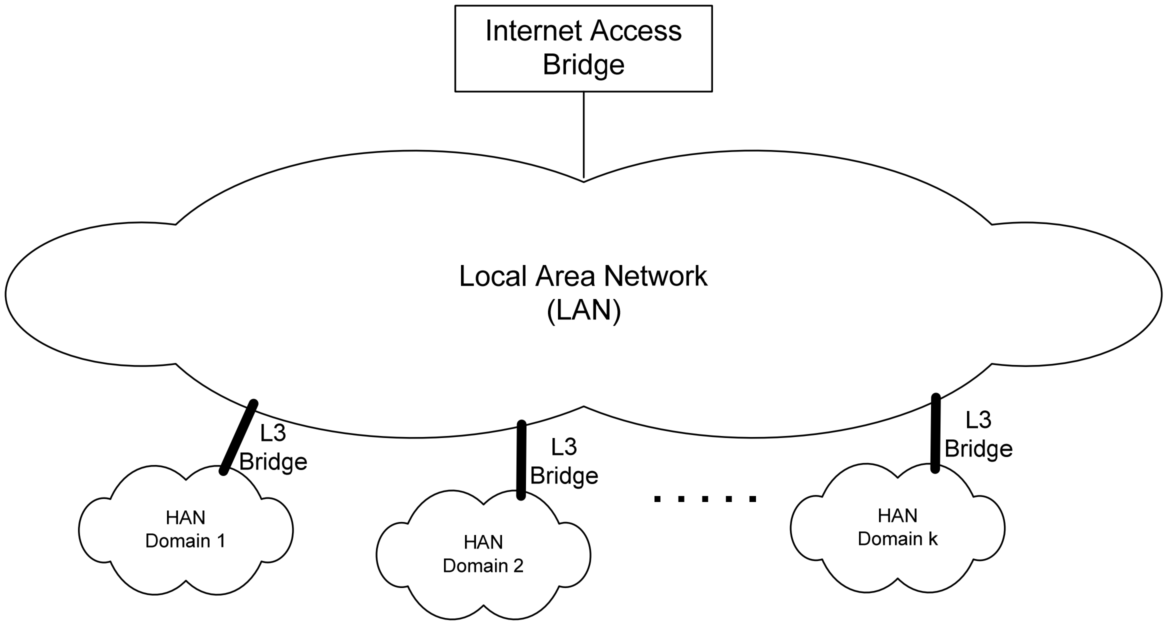

Figure 5-2 – Generic LAN network architecture

5.1.2.1 Generic LAN architecture

The LAN interconnects HAN domains via L3 bridges. Each domain shall be associated with a particular set of ITU-T G.9959 nodes interfacing with the same medium (usually, RF).

The domains may be connected to each other by one or more IDBs. Nodes of different domains may communicate with each other using one or multiple IDBs which may be interconnected by the LAN.

Besides the ITU-T G.9959 domain, the LAN may also connect to alien, non-ITU-T G.9959 domains. These domains are established by non-ITU-T G.9959 technologies, both wired and wireless. Alien domains can be bridged to the ITU-T G.9959 domains using L3 bridges via the LAN. The specification of LAN technologies and bridges to alien domains is beyond the scope of this Recommendation.

5.1.2.2 Generic HAN architecture

The HC-HAN (further called "HAN") is a logical domain. Each domain shall be associated with a particular set of ITU-T G.9959 nodes. A particular node can belong to only 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 domain of a HAN is established using low-power RF communication. One of the HAN domain devices is the domain master, while all other nodes are just called nodes. Two or more HAN domains may overlap: nodes of overlapping domains "see" transmissions of each other and may interfere with each other.

A HAN may be implemented as an alien, non-ITU-T G.9959 domain. These domains can be established using in-home wireline or wireless media. Alien domains can be bridged to the ITU-T G.9959 domains using L3 bridges. The specification of bridges to alien domains is beyond the scope of this Recommendation.

5.2 Reference models

5.2.1 Protocol reference model of a transceiver

The protocol reference model of a transceiver is presented in Figure 5-3. It includes three main reference points: application interface (A-interface), physical medium-independent interface (PMI), and medium-dependent interface (MDI), Figure 5-3. The shaded part of the reference model is defined in this Recommendation.

The MDI is a physical interface defined in terms of the physical signals transmitted over a 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 PMI and A-interface are defined as functional interfaces, in terms of sets of primitives exchanged across the interface.

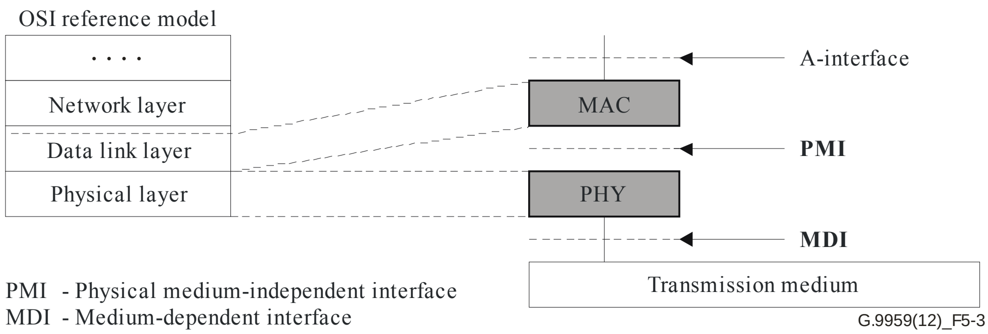

Figure 5-3 – Protocol reference model of an ITU-T G.9959 transceiver

The medium access control layer (MAC) controls access of the node to the medium using the medium access protocols defined. The MAC layer also provides checksum protection to the MAC information.

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

The PHY layer provides encoding of the PHY frame content (header and payload) and modulates the encoded PHY frames for transmission over the medium.

5.2.2 Functional description of the interface

This section contains the functional description of the transceiver interfaces (reference points) based on the protocol reference model presented in Figure 5-4. The interfaces shown in Figure 5-4 are defined in this Recommendation.

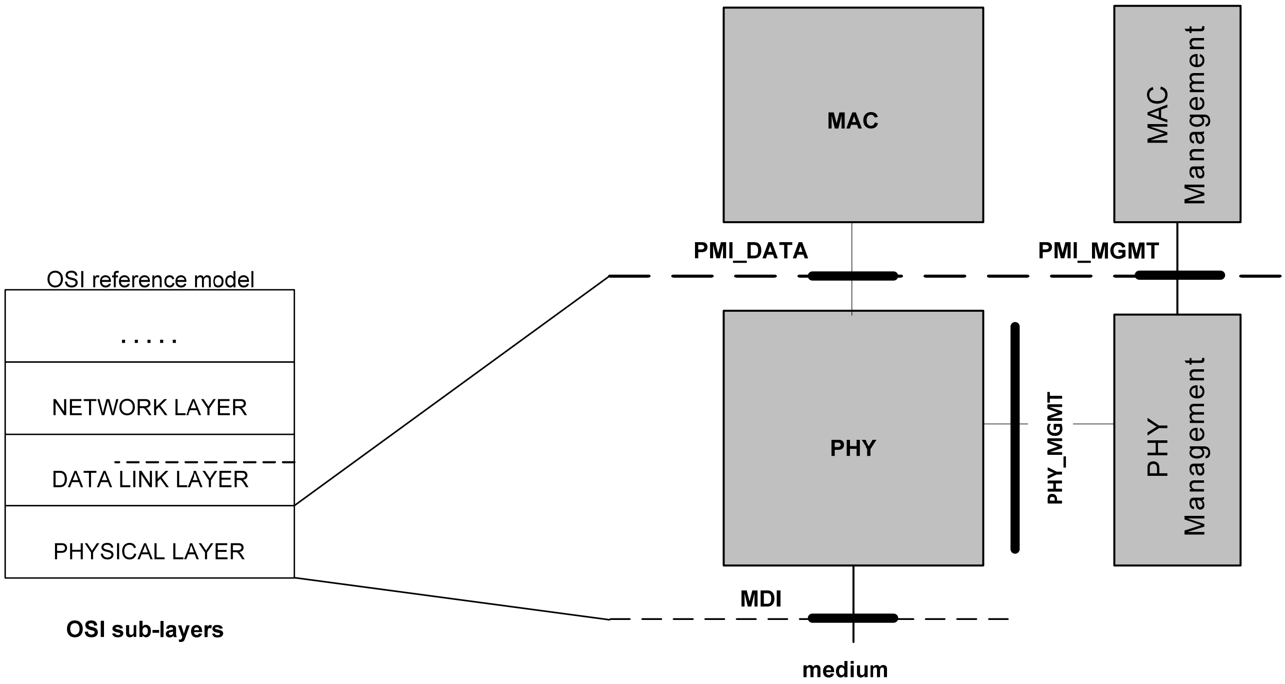

Figure 5-4 − Transceiver reference points related to PHY/MAC

The model in Figure 5-4 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). 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.

5.2.3 Functional model of a transceiver

The functional model of a transceiver is presented in Figure 5-5. It addresses nodes without extended capabilities as well as nodes with extended capabilities such as the domain master.

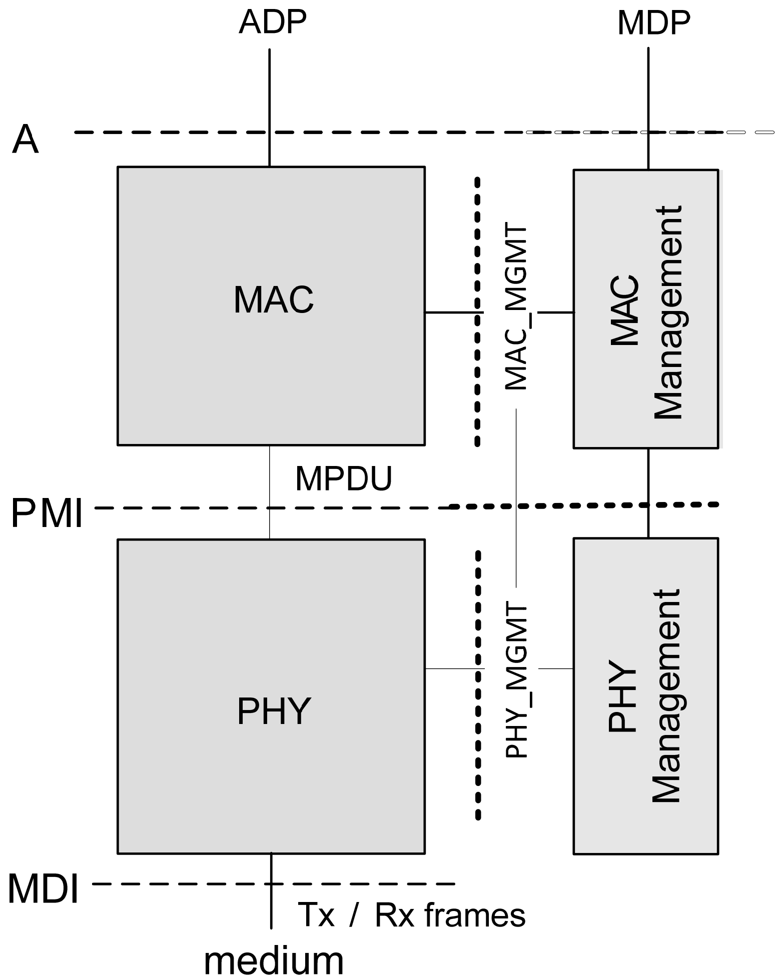

Figure 5-5 − Functional model of an ITU-T G.9959 transceiver

The detailed description of the functional model of the PHY layer is presented in clause 7.1. The detailed description of the functional model of the MAC layer is presented in clause 8.1.

5.3 Operation modes

5.3.1 Operation modes for 'sub 1 GHz' transceivers

The G.9959 nodes operating in sub 1 GHz can operate in two different modes: Always listening (AL) and frequently listening (FL). An ITU-T G.9959 node can operate in either of the two modes or dynamically shift between them.

The FL listening mode is a mode where power consumption is minimized by turning the receiver on and off. In FL mode the receiver is shut down the majority of the time. At a regular interval the receiver is turned on for a short duration to listen. The drawback of FL mode is the increased latency due to the receiver off time.

The alternative AL listening mode is the minimum latency mode where the receiver is turned on at all time (except when transmitting). More energy is consumed if the receiver remains on at all times.

Typically a node that is mains powered will operate all of the time in AL mode. Nodes that are battery powered or where power consumption is critical may operate entirely in FL mode or may operate in FL mode the majority of the time and then occasionally shift to AL mode for a duration of time during which the latency is low to facilitate bandwidth intensive transfers.

6 Profiles

For further study.

7 Physical layer specification

7.1 PHY specifications of 'sub 1 GHz' transceivers

7.1.1 General

7.1.1.1 Features of the PHY layer

The PHY is responsible for the following tasks:

– assignment of an RF profile to a physical channel
– activation and deactivation of the radio transceiver
– data (frames) transmission and reception
– clear channel assessment
– frequency selection
– link quality for received frames

The RF transceiver shall be able to operate in a one, two or three channel configuration located in license-free RF bands such as the ISM bands (refer to clause 7.1.2).

The PHY shall provide two services: (1) the PHY data service accessed through the PHY data (PD) service access point (PD-SAP) and (2) the PHY management service interfacing with the physical layer management entity (PLME) service access point (PLME-SAP). The PHY data service enables the transmission and reception of PHY protocol data units (PPDUs) across the physical radio channel. See clause 7.1.2.4 for a detailed description.

Constants and attributes that are specified and maintained by the PHY are written in the text of this section in *italics*. Constants have a general prefix of "a", e.g., *aMaxPHYPacketSizeRx*, and are listed in Table 7-27. Attributes have a general prefix of "phy", e.g., *phyCurrentChannel*, and are listed in Table 7-28.

7.1.1.2 Data wrapping

The PHY inserts outgoing data into a physical RF frame format. When receiving frames the incoming data is extracted from the RF frame structure and forwarded for the upper layers. Also, refer to clause 7.1.3.1.

The data from the upper layers are passed to the PHY layer as a PHY service data unit (PSDU). The PSDU is prefixed by the PHY with a start header (SHR) and (in some cases) an end header (EHR). The SHR contains the preamble sequence and start-of-frame delimiter (SFD) fields. The preamble sequence enables the RF receiver to achieve symbol synchronization. The SHR, PSDU and EHR together form the PHY protocol data unit (PPDU).

7.1.1.3 Concept of service primitives

This clause provides a brief overview of the concept of service primitives (operations). Refer to [b-ITU-T X.210] for more detailed information. The services of a layer are the capabilities it offers to the user in the next higher layer or sublayer by building its functions on the services of the next lower layer. This concept is illustrated in Figure 7-1, showing the service hierarchy and the relationship of the two correspondent N-users and their associated N-layer (or sublayer) peer protocol entities.

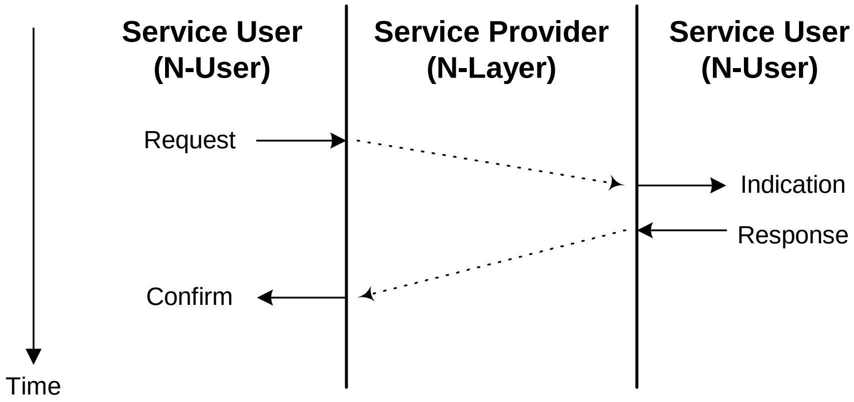

Figure 7-1 – Service primitives

The services are specified by describing the information flow between the N-user and the N-layer. This information flow is modelled by discrete, instantaneous events, which characterize the provision of a service. Each event consists of passing a service primitive from one layer to the other through a service access point (SAP) associated with an N-user. Service primitives convey the required information by providing a particular service. These service primitives are an abstraction because they specify only the provided service rather than the means by which it is provided. This definition is independent of any other interface implementation.

Services are specified by describing the service primitives and parameters that characterize it. A service may have one or more related primitives that constitute the activity that is related to that particular service. Each service primitive may have zero or more parameters that convey the information required to provide the service.

A primitive can be one of four generic types:

– Request: the request primitive is passed from the N-user to the N-layer to request that a service is initiated.

– Indication: the indication primitive is passed from the N-layer to the N-user to indicate an internal N-layer event that is significant to the N-user. This event may be logically related to a remote service request, or it may be caused by an N-layer internal event.

– Response: the response primitive is passed from the N-user to the N-layer to complete a procedure previously invoked by an indication primitive.

– Confirm: the confirm primitive is passed from the N-layer to the N-user to convey the results of one or more associated previous service requests.

7.1.2 Transceiver front-end specifications

7.1.2.1 RF profiles

The possible transceiver usage of RF frequencies and data rates is described in Table 7-1 using RF profiles. An RF profile is a combination of a centre frequency and data rates which can be used at this centre frequency, in a specific region. The list of RF profiles that shall be used by the ITU-T G.9959 transceiver in different world regions is specified in Table 7-1. The list of specific regional frequencies is outside the scope of this Recommendation. A transceiver shall support up to 3 channels. A channel is associated with a single RF profile. Each channel can have a different RF profile assigned to it. Unused channels are disabled by assigning RF profile 0 (a dummy RF profile with no centre frequency or data rate allocated) to this channel. The transceiver shall support one or more of the data rates 9.6 kbit/s (R1), 40 kbit/s (R2) and 100 kbit/s (R3) according to the RF profile assigned.

Table 7-1 – RF profiles

RF Profile	Country/ Market	Centre frequency	R3	R2	R1	FL (Note 1)
0	n/a	n/a				
1	European Union	f_{EU1}	√			
2		f_{EU2}		√	√	
3		f_{EU2}		√		√
4	United States	f_{US1}	√			
5		f_{US2}		√	√	
6		f_{US2}		√		√
7	Hong Kong	f_{HK1}	√			
8		f_{HK1}		√	√	
9		f_{HK1}		√		√
10	Australia and New Zealand	f_{ANZ1}	√			
11		f_{ANZ2}		√	√	
12		f_{ANZ2}		√		√
13	Malaysia	f_{MY1}	√			
14		f_{MY1}		√	√	
15		f_{MY1}		√		√
16	India	f_{IN1}	√			
17		f_{IN1}		√	√	
18		f_{IN1}		√		√
19	Japan	f_{JP1}	√			√
20		f_{JP2}	√			√
21		f_{JP3}	√			√

NOTES:
1. The FL (frequently listening) mode (refer to clause 5.3.1) is only applicable to the marked RF profiles.
2. The AL (always listening) mode can be used with any RF profile.

7.1.2.2 Channel configurations

A compliant node shall operate in one of the three possible configurations as listed in the table below:

Table 7-2 – Channel configurations

Configuration	Num channels	Data rate		
		R3	R2	R1
1	1	–	Ch B	Ch B (Note)
2	2	Ch A	Ch B	Ch B (Note)
3	3	Ch A Ch B Ch C	–	–
NOTE – The R1 data rate is not supported in FL mode (i.e., in this mode configuration 1 shall not include R1). In AL mode both R2 and R1 shall be supported.				

Table 7-1 has the information on specific RF profiles which, in conjunction with Table 7-2, gives the valid configurations for a node.

Example:

Let us assume a specific region in which there are two available frequencies: f_1 and f_2. Let us also assume that the following RF profiles are available (as explained in clause 7.1.2.1):

- One RF profile ("RF profile 1") with frequency f_1 and a single data rate (R3).
- Two RF profiles for f_2:
  - RF profile 2: A dual data rate (R1 and R2) profile.
  - RF profile 3: A single data rate (R2) profile when operating in FL (frequently listening) mode.

According to Table 7-1 this leads to the following possible configurations:

1) A single channel system at f_2 where only R2 is enabled in FL (frequently listening) mode.
2) A single channel system at f_2 where both R1 and R2 are enabled in AL (always listening) mode.
3) A dual channel system where channel B is at f_2 where only R2 is enabled in FL (frequently listening) mode and channel A is at f_1 with only R3 enabled.
4) A dual channel system where channel B is at 868.40 MHz where both R1 and R2 are enabled in AL (always listening) mode and channel A is at f_1 with only R3 enabled.

7.1.2.3 Data rates

The PHY shall support the data rates and the accuracy as listed in the table below.

Table 7-3 – Data rate and accuracy

Rate	Bit rate	Symbol rate	Accuracy
R1	9.6 kbit/s	19.2 kbaud	±27 ppm
R2	40 kbit/s	40 kbaud	±27 ppm
R3	100 kbit/s	100 kbaud	±27 ppm

7.1.2.4 Modulation and encoding

The PHY shall employ frequency shift keying (FSK) for RF modulation of R1 and R2. The PHY shall employ Gaussian frequency shift keying (GFSK) for R3.

Manchester code shall be used for data symbol encoding for R1 and non-return-to-zero for data symbol encoding for R2 and R3.

The modulation and the coding format is summarized in the table below.

Table 7-4 – Modulation and coding format

Data Rate	Modulation	Coding	Separation	Symbols
R1	FSK	Manchester	40 kHz ±10%	Binary
R2	FSK	NRZ	40 kHz ±10%	Binary
R3	GFSK, BT = 0.6	NRZ	58 kHz ±10%	Binary

The mapping of NRZ symbols to the physical medium assuming high side LO is given below. The mapping should be inverted for low side LO:

Table 7-5 – Symbol mapping, for high side LO

Symbol	Frequency
0	$f_{center\ frequency}$ – separation/2
1	$f_{center\ frequency}$ + separation/2

The mapping of Manchester symbols to the physical medium assuming high side LO is given below:

Table 7-6 – Manchester symbols, for high side LO

Symbol	Frequency
0	Transition from ($f_{center\ frequency}$ + separation/2) to ($f_{center\ frequency}$ – separation/2)
1	Transition from ($f_{center\ frequency}$ – separation/2) to ($f_{center\ frequency}$ + separation/2)

7.1.2.5 Transmitter and receiver requirements

Unless stated otherwise, all RF power measurements, either transmit or receive, shall be made at the antenna connector. The measurements shall be made with equipment that is either matched to the impedance of the antenna connector or corrected for any mismatch. For devices without an antenna connector, the measurements shall be interpreted as effective isotropic radiated power (EIRP) (i.e., a 0 dBi gain antenna); and any radiated measurements shall be corrected to compensate for the antenna gain in the implementation.

7.1.2.5.1 Transmit frequency error

Frequency error is defined as the difference between the measured transmitted centre frequency and the ideal centre frequency as described in Table 7-1. A conforming device shall comply with the frequency error requirement over the entire temperature and supply range for the device.

The frequency error shall not exceed ±27 ppm including crystal ageing over five years.

7.1.2.5.2 Transmit power adjustments (conducted)

The transmitter shall transmit at an output power level which shall conform to local regulations and this level is outside the scope of this Recommendation. This output power level is denoted as "nominal".

The transmitter shall be able to adjust its output power in steps of 2 dB in the range between the nominal transmit power down to the (nominal output power – 10 dB).

The transmitter shall be capable of reducing the output power to at least (nominal output power – 20 dB).

7.1.2.5.3 Receiver sensitivity

To ensure a minimum RF link budget the receiver shall be capable, under typical temperature and supply conditions, of receiving a standard test frame at a minimum level as specified in Table 7-8.

Standard test frame and test conditions are described in Table 7-7.

Table 7-7 – Standard test frame

Term	Definition	Conditions
Communication error rate (CER)	Average loss. (command and acknowledgment transmission, i.e., two-way communication)	Average measured over random frame data.
Receiver sensitivity	Threshold input signal power that yields a specific CER	Frame data length = 4 bytes CER < 10% (without retransmission) Power measured at antenna terminals Interference not present

Table 7-8 – Minimum receiver sensitivity

Rate	Minimum receiver sensitivity
R1	–95 dBm
R2	–92 dBm
R3	–89 dBm

7.1.2.5.4 Clear channel assessment

The PHY shall be able to perform a clear channel assessment (CCA) to determine if the RF medium is busy or clear for transmission. If the channel is found to be idle, the PHY may transmit its data.

To comply with local regulations the receiver shall be able to perform LBT (listen before talk) with a threshold of –80 dBm.

7.1.2.5.5 Receiver spurious requirement

The RX spurious requirement restricts the amount of power to be emitted in RX mode. Any unwanted emissions near the centre frequency may degrade other devices ability to receive weak signals.

The RX spurious limit is –70 dBm/100 kHz within ±1 MHz from the regional centre frequency as shown in Figure 7-2.

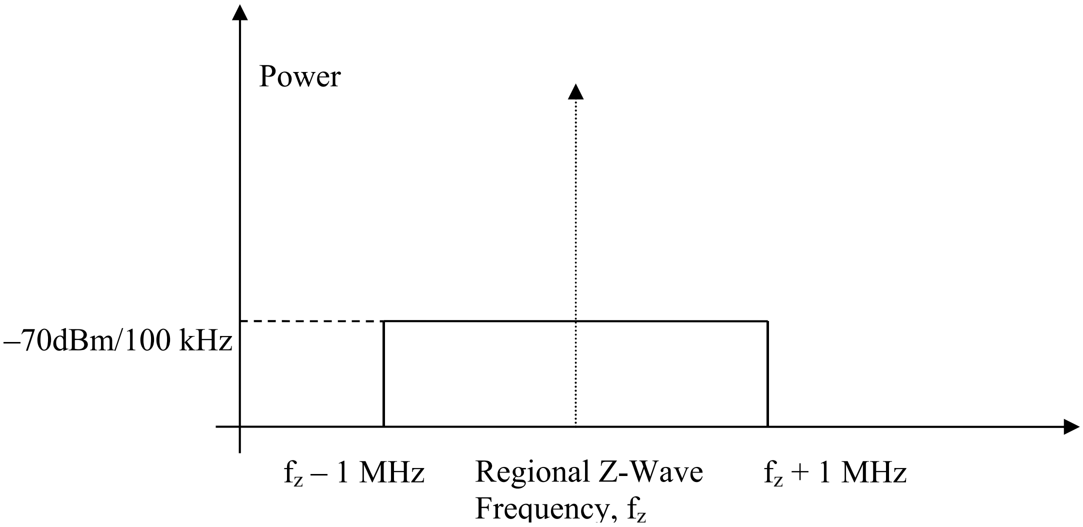

Figure 7-2 – RX spurious limit

7.1.2.5.6 Receiver blocking

Blocking is a measure of the capability of the receiver to receive a wanted modulated signal without exceeding a given degradation due to the presence of an unwanted input signal.

A conforming implementation shall be able to pass a blocking test as described below for all data rates.

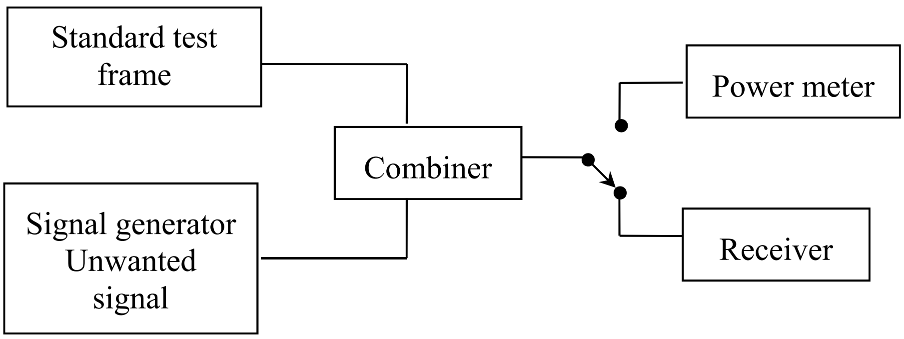

Figure 7-3 – Receiver blocking test definition

Method of measurement:

– Standard test frame is at the nominal frequency with normal modulation. Its power is adjusted down to the sensitivity level +3 dB.

– Unwanted signal is a CW (carrier wave) at a specific offset frequency. Its power is adjusted up until the receiver gets the FER (frame error rate) that correspond to the sensitivity level.

Limits:

Table 7-9 – Blocking limits

Frequency offset	Limits
±1 MHz	–44 dBm
±2 MHz	–34 dBm
±5 MHz	–27 dBm
±10 MHz	–25 dBm

7.1.2.5.7 Receiver maximum input level of desired signal

The receiver maximum input level is the maximum power level of the desired signal, in decibels relative to 1 mW, present at the input of the receiver for which the CER criterion in Table 7-7 is met. A receiver shall have a maximum input level greater than or equal to 0 dBm to be able to sustain "zero" distance between two or more devices.

7.1.2.5.8 TX-to-RX turnaround time

The TX-to-RX turnaround time shall be measured from the trailing edge of the last transmitted symbol until the receiver is ready to begin the reception of the next PHY packet.

The TX-to-RX turnaround time shall be less than *aTurnaroundTimeTXRX* (see Table 7-27).

7.1.2.5.9 RX-to-TX turnaround time

The RX-to-TX turnaround time shall be measured at the receiver from the trailing edge of the last symbol of a received packet until the transmitter is ready to begin transmission.

The RX-to-TX turnaround time shall be more than *aTurnaroundTimeRXTX* (see Table 7-27).

7.1.3 PPDU format

The PPDU packet structure is constructed so that the leftmost field shall be transmitted or received first. Each byte shall be transmitted or received by most significant bit first. All multi-byte fields shall be transmitted or received by least significant byte first.

Each PPDU packet consists of the following basic components:

– a start of frame delimiter (SHR), which allows a receiving node to synchronize and lock onto the bit stream;

– a variable length payload, which carries the MAC sublayer frame;

– an end of frame;

– an end of frame delimiter (HER), which indicates the end of the frame (R1 only).

7.1.3.1 General packet format

The PPDU packet structure shall be formatted as illustrated in Figure 7-4 below.

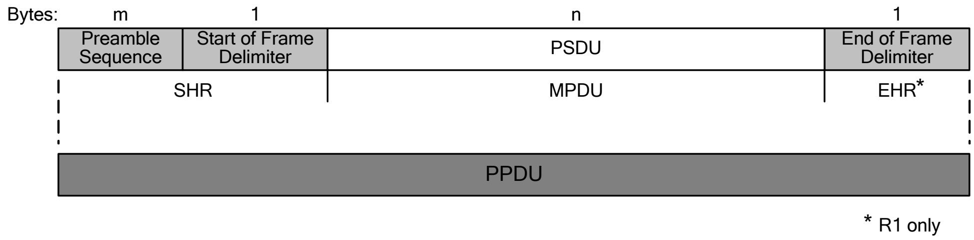

Figure 7-4 – PPDU packet format

7.1.3.2 Preamble field

The preamble sequence field is used by a receiver to obtain symbol and bit synchronization. The preamble field shall be composed of a sequence of bytes containing the binary pattern "01010101". Figure 7-5 shows the Manchester encoded preamble pattern for R1. Figure 7-6 shows the NRZ encoded preamble pattern for R2/R3.

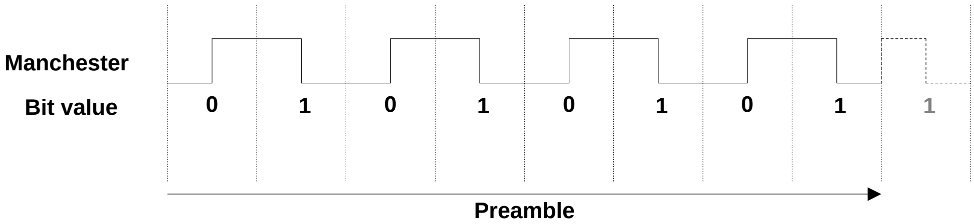

Figure 7-5 – Manchester encoded preamble pattern (R1)

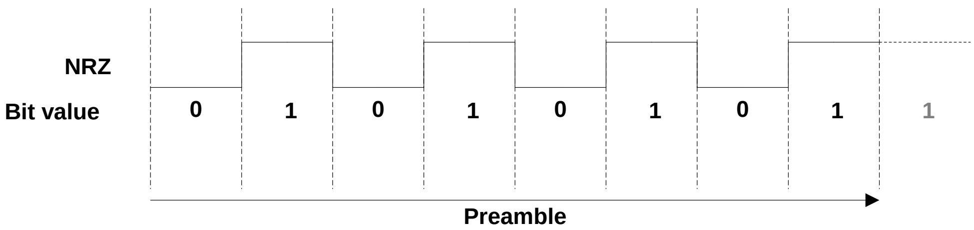

Figure 7-6 – NRZ encoded preamble pattern (R2/R3)

To support an RX round robin in a multichannel system the preamble is extended beyond what would be needed in a single channel system. As the effective bandwidth decreases with increased preamble the preamble can be lower for first transmissions and increased for retransmissions so that the typical bandwidth is maximized. A table of the preamble requirement is given below:

Table 7-10 – Preamble length

Channel configuration	Rate	Preamble length in bytes (min)		
		Singlecast/ broadcast	Multicast	Beam
1 and 2	R1 R2 R3	10 10 40	10 20 40	n/a 8 8
3	R1 R2 R3	n/a n/a 24	n/a n/a 24	n/a n/a 8

7.1.3.3 Start of frame delimiter field

The start of frame delimiter (SFD) is an 8-bit field indicating the end of the synchronization (preamble) field and the start of the packet data. The SFD shall be formatted as illustrated in the table below.

Table 7-11 – Format of the SFD field

Bits	7	6	5	4	3	2	1	0
Data	0	0	0	0	1	1	1	1

7.1.3.4 PSDU field

The PSDU field has a variable length and carries the data of the PHY packet.

7.1.3.5 End of frame delimiter field

The end of frame delimiter (EFD) is only transmitted at R1 (Manchester encoded data rate). The 8 bit symbol is a sequence of 8 Manchester code violations each denoted E. Each violation, E, is defined as a symbol without transition for instance low-low. Figure 7-7 illustrates the transmission of the EFD.

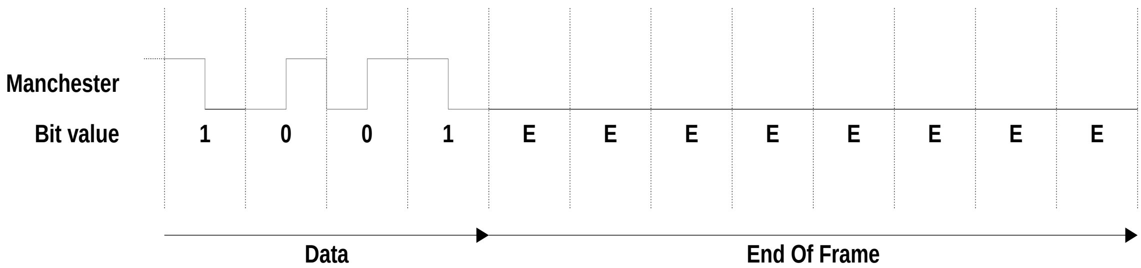

Figure 7-7 – End of frame delimiter pattern

7.1.4 PHY service specifications

The PHY provides an interface between the MAC sublayer and the physical radio channel. The PHY provides two services, accessed through two service access points (SAPs): the PHY data, accessed through the PHY data SAP (PD-SAP), and the PHY management service, accessed through the PHY layer management entity SAP (PLME-SAP). The PLME is responsible for maintaining a database of managed objects pertaining to the PHY. This database is referred to as the PHY management information base (MIB). Figure 7-8 shows the components and interfaces of the PHY.

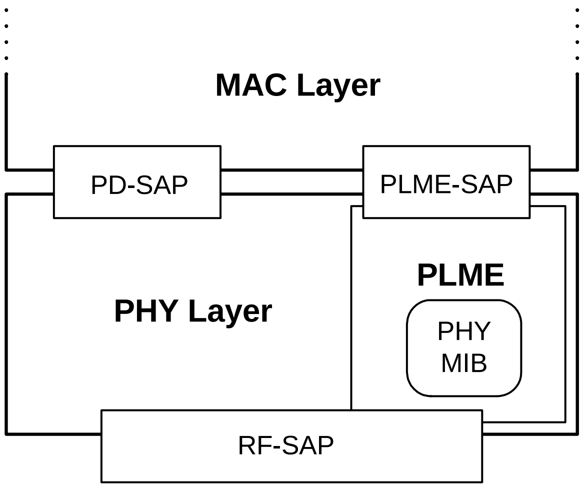

Figure 7-8 – PHY reference model

7.1.4.1 PHY data service

The PD-SAP supports the transport of MPDUs between peer MAC sublayer entities. Table 7-12 lists the primitives supported by the PD-SAP. These primitives are discussed in the clauses referenced in the table.

Table 7-12 – PD-SAP primitives

PD-SAP primitive	Request	Confirm	Indication
PD-DATA	clause 7.1.4.1.3	clause 7.1.4.1.2	clause 7.1.4.1.3

7.1.4.1.1 PD-DATA.request

The PD-DATA.request primitive requests the transfer of an MPDU (i.e., the PSDU) from the MAC sublayer to the local PHY entity.

7.1.4.1.1.1 Semantics of the PHY data request primitive

The semantics of the PD-DATA.request primitive is as follows:

```
PD-DATA.request    (
                   psduChannel,
                   psduRate,
                   psduLength,
                   psdu
                   )
```

Table 7-13 specifies the parameters for the PD-DATA.request primitive.

Table 7-13 – PD-DATA.request parameters

Name	Type	Valid range	Description
psduChannel	Enumeration	Valid channel selection A, B, or C according to the applied RF profile (see Table 7-1)	The physical channel to be used for transmission of the PSDU.
psduRate	Enumeration	R1, R2 or R3. Validity of rate depends on applied RF profile (see Table 7-1)	The data rate to be used for the transmission of the PSDU.
psduLength	Byte	≤ *aMaxPHYPacketSizeRx*, where x is 1, 2 or 3 depending on data rate	The number of bytes contained in the PSDU to be transmitted by the PHY entity.
Psdu	Set of bytes	–	The set of bytes forming the PSDU to be transmitted by the PHY entity.

7.1.4.1.1.2 When generated

The PD-DATA.request primitive is generated by a local MAC sublayer entity and issued to its PHY entity to request the transmission of an MPDU.

7.1.4.1.1.3 Effects on receipt

The receipt of the PD-DATA.request primitive by the PHY entity will cause the transmission of the supplied PSDU. Provided the transmitter is enabled (TX_ON mode), the PHY will first construct a PPDU, containing the supplied PSDU, and then transmit the PPDU. When the PHY entity has completed the transmission, it will issue the PD-DATA.confirm primitive with a status of SUCCESS.

If the PD-DATA.request primitive is received while the receiver is enabled (RX_ON mode) or if the transceiver is disabled (TRX_OFF mode), the PHY entity will issue the PD-DATA.confirm primitive with a status of RX_ON or TRX_OFF, respectively.

7.1.4.1.2 PD-DATA.confirm

The PD-DATA.confirm primitive confirms the end of the transmission of an MPDU (i.e., PSDU) from a local MAC sublayer entity to the physical media.

7.1.4.1.2.1 Semantics of the PHY data confirm primitive

The semantics of the PD-DATA.confirm primitive is as follows:

```
PD-DATA.confirm    (
                   status
                   )
```

Table 7-14 specifies the parameters for the PD-DATA.confirm primitive.

Table 7-14 – PD-DATA.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS, RX_ON or TRX_OFF	The result of the request to transmit a packet.

7.1.4.1.2.2 When generated

The PD-DATA.confirm primitive is generated by the PHY entity and issued to its MAC sublayer entity in response to a PD-DATA.request primitive. The PD-DATA.confirm primitive will return a status of either SUCCESS, indicating that the request to transmit was successful, or an error code of RX_ON or TRX_OFF.

7.1.4.1.2.3 Effects on receipt

The PD-DATA.confirm primitive allows the MAC sublayer entity to take proper action in case the transmission was unsuccessful.

7.1.4.1.3 PD-DATA.indication

The PD-DATA.indication primitive indicates the transfer of an MPDU (i.e., PSDU) from the PHY to the local MAC sublayer entity.

7.1.4.1.3.1 Semantics of the PHY data indication primitive

The semantics of the PD-DATA.indication primitive is as follows:

```
PD-DATA.indication      (
                        psduByte
                        )
```

Table 7-15 specifies the parameters for the PD-DATA.indication primitive.

Table 7-15 – PD-DATA.indication parameters

Name	Type	Valid range	Description
psduByte	Byte	–	One of a set of bytes forming the PSDU to be transmitted by the PHY entity.

7.1.4.1.3.2 When generated

The PD-DATA.indication primitive is generated by the PHY entity and issued to its MAC sublayer entity to transfer a received PSDU.

7.1.4.1.3.3 Effect on receipt

On receipt of the PD-DATA.indication primitive, the MAC sublayer is notified of the arrival of an MPDU across the PHY data service.

7.1.4.2 PHY management service

The PLME-SAP allows the transport of management commands between the MLME and the PLME. Table 7-16 lists the primitives supported by the PLME-SAP. These primitives are discussed in the clauses referenced in the table.

Table 7-16 – PLME-SAP primitives

PLME-SAP Primitive	Request	Confirm	Indication	Response
PLME-SFD	–	–	clause 7.1.4.2.1	–
PLME-GET-CCA	clause 7.1.4.2.2	clause 7.1.4.2.3	–	–
PLME-GET	clause 7.1.4.2.4	clause 7.1.4.2.5	–	–
PLME-SET-TRX-MODE	clause 7.1.4.2.6	clause 7.1.4.2.7	–	–
PLME-SET	clause 7.1.4.2.8	clause 7.1.4.2.9	–	–

7.1.4.2.1 PLME-SFD.indication

The PLME-SFD.indication primitive indicates the reception of a start of frame delimiter from the PHY to the local MAC sublayer.

7.1.4.2.1.1 Semantics for the service primitive

The semantics of the PLME-SFD.indication primitive is as follows:

```
PLME-SFD.indication      (
                         psduChannel,
                         psduRate
                         )
```

The table below specifies the parameters for the PD-DATA.indication primitive.

Table 7-17 – PLME-SFD.indication parameters

Name	Type	Valid range	Description
psduChannel	Enumeration	A, B, or C	The physical channel from which the PSDU was received.
psduRate	Enumeration	R1, R2, or R3	The data rate of the received PSDU.

7.1.4.2.1.2 When generated

The PLME-SFD.indication primitive is generated by the PHY layer management entity (PLME) and issued to the local MAC sublayer through the PLME-SAP whenever a start of frame delimiter is detected by the PHY.

7.1.4.2.1.3 Effect on receipt

The MAC sublayer is notified of the reception of a start of frame delimiter.

7.1.4.2.2 PLME-GET-CCA.request

The PLME-GET-CCA.request primitive requests CCA on a given channel.

7.1.4.2.2.1 Semantics for the service primitive

The semantics of the PLME-GET-CCA.request primitive is as follows:

```
PLME-GET-CCA.request     (
                         channel
                         )
```

The table below specifies the parameters for the PLME-GET-CCA.request primitive.

Table 7-18 – PLME-GET-CCA.request parameters

Name	Type	Valid range	Description
Channel	Enumeration	Valid channel selection A, B, or C according to the applied RF profile (see Table 7-1)	The physical channel on which the CCA is to be performed.

7.1.4.2.2.2 When generated

The PLME-GET-CCA.request primitive is generated by the MLME and issued to its PLME to obtain information from the PHY MIB.

7.1.4.2.2.3 Effect on receipt

On receipt of the PLME-GET-CCA.request primitive, the PLME will attempt to perform a CCA on the given channel. When the CCA is completed, the PLME will issue a PLME-GET-CCA.confirm with a status of CCA_CLEAR, CCA_NOT-CLEAR or ERROR.

7.1.4.2.3 PLME-GET-CCA.confirm

The PLME-GET-CCA.confirm primitive reports the result of a CCA request from the PHY MIB.

7.1.4.2.3.1 Semantics for the service primitive

The semantics of the PLME-GET.confirm primitive is as follows:

```
PLME-GET-CCA.confirm  (
                      status
                      )
```

Table 7-19 specifies the parameters for the PLME-GET-CCA.confirm primitive.

Table 7-19 – PLME-GET-CCA.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	CCA_CLEAR, CCA_NOT_CLEAR or RX_OFF	The result of the CCA request.

7.1.4.2.3.2 When generated

The PLME-GET-CCA.confirm primitive is generated by the PLME and issued to its MLME in response to a PLME-GET-CCA.request primitive. The PLME-GET.CCA.confirm primitive will return a status of either CCA_CLEAR, CCA_NOT-CLEAR or RX_OFF. The RX_OFF status is given if the transceiver is not in RX mode (and unable to perform a CCA).

7.1.4.2.3.3 Effect on receipt

On receipt of the PLME-GET.confirm primitive, the MLME is notified of the results of its request to perform a CCA. If the request was successful the status will be CCA_CLEAR or CCA_NOT_CLEAR depending on channel availability. If the request failed, the status parameter will indicate the error.

7.1.4.2.4 PLME-GET.request

The PLME-GET.request primitive requests information about a given PHY MIB attribute.

7.1.4.2.4.1 Semantics for the service primitive

The semantics of the PLME-GET.request primitive is as follows:

```
PLME-GET.request  (
                  MIBAttribute
                  )
```

Table 7-20 specifies the parameters for the PLME-GET.request primitive.

Table 7-20 – PLME-GET.request parameters

Name	Type	Valid range	Description
MIBAttribute	Enumeration	See Table 7-28	The identifier of the PHY MIB attribute to get.

7.1.4.2.4.2 When generated

The PLME-GET.request primitive is generated by the MLME and issued to its PLME to obtain information from the PHY MIB.

7.1.4.2.4.3 Effect on receipt

On receipt of the PLME-GET.request primitive, the PLME will attempt to retrieve the requested PHY MIB attribute from its database. If the identifier of the MIB attribute is not found in the database, the PLME will issue the PLME-GET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE.

If the requested PHY MIB attribute is found, the PLME will issue the PLME-GET.confirm primitive with a status of SUCCESS.

7.1.4.2.5 PLME-GET.confirm

The PLME-GET.confirm primitive reports the result of an information request from the PHY MIB.

7.1.4.2.5.1 Semantics for the service primitive

The semantics of the PLME-GET.confirm primitive is as follows:

```
PLME-GET.confirm        (
                        status,
                        MIBAttribute,
                        MIBAttributeValue
                        )
```

Table 7-10 specifies the parameters for the PLME-GET.confirm primitive.

Table 7-21 – PLME-GET.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS or UNSUPPORTED_ATTRIBUTE	The result of the request for PHY MIB attribute information.
MIBAttribute	Enumeration	Refer to Table 7-28	The identifier of the PHY MIB attribute to get.
MIBAttributeValue	Various	Attribute specific	The value of the indicated PHY MIB attribute to get.

7.1.4.2.5.2 When generated

The PLME-GET.confirm primitive is generated by the PLME and issued to its MLME in response to a PLME-GET.request primitive. The PLME-GET.confirm primitive will return a status of either SUCCESS, indicating that the request to read a PHY MIB attribute was successful, or an error code of UNSUPPORTED_ATTRIBUTE. The reasons for these status values are fully described in clause 7.1.4.2.3.3.

7.1.4.2.5.3 Effect on receipt

On receipt of the PLME-GET.confirm primitive, the MLME is notified of the results of its request to read a PHY MIB attribute. If the request to read a PHY MIB attribute was successful, the status parameter is set to SUCCESS. Otherwise, the status parameter will indicate the error.

7.1.4.2.6 PLME-SET-TRX-MODE.request

The PLME-SET-TRX-MODE.request primitive requests that the PHY entity changes the internal operating mode of the transceiver. The transceiver will have three main modes:

– transceiver disabled (TRX_OFF)

– transmitter enabled (TX_ON)

– receiver enabled (RX_ON).

7.1.4.2.6.1 Semantics for the service primitive

The semantics of the PLME-SET-TRX-MODE.request primitive is as follows:

```
PLME-SET-TRX-MODE.request    (
                              mode
                              )
```

Table 7-22 specifies the parameters for the PLME-SET-TRX-MODE.request primitive.

Table 7-22 – PLME-SET-TRX-MODE.request parameters

Name	Type	Valid range	Description
Mode	Enumeration	RX_ON, TRX_OFF, FORCE_TRX_OFF, or TX_ON	The new mode in which to configure the transceiver.

7.1.4.2.6.2 When generated

The PLME-SET-TRX-MODE.request primitive is generated by the MLME and issued to its PLME when the current operational mode of the transceiver needs to be changed.

7.1.4.2.6.3 Effect on receipt

On receipt of the PLME-SET-TRX-MODE.request primitive, the PLME will cause the PHY to change to the requested mode. If the mode change is accepted, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status of SUCCESS.

If this primitive requests a mode that the transceiver has already configured, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status indicating the current mode, i.e., RX_ON, TRX_OFF, or TX_ON.

If this primitive is issued with the RX_ON or TRX_OFF argument and the PHY is busy transmitting a PPDU, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status BUSY_TX. When issuing BUSY_TX the PHY will ignore the mode request and the MAC shall reissue the PLME-SET-TRX-MODE.request if a mode shift is still intended.

If this primitive is issued with the TX_ON or TRX_OFF argument and the PHY is in RX_ON mode and is currently receiving data, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status BUSY_RX. When issuing BUSY_RX the PHY will ignore the mode request and the MAC shall reissue the PLME-SET-TRX-MODE.request if a mode shift is still intended.

If this primitive is issued with FORCE_TRX_OFF, the PHY will cause the PHY to go to the TRX_OFF mode irrespective of the mode the PHY is in.

7.1.4.2.7 PLME-SET-TRX-MODE.confirm

The PLME-SET-TRX-MODE.confirm primitive reports the result of a request to change the internal operating mode of the transceiver.

7.1.4.2.7.1 Semantics for the service primitive

The semantics of the PLME-SET-TRX-MODE.confirm primitive is as follows:

```
PLME-SET-TRX-MODE.confirm   (
                            status
                            )
```

Table 7-23 specifies the parameters for the PLME-SET-TRX-MODE.confirm primitive.

Table 7-23 – PLME-SET-TRX-MODE.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS, RX_ON, TRX_OFF, TX_ON, BUSY_RX, or BUSY_TX	The result of the request to change the mode of the transceiver.

7.1.4.2.7.2 When generated

The PLME-SET-TRX-MODE.confirm primitive is generated by the PLME and issued to its MLME after attempting to change the internal operating mode of the transceiver.

7.1.4.2.7.3 Effect on receipt

The MLME is notified of the result of its request to change the internal operating mode of the transceiver. A status value of SUCCESS indicates that the internal operating mode of the transceiver was accepted. A status value of RX_ON, TRX_OFF, or TX_ON indicates that the transceiver is already in the requested internal operating mode. A status value of BUSY_TX is issued when the PHY is requested to change its mode while transmitting. A status value of BUSY_RX is issued when the PHY is requested to change its mode while receiving.

7.1.4.2.8 PLME-SET.request

The PLME-SET.request primitive attempts to set the indicated PHY MIB attribute to the given value.

7.1.4.2.8.1 Semantics for the service primitive

The semantics of the PLME-SET.request primitive is as follows:

```
PLME-SET.request   (
                   MIBAttribute,
                   MIBAttributeValue
                   )
```

Table 7-24 specifies the parameters for the PLME-SET.request primitive.

Table 7-24 – PLME-SET.request parameters

Name	Type	Valid range	Description
MIBAttribute	Enumeration	See clause 7.1.5.2	The identifier of the MIB attribute to set.
MIBAttributeValue	Various	Attribute specific	The value of the indicated MIB attribute to set.

7.1.4.2.8.2 When generated

The PLME-SET.request primitive is generated by the MLME and issued to its PLME to write the indicated PHY MIB attribute.

7.1.4.2.8.3 Effect on receipt

On receipt of the PLME-SET.request primitive, the PLME will attempt to write the given value to the indicated PHY MIB attribute in its database. If the MIBAttribute parameter specifies an attribute that is not found in the database (see Table 7-28), the PLME will issue the PLME-SET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE. If the MIBAttibuteValue parameter specifies a value that is out of the valid range for the given attribute, the PLME will issue the PLME-SET.confirm primitive with a status of INVALID_PARAMETER.

If the requested PHY MIB attribute is successfully written the new value will be taken in use immediately and the PLME will issue the PLME-SET.confirm primitive with a status of SUCCESS.

7.1.4.2.9 PLME-SET.confirm

The PLME-SET.confirm primitive reports the results of the attempt to set an MIB attribute.

7.1.4.2.9.1 Semantics for the service primitive

The semantics of the PLME-SET.confirm primitive is as follows:

```
PLME-SET.confirm (
                 status,
                 MIBAttribute
                 )
```

Table 7-25 specifies the parameters for the PLME-SET.confirm primitive.

Table 7-25 – PLME-SET.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS, UNSUPPORTED_ATTRIBUTE, or INVALID_PARAMETER	The status of the attempt to set the request MIB attribute.
MIBAttribute	Enumeration	Refer to Table 7-28	The identifiers of the MIB attribute being confirmed.

7.1.4.2.9.2 When generated

The PLME-SET.confirm primitive is generated by the PLME and issued to its MLME in response to a PLME-SET.request primitive. The PLME-SET.confirm primitive will return a status of either SUCCESS, indicating that the requested value was written to the indicated PHY MIB attribute, or an error code of UNSUPPORTED_ATTRIBUTE or INVALID_PARAMETER. The reasons for these status values are fully described in clause 7.1.4.2.8.3.

7.1.4.2.9.3 Effect on receipt

On receipt of the PLME-SET.confirm primitive, the MLME is notified of the result of its request to set the value of a PHY MIB attribute. If the requested value was written to the indicated PHY MIB attribute, the status parameter is set to SUCCESS. Otherwise, the status parameter will indicate the error.

7.1.4.3 PHY enumerations description

Table 7-26 shows PHY enumeration values defined in the PHY specification.

Table 7-26 – PHY enumerations description

Enumeration	Description
BUSY	The CCA attempt has detected a busy channel.
BUSY_RX	The transceiver is asked to change its mode while receiving.
BUSY_TX	The transceiver is asked to change its mode while transmitting.
CCA_CLEAR	The CCA indicates a clear channel.
FORCE_TRX_OFF	The transceiver is to be switched off.
IDLE	The CCA has detected an idle channel.
INVALID_PARAMETER	A SET/GET request was issued with a parameter that is out of range.
CCA_NO_CLEAR	The CCA indicates a busy channel.
RX_ON	The transceiver is in or is to be configured into the receiver enabled mode.
RX_OFF	Status from a CCA request when not in RX mode.
SUCCESS	A SET/GET or a transceiver mode change was successful.
TRX_OFF	The transceiver is in or is to be configured into the transceiver disabled mode.
TX_ON	The transceiver is in or is to be configured into the transmitter enabled mode.
UNSUPPORTED_ATTRIBUTE	A SET/GET request was issued with the identifier of an attribute that is not supported.

7.1.5 PHY constants and MIB attributes

This clause specifies the constants and attributes required by the PHY.

7.1.5.1 PHY constants

The constants that define the characteristics of the PHY are presented in Table 7-27. These constants are hardware dependent and cannot be changed during operation.

Table 7-27 – PHY constants

Constants	Description	Value
aMaxPHYPacketSizeR1	The maximum PSDU size (in bytes) the PHY shall be able to receive at the data rate R1.	64 byte
aMaxPHYPacketSizeR2	The maximum PSDU size (in bytes) the PHY shall be able to receive at the data rate R2.	64 byte
aMaxPHYPacketSizeR3	The maximum PSDU size (in bytes) the PHY shall be able to receive at the data rate R3.	170 byte
aTurnaroundTimeTXRX	TX-to-RX maximum turnaround time (see clause 7.1.2.5.8).	1 ms
aTurnaroundTimerRXTX	RX-to-TX minimum turnaround time (see clause 7.1.2.5.9).	1 ms

7.1.5.2 PHY MIB attributes

The PHY management information base (MIB) comprises the attributes required to manage the PHY. Each of these attributes can be read or written using the PLME-GET.request and PLME-SET.request primitives, respectively. The attributes contained in the PHY MIB are presented in Table 7-28.

Table 7-28 – PHY MIB attributes

Attribute	Type	Range	Description
phyCurrentTxChannel	Enumeration	A, B, C	The TX channel to use (refer to Table 7-2).
phyMapChannelA	Enumeration	Available RF profiles (See Table 7-1)	Apply RF profile to channel A.
phyMapChannelB	Enumeration	Available RF profiles (See Table 7-1)	Apply RF profile to channel B.
phyMapChannelC	Enumeration	Available RF profiles (See Table 7-1)	Apply RF profile to channel C.
phyTransmitPower	Enumeration	Valid output power levels (Refer to clause 7.1.2.5.2)	The transmit power to use.

8 MAC layer specification

8.1 MAC specifications of the 'sub 1 GHz' transceivers

8.1.1 General

The MAC layer defined in this specification is a low bandwidth half duplex protocol designed for reliable wireless communication in a low-cost control network. The technology targets real-time applications which are not time critical in nature. The MAC layer enables long battery lifetimes and supports robust mesh routing with low overhead.

The MAC has the following characteristics:

– unique network ID number (HomeID)
– up to 232 nodes in one network
– collision avoidance algorithm
– backoff algorithm
– automatic retransmission for reliable data transfer
– support for low-power operation via dedicated wakeup patterns.

MAC frames carry one small header in order to conserve bandwidth. While presented as one header, a few fields are used by the network layer and higher layers. These fields are carried transparently by the MAC layer. Most of these fields are ignored by the MAC layer.

8.1.1.1 Bootstrapping

Nodes have to be paired with other nodes in order to form a network. This process is known as inclusion and is outside the scope of this Recommendation. A unique 32-bit identifier called the HomeID is used to separate networks from each other.

NodeIDs are unique within a network. A NodeID is an 8 bit short address. A primary node hands out NodeIDs to all other nodes joining the network.

During inclusion, a node uses the uninitialized NodeID (0x00) and a random pseudo-unique HomeID for identification.

8.1.1.2 Features of the MAC layer

The features of the MAC sublayer are channel access, frame validation, acknowledged frame delivery, and retransmission.

The MAC sublayer provides two services: the MAC data service, accessed through the MAC sublayer data service access point (MD-SAP) and the MAC management service interfacing with the MAC layer management entity (MLME) service access point (MLME-SAP). The MAC data service enables the transmission and reception of MAC protocol data units (MPDUs) across the PHY data service.

8.1.1.3 Concept of primitives

This clause provides a brief overview of the concept of service primitives (operations). Refer to [b-ITU-T X.210] for more detailed information.

The services of a layer are the capabilities it offers to the user in the next higher layer or sublayer by building its functions on the services of the next lower layer. This concept is illustrated in Figure 8-1, showing the service hierarchy and the relationship of the two correspondent N-users and their associated N-layer (or sublayer) peer protocol entities.

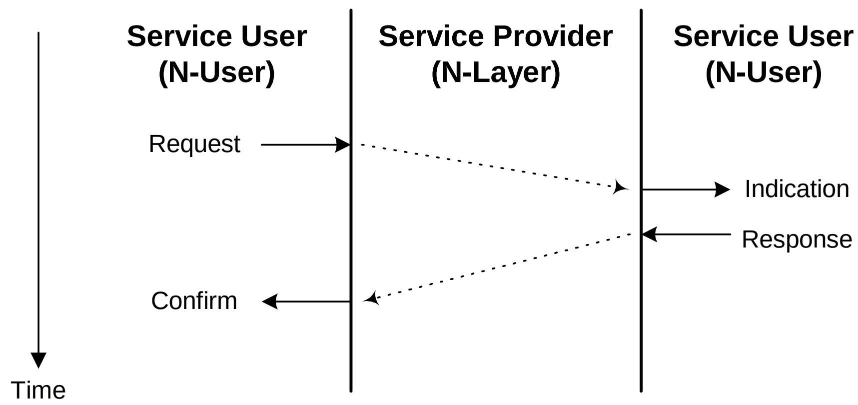

Figure 8-1 – Service primitives

The services are specified by describing the information flow between the N-user and the N-layer. This information flow is modelled by discrete, instantaneous events, which characterize the provision of a service. Each event consists of passing a service primitive from one layer to the other through a service access point (SAP) associated with an N-user. Service primitives convey the required information by providing a particular service. These service primitives are an abstraction because they specify only the provided service rather than the means by which it is provided. This definition is independent of any other interface implementation.

Services are specified by describing the service primitives and parameters that characterize it. A service may have one or more related primitives that constitute the activity that is related to that particular service. Each service primitive may have zero or more parameters that convey the information required to provide the service.

A primitive can be one of four generic types:

– Request: the request primitive is passed from the N-user to the N-layer to request that a service is initiated.
– Indication: the indication primitive is passed from the N-layer to the N-user to indicate an internal N-layer event that is significant to the N-user. This event may be logically related to a remote service request, or it may be caused by an N-layer internal event.
– Response: the response primitive is passed from the N-user to the N-layer to complete a procedure previously invoked by an indication primitive.
– Confirm: the confirm primitive is passed from the N-layer to the N-user to convey the results of one or more associated previous service requests.

8.1.1.4 Functional overview

The MAC features of an ITU-T G.9959 network include data transfer model, frame structure, robustness and power consumption.

8.1.1.4.1 Frame formats

A number of frame formats are defined. In most cases, the MAC layer shall forward frames to the higher layers for further processing. The exception is frames related to acknowledgment.

8.1.1.4.1.1 Singlecast

Figure 8-2 shows the structure of the general frame of the MAC sublayer. The MAC service data unit (MSDU) contains the payload data from the network layer. The MSDU is prepended with a MAC header (MHR) and appended with a MAC footer (MFR). The MHR contains HomeID, source node ID, MAC frame control field, frame length field and destination node ID (see also Figure A.3). The MFR contains a non-correcting frame check sequence (FCS). The MHR, MSDU, and MFR together form the MAC protocol data unit (MPDU).

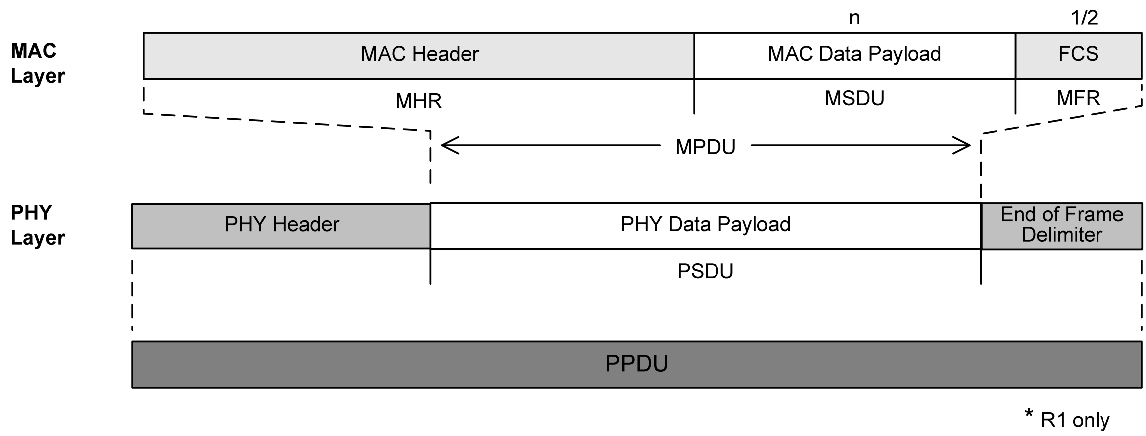

Figure 8-2 – General frame structure

The MPDU is passed to the PHY layer as a PHY service data unit (PSDU). The PSDU is prefixed with a start header (SHR) and an end header (EHR). The SHR contains the preamble sequence and start-of-frame delimiter (SFD) fields. The preamble sequence enables the RF receiver to achieve symbol synchronization. The SHR, PSDU and EHR together form the PHY protocol data unit.

8.1.1.4.1.2 Acknowledgment

The acknowledgment frame structure uses the general frame structure (Figure 8-2) with a zero-length MAC service data unit (MSDU).

8.1.1.4.1.3 Multicast

A multicast header explicitly identifies each target. A multicast frame may only be used in a direct range of destinations.

8.1.1.4.2 Network robustness

The ITU-T G.9959 MAC layer employs various mechanisms to ensure robustness in data transmission. These mechanisms of the MAC layer are backoff algorithm, frame acknowledgment, data verification and frame retransmission. In addition, the PHY layer clear channel assessment function is queried before transmission.

8.1.1.4.2.1 Clear channel assessment

A node shall query the availability of the channel from the PHY layer before transmitting. If the channel is found to be idle, the node may transmit its data. If the channel is found busy, the node shall wait for idle channel before transmitting.

8.1.1.4.2.2 Frame acknowledgment

A successful reception and validation of a frame may be confirmed with an acknowledgment. If the destination node receives a frame containing bit errors, the message is not acknowledged. An acknowledgment request shall be used to indicate the need for acknowledgment.

8.1.1.4.2.3 Retransmission

If two or more nodes transmit their frames simultaneously a collision may occur and the frames may not reach their destinations.

If the source node does not receive an acknowledgment, it shall assume that the transmission was unsuccessful and in response it shall retry the frame transmission up to *aMacMaxFrameRetries* times.

In order to avoid another collision each transmitter shall delay its retransmission with a random delay.

8.1.1.4.2.4 Multi-hop routing

The MAC layer supports multi hop routing. The routing algorithm is outside the scope of this Recommendation (i.e., should be specified at layers above the MAC layer).

8.1.1.4.2.5 Data validation

An 8-bit non-correcting frame check sequence (FCS) mechanism is employed to detect bit errors for data rates R1 and R2. A 16-bit non-correcting frame check sequence (FCS) mechanism is employed to detect bit errors for data rate R3. For a description of the data rates R1, R2 and R3 please refer to clause 7.1.2.3.

8.1.1.4.3 Power consumption considerations

Battery-powered nodes will spend most of their operational life in sleep mode; however, some nodes will periodically wake up and poll other nodes (also known as a mailbox) to determine whether messages are pending.

Other low-power nodes require a faster and more responsive behaviour than can be achieved with a mailbox. Such a node will operate in a mode in which it is frequently listening for messages. Communication with a frequently listening node is described in clause 8.1.1.4.3.1.

8.1.1.4.3.1 Communication with a node in frequently listening mode

Battery-powered devices like drape controllers and electronic door locks should be reachable at any time. This specification defines a transmission mode allowing an originator to send an extended preamble pattern that can be prepended to a singlecast frame. Nodes that are listening at regular short intervals are said to operate in FL (frequently listening) mode.

8.1.2 MAC sublayer service specification

The MAC sublayer provides an interface between the network layer and the PHY layer. The MAC layer management entity (MLME) provides the service interfaces through which layer management functions may be invoked. The MLME is responsible for maintaining a database of managed objects pertaining to the MAC sublayer. This database is referred to as the MAC sublayer management information base (MAC MIB). Figure 8-3 depicts the components and interfaces of the MAC sublayer.

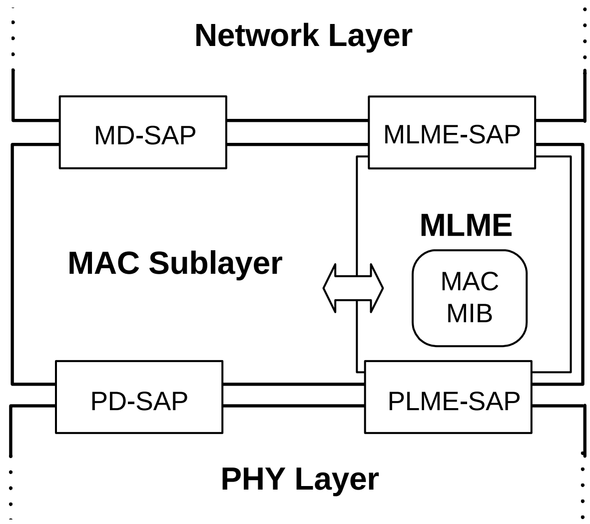

Figure 8-3 – MAC sublayer reference model

The MAC sublayer provides two services to the network layer, accessed through two service access points (SAPs):

– The MAC data service, accessed through the MD-SAP, and

– The MAC management service, accessed through the MLME-SAP.

8.1.2.1 MAC data service

The MD-SAP supports the transport of network layer protocol data units (NPDU) between peer network layer entities. Table 8-1 lists the primitives supported by the MD-SAP. The primitives are discussed in the clauses referenced in the table.

Table 8-1 – MD-SAP primitives

MD-SAP Primitive	Request	Confirm	Indication
MD-DATA	8.1.2.1.1	8.1.2.1.2	8.1.2.1.3

8.1.2.1.1 MD-DATA.request

The MD-DATA.request primitive requests the transfer of an NPDU (i.e., MSDU) from a local network layer entity to one or more peer network layer entities.

8.1.2.1.1.1 Semantics of the service primitive

The semantics of the MD-DATA. request primitive is as follows:

MD-DATA.request (SrcHomeID,
SrcNodeID,
DstNodeID,
msduLength,
msdu,
SequenceNumber,
TxType,
TxOptions,
BeamOption,
RateOption,
ChannelOption,
MulticastOption
)

Table 8-2 specifies the parameters for the MD-DATA.request primitive.

Table 8-2 – MD-Data.request parameters

Name	Type	Valid range	Description
SrcHomeID	Set of bytes	0x00000000 – 0xFFFFFFFF	The 4-byte network identifier of the node from which the MSDU is being transferred.
SrcNodeID	Byte	0x00 .. 0xE8	The 1-byte individual node address of the node from which the MSDU is being transferred. 0x00: Uninitialized NodeID 0x01 – 0xE8: NodeID 0xE9 – 0xFE: Reserved 0xFF: Broadcast NodeID
DstNodeID	Byte	0x00 .. 0xFF	The 1-byte node identifier of the node to which the MSDU is being transferred.
msduLength	Byte	≤ *aMacMaxMSDUSizeX*	The number of bytes contained in the MSDU to be transmitted by the MAC sublayer entity.
msdu	Set of bytes	–	The set of bytes forming the MSDU to be transmitted by the MAC sublayer entity.
SequenceNumber	Set of bits	–	The unique number of this frame. Use same value for retransmissions.
TxType	Byte	0000.xxxx (where x can be 0 or 1)	The transmission type for this MSDU. 0x01 = Singlecast or broadcast transmission 0x02 = Multicast transmission 0x03 = Acknowledge transmission

Table 8-2 – MD-Data.request parameters

Name	Type	Valid range	Description
TxOptions	Byte	0000.xxxx (where x can be 0 or 1)	The transmission options for this MSDU. The control word is formed as a bitwise OR of one or more of the following values: 0x01 = Routed frame transmission 0x02 = Acknowledged transmission 0x04 = Low power transmission (Note 1).
BeamOption	Byte	0x00 .. 0x04	The battery support options for this MSDU. 0x00: No beam 0x01: Repeated beam 0x02: Short Continuous beam 0x03: Long Continuous beam 0x04: Fragmented beam
ChannelOption	Enumeration	A,B, or C (Note 2)	The channel to use for this MSDU.
RateOption	Enumeration	R1, R2 or R3 Validity depends on the applied RF profile for the channel	The data rate to use for this MSDU.
MulticastOption	Byte + Bitmap (30 bytes)	–	If TxType is "Multicast transmission", this bitmap indicates the intended recipients.
NOTE 1 – Bit 2 (0x04) controls which transmission power level the MAC layer requests from the PHY layer. NOTE 2 – Channels A, B and C are preconfigured to RF profiles from the list presented in clause 7.1.2.1 via MAC MIB attributes defined in Table 8-20. Channel configuration MAC MIB profiles map directly to PHY MIB profiles.			

8.1.2.1.1.2 When generated

The MD-DATA.request primitive is generated by a local network layer entity when a data NPDU (i.e., MSDU) is to be transferred to one or more peer network layer entities.

The TxOptions.LowPower is an option targeting the PHY layer. The option makes the MAC choose between the two MAC constants aMacTxPhyPowerLevelLow and aMacTxPhyPowerLevelNormal.

8.1.2.1.1.3 Effects on receipt

The MAC sublayer entity begins the transmission of the supplied MSDU.

The MAC sublayer builds an MPDU to transmit from the supplied parameters. The TxOptions parameters indicate optional parameters on how the MAC sublayer data service transmits the supplied MSDU.

The MAC sublayer shall check for PHY clear channel access (CCA, see clause 8.1.5.1.1). If the macClearChannel attribute is TRUE the MAC sublayer shall enable the transmitter by issuing the PLME-SET-TRX-MODE.request primitive with a mode of TX_ON to the PHY. On receipt of the PLME-SET-TRX-MODE.confirm primitive with a status of either SUCCESS or TX_ON the constructed MPDU may then be transmitted by issuing the PD-DATA.request primitive. Finally, on receipt of the PD-DATA.confirm primitive, the MAC sublayer shall disable the transmitter by issuing the PLME-SET-TRX-MODE.request primitive with a mode of RX_ON to the PHY.

If the TxOptions parameter specifies that acknowledged transmission is required, the MAC sublayer shall enable its receiver immediately following the transmission of the MPDU and wait for an acknowledgment from the recipient for at least *aMacMinAckWaitDuration* symbols. If the MAC sublayer does not receive an acknowledgment within this time, it shall retransmit the MPDU as defined by *aMacMaxFrameRetries*. If the MAC sublayer still does not receive an acknowledgment from the recipient, it shall discard the MSDU and issue the MD-DATA.confirm primitive with a status of NO_ACK.

If the MPDU was successfully transmitted the MAC sublayer will issue the MD-DATA.confirm primitive with a status of SUCCESS.

If the MPDU could not be transmitted due to an occupied channel (clause 8.1.5.1.1) the MAC sublayer shall issue the MD-DATA.confirm primitive with a status of NO_CCA.

If any parameter in the MD-DATA.request primitive is not supported or is out of range, the MAC sublayer shall issue the MD-DATA.confirm primitive with a status of INVALID_PARAMETER.

If the MSDU length is longer than *aMacMaxMSDUSizeX* the MAC sublayer shall issue the MD-DATA.confirm primitive with a status of FRAME_TOO_LONG.

The MD-Data.request parameters are used to construct the MAC Header (MHR) see clause 8.1.3.

8.1.2.1.2 MD-DATA.confirm

The MD-DATA.confirm primitive reports the status of a data NPDU (MSDU) transfer.

8.1.2.1.2.1 Semantics of the service primitive

The semantics of the MD-DATA.confirm primitive is as follows:

```
MD-DATA.confirm  (
                  status
                  )
```

Table 8-3 specifies the parameters for the MD-DATA.confirm primitive.

Table 8-3 – MD-Data.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS, NO_ACK, INVALID_PARAMETER, NO_CCA or FRAME_TOO_LONG	The status of the last MSDU transmission.

8.1.2.1.2.2 When generated

The MD-DATA.confirm primitive is generated by the MAC sublayer entity in response to an MD-DATA.request primitive. The MD-DATA.confirm primitive returns a status of either SUCCESS, NO_ACK, INVALID_PARAMETER, NO_CCA or FRAME_TOO_LONG.

The primitive may be delayed if the MAC has to wait for an acknowledgment. If the frame is transmitted using repeated beam or fragmented beam (refer to clauses 8.1.3.11 and 8.1.3.12) the delay may be more than one second.

8.1.2.1.2.3 Effects on receipt

The network layer of the initiating node is notified of the result of its request to transmit. If the transmission attempt was successful, the status parameter will be set to SUCCESS. Otherwise, the status parameter will indicate the error.

8.1.2.1.3 MD-DATA.indication

The MD-DATA.indication primitive indicates the reception of a data NPDU (i.e., MSDU) from the MAC sublayer to the local network layer entity.

8.1.2.1.3.1 Semantics of the service primitive

The semantics of the MD-DATA.indication primitive is as follows:

```
MD-DATA.indication      (
                        FrmType,
                        SrcHomeID,
                        SrcNodeID,
                        DstNodeID,
                        msduLength,
                        msdu,
                        SequenceNumber,
                        ChannelOption,
                        RateOption,
                        MulticastOption
                        )
```

Table 8-4 specifies the parameters for the MD-DATA.indication primitive.

FrmType indicates if the data delivered to the network layer is a data frame or a beam fragment. Refer to clause 8.1.3.10 for details on beam fragments.

In case of a beam fragment the SrcHomeID field is of the form "0x55xxxxxx" or "0x54xxxxxx", and only the following fields carry data:

* SrcHomeID (Byte 1: BeamTag = 0x55 or 0x54, Byte 2: BeamInfo)

* ChannelOption

* RateOption

All other fields shall be ignored.

The SrcHomeID value "0x55xxxxxx" signals that the BeamInfo byte carries a destination NodeID.

The SrcHomeID value "0x54xxxxxx" signals that the BeamInfo byte carries a value that is reserved for future use. Thus this value shall be ignored.

Table 8-4 – MD-DATA.indication parameters

Name	Type	Valid range	Description
FrmType	Enumeration	0x01 – 0xFF	Code indicating the frame type: 0x00: Uninitialized type0x01: Singlecast frame 0x02: Multicast frame 0x03: Ack frame 0x04 – 0x07: Not to be used (Note) 0x08: Routed frame 0x09: Beam fragment 0x0A – 0xFF: Reserved

Table 8-4 – MD-DATA.indication parameters

Name	Type	Valid range	Description
SrcHomeID	Set of bytes	0x00000000 –0xFFFFFFFF The value 0x00000000 is reserved for un-initialized legacy nodes. No network may use the HomeID 0x00000000 as a network identifier.	The 4-byte network identifier of the node from which the MSDU was received.
SrcNodeID	Byte	0x00 – 0xFF The value 0x00 is reserved for un-initialized nodes.	The 1-byte individual node address of the node from which the MSDU was received. 0x00: Uninitialized NodeID 0x01 – 0xE8: NodeID 0xE9 – 0xFF: Reserved
DstNodeID (Not applicable for multicast frame type)	Byte	0 x 00–0 x FF	The 1-byte individual node address of the node from which the MSDU is being transferred. 0x00: Reserved 0x01 – 0xE8: NodeID 0xE9 – 0xFE: Reserved 0xFF: Broadcast NodeID
msduChannel	Enumeration	A, B, or C	The physical channel from which the MSDU was received.
msduRate	Enumeration	R1, R2, or R3	The bit rate of the received MSDU.
msduLength	Byte	≤ *aMacMaxMSDUSizeX*	The number of bytes contained in the MSDU.
msdu	Set of bytes	–	The set of bytes forming the MSDU.
SequenceNumber	Set of bits	–	The unique number of this frame. Use same value for retransmissions.
MulticastOption	Byte + Bitmap (30 bytes)	–	If TxType is "Multicast transmission", this bitmap indicates the intended recipients.
NOTE – These values shall not be used due to backward compatibility.			

8.1.2.1.3.2 When generated

The MD-DATA.indication primitive is generated by the MAC sublayer on receipt of a frame from the PHY layer. If the frame is free of errors, the frame shall be forwarded to the network layer.

8.1.2.1.3.3 Effects on receipt

The network layer is notified of the arrival of data.

Beam fragments shall also be forwarded to the network layer, which may then forward the beam fragment to higher layers. Higher layers of a node in FL mode may decide to re-enable sleep mode if the NodeID of a beam fragment is intended for another node.

8.1.2.1.4 Data service message sequence charts

Figure 8-4 illustrates the sequence of messages necessary for a successful data transfer between two nodes.

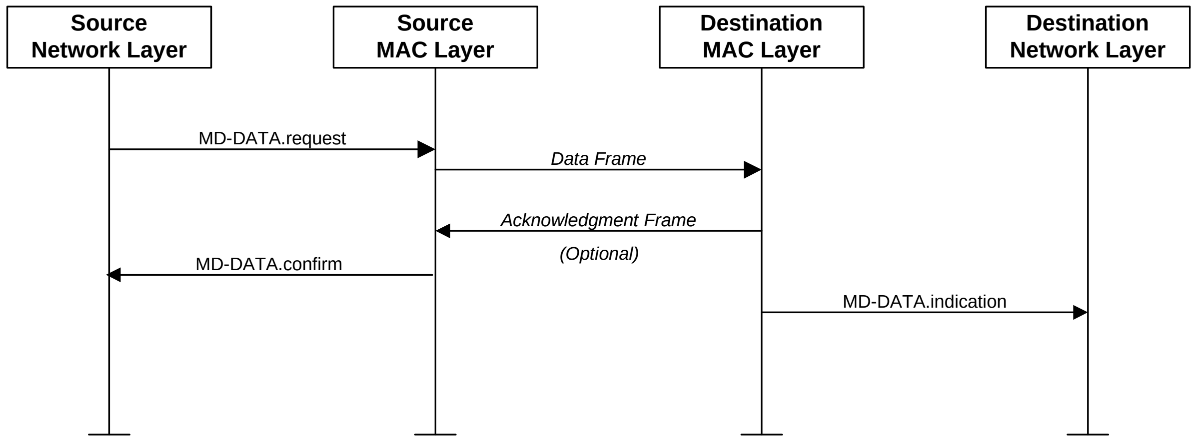

Figure 8-4 – MAC data service message sequence chart

8.1.2.2 MAC management service

The MLME-SAP allows the transport of management commands between the next higher layer and the MLME. Table 8-5 summarizes the primitives supported by the MLME through the MLME-SAP interface.

The primitives are discussed in the clauses referenced in the table.

Table 8-5 – MLME-SAP primitives

MLME-SAP Primitive	Request	Confirm	Indication	Response
MLME-GET	8.1.2.2.1	8.1.2.2.2	–	–
MLME-SET	8.1.2.2.3	8.1.2.2.4	–	–
MLME-RESET	8.1.2.2.5	8.1.2.2.6	–	–

8.1.2.2.1 MLME-GET.request primitives

The MLME-GET.request primitive requests information about a given MAC MIB attribute.

8.1.2.2.1.1 Semantics of the service primitive

The semantics of the MLME-GET.request primitive is as follows:

MLME-GET.request (
MACMACMIBAttribute
)

Table 8-6 specifies the parameters for the MLME-GET.request primitive.

Table 8-6 – MLME-GET.request parameters

Name	Type	Valid range	Description
MACMIBAttribute	Integer	See Table 8-20	The identifier of the MAC MIB attribute to read.

8.1.2.2.1.2 When generated

The MLME-GET.request primitive is generated by the next higher layer and issued to its MLME to obtain information from the MAC MIB.

8.1.2.2.1.3 Effects on receipt

On receipt of the MLME-GET.request primitive, the MLME shall retrieve the requested MAC MIB attribute from its database. If the identifier of the MAC MIB attribute is not found in the database, the MLME shall issue the MLME-GET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE.

If the requested MAC MIB attribute is successfully retrieved, the MLME shall issue the MLME-GET.confirm primitive with a status of SUCCESS and the MAC MIB attribute value.

8.1.2.2.2 MLME-GET.confirm primitives

The MLME-GET.confirm primitive reports the results of an information request from the MAC MIB.

8.1.2.2.2.1 Semantics of the service primitive

The semantics of the MLME-GET.confirm primitive is as follows:

```
MLME-GET.confirm (
                  status,
                  MACMIBAttribute,
                  MACMIBAttributeValue
                  )
```

Table 8-7 specifies the parameters for the MLME-GET.confirm primitive.

Table 8-7 – MLME-GET.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS or UNSUPPORTED_ATTRIBUTE	The result of the request for MAC MIB attribute information.
MACMIBAttribute	Integer	See Table 8-20	The identifier of the MAC MIB attribute that was read.
MACMIBAttributeValue	Various	Attribute specific, see Table 8-20	The value of the indicated MAC MIB attribute that was read.

8.1.2.2.2.2 When generated

The MLME-GET.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-GET.request primitive. This primitive returns a status of either SUCCESS, indicating that the request to read a MAC MIB attribute was successful, or an error code of UNSUPPORTED_ATTRIBUTE.

8.1.2.2.2.3 Effects on receipt

The MLME-GET.confirm primitive reports the results of the request to read a MAC MIB attribute. If the request to read a MAC MIB attribute was successful, the status parameter will be set to SUCCESS. Otherwise, the status parameter indicates the error.

8.1.2.2.3 MLME-SET.request primitives

The MLME-SET.request primitive attempts to write the given value to the indicated MAC MIB attribute.

8.1.2.2.3.1 Semantics of the service primitive

The semantics of the MLME-SET.request primitive is as follows:

```
MLME-SET.request  (
                  MACMIBAttribute,
                  MACMIBAttributeValue
                  )
```

Table 8-8 specifies the parameters for the MLME-SET.request primitive.

Table 8-8 – MLME-SET.request parameters

Name	Type	Valid range	Description
MACMIBAttribute	Integer	See Table 8-20	The identifier of the MAC MIB attribute to write.
MACMIBAttributeValue	Various	Attribute specific, see Table 8-20	The value to write to the indicated MAC MIB attribute.

8.1.2.2.3.2 When generated

The MLME-SET.request primitive is generated by the next higher layer and issued to its MLME to write the indicated MAC MIB attribute.

8.1.2.2.3.3 Effects on receipt

On receipt of the MLME-SET.request primitive, the MLME shall write the given value to the indicated MAC MIB attribute in its database. If the MACMIBAttribute parameter specifies an attribute that is not found in Table 8-20, the MLME shall issue the MLME-SET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE. If the MACMIBAttributeValue parameter specifies a value that is out of the valid range for the given attribute, the MLME shall issue the MLME-SET.confirm primitive with a status of INVALID_PARAMETER.

If the requested MAC MIB attribute is successfully written, the MLME shall issue the MLME-SET.confirm primitive with a status of SUCCESS.

8.1.2.2.4 MLME-SET.confirm primitives

The MLME-SET.confirm primitive reports the results of an attempt to write a value to a MAC MIB attribute.

8.1.2.2.4.1 Semantics of the service primitive

The semantics of the MLME-SET.confirm primitive is as follows:

```
MLME-SET.confirm        (
                        status,
                        MACMIBAttribute
                        )
```

Table 8-9 specifies the parameters for the MLME-SET.confirm primitive.

Table 8-9 – MLME-SET.confirm parameters

Name	Type	Valid range	Description
status	Enumeration	SUCCESS, UNSUPPORTED_ATTRIBUTE, or INVALID_PARAMETER	The result of the request to write the MAC MIB attribute.
MACMIBAttribute	Integer	Table 8-20	The identifier of the MAC MIB attribute that was written.

8.1.2.2.4.2 When generated

The MLME-SET.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-SET.request primitive. The MLME-SET.confirm primitive returns a status of either SUCCESS, indicating that the requested value was written to the indicated MAC MIB attribute, or an error code of UNSUPPORTED_ATTRIBUTE or INVALID_PARAMETER.

8.1.2.2.4.3 Effects on receipt

On receipt of the MLME-SET.confirm primitive, the next higher layer is notified of the result of its request to set the value of a MAC MIB attribute. If the requested value was written to the indicated MAC MIB attribute, the status parameter shall be set to SUCCESS. Otherwise, the status parameter indicates the error.

8.1.2.2.5 MLME-RESET.request primitives

The MLME-RESET.request primitive allows the next higher layer to request that the MLME performs a reset operation.

8.1.2.2.5.1 Semantics of the service primitive

The semantics of the MLME-RESET.request primitive is as follows:

```
MLME-RESET.request      (
                        SetDefaultMIB
                        )
```

Table 8-10 specifies the parameter for the MLME-RESET.request primitive.

Table 8-10 – MLME-RESET.request parameters

Name	Type	Valid range	Description
SetDefaultMIB	Boolean	TRUE or FALSE	If TRUE, the MAC sublayer is reset and all MAC MIB attributes are set to their default values.

8.1.2.2.5.2 When generated

The MLME-RESET.request primitive is generated by the next higher layer and issued to its MLME to request a reset of the MAC sublayer to its initial conditions. The MLME-RESET.request primitive is issued when a node is excluded from a network.

8.1.2.2.5.3 Effects on receipt

On receipt of the MLME-RESET.request primitive, the MLME shall issue the PLME-SET-TRX-STATE.request primitive with a state of TRX_OFF. On receipt of the PLME-SET-TRX-STATE.confirm primitive, the MAC sublayer shall then be set to its initial conditions, clearing all internal variables to their default values. If the SetDefaultMIB parameter is set to TRUE, the MAC MIB attributes are set to their default values.

If the PLME-SET-TRX-STATE.confirm primitive is successful, the MLME shall issue the MLME-RESET.confirm primitive with the status of SUCCESS. Otherwise, the MLME shall issue the MLME-RESET.confirm primitive with the status of DISABLE_TRX_FAILURE.

8.1.2.2.6 MLME-RESET.confirm primitives

The MLME-RESET.confirm primitive reports the results of the reset operation.

8.1.2.2.6.1 Semantics of the service primitive

The semantics of the MLME-RESET.confirm primitive is as follows:

```
MLME-RESET.confirm      (
                        status
                        )
```

Table 8-11 specifies the parameter for the MLME-RESET.confirm primitive.

Table 8-11 – MLME-RESET.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS or DISABLE_TRX_FAILURE	The result of the reset operation.

8.1.2.2.6.2 When generated

The MLME-RESET.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-RESET.request primitive and following the receipt of the PLME-SET-TRX-STATE.confirm primitive.

8.1.2.2.6.3 Effects on receipt

On receipt of the MLME-RESET.confirm primitive, the next higher layer is notified of its request to reset the MAC sublayer. This primitive returns a status of SUCCESS if the request to reset the MAC sublayer was successful. Otherwise, the status is set to DISABLE_TRX_FAILURE, indicating that the attempt to disable the transceiver was unsuccessful.

8.1.2.3 MAC enumeration description

This clause explains the meaning of the enumerations used in the primitives defined in the MAC sublayer specification. Table 8-12 shows a description of the MAC enumeration values.

Table 8-12 – MAC enumerations description

Enumeration	Description
SUCCESS	The requested operation was completed successfully.
DISABLE_TRX_FAILURE	The attempt to disable the transceiver has failed.
FRAME_TOO_LONG	The frame has a length that is greater than *aMacMaxMSDUSizeX*.
INVALID_PARAMETER	A parameter in the primitive is out of the valid range.
NO_ACK	No acknowledgment was received after *aMacMaxFrameRetries*.
NO_CCA	No clear channel access was possible after a period of *macCCARetryDuration*.
UNSUPPORTED_ATTRIBUTE	A SET/GET request was issued with the identifier of a MAC MIB attribute that is not supported.

8.1.3 MAC frame format

This clause specifies the format of the MAC frame (MPDU). Each MAC frame consists of the following basic components:

1. An MHR, which comprises addresses, frame control and length information.
2. A MAC data payload, of variable length, which contains information specific to the frame type.
   – Acknowledgment frames do not contain a payload.
3. An MFR, which contains a FCS.

The frames in the MAC sublayer are defined as a sequence of fields. All frame formats in this clause are depicted in the order in which they are transmitted by the PHY, from left to right, where the leftmost bit (bit 7) is transmitted first in time. Bits within each field are numbered from $k – 1$ (leftmost and most significant) down to 0 (rightmost and least significant), where the length of the field is k bits. If fields are longer than a single byte the most significant part is transmitted first.

The standard MAC frame format comprises an MHR, a MAC payload, and an MFR. The fields of the MHR appear in a fixed order. The general MAC frame shall be formatted as illustrated in Figure 8-5.

The MAC frame formats for channel configurations 1, 2 and channel configuration 3 are different. The differences are explained in the following clauses.

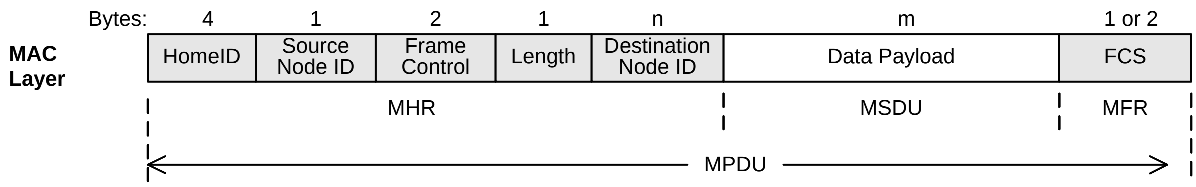

Figure 8-5 – General MAC frame format (channel configurations 1, 2)

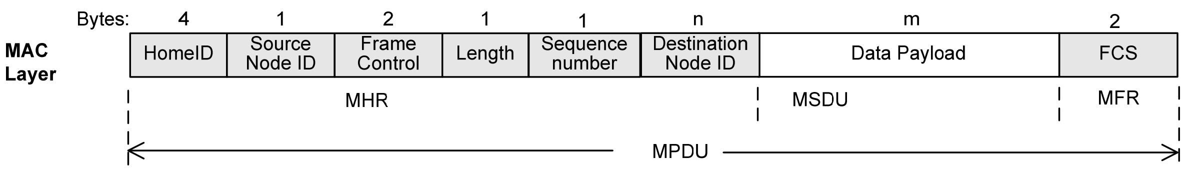

Figure 8-6 – General MAC frame format (channel configuration 3)

Multicast frames use a special format for addressing.

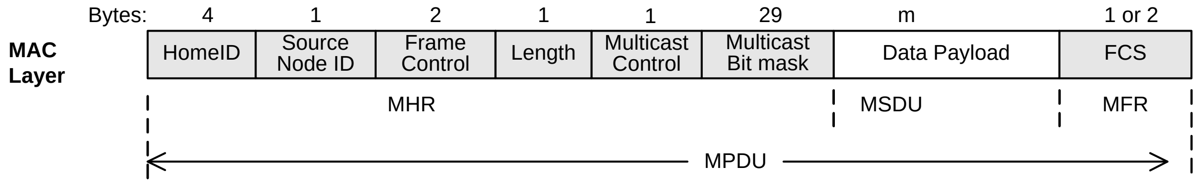

Figure 8-7 – Multicast MAC frame format (channel configurations 1, 2)

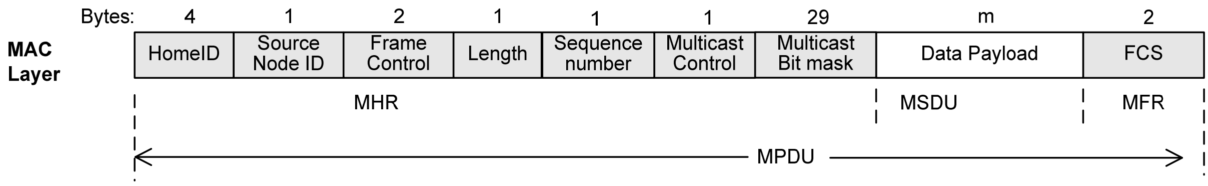

Figure 8-8 – Multicast MAC frame format (channel configuration 3)

Refer to clause 8.1.3.6.1 for details on transmission and processing of multicast frames.

8.1.3.1 HomeID

The HomeID identifier field is 4 bytes in length and specifies the unique network identifier. All nodes in a network have the same HomeID. The HomeID is assigned by a primary node during inclusion; refer to clause 8.1.1.1.

Byte \ Bit	7	6	5	4	3	2	1	0
0	HomeID (0) MSB							
1	HomeID (1)							
2	HomeID (2)							
3	HomeID (3) LSB							

Figure 8-9 – Format of a HomeID field

The MAC shall support configuration of a promiscuous mode in which all frames are forwarded to higher layers. Refer to clause 8.1.5.1.3.

8.1.3.2 Source NodeID

The source NodeID is an 8 bit unique identifier of a node. Together with the HomeID, the source NodeID identifies the node that originated the frame.

The NodeID is assigned by a primary node during inclusion; refer to clause 8.1.1.1.

Byte \ Bit	7	6	5	4	3	2	1	0
4	Source NodeID							

Figure 8-10 – Format of a source NodeID field

The source NodeID range is as shown in Table 8-13.

Table 8-13 – Source NodeID

NodeID	Node type
0x00	Uninitialized NodeID
0x01 – 0xE8	NodeID
0xE9 – 0xFF	Reserved

8.1.3.3 Frame control

The frame control field is 16 bits in length and contains information defining the frame type, addressing fields, and other control flags. The frame control field shall be formatted as illustrated in Figures 8-11 and 8-12.

Byte \ Bit	7	6	5	4	3	2	1	0
5	Routed	Ack Req	Low Power	Speed modified	Header Type			
6	Reserved	Beam control		Reserved	Sequence Number			

Figure 8-11 – Format of a frame control field (channel configurations 1, 2)

Byte \ Bit	7	6	5	4	3	2	1	0
5	Ack Req	Low Power	Reserved		Header Type			
6	Reserved	Beam control			Reserved			

Figure 8-12 – Format of a frame control field (channel configuration 3)

8.1.3.3.1 Routed subfield (channel configurations 1, 2 only)

The routed subfield is 1 bit in length. It shall be set to 0 when the frame is not routed and set to 1 when the frame is routed.

The information is used by the upper layers.

8.1.3.3.2 Ack Req Subfield

The Ack Req subfield is 1 bit in length. It shall be set to 1 when the source node wants the destination node to acknowledge the frame and set to 0 when no acknowledgment is needed. The information is used by the MAC layer to enable acknowledgment timeout and potential retransmission.

The MAC layer shall be able to react to an acknowledgment request in an incoming singlecast frame.

8.1.3.3.3 Low power subfield

The low power subfield is 1 bit in length. It shall be set by the source node. The information is used to configure the PHY prior to frame transmission. The bit informs a destination node that the actual frame was transmitted using low power.

A destination node may return an acknowledgment frame in low power in response to the low-power bit.

8.1.3.3.4 Speed modified subfield (channel configurations 1, 2 only)

The speed modified subfield is 1 bit in length. It shall be set to 1 if a frame is sent at a lower speed than supported by the source and destination. The field shall not be used for routed and multicast frames. The field shall be set to 0 if not used.

8.1.3.3.5 Header type subfield

The header type subfield defines the frame header type.

Table 8-14 – Header type

Header type	Frame
0x1	Singlecast frame
0x2	Multicast frame
0x3	Acknowledgment
0x4..0x7	Not to be used (Note)
0x8	Routed frame (channel configuration 3 only)
0x9..0xB	Not to be used (Note)
0xC..0xF	*Reserved*
NOTE – These fields shall not be used due to backward compatibility.	

A broadcast frame is a singlecast frame (header type 0x1) carrying destination NodeID = 0xFF; see clause 8.1.3.6.

8.1.3.3.6 Beam control

The interpretation of this sub-field shall depend on the channel configuration being used.

For nodes operating in channel configuration 1 or 2, bit 6 shall signal that the sending FL node shall be woken up with a repeated beam or a long continuous beam. Bit 5 shall signal that the sending FL node shall be woken up with a repeated beam or a short continuous beam.

For nodes operating in channel configuration 3, bit 6 shall signal that the sending FL node shall be woken up with a fragmented beam. Bit 5 is reserved.

8.1.3.3.7 Sequence number (channel configurations 1, 2 only)

The sequence number field is a number provided by higher layers when transmitting. The PHY and MAC layers shall forward the sequence number value unmodified. Upon reception of a frame, if acknowledgment is requested, the MAC layer shall return the received sequence number value in the acknowledgment frame.

The acknowledgment mechanism described in clause 8.1.5.1.4 shall check that the received sequence number in an acknowledgment matches the transmitted sequence number value.

The valid range is 0x1-0xf.

In order to support legacy implementations, a transmitting MAC entity shall accept the sequence number value zero in an acknowledgment frame irrespective of the sequence number transmitted. An ITU-T G.9959 MAC entity shall be able to return the sequence number value zero in response to an incoming singlecast frame with the sequence number value zero.

8.1.3.4 Length

The length field is 1 byte in length and indicates the length of the whole MPDU in bytes.

Byte \ Bit	7	6	5	4	3	2	1	0
7	Length							

Figure 8-13 – Format of a length field

The length is limited to *aMacMaxMSDUSizeX*. The actual values can be found in Table 8-18. A receiving node shall not accept more bytes than the maximum length allowed for the actual data rate.

8.1.3.5 Sequence number (channel configuration 3 only)

The sequence number field is a number provided by higher layers when transmitting. The PHY and MAC layers shall forward the sequence number value unmodified. Upon reception of a frame, if acknowledgment is requested, the MAC layer shall return the received sequence number value in the acknowledgment frame.

The acknowledgment mechanism described in clause 8.1.5.1.4 shall check that the received sequence number in an acknowledgment matches the transmitted sequence number value.

The valid range is 0x00-0xff.

8.1.3.6 Destination NodeID

The destination NodeID is used to address individual nodes. The NodeID allocation process is described in clause 8.1.1.1. Together with the HomeID the destination NodeID identifies the target node of the frame.

The MAC shall support the configuration of a promiscuous mode; forwarding all frames to higher layers.

The destination NodeID range is as shown in Table 8-15.

Table 8-15 – Destination NodeID

NodeID	Node type
0x00	Uninitialized NodeID
0x01 – 0xE8	NodeID
0xE9 – 0xFE	Reserved
0xFF	Broadcast NodeID

The destination NodeID 0xFF is a special broadcast value. The MAC layer shall always forward a broadcast frame to higher layers.

8.1.3.6.1 Multicast destination address

Multicast frames carry a destination bit map as shown in Figure 8-14.

Byte \ Bit	7	6	5	4	3	2	1	0
8	Address Offset			Number of Mask Bytes				
9	Mask Byte 0							
10	Mask Byte 1							
11	Mask Byte 2							
...	....							
37	Mask Byte 28							

Figure 8-14 – Multicast destination address format

A multicast frame cannot be acknowledged. The ACK request bit in the frame control field shall always be set to zero (see clause 8.1.3.3). A transmitting node shall set the address offset field to zero and mask bytes field to 29. A receiving node shall accept the values of the address offset field outlined in Table 8-16.

The address offset field is a 3 bit field. The encoding of the field is outlined in Table 8-16.

Table 8-16 – Multicast header address offset field

Address offset value	Bit address offset
0	0
1	32
2	64
3	96
4	128
5	160
6	192
7	224

Each bit of a mask byte represents a NodeID. Mask byte 0 represents NodeIDs 1 to 8.

As an example a mask byte 0 value of 0xC5 (in binary: 11000101) covers NodeIDs 1, 3, 7 and 8 while a mask byte 1 value of 0xC5 (in binary: 11000101) covers NodeIDs 9, 11, 15 and 16.

The destination NodeID can be calculated using the following formula.

$$\text{Destination NodeID} = \text{Bit Address offset} + \text{Mask Byte Number} \times 8 + \text{Mask Bit Number} + 1$$

8.1.3.7 Data payload

The frame payload field has a variable length and contains information specific to individual frames. An acknowledgment frame has no data payload.

8.1.3.8 FCS

An 8-bit non-correcting frame checksum is used for checking frame integrity for R1 and R2 data rates. The checksum is calculated from the HomeID field and to the data payload field as shown in Figure 8-15.

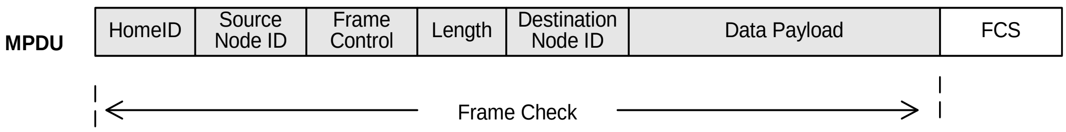

Figure 8-15 – Frame check coverage

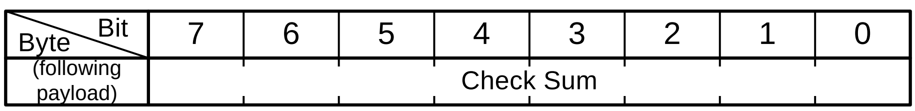

Figure 8-16 – Format of checksum field

The checksum is calculated as an odd (XOR'ed) checksum as shown in the following algorithm.

```
BYTE GenerateCheckSum(BYTE *Data, BYTE Length)
```

```
{
  BYTE CheckSum = 0xFF;
  for (; Length > 0; Length--)
  {
    CheckSum ^= *Data++;
  }
  return CheckSum;
}
```

8.1.3.9 16-bit CRC

A 16-bit non-correcting frame CRC is used for checking frame integrity for R3 data rate.

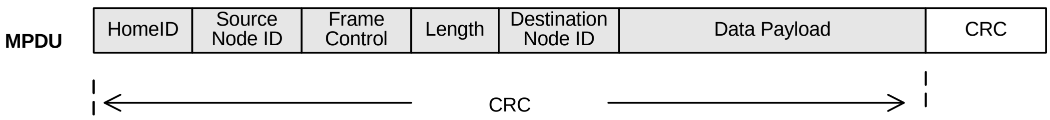

Figure 8-17 – CRC coverage

The CRC-16 generator polynomial is:

P(x) = x16+x12+x5+1, also known as CRC-CCITT

The CRC 16 is calculated over the whole frame, except for the preamble, SOF, and the CRC-16 field.

The CRC-16 generator is initialized to 1D0Fh before applying the first frame byte of a frame and no bits are appended to the frame data.

8.1.3.10 Beam frame format

Beam frames are used for fragmented beams as well as for repeated beams. In both cases a high number of beam frames are transmitted back to back. In case of repeated beams, a singlecast frame follows the beam.

Beam frames are used to get the attention of nodes operating in frequently listening (FL) mode with low latency. Each beam frame carries a preamble pattern and an SOF field, just like the field prepended by the PHY to the general MAC packet format. The SOF is followed by two bytes; the beam tag and the beam info. A receiving node may distinguish a general MAC packet from a beam frame by inspecting the MS byte of the HomeID. If this byte carries a beam tag (refer to Table 8-17) this is a beam frame and the following byte carries the beam info field.

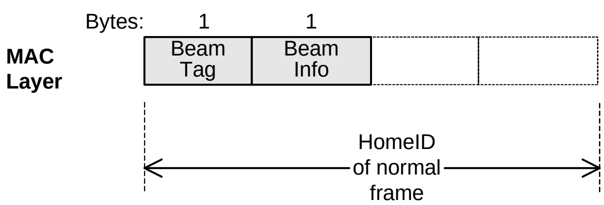

Figure 8-18 – Beam frame

The preamble pattern length depends on the actual RF profiles used. Refer to clauses 7.1.2.1 and 7.1.3.2. A node shall listen long enough to detect a beam frame during worst case conditions around the SOF, beam tag and beam info. In a worst case situation, a node in FL mode starts listening just shortly before the preamble pattern ends. In this case the node has to wait for preamble detection during the SOF, beam tag and beam info fields before another preamble pattern starts. Only after achieving lock to the preamble, the SOF and beam tag fields may be correctly decoded.

Each beam frame may carry one byte value. The beam tag value indicates the type of data carried in the beam info byte.

Table 8-17 – Beam info data type

Beam tag value	Beam info data type	Description
0x55	NodeID (values 1..232, 255)	High priority field. Shall be transmitted in every second beam frame. May be transmitted in every beam frame.
0x54	Low priority beam info	Reserved for future use.

The beam tag value 0x54 allows for future extensions.

If a node in FL mode wakes up and detects a beam frame, the node shall evaluate the encapsulated information in order to determine if it can return to sleep, or if it has to stay awake.

8.1.3.11 Fragmented beam format

A fragmented beam comprises a number of beam fragments. A beam fragment comprises a number of beam frames spanning a period of 100 ms. The fragment duration should be as close to 100 ms as possible without exceeding 100 ms. Beam frames shall be sent back to back to ensure that the beam fragment can be detected by a node waking up at any moment during the duration of the beam fragment.

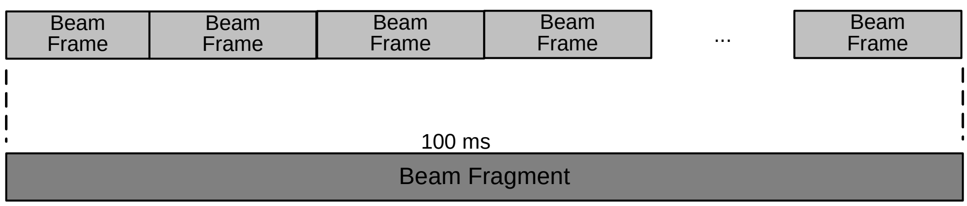

Figure 8-19 – Format of the beam fragment

A fragmented beam is constructed of 100 ms beam fragments separated by 100 ms pauses. This duty cycle complies with regional requirements for the 900 MHz ISM band. A receiving node may acknowledge the reception of a beam fragment. Fragmented beams shall be sent using the R3 data rate. A receiver shall be able to monitor multiple channels for beam frames.

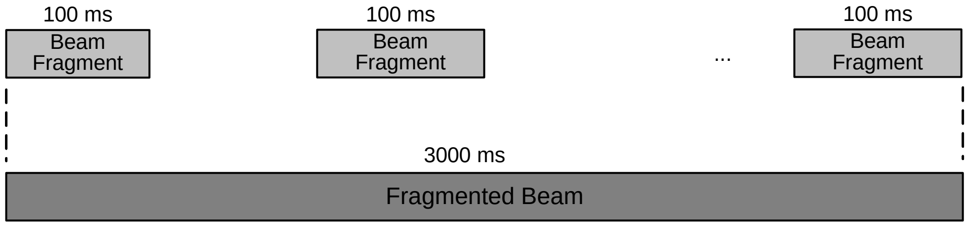

Figure 8-20 – A fragmented beam

Proper TX scheduling allows a sending node to reach sleeping nodes that use a range of wakeup intervals 100 ms, 150 ms, 250 ms, 500 ms, 700 ms, 900 ms. The chosen wakeup interval is a trade-off between battery lifetime and response time.

A fragmented beam may address any NodeID; also the broadcast NodeID 0xFF. A full fragmented beam spans 3000 ms. A singlecast frame follows the fragmented beam. A receiving node may interrupt the transmission of a fragmented beam by acknowledging a beam fragment. The originating node shall then transmit the frame to the receiving node if the source HomeID matches the HomeID of the beaming node. A receiving node shall not acknowledge a fragmented beam targeting the broadcast address. A receiving node shall apply a random delay of 0..50 ms to the ack message to prevent collisions with identical NodeIDs of FL nodes in other networks.

8.1.3.12 Repeated beam format

A repeated beam is an up to 3000 ms long series of beams. Each beam is a 75 ms long beam fragment followed by a singlecast frame. A new beam is transmitted every 150 ms; measured from the start of the previous beam. If the frame is acknowledged, the repeated beam transmission is terminated.

Figure 8-21 – A single beam

A repeated beam may address any NodeID, including the broadcast NodeID 0xFF.

A receiving node shall compare the NodeID in the beam frame with its own NodeID and if there is a match the receiving node shall configure its radio to receive singlecast frames.

8.1.3.13 Continuous beam format

A continuous beam is a series of beam frames spanning a fixed period of time. The beam frames shall be sent back to back to prevent other transmitters from interrupting the continuous beam. The continuous beam shall be sent using the R2 data rate.

A continuous beam may address any NodeID; also the broadcast NodeID 0xFF.

There are two types of continuous beams: long and short. A long continuous beam may last for a maximum 1160 ms, and a short one may last for maximum 300 ms. The recommended duration is 1100 ms for a long continuous beam and 275 ms for a short continuous beam. A continuous beam shall always be followed by a singlecast frame.

A receiving node shall match the NodeID in the beam frame with its own NodeID and if there is a match the receiving node shall configure its radio to receive singlecast frames.

8.1.3.14 Beam formats on channel configurations 1 and 2

FL nodes operating in channel configuration 1 or 2 shall:

- be able to receive the continuous beam format and the repeated beam format on R2;
- be able to transmit either the repeated beam format or the continuous beam format on R2. It is recommended that the repeated beam format is transmitted.

8.1.3.15 Beam formats on channel configuration 3

FL nodes operating in channel configuration 3 shall be able to receive and transmit the fragmented beam format.

8.1.4 MAC constants and MIB attributes

This clause specifies the constants and attributes required by the MAC.

8.1.4.1 MAC constants

The constants that define the characteristics of the MAC are presented in Table 8-18. These constants are hardware dependent and cannot be changed during operation.

Table 8-18 – MAC constants in general

Constants	Description	Value
aMacMaxMSDUSizeR1	The maximum size, in bytes, of a singlecast/broadcast payload for data rate R1	54 bytes
aMacMaxMSDUSizeMultiR1	The maximum size, in bytes, of a multicast payload for data rate R1	25 bytes
aMacMaxMSDUSizeR2	The maximum size, in bytes, of a singlecast/broadcast payload for data rate R2	54 bytes
aMacMaxMSDUSizeMultiR2	The maximum size, in bytes, of a payload for multicast data rate R2	25 bytes
aMacMaxMSDUSizeR3	The maximum size, in bytes, of a singlecast/broadcast payload for data rate R3	158 bytes
aMacMaxMSDUSizeMultiR3	The maximum size, in bytes, of a multicast payload for data rate R3	129 bytes
aMacMinAckWaitDuration	The minimum Ack wait duration	aTurnaroundTimeRxTx + (aMacTransferAckTimeTX*(1/data rate))
aMacMaxFrameRetries	The number of retries after a transmission failure	2
aMacMinCCARetryDuration	The minimum duration of clear channel access assessment	1100 ms

Table 8-18 – MAC constants in general

Constants	Description	Value
aMacTxPhyPowerLevelLow	The constant value used by the MAC layer to configure the PHY MIB parameter phyTransmitPower prior to a low-power transmission.	Implementation dependent.
aMacTxPhyPowerLevelNormal	The constant value used by the MAC layer to configure the PHY MIB parameter phyTransmitPower prior to a normal power transmission.	Implementation dependent.

The term *aMacMaxMSDUSizeX* is used in text that applies to all data rates (*aMacMaxMSDUSizeR1, aMacMaxMSDUSizeR2, aMacMaxMSDUSizeR3, aMacMaxMSDUSizeMultiR1, aMacMaxMSDUSizeMultiR2, aMacMaxMSDUSizeMultiR3).*

Table 8-19 outlines the constant values to use for individual transmissions. Three transmission events are defined: TX1, TX2 and TX3. TX1 is the initial transmission while TX2 and TX3 are the optional first and second retransmissions. Only TX1 is defined for acknowledgment timing constants.

Table 8-19 – MAC constants for message transfer

Constants	Description	Value, R1	Value, R2	Value, R3; CC 1 and 2	Value, R3; CC 3
aMacTransferAckTimeTX (ch)	The number of symbols it takes to transmit an acknowledgment frame including preamble.	168 bits	248 bits	416 bits	296 bits
aMacTypicalFrameLengthTX(ch)	The number of symbols of a singlecast frame with a data payload of 4 bytes.	200 bits	280 bits	448 bits	328 bits
aMacMinRetransmitDelay	Random backoff shall be higher than this value.	10 ms			
aMacMaxRetransmitDelay	Random backoff shall be lower than this value.	40 ms			

The MAC determines if the PHY is configured for channel configurations 1, 2 or channel configuration 3 operation by evaluating the RF profile mappings configured via the *macMapPhyChannel** MAC MIB parameters and the RF profile table defined for the PHY.

8.1.4.2 MAC MIB attributes

The MAC management information base (MAC MIB) comprises the attributes required to manage the MAC of a node. Each of these attributes can be read or written using the MLME-GET.request and MLME-SET.request primitives, respectively. The attributes contained in the MAC MIB are presented in Table 8-20.

Table 8-20 – MAC MIB attributes

Attribute	Type	Range	Description	Default
macCCARetryDuration	Integer	aMacMinCCARetryDuration – infinite	The duration of clear channel access assessment.	~ 1100 ms
macHomeID	Set of bytes	0x00000000 – 0xFFFFFFFF	The HomeID is the unique network identifier.	(random)
macNodeID	Byte	0x00– 0xFF	The NodeID is the address of the individual nodes in a network; they are only unique within a network defined by a unique HomeID.	0x00
macPromiscuousMode	Boolean	TRUE or FALSE	This indicates whether the MAC sublayer is in a promiscuous (receive all) mode. A value of TRUE indicates that the MAC sublayer accepts all frames received from the PHY.	FALSE
macRxOnWhenIdle	Boolean	TRUE or FALSE	This indicates whether the MAC sublayer is to enable its receiver at any time.	FALSE
phyCurrentTxChannel	Enumeration	A, B, C	The TX channel to use (see clause 7.1.2.2).	A
macMapPhyChannelA	Integer	Available RF profiles (Refer to clause 7.1.2.1)	Apply RF profile to channel A. Parameter maps directly to the PHY MIB parameter phyMapChannelA.	–

Table 8-20 – MAC MIB attributes

Attribute	Type	Range	Description	Default
macMapPhyChannelB	Integer	Available RF profiles (Refer to clause 7.1.2.1)	Apply RF profile to channel B. Parameter maps directly to the PHY MIB parameter phyMapChannelB.	–
macMapPhyChannelC	Integer	Available RF profiles (Refer to clause 7.1.2.1)	Apply RF profile to channel C. Parameter maps directly to the PHY MIB parameter phyMapChannelC.	–

8.1.5 MAC functional description

This clause provides a detailed description of the MAC functionality. Throughout this clause, the receipt of a frame is defined as the successful receipt of an incoming frame from the RF media to the PHY and the successful verification of the FCS by the MAC sublayer, as described in clause 8.1.3.8.

8.1.5.1 Transmission, reception and acknowledgment

This clause describes the fundamental procedures for transmission, reception, and acknowledgment.

8.1.5.1.1 Clear channel access

The MAC layer shall request the channel status from the PHY layer.

A PLME-GET-CCA.request message is used to evaluate the channel. A PLME-GET-CCA.confirm message returns the current channel status.

If the MAC sublayer finds the channel busy for a period of *macCCARetryDuration* the frame transmission has failed. This is indicated to the network layer via the MD-DATA.confirm primitive with a status of NO_CCA (see clause 8.1.2.1.2).

8.1.5.1.2 Transmission

To avoid RF collisions the MAC sublayer performs a CCA before transmitting a frame. If the channel is found idle the frame is transmitted. The source HomeID and source NodeID field shall contain the address of the node sending the frame and the destination NodeID field shall contain the address of the recipient. In addition the frame type and frame length is generated as a part of the MPDU.

8.1.5.1.3 Reception and rejection

Each node may choose whether the MAC sublayer is to enable its receiver during idle periods. During these idle periods, the MAC sublayer shall still service transceiver task requests from the network layer. A transceiver task shall be defined as a transmission request, a reception request, or a clear channel access detection. On completion of each transceiver task, the MAC sublayer shall request that the PHY enables or disables its receiver, depending on whether *macRxOnWhenIdle* is set to TRUE or FALSE, respectively.

Due to the nature of radiocommunications, a node with its receiver enabled will be able to receive and decode transmissions from all nodes that are operating on the same channel(s). The MAC sublayer shall, therefore, be able to filter incoming frames and present only the frames that are of interest to the upper layers.

The MAC sublayer shall discard all received frames that do not contain a correct value in their FCS field in the MFR, according to the algorithm described in clause 8.1.3.8. The next level of filtering shall depend on whether the MAC sublayer is currently operating in promiscuous mode or not. In promiscuous mode, the MAC sublayer shall pass all frames directly to the network layer.

If the MAC sublayer is not in promiscuous mode (i.e., *macPromiscuousMode* is set to FALSE), it shall only accept frames (i.e., issue an MD-DATA.indication to the network layer) if the frame header contains the HomeID and NodeID of the receiving node. The filter shall accept frames if addressed to the broadcast address or if the NodeID is included in a multicast header.

If the frame type subfield indicates a singlecast frame and the acknowledgment request subfield of the frame control field is set to 1, the MAC sublayer shall send an acknowledgment frame.

The MAC sublayer shall be able to receive beam fragments and forward such datagrams to higher layers.

8.1.5.1.3.1 RX filtering

A frame shall be discarded if the received frame has a wrong checksum value (applies to R1 and R2) or a wrong CRC16 value (applies to R3).

A frame shall be discarded if the received frame has a length field less than 9 or greater than the maximum size values indicated in Table 8-18.

8.1.5.1.4 Use of acknowledgment frames

A singlecast frame may be sent with the acknowledgment request subfield of the frame control field set to 1. Any multicast or broadcast frame shall be sent with the acknowledgment request subfield set to 0.

Sequence number checking shall be done as part of the acknowledgment handling. Refer to clause 8.1.3.3.7.

8.1.5.1.4.1 No acknowledgment

A frame transmitted with its acknowledgment request subfield set to 0 shall not be acknowledged by its intended recipient. The originating node shall assume that the transmission of the frame was successful. The message sequence chart in Figure 8-22 shows the scenario for transmitting a single frame of data from an originator to a recipient without requiring an acknowledgment.

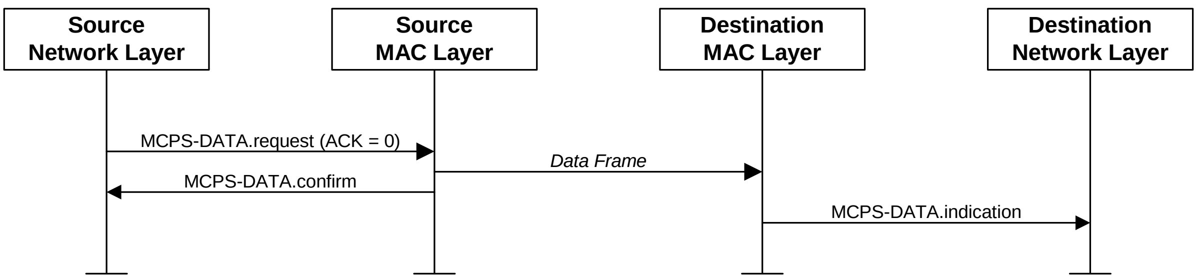

Figure 8-22 – Successful data transmission without an acknowledgment

8.1.5.1.4.2 Acknowledgment

A singlecast frame transmitted with the acknowledgment request subfield of its frame control field set to 1 shall be acknowledged by the recipient. If the intended recipient correctly receives the frame, it shall return an acknowledgment frame. Only the singlecast frame may be sent with the acknowledgment request subfield set to 1. For other frame types the acknowledgment request subfield shall be ignored by the intended recipient.

The transmission of an acknowledgment frame shall commence *aTurnaroundTimeRXTX* symbols after the reception of the last symbol of the frame. Refer to clause 7.1.2.5.9.

The message sequence chart in Figure 8-23 shows the transmission of a singlecast frame with an acknowledgment frame.

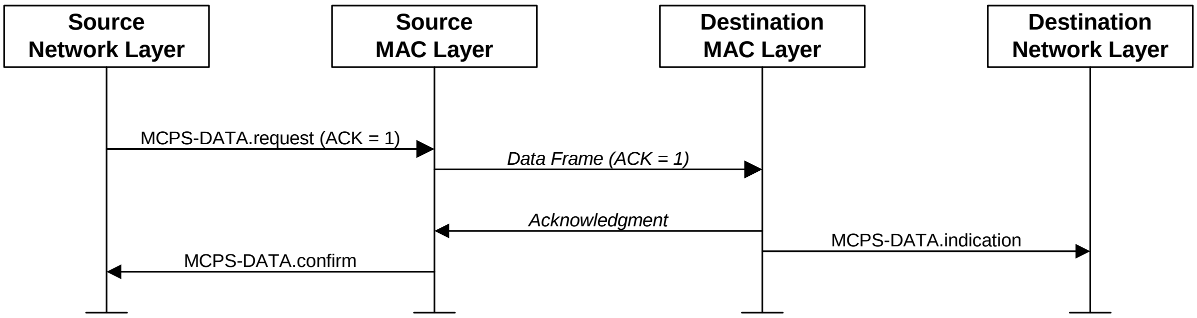

Figure 8-23 – Successful data transmission with an acknowledgment frame

8.1.5.1.4.3 Retransmission

A node that sends a singlecast frame with its acknowledgment request subfield set to 1 shall wait for minimum *aMacMinAckWaitDuration* symbols for the corresponding acknowledgment frame to be received. If an acknowledgment frame is received within *aMacMinAckWaitDuration* symbols and contains the same HomeID and the correct source NodeID, the transmission is considered successful, and no further action shall be taken by the originator. If an acknowledgment frame is not received within *aMacMinAckWaitDuration* symbols the transmission attempt has failed. The originator shall repeat the process of transmitting the frame and waiting for the acknowledgment frame up to *aMacMaxFrameRetries* times. Before retransmitting, the node shall wait for a random backoff period (see clause 8.1.5.1.4.4).

If an acknowledgment frame is still not received after *aMacMaxFrameRetries* retransmissions, the MAC sublayer shall assume the transmission has failed and notify the network layer of the failure. This is done via the MD-DATA.confirm primitive with a status of NO_ACK.

8.1.5.1.4.4 Random backoff

If a singlecast frame with its acknowledgment request subfield set to 1 or the corresponding acknowledgment frame is lost or corrupted, the singlecast frame shall be retransmitted. The MAC layer collision avoidance mechanism prevents nodes from retransmitting at the same time. The random delay is calculated as a period in the interval *aMacMinRetransmitDelay* .. *aMacMaxRetransmitDelay;* Refer to Table 8-19.

8.1.5.1.5 Idle mode

If the MLME is requested to set *macRxOnWhenIdle* to TRUE the PHY will enter RX mode and stay in RX mode when a frame has been transmitted (always listening). This request is achieved when the MLME issues the PLME-SET-TRX-STATE.request primitive with a state of RX_ON.

If the MLME is requested to set *macRxOnWhenIdle* to FALSE, the PHY will disable its receiver when a frame has been transmitted. This is achieved by the MLME issuing the PLME-SET-TRX-STATE.request primitive with a state of TRX_OFF.

8.1.5.2 Transmission scenarios

Due to the imperfect nature of the radio medium, a transmitted frame does not always reach its intended destination. Figures 8-24 to 8-26 illustrates three different data transmission scenarios:

– ***Successful data frame transmission.*** The originating MAC sublayer transmits the frame to the recipient via the PHY data service. In waiting for an acknowledgment frame, the originating MAC sublayer starts a timer that will expire after *aMacMinAckWaitDuration* symbols. The recipient MAC sublayer receives the frame, sends an acknowledgment frame back to the originator, and passes the frame to the next higher layer. The originating MAC sublayer receives the acknowledgment frame from the recipient before its timer expires and then disables and resets the timer. The data transfer is now complete, and the originating MAC sublayer issues a success confirmation to the next higher layer.

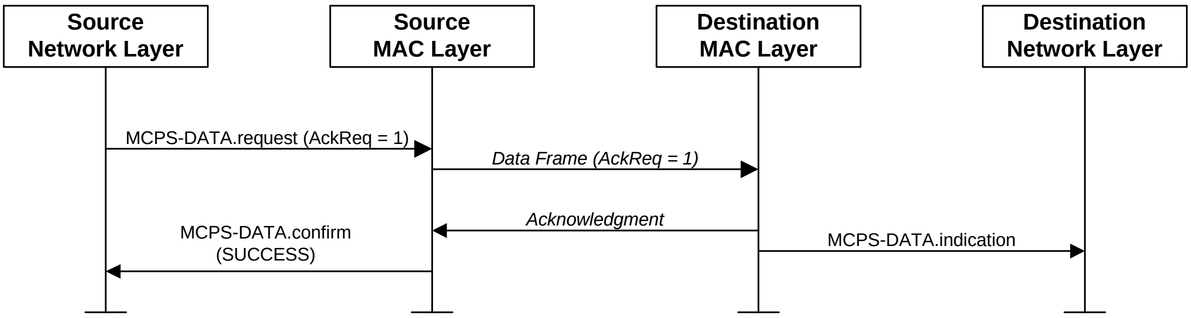

Figure 8-24 – Successful frame transmission scenario

– ***Lost data frame transmission.*** The originating MAC sublayer transmits the frame to the recipient via the PHY data service. In waiting for an acknowledgment frame, the originating MAC sublayer starts a timer that will expire after *aMacMinAckWaitDuration* symbols. The recipient MAC sublayer does not receive the frame and does not therefore return an acknowledgment frame. The timer of the originating MAC sublayer expires before an acknowledgment frame is received. The data transfer has failed and the originator retransmits the frame. This sequence is repeated up to *aMacMaxFrameRetries* times. If a data transfer attempt fails a total of (1 +*aMacMaxFrameRetries*) times, the originating MAC sublayer will issue a failure confirmation to the network layer.

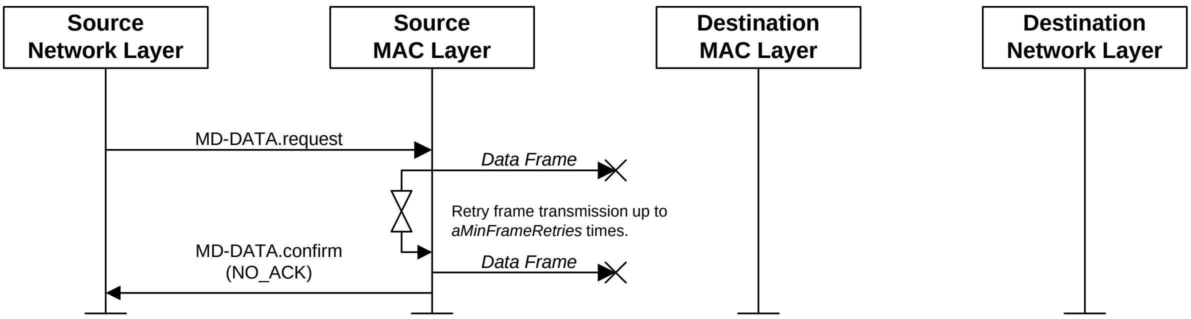

Figure 8-25 – Lost frame transmission scenario

– ***Lost acknowledgment frame transmission.*** The originating MAC sublayer transmits the frame to the recipient via the PHY data service. In waiting for an acknowledgment frame, the originating MAC sublayer starts a timer that will expire after *aMacMinAckWaitDuration* symbols. The recipient MAC sublayer receives the frame, sends an acknowledgment frame back to the originator, and passes the frame to the next network layer. The originating MAC sublayer does not receive the acknowledgment frame and its timer expires. The data transfer has failed, and the originator will retransmit the data. If a data transfer attempt fails a total of (1 + *aMacMaxFrameRetries*) times, the MAC sublayer will issue a failure confirmation to the next higher layer.

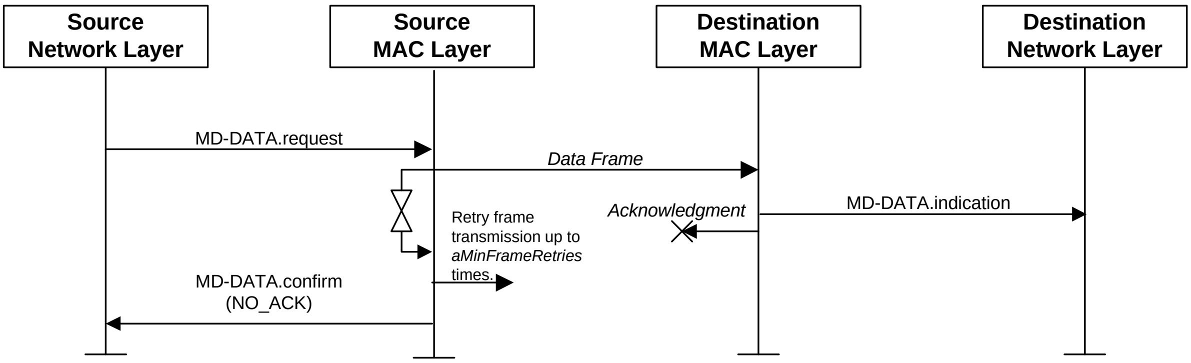

Figure 8-26 – Lost acknowledgment frame transmission scenario

Annex A

The non-radio related aspects of the Z-Wave PHY and MAC Specifications

(This annex forms an integral part of this Recommendation.)

A.1 Introduction

A.1.1 General

Z-Wave is a low-power, low-cost wireless technology enabling consumer-grade products with networked features. Examples include remote controlled light dimmers, networked temperature sensors, electronic door locks and AV systems.

This annex presents the physical (PHY) layer to be implemented by a network device operating in a Z-Wave network.

Low-power wireless radio networks suffer from frequent packet drops due to fading effects, spurious noise and reflections. Z-Wave provides low-level retransmissions to recover from such situations. Network layers may employ mesh routing to extend networks beyond what is possible in direct range. The Z-Wave protocol is one such example.

A.1.2 Technology overview

NOTE – The information in this clause is for informational purposes only and contains features that are outside of the scope of this Recommendation.

The Z-Wave framework incorporates its own network, transport and application layers. These should not be confused with the network and transport layers defined by IETF for IP transport. The Z-Wave framework supports IP transport as well as IP routing protocols.

Nodes receive a unique address during network inclusion. A network may hold up to 232 nodes. A network ID is used to identify a specific network. Network and node addresses are auto-configured with no user intervention.

The Z-Wave network layer defines the Z-Wave routing protocol which is a source routing protocol. The Z-Wave routing protocol allows an originator to reach a destination node via one or more repeaters. A reactive discovery algorithm allows originators to determine new source routes when required.

Z-Wave uses a special frame format for multicast messages. The Z-Wave multicast frame header carries a complete destination bit map in the frame header.

The Z-Wave application layer defines a wide range of device and command classes for various devices such as lamps, door locks and temperature sensors.

A.2 General description

The Z-Wave protocol is a low bandwidth half duplex protocol designed for reliable wireless communication in a low-cost control network with limited power and non-time critical and relaxed time critical data transfer requirements. The main purpose of the protocol is to communicate short control messages in a reliable manner from one node to one or more nodes in the network. The main objectives of a Z-Wave network are ease of installation, interoperability, reliable data transfer, very low cost and long battery lifetime while maintaining a simple and flexible protocol. The Z-Wave protocol is designed for residential and light commercial control and status reading applications such as meter reading, lighting and appliance control, HVAC, access control, intruder sensors, etc.

Some of the characteristics of a Z-Wave PHY/MAC are:

- use of license-free RF bands such as the ISM frequency band
- multichannel robustness with 2- or 3-channel operation
- data rates of 9.6 kbit/s, 40 kbit/s and 100 kbit/s
- mesh network operation with:
  – unique network ID numbers (HomeID)
  – up to 232 nodes in one network
- fully acknowledged protocol for reliable data transfer
- collision avoidance algorithm including:
  – Clear channel assessment
  – Backoff algorithm
- automatic retransmission for reliable data transfer
- support for low-power operation via dedicated beaming patterns.

The Z-Wave protocol (using routing algorithms which are beyond the scope of this Recommendation) assures full home coverage. Z-Wave frames are routed around radio dead spots and signal reflections thereby securing a highly robust communication.

A.2.1 Network topology

A.2.1.1 Network components

Z-Wave nodes can be divided into two subgroups: normal nodes and low-power nodes. The normal nodes are typically in receive mode at all times. The low-power nodes spend the majority of the operating life in power down and wake up at regular short intervals to minimize power consumption. The latter type is denoted frequently listening (FL) nodes.

A.2.1.2 Network topology

NOTE – The network formation is handled by the network layer, which is out of the scope of this Recommendation. However, this section provides a brief overview of how the supported network topology is formed.

In a mesh network topology, each device is capable of communicating with another device within its radio sphere of influence. A node can send and receive messages and can also relay messages for its neighbours. This relay process enables wireless messages to reach its ultimate destination, passing through intermediate nodes. Introducing routing provides for a redundant and more reliable network. An example of a mesh network is shown in Figure A.1.

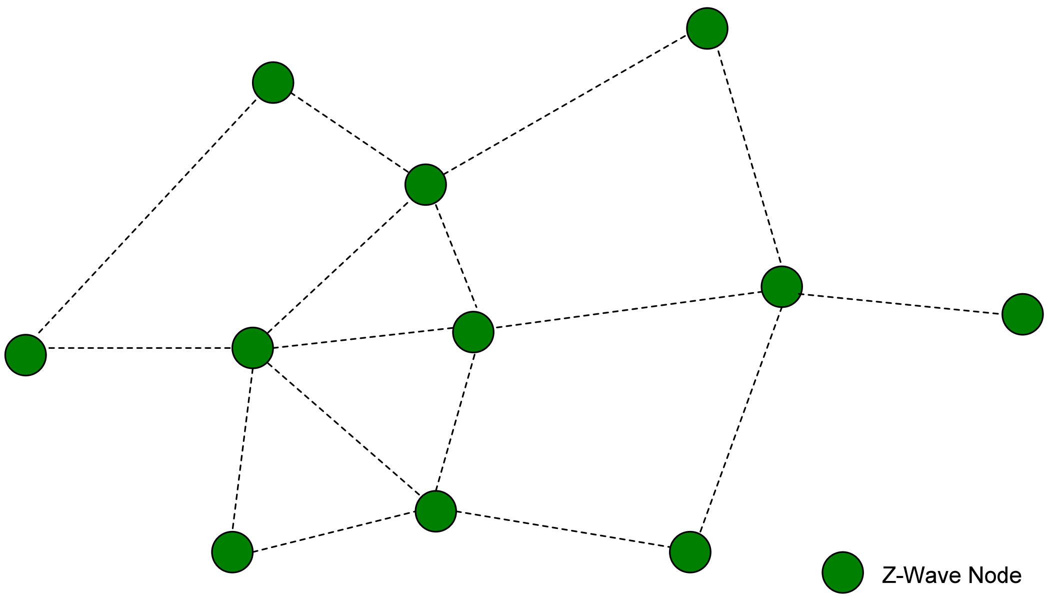

Figure A.1 – Mesh network topology

Z-Wave uses source routing. The routes may be established on a proactive basis or via on-demand discovery. If messages are lost due to fading effects and/or interference, the node may retransmit messages and if still failing, use alternative routes. New alternative routes may be discovered if needed.

A.2.1.3 Z-Wave bootstrapping

Nodes have to be paired with other nodes in order to form a network. This process is known as inclusion. A unique 32-bit identifier called the HomeID is used to separate networks from each other.

NodeIDs are unique within a network. A NodeID is an 8 bit short address. The first node hands out NodeIDs to all other nodes joining the network.

During inclusion, a node uses the initialized NodeID (0x00) and a random pseudo-unique HomeID for identification.

A.2.2 Network architecture

The Z-Wave architecture is defined in terms of layers. Each layer is responsible for one part of the operation and offers services to the higher layers: a data entity provides a data transmission service and a management entity provides other services related to the actual layer. The data and management entities define the logical links between the layers.

A Z-Wave node implements the PHY layer, which contains the RF transceiver along with its low-level control mechanism, a MAC layer that provides access to the physical channel for all types of transfers, a network layer controlling message routing and a combined application layer that collapses the OSI stack layers transport, session and presentation.

Figure A.2 shows these layers in a graphical representation.

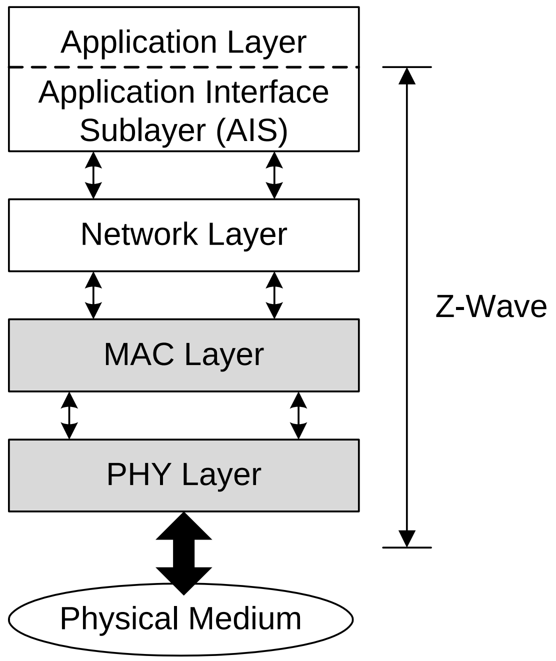

Figure A.2 – Z-Wave layers

This annex specifies the PHY and MAC layers. Upper layers are outside the scope of this Recommendation.

A.2.2.1 PHY layer

The features of the PHY are activation and deactivation of the RF transceiver, frequency selection, and transmitting as well as receiving frames. The RF receiver is able to perform a clear channel assessment. The RF transceiver operates in a one, two- or three-channel configuration located in the license-free ISM frequency bands.

The PHY provides two services: (1) the PHY data service accessed through the PHY data (PD) service access point (PD-SAP) and (2) the PHY management service interfacing with the physical layer management entity (PLME) service access point (PLME-SAP). The PHY data service enables the transmission and reception of PHY protocol data units (PPDUs) across the physical radio channel.

Clause A.3 contains the full specifications of the PHY layer.

A.2.2.2 MAC layer

The features of the MAC sublayer are channel access, frame validation, acknowledged frame delivery, and retransmissions.

The MAC sublayer provides two services: the MAC data service, accessed through the MAC sublayer data service access point (MD-SAP) and the MAC management service interfacing with the MAC layer management entity (MLME) service access point (MLME-SAP). The MAC data service enables the transmission and reception of MAC protocol data units (MPDUs) across the PHY data service.

Section A-6 contains the full specifications of the MAC layer.

A.2.3 Functional overview

A brief overview of the general functions of a Z-Wave network is given in the following sections and includes information on the data transfer model, the frame structure, robustness and power consumption considerations.

A.2.3.1 Frame formats

A number of frame formats are defined. In most cases, the MAC layer shall forward frames to the higher layers for further processing. The exception is frames related to acknowledgment.

A.2.3.1.1 Singlecast

Figure A.3 shows the structure of the general frame of the MAC sublayer. The MAC service data unit (MSDU) contains the payload data from the network layer. The MSDU is prepended with a MAC header (MHR) and appended with a MAC footer (MFR). The MHR contains HomeID, source node ID, MAC frame control field, frame length field and destination node ID. The MFR contains a non-correcting frame check sequence (FCS). The MHR, MSDU, and MFR together form the MAC protocol data unit (MPDU).

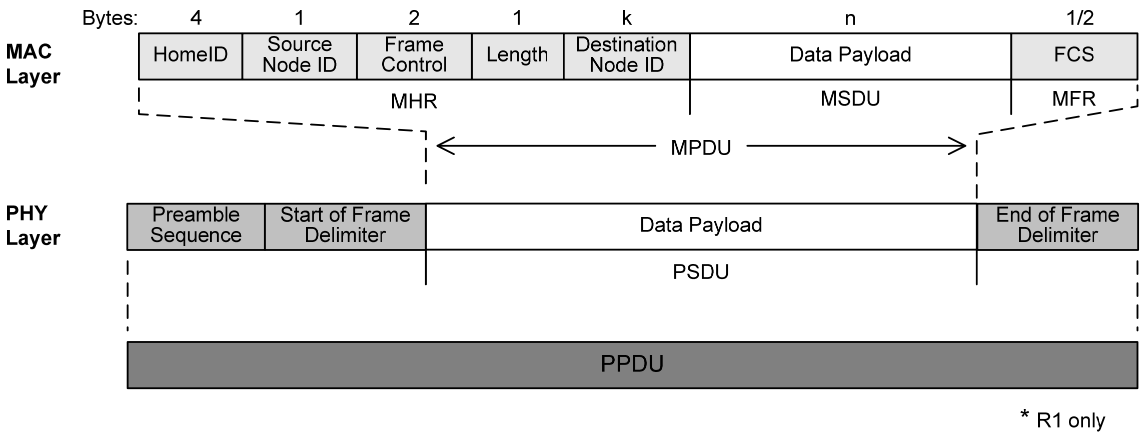

Figure A.3 – General frame structure

The MPDU is passed to the PHY layer as a PHY service data unit (PSDU). The PSDU is prefixed with a start header (SHR) and an end header (EHR). The SHR contains the preamble sequence and start-of-frame delimiter (SFD) fields. The preamble sequence enables the RF receiver to achieve symbol synchronization. The SHR, PSDU and EHR together form the PHY protocol data unit.

A.2.3.1.2 Acknowledgment

The acknowledgment frame structure uses the general frame structure (Figure A.3) with a zero-length MAC service data unit (MSDU).

A.2.3.1.3 Multicast

A Z-Wave multicast header explicitly identifies each target. A multicast frame may only be used in direct range of destinations.

A.2.3.2 Network robustness

The Z-Wave protocol employs various mechanisms to ensure robustness in the data transmission. These mechanisms of the MAC layer are backoff algorithm, frame acknowledgment, data verification and frame retransmission. In addition, the PHY layer clear channel assessment function is queried before transmission.

A.2.3.2.1 Clear channel assessment

A node shall query the availability of the channel from the PHY layer before transmitting. If the channel is found to be idle, the node may transmit its data. If the channel is found busy, the node shall wait for idle channel before transmitting.

A.2.3.2.2 Frame acknowledgment

A successful reception and validation of a frame can be optionally confirmed with an acknowledgment. If the destination node receives a frame containing bit errors, the message is not acknowledged. An acknowledgment request may be carried in the frame that needs acknowledgment.

A.2.3.2.3 Retransmission

If two or more nodes transmit their frames simultaneously a collision may occur and the frames may not reach their destinations.

If the source node does not receive an acknowledgment, it assumes that the transmission was unsuccessful and retries the frame transmission up to *aMacMaxFrameRetries* times.

In order to avoid another collision each transmitter delays its retransmission with a random delay.

A.2.3.2.4 Multi-hop routing

The MAC layer supports multi hop routing. The routing algorithm is outside the scope of this Recommendation (i.e., should be specified at layers above the MAC layer).

A.2.3.2.5 Data validation

An 8-bit non-correcting frame check sequence (FCS) mechanism is employed to detect bit errors for data rates R1 and R2. A 16-bit non-correcting frame check sequence (FCS) mechanism is employed to detect bit errors for data rate R3. For a description of the data rates R1, R2 and R3 please refer to clause A.3.1.3.

A.2.3.3 Power consumption considerations

Battery-powered nodes will spend most of their operational life in a sleep mode; however, some nodes shall periodically wake up and poll other nodes (also known as a mailbox) to determine whether messages are pending.

Other low-power nodes require a faster and more responsive behaviour than can be achieved with a mailbox. These notes will be frequently listening for messages. Communication with this type of nodes is described in the next section.

A.2.3.3.1 Communication with frequently listening nodes

Battery-powered devices like drape controllers and electronic door locks should be reachable at any time. This specification defines a transmission mode allowing an originator to send an extended preamble pattern that can be prepended to a singlecast frame. Nodes that are listening at regular short intervals are denoted FLN (frequently listening node).

A.2.4 Concept of primitives

This clause provides a brief overview of the concept of service primitives (operations). Refer to [b-ITU-T X.210] for more detailed information.

The services of a layer are the capabilities it offers to the user in the next higher layer or sublayer by building its functions on the services of the next lower layer. This concept is illustrated in Figure A.4, showing the service hierarchy and the relationship of the two correspondent N-users and their associated N-layer (or sublayer) peer protocol entities.

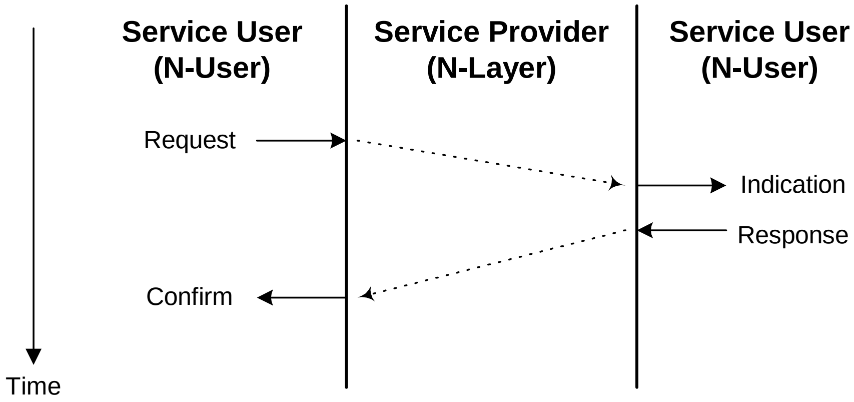

Figure A.4 – Service primitives

The services are specified by describing the information flow between the N-user and the N-layer. This information flow is modelled by discrete, instantaneous events, which characterize the provision of a service. Each event consists of passing a service primitive from one layer to the other through a service access point (SAP) associated with an N-user. Service primitives convey the required information by providing a particular service. These service primitives are an abstraction because they specify only the provided service rather than the means by which it is provided. This definition is independent of any other interface implementation.

Services are specified by describing the service primitives and parameters that characterize it. A service may have one or more related primitives that constitute the activity that is related to that particular service. Each service primitive may have zero or more parameters that convey the information required to provide the service.

A primitive can be one of four generic types:

- Request: the request primitive is passed from the N-user to the N-layer to request that a service is initiated.
- Indication: the indication primitive is passed from the N-layer to the N-user to indicate an internal N-layer event that is significant to the N-user. This event may be logically related to a remote service request, or it may be caused by an N-layer internal event.
- Response: the response primitive is passed from the N-user to the N-layer to complete a procedure previously invoked by an indication primitive.
- Confirm: the confirm primitive is passed from the N-layer to the N-user to convey the results of one or more associated previous service requests.

A.3 Physical (PHY) layer specification

This clause specifies the PHY. The PHY is responsible for the following tasks:

- Assignment of an RF profile to a physical channel
- Activation and deactivation of the radio transceiver
- Data transmission and reception
- Clear channel assessment
- Frequency selection
- Link quality for received frames

Constants and attributes that are specified and maintained by the PHY are written in the text of this clause in *italics*. Constants have a general prefix of "a", e.g., *aMaxPHYPacketSizeRx*, and are listed in Table A.27. Attributes have a general prefix of "phy", e.g., *phyCurrentChannel*, and are listed in Table A.28.

A.3.1 Transceiver front end specifications

This clause specifies requirements of the PHY.

The Z-Wave specification shall conform to established regulations in Europe, United States, and other world regions.

A.3.1.1 RF profiles

The possible transceiver usage of RF frequencies and data rates is described using RF profiles. An RF profile is a combination of a centre frequency and data rates, which can be used at this centre frequency, in a specific region. The list of RF profiles that are specified for the Z-Wave transceiver in different world regions is specified in Table A.1. The list of specific regional frequencies is outside the scope of this Recommendation. A transceiver shall support up to 3 channels. A channel is associated with a single RF profile. Each channel can have a different RF profile assigned to it. Unused channels are disabled by assigning RF profile 0 (a dummy RF profile with no centre frequency or data rate allocated) to this channel. The transceiver shall support one or more of the data rates 9.6 kbit/s (R1), 40 kbit/s (R2) and 100 kbit/s (R3) according to the RF profile assigned.

Table A.1 – RF profiles and frequencies

RF Profile	Country/ Market	Centre frequency	R3	R2	R1	FL (Note 1)
0	n/a	n/a				
1	European Union	f_{EU1}	√			
2		f_{EU2}		√	√	
3		f_{EU2}		√		√
4	United States	f_{US1}	√			
5		f_{US2}		√	√	
6		f_{US2}		√		√
7	Hong Kong	f_{HK1}	√			
8		f_{HK1}		√	√	
9		f_{HK1}		√		√
10	Australia and New Zealand	f_{ANZ1}	√			
11		f_{ANZ2}		√	√	
12		f_{ANZ2}		√		√
13	Malaysia	f_{MY1}	√			
14		f_{MY1}		√	√	
15		f_{MY1}		√		√
16	India	f_{IN1}	√			
17		f_{IN1}		√	√	
18		f_{IN1}		√		√

Table A.1 – RF profiles and frequencies

RF Profile	Country/ Market	Centre frequency	R3	R2	R1	FL (Note 1)
19	Japan	f_{JP1}	√			√
20		f_{JP2}	√			√
21		f_{JP3}	√			√
NOTE 1 – FL (frequently listening) nodes (refer to clause A.2.1.1) are only applicable to the marked RF profiles. NOTE 2 – AL (always listening) nodes can be used with any RF profile.						

A.3.1.2 Transmit frequency error

Frequency error is defined as the difference between the measured transmitted centre frequency and the ideal centre frequency. A conforming device shall comply with the frequency error requirement over the entire temperature and supply range for the device.

The frequency error shall not exceed ±27 ppm including crystal ageing over five years.

A.3.1.3 RF data rate

The PHY shall support the data rates and the accuracy as listed in the table below:

Table A.2 – Data rate and accuracy

Rate	Bit rate	Symbol rate	Accuracy
R1	9.6 kbit/s	19.2 kbaud	±27 ppm
R2	40 kbit/s	40 kbaud	±27 ppm
R3	100 kbit/s	100 kbaud	±27 ppm

A.3.1.4 Channel configuration

A compliant node shall operate in one of the three possible configurations as listed in the table below:

Table A.3 – Channel configurations

Configuration	Num channels	Data rate		
		R3	R2	R1
1	1	–	Ch B	Ch B (Note)
2	2	Ch A	Ch B	Ch B (Note)
3	3	Ch A Ch B Ch C	–	–
NOTE – The R1 data rate is not supported by FL nodes (i.e., for these nodes configuration 1 shall not include R1). AL nodes shall support both R2 and R1.				

Table A.1 with the information on specific RF profiles and Table A.3 in conjunction gives the valid configurations for a node.

Example:

Let us assume a specific region in which there are two available frequencies: f_1 and f_2. Let us also assume that the following RF profiles are available (as explained in clause A.3.1.1):

- One RF profile ("RF profile 1") with frequency of f_1 and a single data rate (R3).
- Two RF profiles for f_2:
  - RF profile 2: A dual data rate (R1 and R2) profile
  - RF profile 3: A single data rate (R2) profile in a FL (frequently listening) device.

According to Table A.3 this leads to the following possible configurations:

1) A single channel system at f_2 where only R2 is enabled for an FL (frequently listening) node.
2) A single channel system at f_2 where both R1 and R2 are enabled for an AL (always listening) node.
3) A dual channel system where channel B is at f_2 where only R2 is enabled for an FL (frequently listening) node and channel A is at f_1 with only R3 enabled.
4) A dual channel system where channel B is at f_2 where both R1 and R2 are enabled for an AL (always listening) node and channel A is at f_1 with only R3 enabled.

A.3.1.5 Modulation and coding

The PHY shall employ frequency shift keying (FSK) for RF modulation of R1 and R2. The PHY shall employ Gaussian frequency shift keying (GFSK) for R3.

Manchester code shall be used for data symbol encoding for R1 and non-return-to-zero for data symbol encoding for R2 and R3.

The modulation and the coding format are summarized in the table below:

Table A.4 – Modulation and coding format

Data Rate	Modulation	Coding	Separation	Symbols
R1	FSK	Manchester	40 kHz ±10%	Binary
R2	FSK	NRZ	40 kHz ±10%	Binary
R3	GFSK, BT = 0.6	NRZ	58 kHz ±10%	Binary

The mapping of NRZ symbols to the physical medium assuming high side LO (The mapping should be inverted for low side LO) is given below:

Table A.5 – Symbol mapping, for high side LO

Symbol	Frequency
0	$f_{center\ frequency}$ – separation/2
1	$f_{center\ frequency}$ + separation/2

The mapping of Manchester symbols to the physical medium assuming high side LO (The mapping should be inverted for low side LO) is given below:

Table A.6 – Manchester symbols, for high side LO

Symbol	Frequency
0	Transition from ($f_{center\ frequency}$ + separation/2) to ($f_{center\ frequency}$ – separation/2)
1	Transition from ($f_{center\ frequency}$ – separation/2) to ($f_{center\ frequency}$ + separation/2)

A.3.1.6 RF power measurement

Unless stated otherwise, all RF power measurements, either transmit or receive, shall be made at the antenna connector. The measurements shall be made with equipment that is either matched to the impedance of the antenna connector or corrected for any mismatch. For devices without an antenna connector, the measurements shall be interpreted as effective isotropic radiated power (EIRP) (i.e., a 0 dBi gain antenna); and any radiated measurements shall be corrected to compensate for the antenna gain in the implementation.

A.3.1.7 Transmit power adjustments (conducted)

The transmitter shall transmit at an output power level which shall conform to local regulations and this level is outside the scope of this Recommendation. This output power level is denoted as "nominal".

The transmitter shall be able to adjust its output power in steps of 2 dB in the range between the nominal transmit power down to the (nominal output power – 10 dB).

The transmitter shall be capable of reducing the output power to at least (nominal output power – 20 dB).

A.3.1.8 Receiver sensitivity

To ensure a minimum RF link budget the receiver shall be capable under typical temperature and supply conditions of receiving a standard test frame at a minimum level as specified in Table A.8.

Standard test frame and test conditions are described in Table A.7.

Table A.7 – Standard test frame

Term	Definition	Conditions
Communication error rate (CER)	Average loss. (command and acknowledgment transmission, i.e., two-way communication)	Average measured over random frame data.
Receiver sensitivity	Threshold input signal power that yields a specific CER	Frame data length = 4 bytes CER < 10% (without retransmission) Power measured at antenna terminals Interference not present

Table A.8 – Minimum receiver sensitivity

Bit rate	Minimum receiver sensitivity
R1	–95 dBm
R2	–92 dBm
R3	–89 dBm

A.3.1.9 Clear channel assessment

The PHY shall be able to perform a clear channel assessment (CCA) to determine if the RF medium is busy or clear for transmission.

To comply with local regulations the receiver shall be able to perform LBT (listen before talk) with a threshold of –80 dBm.

A.3.1.10 Receiver spurious requirement

The RX spurious requirement restricts the amount of power to be emitted in RX mode by Z-Wave devices. Any unwanted emission near the Z-Wave frequency may degrade other Z-Wave devices ability to receive weak signals.

The RX spurious limit is –70 dBm/100 kHz within ±1 MHz from the regional Z-Wave frequency as shown in the figure below:

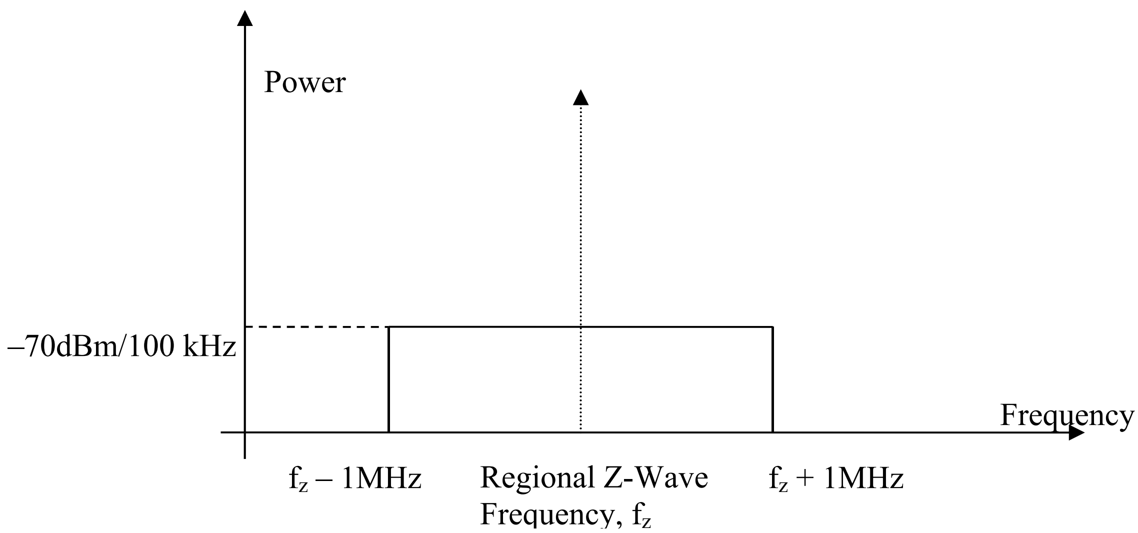

Figure A.5 – Receiver spurious requirements

A.3.1.11 Receiver blocking

Blocking is a measure of the capability of the receiver to receive a wanted modulated signal without exceeding a given degradation due to the presence of an unwanted input signal.

A conforming implementation shall be able to pass a blocking test as described below for all bit rates.

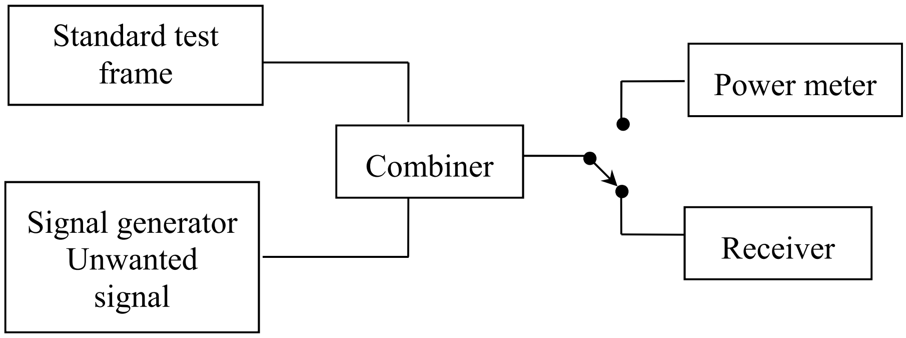

Figure A.6 – Receiver blocking test definitions

Method of measurement:

– Standard test frame is at the nominal frequency with normal modulation. Its power is adjusted down to the sensitivity level +3 dB.

– Unwanted signal is a CW (carrier wave) at a specific offset frequency. Its power is adjusted up until the receiver gets the FER (frame error rate) that correspond to the sensitivity level.

Limits:

Table A.9 – Blocking limits

Frequency offset	Limits
±1 MHz	–44 dBm
±2 MHz	–34 dBm
±5 MHz	–27 dBm
±10 MHz	–25 dBm

A.3.1.12 Receiver maximum input level of desired signal

The receiver maximum input level is the maximum power level of the desired signal, in decibels relative to 1 mW, present at the input of the receiver for which the CER criterion in Table A.7 is met. A receiver shall have a maximum input level greater than or equal to 0 dBm (Z-Wave devices should be able to sustain "zero" distance between two or more devices).

A.3.1.13 TX-to-RX turnaround time

The TX-to-RX turnaround time shall be measured from the trailing edge of the last transmitted symbol until the receiver is ready to begin the reception of the next PHY packet.

The TX-to-RX turnaround time shall be less than *aTurnaroundTimeTXRX* (see Table A.27).

A.3.1.14 RX-to-TX turnaround time

The RX-to-TX turnaround time shall be measured at the receiver from the trailing edge of the last symbol of a received packet until the transmitter is ready to begin transmission.

The RX-to-TX turnaround time shall be more than *aTurnaroundTimeRXTX* (see Table A.27).

A.3.2 PPDU format

This clause specifies the format of the PPDU packet.

The PPDU packet structure is presented so that the leftmost field shall be transmitted or received first.

Each byte shall be transmitted or received most significant bit first.

All multi-byte fields shall be transmitted or received least significant byte first.

Each PPDU packet consists of the following basic components:

– an SHR, which allows a receiving node to synchronize and lock onto the bit stream;
– a variable length payload, which carries the MAC sublayer frame;
– an EHR, which indicates the end of the frame (R1 only).

A.3.2.1 General packet format

The PPDU packet structure shall be formatted as illustrated in Figure A.5:

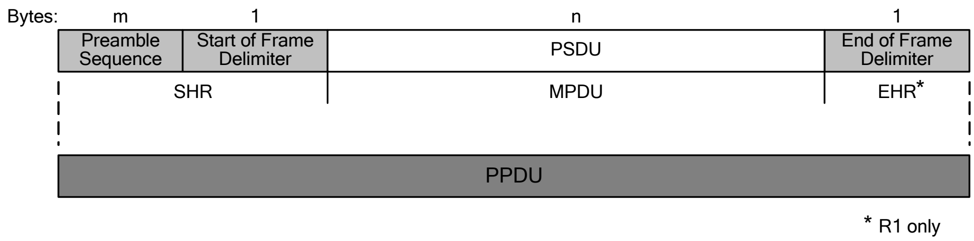

Figure A.7 – PPDU packet format

A.3.2.1.1 Preamble field

The preamble sequence field is used by a receiver to obtain symbol and bit synchronization. The preamble field shall be composed of a sequence of bytes containing the binary pattern "01010101". Figure A.8 shows the Manchester encoded preamble pattern for R1. Figure A.9 shows the NRZ encoded preamble pattern for R2/R3.

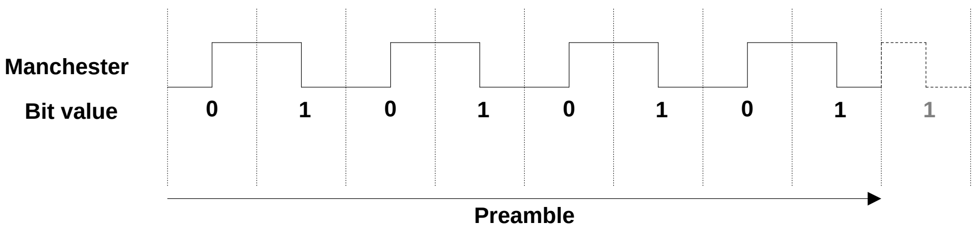

Figure A.8 – Manchester encoded preamble pattern (R1)

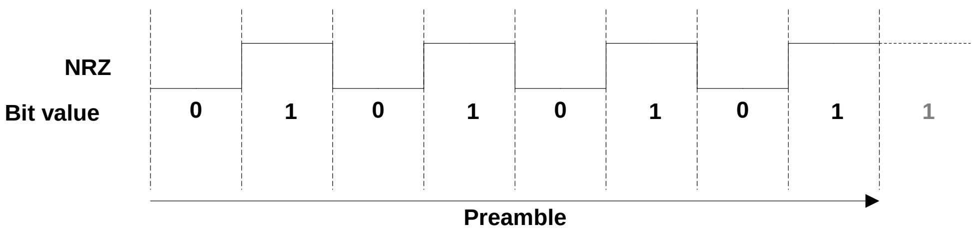

Figure A.9 – NRZ encoded preamble pattern (R2/R3)

To support an RX round robin in a multichannel system the preamble is extended beyond what would be needed in a single channel system. As the effective bandwidth decreases with increased preamble the preamble can be lower for first transmissions and increased for retransmissions so that the typical bandwidth is maximized. A table of Z-Wave preamble requirements is given in Table A.10.

Table A.10 – Preamble length

Channel configuration	Rate	Preamble length in bytes (min)		
		Singlecast/ broadcast	Multicast	Beam
1 and 2	R1 R2 R3	10 10 40	10 20 40	n/a 8 8
3	R1 R2 R3	n/a n/a 24	n/a n/a 24	n/a n/a 8

A.3.2.1.2 Start of frame delimiter (SFD) field

The start of frame delimiter (SFD) is an 8-bit field indicating the end of the synchronization (preamble) field and the start of the packet data. The SFD shall be formatted as illustrated in Table A.11.

Table A.11 – Format of the SFD field

Bits	7	6	5	4	3	2	1	0
Data	0	0	0	0	1	1	1	1

A.3.2.1.3 PSDU field

The PSDU field has a variable length and carries the data of the PHY packet.

A.3.2.1.4 End of frame delimiter (EFD) field

The end of frame delimiter (EFD) is only transmitted at R1 (Manchester encoded data rate). The 8 bit symbol is a sequence of 8 Manchester code violations each denoted E. Each violation, E, is defined as a symbol without transition for instance low-low. Figure A.10 illustrates the transmission of the EFD.

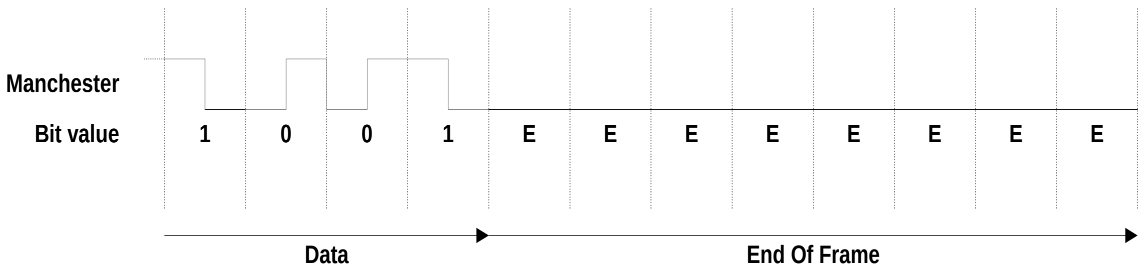

Figure A.10 – End of frame delimiter (EFD) pattern (R2/R3)

A.3.3 PHY service specifications

The PHY provides an interface between the MAC sublayer and the physical radio channel. The PHY provides two services, accessed through two service access points (SAPs): the PHY data, accessed through the PHY data SAP (PD-SAP), and the PHY management service, accessed through the PHY layer management entity SAP (PLME-SAP). The PLME is responsible for maintaining a database of managed objects pertaining to the PHY. This database is referred to as the PHY management information base (MIB). Figure A.11 shows the components and interfaces of the PHY.

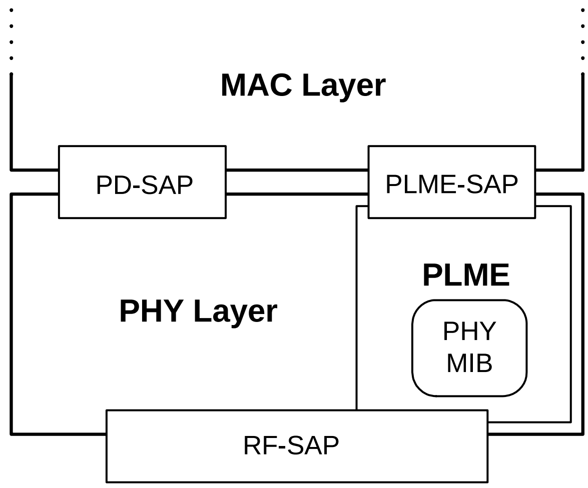

Figure A.11 – PHY reference model

A.3.3.1 PHY data service

The PD-SAP supports the transport of MPDUs between peer MAC sublayer entities. Table A.12 lists the primitives supported by the PD-SAP. These primitives are discussed in the clauses referenced in the table.

Table A.12 – PD-SAP primitives

PD-SAP primitive	Request	Confirm	Indication
PD-DATA	A.3.3.1.1	A.3.3.1.2	A.3.3.1.3

A.3.3.1.1 PD-DATA.request

The PD-DATA.request primitive requests the transfer of an MPDU (i.e., the PSDU) from the MAC sublayer to the local PHY entity.

A.3.3.1.1.1 Semantics of the PHY data request primitive

The semantics of the PD-DATA.request primitive is as follows:

```
PD-DATA.request   (
                  psduChannel,
                  psduRate,
                  psduLength,
                  psdu
                  )
```

Table A.13 specifies the parameters for the PD-DATA.request primitive:

Table A.13 – PD-DATA.request parameters

Name	Type	Valid range	Description
psduChannel	Enumeration	Valid channel selection A, B, or C according to the applied RF profile (see Table A.1)	The physical channel to be used for transmission of the PSDU.
psduRate	Enumeration	R1, R2 or R3. Validity of rate depends on applied RF profile (see Table A.1)	The bit rate to be used for the transmission of the PSDU.
psduLength	Byte	≤ *aMaxPHYPacketSizeRx*, where x is 1, 2 or 3 depending on data rate	The number of bytes contained in the PSDU to be transmitted by the PHY entity.
psdu	Set of bytes	–	The set of bytes forming the PSDU to be transmitted by the PHY entity.

A.3.3.1.1.2 When generated

The PD-DATA.request primitive is generated by a local MAC sublayer entity and issued to its PHY entity to request the transmission of an MPDU.

A.3.3.1.1.3 Effects on receipt

The receipt of the PD-DATA.request primitive by the PHY entity will cause the transmission of the supplied PSDU. Provided the transmitter is enabled (TX_ON mode), the PHY will first construct a PPDU, containing the supplied PSDU, and then transmit the PPDU. When the PHY entity has completed the transmission, it will issue the PD-DATA.confirm primitive with a status of SUCCESS.

If the PD-DATA.request primitive is received while the receiver is enabled (RX_ON mode) or if the transceiver is disabled (TRX_OFF mode), the PHY entity will issue the PD-DATA.confirm primitive with a status of RX_ON or TRX_OFF, respectively.

A.3.3.1.2 PD-DATA.confirm

The PD-DATA.confirm primitive confirms the end of the transmission of an MPDU (i.e., PSDU) from a local MAC sublayer entity to the physical media.

A.3.3.1.2.1 Semantics of the PHY data confirm primitive

The semantics of the PD-DATA.confirm primitive is as follows:

```
PD-DATA.confirm   (
                  Status
                  )
```

Table A.14 specifies the parameters for the PD-DATA.confirm primitive:

Table A.14 – PD-DATA.confirm parameters

Name	Type	Valid range	Description
status	Enumeration	SUCCESS, RX_ON or TRX_OFF	The result of the request to transmit a packet.

A.3.3.1.2.2 When generated

The PD-DATA.confirm primitive is generated by the PHY entity and issued to its MAC sublayer entity in response to a PD-DATA.request primitive. The PD-DATA.confirm primitive will return a status of either SUCCESS, indicating that the request to transmit was successful, or an error code of RX_ON or TRX_OFF.

A.3.3.1.2.3 Effects on receipt

The PD-DATA.confirm primitive allows the MAC sublayer entity to take proper action in case the transmission was unsuccessful.

A.3.3.1.3 PD-DATA.indication

The PD-DATA.indication primitive indicates the transfer of an MPDU (i.e., PSDU) from the PHY to the local MAC sublayer entity.

A.3.3.1.3.1 Semantics of the PHY data indication primitive

The semantics of the PD-DATA.indication primitive is as follows:

```
PD-DATA.indication      (
                        psduByte
                        )
```

Table A.15 specifies the parameters for the PD-DATA.indication primitive:

Table A.15 – PD-DATA.indication parameters

Name	Type	Valid range	Description
psduByte	Byte	–	One of a set of bytes forming the PSDU to be transmitted by the PHY entity.

A.3.3.1.3.2 When generated

The PD-DATA.indication primitive is generated by the PHY entity and issued to its MAC sublayer entity to transfer a received PSDU.

A.3.3.1.3.3 Effect on receipt

On receipt of the PD-DATA.indication primitive, the MAC sublayer is notified of the arrival of an MPDU across the PHY data service.

A.3.3.2 PHY management service

The PLME-SAP allows the transport of management commands between the MLME and the PLME. Table A.16 lists the primitives supported by the PLME-SAP. These primitives are discussed in the clauses referenced in the table.

Table A.16 – PLME-SAP primitives

PLME-SAP Primitive	Request	Confirm	Indication	Response
PLME-SFD	–	–	A.3.3.2.1	–
PLME-GET-CCA	A.3.3.2.2	A.3.3.2.3	–	–
PLME-GET	A.3.3.2.4	A.3.3.2.5	–	–
PLME-SET-TRX-MODE	A.3.3.2.6	A.3.3.2.7	–	–
PLME-SET	A.3.3.2.8	A.3.3.2.9	–	–

A.3.3.2.1 PLME-SFD.indication

The PLME-SFD.indication primitive indicates the reception of a start of frame delimiter from the PHY to the local MAC sublayer.

A.3.3.2.1.1 Semantics for the service primitive

The semantics of the PLME-SFD.indication primitive is as follows:

```
PLME-SFD.indication     (
                        psduChannel,
                        psduRate
                        )
```

The table below specifies the parameters for the PD-DATA.indication primitive.

Table A.17 – PLME-SFD.indication parameters

Name	Type	Valid range	Description
psduChannel	Enumeration	A, B, or C	The physical channel from which the PSDU was received.
psduRate	Enumeration	R1, R2, or R3	The bit rate of the received PSDU.

A.3.3.2.1.2 When generated

The PLME-SFD.indication primitive is generated by the PHY layer management entity (PLME) and issued to the local MAC sublayer through the PLME-SAP whenever a start of frame delimiter is detected by the PHY.

A.3.3.2.1.3 Effect on receipt

The MAC sublayer is notified of the reception of a start of frame delimiter.

A.3.3.2.2 PLME-GET-CCA.request

The PLME-GET-CCA.request primitive requests CCA on a given channel.

A.3.3.2.2.1 Semantics for the service primitive

The semantics of the PLME-GET-CCA.request primitive is as follows:

```
PLME-GET-CCA.request    (
                        channel
                        )
```

The table below specifies the parameters for the PLME-GET-CCA.request primitive.

Table A.18 – PLME-GET-CCA.request parameters

Name	Type	Valid range	Description
channel	Enumeration	Valid channel selection A, B, or C according to the applied RF profile (see Table A.1)	The physical channel on which the CCA shall be performed.

A.3.3.2.2.2 When generated

The PLME-GET-CCA.request primitive is generated by the MLME and issued to its PLME to obtain information from the PHY MIB.

A.3.3.2.2.3 Effect on receipt

On receipt of the PLME-GET-CCA.request primitive, the PLME will attempt to perform a CCA on the given channel. When the CCA is completed the PLME will issue a PLME-GET-CCA.confirm with a status of CCA_CLEAR, CCA_NOT-CLEAR or ERROR.

A.3.3.2.3 PLME-GET-CCA.confirm

The PLME-GET-CCA.confirm primitive reports the result of a CCA request from the PHY MIB.

A.3.3.2.3.1 Semantics for the service primitive

The semantics of the PLME-GET.confirm primitive is as follows:

```
PLME-GET-CCA.confirm   (
                       status
                       )
```

Table A.19 specifies the parameters for the PLME-GET-CCA.confirm primitive.

Table A.19 – PLME-GET-CCA.confirm parameters

Name	Type	Valid range	Description
status	Enumeration	CCA_CLEAR, CCA_NOT_CLEAR or RX_OFF	The result of the CCA request.

A.3.3.2.3.2 When generated

The PLME-GET-CCA.confirm primitive is generated by the PLME and issued to its MLME in response to a PLME-GET-CCA.request primitive. The PLME-GET.CCA.confirm primitive will return a status of either CCA_CLEAR, CCA_NOT-CLEAR or RX_OFF. The RX_OFF status is given if the transceiver is not in RX mode (and unable to perform a CCA).

A.3.3.2.3.3 Effect on receipt

On receipt of the PLME-GET.confirm primitive, the MLME is notified of the results of its request to perform a CCA. If the request was successful the status will be CCA_CLEAR or CCA_NOT_CLEAR depending on channel availability. If the request failed, the status parameter will indicate the error.

A.3.3.2.4 PLME-GET.request

The PLME-GET.request primitive requests information about a given PHY MIB attribute.

A.3.3.2.4.1 Semantics for the service primitive

The semantics of the PLME-GET.request primitive is as follows:

```
PLME-GET.request  (
                  MIBAttribute
                  )
```

Table A.20 specifies the parameters for the PLME-GET.request primitive.

Table A.20 – PLME-GET.request parameters

Name	Type	Valid range	Description
MIBAttribute	Enumeration	See Table A.28	The identifier of the PHY MIB attribute to get.

A.3.3.2.4.2 When generated

The PLME-GET.request primitive is generated by the MLME and issued to its PLME to obtain information from the PHY MIB.

A.3.3.2.4.3 Effect on receipt

On receipt of the PLME-GET.request primitive, the PLME will attempt to retrieve the requested PHY MIB attribute from its database. If the identifier of the MIB attribute is not found in the database, the PLME will issue the PLME-GET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE.

If the requested PHY MIB attribute is found, the PLME will issue the PLME-GET.confirm primitive with a status of SUCCESS.

A.3.3.2.5 PLME-GET.confirm

The PLME-GET.confirm primitive reports the result of an information request from the PHY MIB.

A.3.3.2.5.1 Semantics for the service primitive

The semantics of the PLME-GET.confirm primitive is as follows:

```
PLME-GET.confirm (
                  status,
                  MIBAttribute,
                  MIBAttributeValue
                  )
```

Table A.21 specifies the parameters for the PLME-GET.confirm primitive.

Table A.21 – PLME-GET.confirm parameters

Name	Type	Valid range	Description
status	Enumeration	SUCCESS or UNSUPPORTED_ATTRIBUTE	The result of the request for PHY MIB attribute information.
MIBAttribute	Enumeration	See clause A.3.4.2	The identifier of the PHY MIB attribute to get.
MIBAttributeValue	Various	Attribute specific	The value of the indicated PHY MIB attribute to get.

A.3.3.2.5.2 When generated

The PLME-GET.confirm primitive is generated by the PLME and issued to its MLME in response to a PLME-GET.request primitive. The PLME-GET.confirm primitive will return a status of either SUCCESS, indicating that the request to read a PHY MIB attribute was successful, or an error code of UNSUPPORTED_ATTRIBUTE. The reasons for these status values are fully described in clause A.3.3.2.4.3.

A.3.3.2.5.3 Effect on receipt

On receipt of the PLME-GET.confirm primitive, the MLME is notified of the results of its request to read a PHY MIB attribute. If the request to read a PHY MIB attribute was successful, the status parameter is set to SUCCESS. Otherwise, the status parameter will indicate the error.

A.3.3.2.6 PLME-SET-TRX-MODE.request

The PLME-SET-TRX-MODE.request primitive requests that the PHY entity changes the internal operating mode of the transceiver. The transceiver will have three main modes:

– transceiver disabled (TRX_OFF)

– transmitter enabled (TX_ON)

– receiver enabled (RX_ON).

A.3.3.2.6.1 Semantics for the service primitive

The semantics of the PLME-SET-TRX-MODE.request primitive is as follows:

```
PLME-SET-TRX-MODE.request    (
                              mode
                              )
```

Table A.22 specifies the parameters for the PLME-SET-TRX-MODE.request primitive.

Table A.22 – PLME-SET-TRX-MODE.request parameters

Name	Type	Valid range	Description
mode	Enumeration	RX_ON, TRX_OFF, FORCE_TRX_OFF, or TX_ON	The new mode in which to configure the transceiver.

A.3.3.2.6.2 When generated

The PLME-SET-TRX-MODE.request primitive is generated by the MLME and issued to its PLME when the current operational mode of the transceiver needs to be changed.

A.3.3.2.6.3 Effect on receipt

On receipt of the PLME-SET-TRX-MODE.request primitive, the PLME will cause the PHY to change to the requested mode. If the mode change is accepted, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status of SUCCESS.

If this primitive requests a mode that the transceiver has already configured, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status indicating the current mode, i.e., RX_ON, TRX_OFF, or TX_ON.

If this primitive is issued with the RX_ON or TRX_OFF argument and the PHY is busy transmitting a PPDU, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status BUSY_TX. When issuing BUSY_TX the PHY will ignore the mode request and the MAC shall reissue the PLME-SET-TRX-MODE.request if a mode shift is still intended.

If this primitive is issued with the TX_ON or TRX_OFF argument and the PHY is in RX_ON mode and is currently receiving data, the PHY will issue the PLME-SET-TRX-MODE.confirm primitive with a status BUSY_RX. When issuing BUSY_RX the PHY will ignore the mode request and the MAC shall reissue the PLME-SET-TRX-MODE.request if a mode shift is still intended.

If this primitive is issued with FORCE_TRX_OFF, the PHY will cause the PHY to go to the TRX_OFF mode irrespective of the mode the PHY is in.

A.3.3.2.7 PLME-SET-TRX-MODE.confirm

The PLME-SET-TRX-MODE.confirm primitive reports the result of a request to change the internal operating mode of the transceiver.

A.3.3.2.7.1 Semantics for the service primitive

The semantics of the PLME-SET-TRX-MODE.confirm primitive is as follows:

```
PLME-SET-TRX-MODE.confirm   (
                            status
                            )
```

Table A.23 specifies the parameters for the PLME-SET-TRX-MODE.confirm primitive.

Table A.23 – PLME-SET-TRX-MODE.confirm parameters

Name	Type	Valid range	Description
status	Enumeration	SUCCESS, RX_ON, TRX_OFF, TX_ON, BUSY_RX, or BUSY_TX	The result of the request to change the mode of the transceiver.

A.3.3.2.7.2 When generated

The PLME-SET-TRX-MODE.confirm primitive is generated by the PLME and issued to its MLME after attempting to change the internal operating mode of the transceiver.

A.3.3.2.7.3 Effect on receipt

The MLME is notified of the result of its request to change the internal operating mode of the transceiver. A status value of SUCCESS indicates that the internal operating mode of the transceiver was accepted. A status value of RX_ON, TRX_OFF, or TX_ON indicates that the transceiver is already in the requested internal operating mode. A status value of BUSY_TX is issued when the PHY is requested to change its mode while transmitting. A status value of BUSY_RX is issued when the PHY is requested to change its mode while receiving.

A.3.3.2.8 PLME-SET.request

The PLME-SET.request primitive attempts to set the indicated PHY MIB attribute to the given value.

A.3.3.2.8.1 Semantics for the service primitive

The semantics of the PLME-SET.request primitive is as follows:

```
PLME-SET.request   (
                   MIBAttribute,
                   MIBAttributeValue
                   )
```

Table A.24 specifies the parameters for the PLME-SET.request primitive.

Table A.24 – PLME-SET.request parameters

Name	Type	Valid range	Description
MIBAttribute	Enumeration	See clause A.3.4.2	The identifier of the MIB attribute to set.
MIBAttributeValue	Various	Attribute specific	The value of the indicated MIB attribute to set.

A.3.3.2.8.2 When generated

The PLME-SET.request primitive is generated by the MLME and issued to its PLME to write the indicated PHY MIB attribute.

A.3.3.2.8.3 Effect on receipt

On receipt of the PLME-SET.request primitive, the PLME will attempt to write the given value to the indicated PHY MIB attribute in its database. If the MIBAttribute parameter specifies an attribute that is not found in the database (see Table A.28), the PLME will issue the PLME-SET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE. If the MIBAttibuteValue parameter specifies a value that is out of the valid range for the given attribute, the PLME will issue the PLME-SET.confirm primitive with a status of INVALID_PARAMETER.

If the requested PHY MIB attribute is successfully written the new value will be taken in use immediately and the PLME will issue the PLME-SET.confirm primitive with a status of SUCCESS.

A.3.3.2.9 PLME-SET.confirm

The PLME-SET.confirm primitive reports the results of the attempt to set an MIB attribute.

A.3.3.2.9.1 Semantics for the service primitive

The semantics of the PLME-SET.confirm primitive is as follows:

```
PLME-SET.confirm (
                  status,
                  MIBAttribute
                  )
```

Table A.25 specifies the parameters for the PLME-SET.confirm primitive.

Table A.25 – PLME-SET.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS, UNSUPPORTED_ATTRIBUTE, or INVALID_PARAMETER	The status of the attempt to set the request MIB attribute.
MIBAttribute	Enumeration	See clause A.3.4.2	The identifiers of the MIB attribute being confirmed.

A.3.3.2.9.2 When generated

The PLME-SET.confirm primitive is generated by the PLME and issued to its MLME in response to a PLME-SET.request primitive. The PLME-SET.confirm primitive will return a status of either SUCCESS, indicating that the requested value was written to the indicated PHY MIB attribute, or an error code of UNSUPPORTED_ATTRIBUTE or INVALID_PARAMETER. The reasons for these status values are fully described in clause A.3.3.2.8.3.

A.3.3.2.9.3 Effect on receipt

On receipt of the PLME-SET.confirm primitive, the MLME is notified of the result of its request to set the value of a PHY MIB attribute. If the requested value was written to the indicated PHY MIB attribute, the status parameter is set to SUCCESS. Otherwise, the status parameter will indicate the error.

A.3.3.3 PHY enumerations description

Table A.26 shows PHY enumeration values defined in the PHY specification.

Table A.26 – PHY enumerations description

Enumeration	Description
BUSY	The CCA attempt has detected a busy channel.
BUSY_RX	The transceiver is asked to change its mode while receiving.
BUSY_TX	The transceiver is asked to change its mode while transmitting.
CCA_CLEAR	The CCA indicates a clear channel.
FORCE_TRX_OFF	The transceiver is to be switched off.
IDLE	The CCA has detected an idle channel.
INVALID_PARAMETER	A SET/GET request was issued with a parameter that is out of range.
CCA_NO_CLEAR	The CCA indicates a busy channel.
RX_ON	The transceiver is in or is to be configured into the receiver enabled mode.
RX_OFF	Status from a CCA request when not in RX mode .
SUCCESS	A SET/GET or a transceiver mode change was successful.
TRX_OFF	The transceiver is in or is to be configured into the transceiver disabled mode.
TX_ON	The transceiver is in or is to be configured into the transmitter enabled mode.
UNSUPPORTED_ATTRIBUTE	A SET/GET request was issued with the identifier of an attribute that is not supported.

A.3.4 PHY constants and MIB attributes

This clause specifies the constants and attributes required by the PHY.

A.3.4.1 PHY constants

The constants that define the characteristics of the PHY are presented in Table A.27. These constants are hardware dependent and cannot be changed during operation.

Table A.27 – PHY constants

Constants	Description	Value
aMaxPHYPacketSizeR1	The maximum PSDU size (in bytes) the PHY shall be able to receive at the data rate R1	64 byte
aMaxPHYPacketSizeR2	The maximum PSDU size (in bytes) the PHY shall be able to receive at the data rate R2	64 byte
aMaxPHYPacketSizeR3	The maximum PSDU size (in bytes) the PHY shall be able to receive at the data rate R3	170 byte
aTurnaroundTimeTXRX	TX-to-RX maximum turnaround time (see clause A.3.1.13)	1 ms
aTurnaroundTimerRXTX	RX-to-TX minimum turnaround time (see clause A.3.1.14)	1 ms

A.3.4.2 PHY MIB attributes

The PHY management information base (MIB) comprises the attributes required to manage the PHY. Each of these attributes can be read or written using the PLME-GET.request and PLME-SET.request primitives, respectively. The attributes contained in the PHY MIB are presented in Table A.28.

Table A.28 – PHY MIB attributes

Attribute	Type	Range	Description
phyCurrentTxChannel	Enumeration	A, B, C	The TX channel to use (see Table A.1).
phyMapChannelA	Enumeration	Available RF profiles (Refer to Table A.1)	Apply RF profile to channel A
phyMapChannelB	Enumeration	Available RF profiles (Refer to Table A.1)	Apply RF profile to channel B
phyMapChannelC	Enumeration	Available RF profiles (Refer to Table A.1)	Apply RF profile to channel C
phyTransmitPower	Enumeration	Valid output power levels (Refer to clause A.3.1.7)	The transmit power to use

A.4 Medium access (MAC) layer specification

The MAC sublayer handles all access to the physical layer and is responsible for the following tasks:

- Frame acknowledgment
- Retransmission
- Providing a reliable link between two peer MAC entities.

Constants and attributes that are specified and maintained by the MAC sublayer are written in the text of this clause in italics. Constants have a general prefix of "a", e.g., *aMacMaxFrameRetries*, and are listed in Tables A.45 and A.46. Attributes have a general prefix of "mac", e.g., *macHomeID*, and are listed in Table A.45.

A.4.1 MAC sublayer service specification

The MAC sublayer provides an interface between the network layer and the PHY layer. The MAC layer management entity (MLME) provides the service interfaces through which layer management functions may be invoked. The MLME is responsible for maintaining a database of managed objects pertaining to the MAC sublayer. This database is referred to as the MAC sublayer management information base (MAC MIB). Figure A.12 – depicts the components and interfaces of the MAC sublayer.

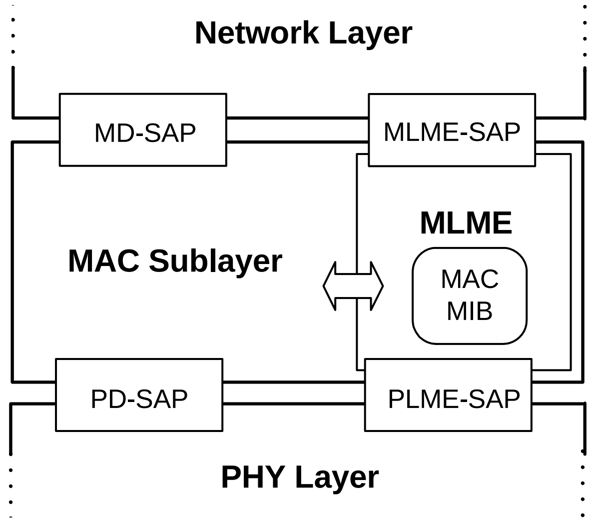

Figure A.12 – MAC sublayer reference model

The MAC sublayer provides two services to the network layer, accessed through two Service Access Points (SAPs):

– The MAC data service, accessed through the MD-SAP, and

– The MAC management service, accessed through the MLME-SAP.

A.4.1.1 MAC data service

The MD-SAP supports the transport of network layer protocol data units (NPDU) between peer network layer entities. Table A.29 lists the primitives supported by the MD-SAP. The primitives are discussed in the clauses referenced in the table.

Table A.29 – MD-SAP primitives

MD-SAP Primitive	Request	Confirm	Indication
MD-DATA	A.4.1.1.1	A.4.1.1.2	A.4.1.1.3

A.4.1.1.1 MD-DATA.request

The MD-DATA.request primitive requests the transfer of an NPDU (i.e., MSDU) from a local network layer entity to one or more peer network layer entities.

A.4.1.1.1.1 Semantics of the service primitive

The semantics of the MD-DATA. request primitive is as follows:

```
MD-DATA.request    (
                   SrcHomeID,
                   SrcNodeID,
                   DstNodeID,
                   msduLength,
                   msdu,
                   SequenceNumber,
                   TxType,
                   TxOptions,
                   BeamOption,
                   RateOption,
                   ChannelOption,
```

MulticastOption
)

Table A.30 specifies the parameters for the MD-DATA.request primitive:

Table A.30 – MD-Data.request parameters

Name	Type	Valid range	Description
SrcHomeID	Set of bytes	0x00000000 – 0xFFFFFFFF	The 4-byte network identifier of the node from which the MSDU is being transferred.
SrcNodeID	Byte	0x00 .. 0xE8	The 1-byte individual node address of the node from which the MSDU is being transferred. 0x00: Uninitialized Z-Wave Node ID 0x01 – 0xE8: NodeID 0xE9 – 0xFE: Reserved 0xFF: Broadcast NodeID
DstNodeID	Byte	0x00 .. 0xFF	The 1-byte node identifier of the node to which the MSDU is being transferred.
msduLength	Byte	≤ *aMacMaxMSDUSizeX*	The number of bytes contained in the MSDU to be transmitted by the MAC sublayer entity.
msdu	Set of bytes	–	The set of bytes forming the MSDU to be transmitted by the MAC sublayer entity.
SequenceNumber	Set of bits	–	The unique number of this frame. Use same value for retransmissions.
TxType	Byte	0000.xxxx	The transmission type for this MSDU. 0x01 = Singlecast or broadcast transmission 0x02 = Multicast transmission 0x03 = Acknowledge transmission
TxOptions	Byte	0000.xxxx (where x can be 0 or 1)	The transmission options for this MSDU. The control word is formed as a bit wise OR of one or more of the following values: 0x01 = Routed frame transmission 0x02 = Acknowledged transmission. 0x04 = Low power transmission (Note 1)
BeamOption	Byte	0x00 .. 0x04	The battery support options for this MSDU. 0x00: No beam 0x01: Repeated beam 0x02: Short Continuous beam 0x03: Long Continuous beam 0x04: Fragmented beam
ChannelOption	Enumeration	A,B, or C (Note 2)	The channel to use for this MSDU.
RateOption	Enumeration	R1, R2 or R3 Validity depends on the applied RF profile for the channel	The data rate to use for this MSDU.

Table A.30 – MD-Data.request parameters

Name	Type	Valid range	Description
MulticastOption	Byte + Bitmap (30 bytes)	–	If TxType is "Multicast transmission", this bitmap indicates the intended recipients.
NOTE 1 – Bit 2 (0x04) controls which transmission power level the MAC layer requests from the PHY layer. NOTE 2 – Channels A, B and C are preconfigured to RF profiles from the list presented in clause 7.1.2.1 via MAC MIB attributes defined in Table A-47. Channel configuration MAC MIB profiles map directly to PHY MIB profiles.			

A.4.1.1.1.2 When generated

The MD-DATA.request primitive is generated by a local network layer entity when a data NPDU (i.e., MSDU) is to be transferred to one or more peer network layer entities.

A.4.1.1.1.3 Effects on receipt

The MAC sublayer entity begins the transmission of the supplied MSDU.

The MAC sublayer builds an MPDU to transmit from the supplied parameters. The TxOptions parameters indicate optional parameters on how the MAC sublayer data service transmits the supplied MSDU.

The MAC sublayer checks for a clear channel access (CCA, see A.4.4.1.1). If the macClearChannel attribute is TRUE the MAC sublayer enables the transmitter by issuing the PLME-SET-TRX-MODE.request primitive with a mode of TX_ON to the PHY. On receipt of the PLME-SET-TRX-MODE.confirm primitive with a status of either SUCCESS or TX_ON the constructed MPDU is then transmitted by issuing the PD-DATA.request primitive. Finally, on receipt of the PD-DATA.confirm primitive, the MAC sublayer disables the transmitter by issuing the PLME-SET-TRX-MODE.request primitive with a mode of RX_ON to the PHY.

If the TxOptions parameter specifies that acknowledged transmission is required, the MAC sublayer will enable its receiver immediately following the transmission of the MPDU and wait for an acknowledgment from the recipient for at minimum *aMacMinAckWaitDuration* symbols. If the MAC sublayer does not receive an acknowledgment within this time, it will retransmit as defined by *aMacMaxFrameRetries*. If the MAC sublayer still does not receive an acknowledgment from the recipient, it will discard the MSDU and issue the MD-DATA.confirm primitive with a status of NO_ACK.

The TxOptions.LowPower is an option targeting the PHY layer. The option makes the MAC choose between the two MAC constants aMacTxPhyPowerLevelLow and aMacTxPhyPowerLevelNormal.

If the MPDU was successfully transmitted the MAC sublayer will issue the MD-DATA.confirm primitive with a status of SUCCESS.

If the MPDU could not be transmitted due to an occupied channel (clause A.4.4.1.1) the MAC sublayer will issue the MD-DATA.confirm primitive with a status of NO_CCA.

If any parameter in the MD-DATA.request primitive is not supported or is out of range, the MAC sublayer will issue the MD-DATA.confirm primitive with a status of INVALID_PARAMETER.

If the MSDU length is longer than *aMacMaxMSDUSizeX* the MAC sublayer will issue the MD-DATA.confirm primitive with a status of FRAME_TOO_LONG.

The MD-Data.request parameters are used to construct the MAC Header (MHR) see clause A.4.2.

A.4.1.1.2 MD-DATA.confirm

The MD-DATA.confirm primitive reports the status of a data NPDU (MSDU) transfer.

A.4.1.1.2.1 Semantics of the service primitive

The semantics of the MD-DATA.confirm primitive is as follows:

MD-DATA.confirm (
status
)

Table A.31 specifies the parameters for the MD-DATA.confirm primitive.

Table A.31 – MD-Data.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS, NO_ACK, INVALID_PARAMETER, NO_CCA or FRAME_TOO_LONG	The status of the last MSDU transmission.

A.4.1.1.2.2 When generated

The MD-DATA.confirm primitive is generated by the MAC sublayer entity in response to an MD-DATA.request primitive. The MD-DATA.confirm primitive returns a status of either SUCCESS, NO_ACK, INVALID_PARAMETER, NO_CCA or FRAME_TOO_LONG.

The primitive may be delayed if the MAC has to wait for an acknowledgment. If the frame is transmitted using repeated beam or fragmented beam the delay may be more than one second.

A.4.1.1.2.3 Effects on receipt

The network layer of the initiating node is notified of the result of its request to transmit. If the transmission attempt was successful, the status parameter will be set to SUCCESS. Otherwise, the status parameter will indicate the error.

A.4.1.1.3 MD-DATA.indication

The MD-DATA.indication primitive indicates the reception of a data NPDU (i.e., MSDU) from the MAC sublayer to the local network layer entity.

A.4.1.1.3.1 Semantics of the service primitive

The semantics of the MD-DATA.indication primitive is as follows:

MD-DATA.indication (
FrmType
SrcHomeID,
SrcNodeID,
DstNodeID,
msduLength,
msdu,
SequenceNumber,
ChannelOption,
RateOption,
MulticastOption
)

Table A.32 specifies the parameters for the MD-DATA.indication primitive.

FrmType indicates if the data delivered to the network layer is a data frame or a beam fragment.

In case of a beam fragment the SrcHomeID field is of the form "0x55xxxxxx" or "0x54xxxxxx", and only the following fields carry data:

* SrcHomeID (Byte 1: BeamTag = 0x55 or 0x54, Byte 2: BeamInfo)

* ChannelOption

* RateOption

All other fields should be ignored.

The SrcHomeID value "0x55xxxxxx" signals that the BeamInfo byte carries a destination NodeID. The SrcHomeID value "0x54xxxxxx" signals that the BeamInfo byte carries a value that is reserved for future use.

Table A.32 – MD-DATA.indication parameters

Name	Type	Valid range	Description
FrmType	Enumeration	0x01 – 0xFF	Code indicating the frame type: 0x00: Uninitialized type 0x01: Singlecast frame 0x02: Multicast frame 0x03: Ack frame 0x04 – 0x07: Not to be used (Note) 0x08: Routed frame 0x09: Beam fragment 0x0A – 0xFF: Reserved
SrcHomeID	Set of bytes	0x00000000 –0xFFFFFFFF The value 0x00000000 is reserved for un-initialized legacy nodes. No network may use the HomeID 0x00000000 as a network identifier.	The 4-byte network identifier of the node from which the MSDU was received.
SrcNodeID	Byte	0x00 – 0xFF The value 0x00 is reserved for un-initialized nodes.	The 1-byte individual node address of the node from which the MSDU was received. 0x00: Uninitialized NodeID 0x01 – 0xE8: NodeID 0xE9 – 0xFF: Reserved
DstNodeID (Not applicable for multicast frame type)	Byte	0 x 00–0 x FF	The 1-byte individual node address of the node from which the MSDU is being transferred. 0x00: Uninitialized NodeID 0x01 – 0xE8: NodeID 0xE9 – 0xFE: Reserved 0xFF: Broadcast NodeID

Table A.32 – MD-DATA.indication parameters

Name	Type	Valid range	Description
msduChannel	Enumeration	A, B, or C	The physical channel from which the MSDU was received.
msduRate	Enumeration	R1, R2, or R3	The bit rate of the received MSDU.
msduLength	Byte	≤ *aMacMaxMSDUSizeX*	The number of bytes contained in the MSDU.
Msdu	Set of bytes	–	The set of bytes forming the MSDU.
SequenceNumber	Set of bits	–	The unique number of this frame. Use same value for retransmissions.
MulticastOption	Byte + Bitmap (30 bytes)	–	If TxType is "Multicast transmission", this bitmap indicates the intended recipients.
NOTE – These fields shall not be used due to backward compatibility.			

A.4.1.1.3.2 When generated

The MD-DATA.indication primitive is generated by the MAC sublayer on receipt of a frame from the PHY layer. If the frame is free of errors, the frame is forwarded to the Z-Wave network layer.

A.4.1.1.3.3 Effects on receipt

The network layer is notified of the arrival of data.

Also beam fragments shall be forwarded to the network layer, which may then forward the beam fragment to higher layers. Higher layers of an FLN node may decide to re-enable sleep mode if the NodeID of a beam fragment is intended for another node.

A.4.1.1.4 Data service message sequence charts

Figure A.13 illustrates the sequence of messages necessary for a successful data transfer between two nodes.

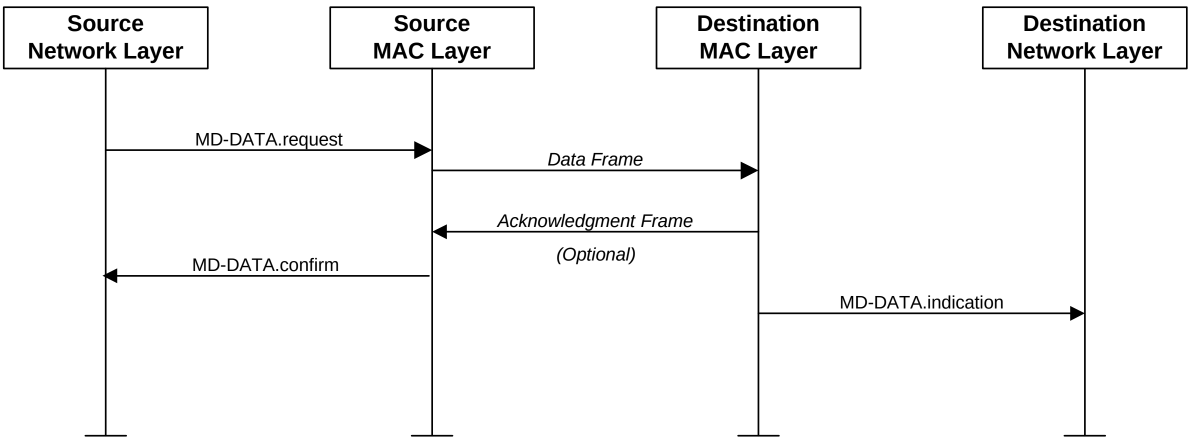

Figure A.13 – MAC data service message sequence chart

A.4.1.2 MAC management service

The MLME-SAP allows the transport of management commands between the next higher layer and the MLME. Table A.33 summarizes the primitives supported by the MLME through the MLME-SAP interface.

The primitives are discussed in the clauses referenced in the table.

Table A.33 – MLME-SAP primitives

MLME-SAP Primitive	Request	Confirm	Indication	Response
MLME-GET	A.4.1.2.1	A.4.1.2.2	–	–
MLME-SET	A.4.1.2.3	A.4.1.2.4	–	–
MLME-RESET	A.4.1.2.5	A.4.1.2.6	–	–

A.4.1.2.1 MLME-GET.request primitives

The MLME-GET.request primitive requests information about a given MAC MIB attribute.

A.4.1.2.1.1 Semantics of the service primitive

The semantics of the MLME-GET.request primitive is as follows:

MLME-GET.request (
MACMACMIBAttribute
)

Table A.33a specifies the parameters for the MLME-GET.request primitive.

Table A.33a – MLME-GET.request parameters

Name	Type	Valid range	Description
MACMIBAttribute	Integer	See Table A.47	The identifier of the MAC MIB attribute to read.

A.4.1.2.1.2 When generated

The MLME-GET.request primitive is generated by the next higher layer and issued to its MLME to obtain information from the MAC MIB.

A.4.1.2.1.3 Effects on receipt

On receipt of the MLME-GET.request primitive, the MLME attempts to retrieve the requested MAC MIB attribute from its database. If the identifier of the MAC MIB attribute is not found in the database, the MLME will issue the MLME-GET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE.

If the requested MAC MIB attribute is successfully retrieved, the MLME will issue the MLME-GET.confirm primitive with a status of SUCCESS and the MAC MIB attribute value.

A.4.1.2.2 MLME-GET.confirm primitives

The MLME-GET.confirm primitive reports the results of an information request from the MAC MIB.

A.4.1.2.2.1 Semantics of the service primitive

The semantics of the MLME-GET.confirm primitive is as follows:

```
MLME-GET.confirm        (
                        status,
                        MACMIBAttribute,
                        MACMIBAttributeValue
                        )
```

Table A.34 specifies the parameters for the MLME-GET.confirm primitive.

Table A.34 – MLME-GET.confirm parameters

Name	Type	Valid range	Description
Status	Enumeration	SUCCESS or UNSUPPORTED_ATTRIBUTE	The result of the request for MAC MIB attribute information.
MACMIBAttribute	Integer	See Table A.47	The identifier of the MAC MIB attribute that was read.
MACMIBAttributeValue	Various	Attribute specific, See Table A.47	The value of the indicated MAC MIB attribute that was read.

A.4.1.2.2.2 When generated

The MLME-GET.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-GET.request primitive. This primitive returns a status of either SUCCESS, indicating that the request to read a MAC MIB attribute was successful, or an error code of UNSUPPORTED_ATTRIBUTE.

A.4.1.2.2.3 Effects on receipt

The MLME-GET.confirm primitive reports the results of the request to read a MAC MIB attribute. If the request to read a MAC MIB attribute was successful, the status parameter will be set to SUCCESS. Otherwise, the status parameter indicates the error.

A.4.1.2.3 MLME-SET.request primitives

The MLME-SET.request primitive attempts to write the given value to the indicated MAC MIB attribute.

A.4.1.2.3.1 Semantics of the service primitive

The semantics of the MLME-SET.request primitive is as follows:

```
MLME-SET.request  (
                  MACMIBAttribute,
                  MACMIBAttributeValue
                  )
```

Table A.35 specifies the parameters for the MLME-SET.request primitive.

Table A.35 – MLME-SET.request parameters

Name	Type	Valid range	Description
MACMIBAttribute	Integer	See Table A.47	The identifier of the MAC MIB attribute to write.
MACMIBAttributeValue	Various	Attribute specific, See Table A.47	The value to write to the indicated MAC MIB attribute.

A.4.1.2.3.2 When generated

The MLME-SET.request primitive is generated by the next higher layer and issued to its MLME to write the indicated MAC MIB attribute.

A.4.1.2.3.3 Effects on receipt

On receipt of the MLME-SET.request primitive, the MLME attempts to write the given value to the indicated MAC MIB attribute in its database. If the MACMIBAttribute parameter specifies an attribute that is not found in the database Table A.47, the MLME will issue the MLME-SET.confirm primitive with a status of UNSUPPORTED_ATTRIBUTE. If the MACMIBAttributeValue parameter specifies a value that is out of the valid range for the given attribute, the MLME will issue the MLME-SET.confirm primitive with a status of INVALID_PARAMETER.

If the requested MAC MIB attribute is successfully written, the MLME will issue the MLME-SET.confirm primitive with a status of SUCCESS.

A.4.1.2.4 MLME-SET.confirm primitives

The MLME-SET.confirm primitive reports the results of an attempt to write a value to a MAC MIB attribute.

A.4.1.2.4.1 Semantics of the service primitive

The semantics of the MLME-SET.confirm primitive is as follows:

```
MLME-SET.confirm     (
                     status,
                     MACMIBAttribute
                     )
```

Table A.36 specifies the parameters for the MLME-SET.confirm primitive.

Table A.36 – MLME-SET.confirm parameters

Name	Type	Valid range	Description
status	Enumeration	SUCCESS, UNSUPPORTED_ATTRIBUTE, or INVALID_PARAMETER	The result of the request to write the MAC MIB attribute.
MACMIBAttribute	Integer	Table A.47	The identifier of the MAC MIB attribute that was written.

A.4.1.2.4.2 When generated

The MLME-SET.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-SET.request primitive. The MLME-SET.confirm primitive returns a status of either SUCCESS, indicating that the requested value was written to the indicated MAC MIB attribute, or an error code of UNSUPPORTED_ATTRIBUTE or INVALID_PARAMETER.

A.4.1.2.4.3 Effects on receipt

On receipt of the MLME-SET.confirm primitive, the next higher layer is notified of the result of its request to set the value of a MAC MIB attribute. If the requested value was written to the indicated MAC MIB attribute, the status parameter will be set to SUCCESS. Otherwise, the status parameter indicates the error.

A.4.1.2.5 MLME-RESET.request primitives

The MLME-RESET.request primitive allows the next higher layer to request that the MLME performs a reset operation.

A.4.1.2.5.1 Semantics of the service primitive

The semantics of the MLME-RESET.request primitive is as follows:

```
MLME-RESET.request      (
                        SetDefaultMIB
                        )
```

Table A.37 specifies the parameter for the MLME-RESET.request primitive.

Table A.37 – MLME-RESET.request parameters

Name	Type	Valid range	Description
SetDefaultMIB	Boolean	TRUE or FALSE	If TRUE, the MAC sublayer is reset and all MAC MIB attributes are set to their default values.

A.4.1.2.5.2 When generated

The MLME-RESET.request primitive is generated by the next higher layer and issued to its MLME to request a reset of the MAC sublayer to its initial conditions. The MLME-RESET.request primitive is issued when a node is excluded from a network.

A.4.1.2.5.3 Effects on receipt

On receipt of the MLME-RESET.request primitive, the MLME issues the PLME-SET-TRX-STATE.request primitive with a state of TRX_OFF. On receipt of the PLME-SET-TRX-STATE.confirm primitive, the MAC sublayer is then set to its initial conditions, clearing all internal variables to their default values. If the SetDefaultMIB parameter is set to TRUE, the MAC MIB attributes are set to their default values.

If the PLME-SET-TRX-STATE.confirm primitive is successful, the MLME will issue the MLME-RESET.confirm primitive with the status of SUCCESS. Otherwise, the MLME issues the MLME-RESET.confirm primitive with the status of DISABLE_TRX_FAILURE.

A.4.1.2.6 MLME-RESET.confirm primitives

The MLME-RESET.confirm primitive reports the results of the reset operation.

A.4.1.2.6.1 Semantics of the service primitive

The semantics of the MLME-RESET.confirm primitive is as follows:

```
MLME-RESET.confirm      (
                        status
                        )
```

Table A.38 specifies the parameter for the MLME-RESET.confirm primitive.

Table A.38 – MLME-RESET.confirm parameters

Name	Type	Valid range	Description
status	Enumeration	SUCCESS or DISABLE_TRX_FAILURE	The result of the reset operation.

A.4.1.2.6.2 When generated

The MLME-RESET.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-RESET.request primitive and following the receipt of the PLME-SET-TRX-STATE.confirm primitive.

A.4.1.2.6.3 Effects on receipt

On receipt of the MLME-RESET.confirm primitive, the next higher layer is notified of its request to reset the MAC sublayer. This primitive returns a status of SUCCESS if the request to reset the MAC sublayer was successful. Otherwise, the status is set to DISABLE_TRX_FAILURE, indicating that the attempt to disable the transceiver was unsuccessful.

A.4.1.3 MAC enumeration description

This clause explains the meaning of the enumerations used in the primitives defined in the MAC sublayer specification. Table A.39 shows a description of the MAC enumeration values.

Table A.39 – MAC enumerations description

Enumeration	Description
SUCCESS	The requested operation was completed successfully.
DISABLE_TRX_FAILURE	The attempt to disable the transceiver has failed.
FRAME_TOO_LONG	The frame has a length that is greater than *aMacMaxMSDUSizeX*.
INVALID_PARAMETER	A parameter in the primitive is out of the valid range.
NO_ACK	No acknowledgment was received after*aMacMaxFrameRetries*.
NO_CCA	No clear channel access was possible after a period of *.macCCARetryDuration.*
UNSUPPORTED_ATTRIBUTE	A SET/GET request was issued with the identifier of a MAC MIB attribute that is not supported.

A.4.2 MAC frame formats

This clause specifies the format of the MAC frame (MPDU). Each MAC frame consists of the following basic components:

1. An MHR, which comprises address, frame control and length information.
2. A MAC data payload, of variable length, which contains information specific to the frame type.
   – Acknowledgment frames do not contain a payload.
3. An MFR, which contains a FCS.

The frames in the MAC sublayer are described as a sequence of fields in a specific order. All frame formats in this clause are depicted in the order in which they are transmitted by the PHY, from left to right, where the leftmost bit (bit 7) is transmitted first in time. Bits within each field are numbered from k – 1 (leftmost and most significant) down to 0 (rightmost and least significant), where the length of the field is *k* bits. If fields are longer than a single byte the most significant byte is transmitted first.

The standard MAC frame format is composed of an MHR, a MAC payload, and an MFR. The fields of the MHR appear in a fixed order. The general MAC frame shall be formatted as illustrated in Figure A.14.

The MAC frame format for Channel Configurations 1, 2 and Channel Configuration 3 nodes is not identical. In the following description of the frame format all differences are described in a Channel Configurations 1, 2 and a Channel Configuration 3 section.

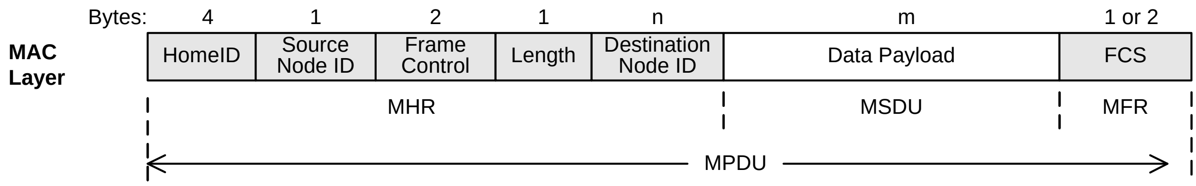

Figure A.14 – General MAC frame format (Channel Configurations 1, 2 only)

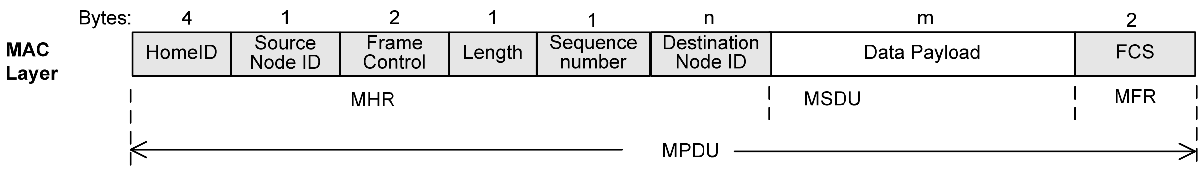

Figure A.15 – General MAC frame format (channel configuration 3 only)

Multicast frames use a special format for addressing.

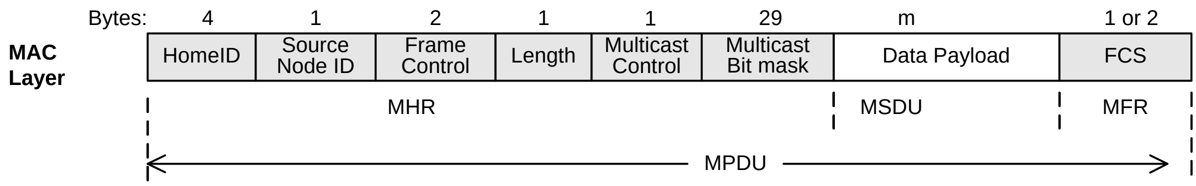

Figure A.16 – Multicast MAC frame format (channel configurations 1, 2 only)

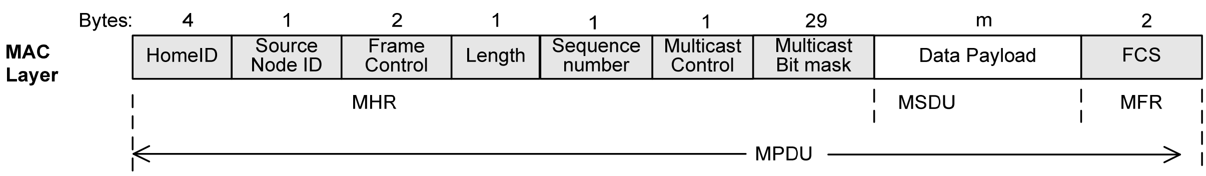

Figure A.17 – Multicast MAC frame format (channel configuration 3)

Refer to clause A.4.2.5.1 for details on transmission and processing of multicast frames.

A.4.2.1 HomeID

The HomeID identifier field is 4 bytes in length and specifies the unique network identifier. All nodes in a network have the same HomeID.

The HomeID is assigned by a primary node during inclusion.

Byte \ Bit	7	6	5	4	3	2	1	0
0	HomeID (0) MSB							
1	HomeID (1)							
2	HomeID (2)							
3	HomeID (3) LSB							

Figure A.18 – Format of a HomeID field

The MAC shall support configuration of a promiscuous mode; forwarding all frames to higher layers. Refer to clause A.4.4.1.3.

A.4.2.2 Source NodeID

The source NodeID is an 8 bit unique identifier of a node. Together with the HomeID, the source NodeID identifies the node that originated the frame.

The NodeID is assigned by a primary node during inclusion.

Byte \ Bit	7	6	5	4	3	2	1	0
4	Source NodeID							

Figure A.19 – Format of a source NodeID field

The source NodeID range is as shown in Table A.40:

Table A.40 – Source NodeID

NodeID	Node type
0x00	Uninitialized NodeID
0x01 – 0xE8	NodeID
0xE9 – 0xFE	Reserved

A.4.2.3 Frame control

The frame control field is 16 bits in length and contains information defining the frame type, addressing fields, and other control flags. The frame control field shall be formatted as illustrated in Figures A.20 and A.21:

Byte \ Bit	7	6	5	4	3	2	1	0
5	Routed	Ack Req	Low Power	Speed modified	Header Type			
6	Reserved	Beam control		Reserved	Sequence Number			

Figure A.20 – Format of a frame control field (channel configurations 1, 2 only)

Byte \ Bit	7	6	5	4	3	2	1	0
5	Ack Req	Low Power	Reserved		Header Type			
6	Reserved	Beam control			Reserved			

Figure A.21 – Format of a frame control field (channel configuration 3)

A.4.2.3.1 Routed subfield (channel configurations 1, 2 only)

The routed subfield is 1 bit in length and shall be set to 0 when the frame is not routed and set to 1 when the frame is routed.

The information is used by the upper layers.

A.4.2.3.2 Ack Req subfield

The Ack Req subfield is 1 bit in length and set to 1 when the source node wants the destination node to acknowledge the frame, and the bit is set to 0 when no acknowledgment is needed.

The information is used by the MAC layer to enable acknowledgment timeout and potential retransmission.

A.4.2.3.3 Low power subfield

The Low Power subfield is 1 bit in length and is set by the source node. The information is used to configure the PHY prior to frame transmission. The bit informs a destination node that the actual frame was transmitted using low power.

A destination node may return an acknowledgment frame in low power in response to the low-power bit.

A.4.2.3.4 Speed modified subfield (Channel Configurations 1, 2 only)

Speed modified (1 bit)

Set to 1 if the singlecast frame is sent at a lower speed than supported by the source and destination. The field is not used for routed and multicast frames. The field shall be set to 0 if not used.

A.4.2.3.5 Header type subfield

The header type subfield defines the frame header type.

Table A.41 – Header type

Header type	Frame
0x1	Singlecast frame
0x2	Multicast frame
0x3	Transfer acknowledge
0x4..0x7	Not to be used (Note)
0x8	Routed frame (channel configuration 3 only)
0x9..0xB	Not to be used (Note)
0xC..0xF	*Reserved*
NOTE – These fields shall not be used due to backward compatibility.	

A broadcast frame is a singlecast frame (header type 0x1) carrying destination NodeID = 0xFF; see clause A.4.2.6.

A.4.2.3.6 Beam control

The interpretation of this sub-field shall depend on the channel configuration being used. For nodes operating in channel configuration 1 or 2, bit 6 shall signal that the sending FL node shall be woken up with a long repeated or continuous beam. Bit 5 shall signal that the sending FL node shall be woken up with a short repeated or continuous beam.

For nodes operating in channel configuration 3, bit 6 shall signal that the sending FL node shall be woken up with a fragmented beam. Bit 5 is reserved.

A.4.2.3.7 Sequence number (channel configurations 1, 2 only)

The sequence number field is a number provided by higher layers when transmitting. The PHY and MAC layers shall forward the sequence number value unmodified. Upon reception of a frame, if acknowledgment is requested, the MAC layer shall return the received sequence number value in the acknowledgment frame.

The acknowledgment mechanism described in clause A.4.4.1.4 shall check that the received sequence number in an acknowledgment matches the transmitted sequence number value.

The valid range is 0x1-0xf.

In order to support legacy implementations, a transmitting MAC entity shall accept the sequence number value zero in an acknowledgment frame irrespective of the sequence number transmitted. An ITU-T G.9959 MAC entity shall be able to return the sequence number value zero in response to an incoming singlecast frame with the sequence number value zero.

A.4.2.4 Length

The Length field is 1 byte in length and indicates the length of the whole MPDU in bytes.

Byte \ Bit	7	6	5	4	3	2	1	0
7	Length							

Figure A.22 – Format of a length field

The length is limited to *aMacMaxMSDUSizeX*. The actual limits can be found in Tables A.45 and A.46. A receiving node shall not read more bytes than the maximum length allowed for the actual data rate.

A.4.2.5 Sequence number (channel configuration 3 only)

The sequence number field is a number provided by higher layers when transmitting. The PHY and MAC layers shall forward the sequence number value unmodified. Upon reception of a frame, if acknowledgment is requested, the MAC layer shall return the received sequence number value in the acknowledgment frame.

The acknowledgment mechanism described in clause A.4.4.1.4 shall check that the received sequence number in an acknowledgment matches the transmitted sequence number value.

The valid range is 0x00-0xff.

A.4.2.6 Destination NodeID

The destination NodeID is used to address individual nodes. The NodeID allocation process is described in clause A.2.1.3. Together with the HomeID the destination NodeID identifies the target node of the frame.

The MAC shall support the configuration of a promiscuous mode; forwarding all frames to higher layers.

The destination NodeID range is as shown in Table A.42:

Table A.42 – Destination NodeID

NodeID	Node type
0x00	Uninitialized NodeID
0x01 – 0xE8	NodeID
0xE9 – 0xFE	Reserved
0xFF	Broadcast NodeID

The destination NodeID 0xFF is a special broadcast value. The MAC layer shall always forward a broadcast frame to higher layers.

A.4.2.6.1 Multicast destination address

Multicast frames carry a destination bit map as shown in Figure A.20:

Byte \ Bit	7	6	5	4	3	2	1	0
8	Address Offset			Number of Mask Bytes				
9	Mask Byte 0							
10	Mask Byte 1							
11	Mask Byte 2							
...	....							
37	Mask Byte 28							

Figure A.23 – Multicast destination address format

A multicast frame cannot be acknowledged. The ACK request bit in the frame control field shall always be set to zero (see clause A.4.2.3).

A transmitting node shall set the address offset field to zero and 'Number of Mask Bytes' field to 29. A receiving node shall accept the values of the address offset field outlined in Table A.43.

The address offset field is a 3 bit field. The encoding of the field is outlined in Table A.43:

Table A.43 – Multicast header address offset field

Address offset value	Bit address offset
0	0
1	32
2	64
3	96
4	128
5	160
6	192
7	224

Each bit of a mask byte represents a node. Mask Byte 0 represents the NodeIDs 1 to 8.

As an example a mask byte 0 value of 0xC5 (in binary: 11000101) covers nodeIDs 1, 3, 7 and 8 while a mask byte 1 value of 0xC5 (in binary: 11000101) covers nodeIDs 9, 11, 15 and 16.

The destination NodeID can be calculated using the following formula.

$$\text{Destination NodeID} = \text{Bit Address offset} + \text{Mask Byte Number} \times 8 + \text{Mask Bit Number} + 1$$

A.4.2.7 Data payload

The frame payload field has a variable length and contains information specific to individual frames. An acknowledgment frame has no data payload.

A.4.2.8 FCS

An 8-bit non-correcting frame checksum is used for checking frame integrity on R1 and R2 data rates. The checksum is calculated over the fields starting at the HomeID field and ending at the data payload field as shown in Figure A.24.

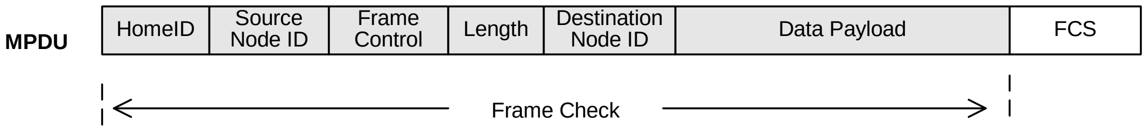

Figure A.24 – Frame check coverage

Byte \ Bit	7	6	5	4	3	2	1	0
(following payload)	Check Sum							

Figure A.25 – Format of check sum field

The checksum is calculated as an odd (XOR'ed) checksum as shown in the following algorithm.

```
BYTE GenerateCheckSum(BYTE *Data, BYTE Length)
{
  BYTE CheckSum = 0xFF;
  for (; Length > 0; Length--)
  {
    CheckSum ^= *Data++;
  }
  return CheckSum;
}
```

A.4.2.9 CRC

A 16-bit non-correcting frame CRC is used for checking frame integrity on R3 data rate.

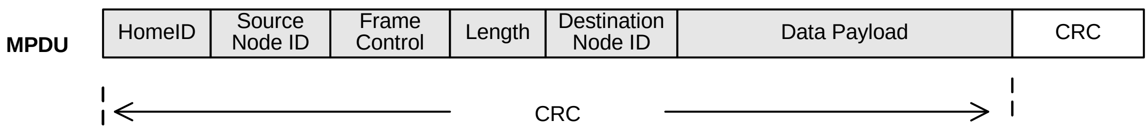

Figure A.26 – CRC coverage

The CRC-16 generator polynomial is:

$$P(x) = x^{16}+x^{12}+x^{5}+1$$, also known as CRC-CCITT

The CRC 16 is calculated over the whole frame, except for the preamble, SOF, and the CRC-16 field.

The CRC-16 generator is initialized to 1D0Fh before applying the first frame byte of a frame and no bits are appended the frame data.

A.4.2.10 Beam frame format

Beam frames are used to get the attention of battery-powered nodes with low latency. Each beam frame carries a preamble pattern and a SOF field, just like the field prepended by the PHY to the general MAC packet format. The SOF is followed by two bytes; the beam tag and the beam info. A receiving node may distinguish a general MAC packet from a beam frame by inspecting the MS byte of the HomeID. If this byte carries a beam tag (refer to Table 8-17) this is a beam frame and the following byte carries the beam info field.

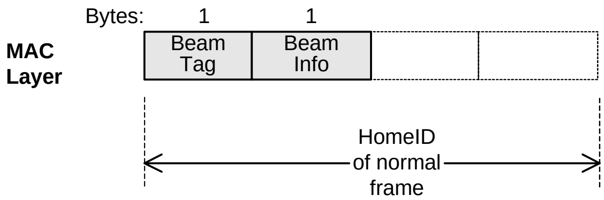

Figure A.27 – Beam frame

The preamble pattern length depends on the actual RF profiles used. Refer to Table A.1. A node shall listen long enough to detect a beam frame during worst case conditions around the SOF, beam tag and beam info. In a worst case situation, an FLN starts listening just shortly before the preamble pattern ends. Therefore the FLN has to wait for preamble detection during the SOF, beam tag and beam info fields before another preamble pattern starts. Only after achieving locking to the preamble, the SOF and beam tag fields may be correctly decoded.

Beam frames are used for fragmented beams as well as for repeated beams. In both cases a high number of beam frames are transmitted back to back. In case of repeated beams, a singlecast frame follows the beam.

Each beam frame may carry one byte value. The beam tag value indicates the type of data carried in the beam info byte.

Table A.44 – Beam info data type

Beam tag value	Beam info data type	Description
0x55	NodeID (values 1..232, 255)	High priority field. Shall be transmitted in every second beam frame. May be transmitted in every beam frame.
0x54	Low priority beam info	Reserved for future use.

The beam tag value 0x54 allows for future extensions.

If a battery-operated node wakes up and detects a beam frame, the node shall evaluate the encapsulated information in order to determine if it can return to sleep, or if it has to stay awake.

A.4.2.11 Fragmented beam format

A fragmented beam comprises a number of beam fragments. A beam fragment comprises a number of beam frames spanning a period of 100 ms. The fragment duration shall be as close to 100 ms as possible without exceeding 100 ms. Beam frames shall be sent back to back to ensure that the beam fragment can be detected by a node waking up at any moment during the duration of the beam fragment.

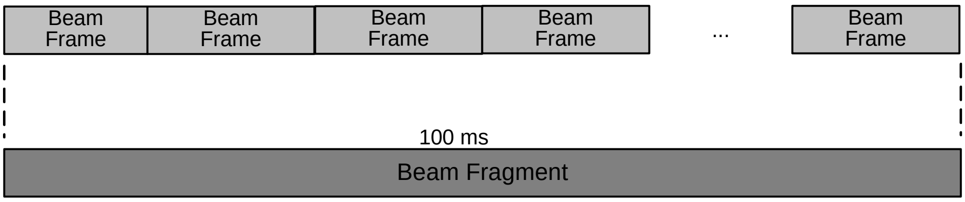

Figure A.28 – Format of the beam fragment

A fragmented beam is constructed of 100 ms beam fragments separated by 100 ms pauses. The duty cycle complies with regional requirements for the 900 MHz ISM band. A receiving node may acknowledge the reception of a beam fragment. Fragmented beams shall be sent using the R3 data rate. A receiver shall be able to monitor multiple channels for beam frames.

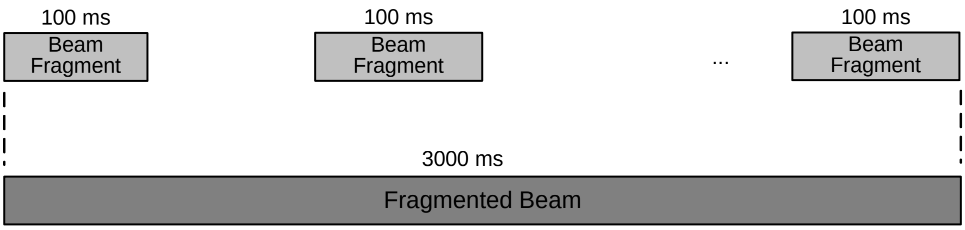

Figure A.29 – A fragmented beam

Proper tx scheduling allows a sending node to reach sleeping nodes using a range of wakeup intervals 100 ms, 150 ms, 250 ms, 500 ms, 700 ms, 900 ms. The chosen wakeup interval is a compromise between battery lifetime and response time.

A fragmented beam may address any NodeID; also the broadcast NodeID 0xFF. A full fragmented beam spans 3000 ms. A Z-Wave singlecast frame follows the fragmented beam. A receiving node may interrupt the transmission of a fragmented beam by acknowledging a beam fragment. The originating node shall then transmit the Z-Wave frame to the receiving node if the source HomeID matches the HomeID of the beaming node. A receiving node shall not acknowledge a fragmented beam targeting the broadcast address. A receiving node shall apply a random delay of 0..50 ms to the ack message to prevent collisions with identical NodeIDs in other networks.

A.4.2.12 Repeated beam format

A repeated beam is an up to 3000 ms long series of beams. Each beam is a 75 ms long beam fragment followed by a singlecast Z-Wave frame. A new beam is transmitted every 150 ms; measured from the start of the previous beam. If the frame is acknowledged, the repeated beam transmission is terminated.

Figure A.30 – A single beam

A repeated beam may address any NodeID, including the broadcast NodeID 0xFF.

A receiving node shall compare the NodeID in the beam frame with its own NodeID and if there is a match the receiving node shall configure its radio to receive singlecast frames.

A.4.2.13 Continuous beam format

A continuous beam is a series of beam frames spanning a fixed period of time. The beam frames shall be sent back to back to prevent other transmitters from interrupting the continuous beam. The continuous beam shall be sent using the R2 data rate.

A continuous beam may address any NodeID; also the broadcast NodeID 0xFF.

There are two types of continuous beams: long and short. A long continuous beam may last for a maximum 1160 ms, and a short one may last for maximum 300 ms. The recommended duration is 1100 ms for a long continuous beam and 275 ms for a short continuous beam. A continuous beam is always followed by a singlecast frame.

A receiving node shall match the NodeID in the beam frame with its own NodeID and if there is a match the receiving node shall configure its radio to receive singlecast frames.

A.4.2.14 Beam formats on channel configurations 1 and 2

An FL node operating in channel configuration 1 or 2 shall:

- be able to receive the continuous beam format and the repeated beam format on R2;
- be able to transmit either the repeated beam format or the continuous beam format on R2. It is recommended that the repeated beam format is transmitted.

A.4.2.15 Beam formats on channel configuration 3

FL nodes operating in channel configuration 3 shall be able to both receive and transmit the fragmented beam format.

A.4.3 MAC constants and MIB attributes

This clause specifies the constants and attributes required by the MAC.

A.4.3.1 MAC constants

The constants that define the characteristics of the MAC are presented in Table A.45. These constants are hardware dependent and cannot be changed during operation.

Table A.45 – MAC constants in general

Constants	Description	Value
aMacMaxMSDUSizeR1	The maximum size, in bytes, of a singlecast/broadcast payload for data rate R1.	54 bytes
aMacMaxMSDUSizeMultiR1	The maximum size, in bytes, of a multicast payload for data rate R1.	25 bytes
aMacMaxMSDUSizeR2	The maximum size, in bytes, of a singlecast/broadcast payload for data rate R2.	54 bytes
aMacMaxMSDUSizeMultiR2	The maximum size, in bytes, of a payload for multicast data rate R2.	25 bytes
aMacMaxMSDUSizeR3	The maximum size, in bytes, of a singlecast/broadcast payload for data rate R3.	158 bytes
aMacMaxMSDUSizeMultiR3	The maximum size, in bytes, of a multicast payload for data rate R3.	129 bytes
aMacMinAckWaitDuration	The minimum Ack wait duration.	aTurnaroundTimeRxTx + (aMacTransferAckTimeTX*(1/data rate))
aMacMaxFrameRetries	The number of retries after a transmission failure.	2
aMacMinCCARetryDuration	The minimum duration of clear channel access assessment.	1100 ms
aMacTxPhyPowerLevelLow	The constant value used by the MAC layer to configure the PHY MIB parameter phyTransmitPower prior to a low-power transmission.	Implementation dependent.
aMacTxPhyPowerLevelNormal	The constant value used by the MAC layer to configure the PHY MIB parameter phyTransmitPower prior to a normal power transmission.	Implementation dependent.

The term *aMacMaxMSDUSizeX* is used in text that applies to all data rates (*aMacMaxMSDUSizeR1, aMacMaxMSDUSizeR2, aMacMaxMSDUSizeR3, aMacMaxMSDUSizeMultiR1, aMacMaxMSDUSizeMultiR2, aMacMaxMSDUSizeMultiR3)*.

Retransmitted frames may cause prolonged transmission times as the retransmission uses longer preambles in certain transmission modes.

Table A.46 outlines the constant values to use for individual transmissions. Three transmission events are defined: TX1, TX2 and TX3. TX1 is the original transmission while TX2 and TX3 are the optional first and second retransmissions. Only TX1 is defined for acknowledgment timing constants

Table A.46 – MAC constants for message transfer

Constants	Description	Value, R1	Value, R2	Value, R3; CC 1 and 2	Value, R3; CC 3
aMacTransferAckTimeTX (ch)	The number of symbols it takes to transmit an acknowledgment frame including preamble.	168 bits	248 bits	416 bits	296 bits
aMacTypicalFrameLengthTX (ch)	The number of symbols of a singlecast frame with a data payload of 4 bytes.	200 bits	280 bits	448 bits	328 bits
aMacMinRetransmitDelay	Random backoff shall be higher than this value.	10 ms			
aMacMaxRetransmitDelay	Random backoff shall be lower than this value.	40 ms			

The MAC determines if the PHY is configured for channel configurations 1, 2 or channel configuration 3 operation by evaluating the RF profile mappings configured via the *macMapPhyChannel** MAC MIB parameters and the RF profile table defined for the PHY.

A.4.3.2 MAC MIB attributes

The MAC management information base (MAC MIB) comprises the attributes required to manage the MAC of a node. Each of these attributes can be read or written using the MLME-GET.request and MLME-SET.request primitives, respectively. The attributes contained in the MAC MIB are presented in Table A.47.

Table A.47 – MAC MIB attributes

Attribute	Type	Range	Description	Default
macCCARetryDuration	Integer	*aMacMinCCARetryDuration – infinite*	The duration of clear channel access assessment.	~ 1100 ms
macHomeID	Set of bytes	0x00000000 – 0xFFFFFFFF	The HomeID is the unique network identifier.	(random)

Table A.47 – MAC MIB attributes

Attribute	Type	Range	Description	Default
macNodeID	Byte	0x00– 0xFF	The NodeID is the address of the individual nodes in a network; they are only unique within a network defined by a unique HomeID.	0x00
macPromiscuousMode	Boolean	TRUE or FALSE	This indicates whether the MAC sublayer is in a promiscuous (receive all) mode. A value of TRUE indicates that the MAC sublayer accepts all frames received from the PHY.	FALSE
macRxOnWhenIdle	Boolean	TRUE or FALSE	This indicates whether the MAC sublayer is to enable its receiver at any time.	FALSE
phyCurrentTxChannel	Enumeration	A, B, C	The TX channel to use (see Table A.1).	A
macMapPhyChannelA	Integer	Available RF profiles (Table A.1)	Apply RF profile to channel A. Parameter maps directly to the PHY MIB parameter *phyMapChannelA*	–
macMapPhyChannelB	Integer	Available RF profiles (Table A.1)	Apply RF profile to channel B. Parameter maps directly to the PHY MIB parameter *phyMapChannelB*	–
macMapPhyChannelC	Integer	Available RF profiles (Table A.1)	Apply RF profile to channel C. Parameter maps directly to the PHY MIB parameter *phyMapChannelC*	–

A.4.4 MAC functional description

This clause provides a detailed description of the MAC functionality. Throughout this clause, the receipt of a frame is defined as the successful receipt of an incoming frame from the RF media to the PHY and the successful verification of the FCS by the MAC sublayer, as described in clause A.4.2.8.

A.4.4.1 Transmission, reception and acknowledgment

This clause describes the fundamental procedures for transmission, reception, and acknowledgment.

A.4.4.1.1 Clear channel access

The MAC layer shall request the channel status from the PHY layer.

A PLME-GET-CCA.request message is used to evaluate the channel. A PLME-GET-CCA.confirm message returns the current channel status.

If the MAC sublayer finds the channel busy for a period of *macCCARetryDuration* times the frame transmission is failed. This is indicated to the network layer via the MD-DATA.confirm primitive with a status of NO_CCA (see clause A.4.1.1.2).

A.4.4.1.2 Transmission

To avoid RF collisions the MAC sublayer performs a CCA before transmitting a frame. If the channel is found idle the frame is transmitted. The source HomeID and source NodeID field shall contain the address of the node sending the frame and the destination HomeID address field shall contain the address of the recipient. In addition the frame type and frame length is generated as a part if the MPDU.

A.4.4.1.3 Reception and rejection

Each node may choose whether the MAC sublayer is to enable its receiver during idle periods. During these idle periods, the MAC sublayer shall still service transceiver task requests from the network layer. A transceiver task shall be defined as a transmission request, a reception request, or a clear channel access detection. On completion of each transceiver task, the MAC sublayer shall request that the PHY enables or disables its receiver, depending on whether *macRxOnWhenIdle* is set to TRUE or FALSE, respectively.

Due to the nature of radiocommunications, a node with its receiver enabled will be able to receive and decode transmissions from all nodes that are operating on the same channel(s). The MAC sublayer shall, therefore, be able to filter incoming frames and present only the frames that are of interest to the upper layers.

The MAC sublayer shall discard all received frames that do not contain a correct value in their FCS field in the MFR, according to the algorithm described in clause A.4.2.8. The next level of filtering shall depend on whether the MAC sublayer is currently operating in promiscuous mode or not. In promiscuous mode, the MAC sublayer shall pass all frames directly to the network layer without applying any more filtering.

If the MAC sublayer is not in promiscuous mode (i.e., *macPromiscuousMode* is set to FALSE), it shall only accept frames (i.e., issue an MD-DATA.indication to the network layer) if the frame header contains the HomeID address and NodeID address of the receiving node. If the frame is not addressed to the receiving node the frame is discarded. The filter shall accept frames if addressed to the broadcast address or if the NodeID is included in a multicast header.

If the frame type subfield indicates a Singlecast Frame and the acknowledgment request subfield of the frame control field is set to 1, the MAC sublayer shall send an acknowledgment frame.

The MAC sublayer shall be able to receive beam fragments and forward such datagrams to higher layers.

A.4.4.1.3.1 RX filtering

A frame shall be discarded if the received frame has a wrong checksum value (applies to R1 and R2) or a wrong CRC16 value (applies to R3).

A frame shall be discarded if the received frame has a length field less than 9 or greater than the maximum size values indicated in Tables A.45 and A.46.

A.4.4.1.4 Use of acknowledgment frames

A singlecast frame may be sent with the acknowledgment request subfield of the frame control field set to 1. Any multicast or broadcast frame shall be sent with the acknowledgment request subfield set to 0.

Sequence number checking shall be applied to acknowledgment handling. Refer to clause A.4.2.3.7.

A.4.4.1.4.1 No acknowledgment

A frame transmitted with its acknowledgment request subfield set to 0 shall not be acknowledged by its intended recipient. The originating node shall assume that the transmission of the frame was successful. The message sequence chart in Figure A.31 – shows the scenario for transmitting a single frame of data from an originator to a recipient without requiring an acknowledgment.

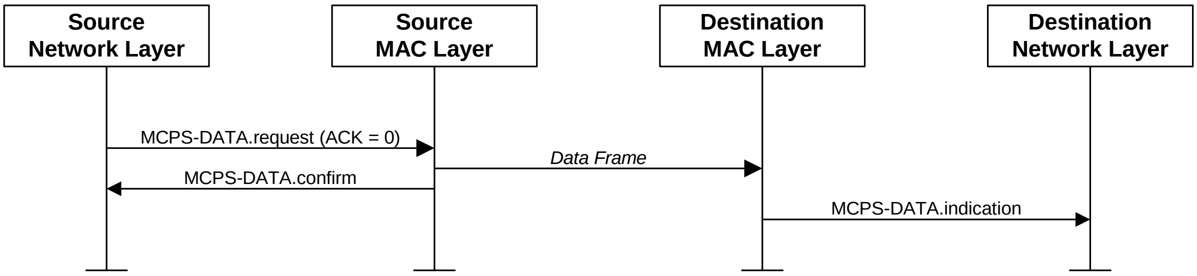

Figure A.31 – Successful data transmission without an acknowledgment

A.4.4.1.4.2 Acknowledgment

A singlecast frame transmitted with the acknowledgment request subfield of its frame control field set to 1 shall be acknowledged by the recipient. If the intended recipient correctly receives the frame, it shall return an acknowledgment frame. Only the singlecast frame may be sent with the acknowledgment request subfield set to 1. For other frame types the acknowledgment request subfield shall be ignored by the intended recipient.

The transmission of an acknowledgment frame shall commence *aTurnaroundTimeRXTX* symbols after the reception of the last symbol of the frame. Refer to clause A.3.1.14.

The message sequence chart in Figure A.32 – shows the transmission of a singlecast frame with an acknowledgment frame.

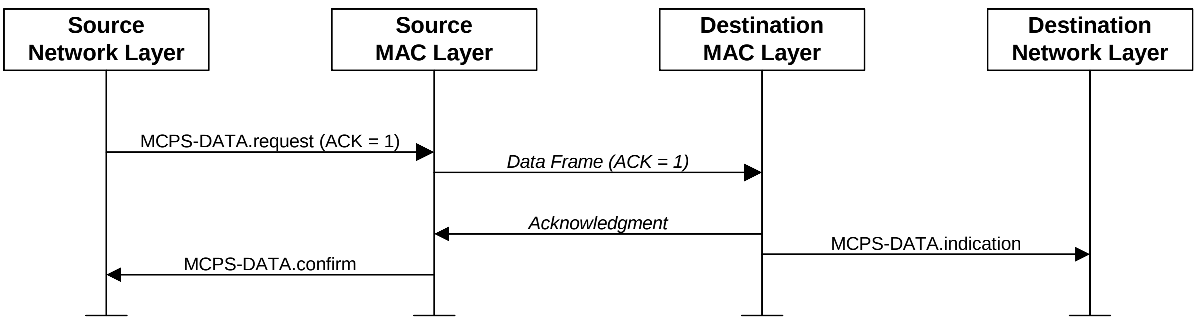

Figure A.32 – Successful data transmission with an acknowledgment frame

A.4.4.1.4.3 Retransmission

A node that sends a singlecast frame with its acknowledgment request subfield set to 1 shall wait for minimum *aMacMinAckWaitDuration* symbols for the corresponding acknowledgment frame to be received. If an acknowledgment frame is received within *aMacMinAckWaitDuration* symbols and contains the same HomeID and the correct source NodeID, the transmission is considered successful, and no further action shall be taken by the originator. If an acknowledgment frame is not received within *aMacMinAckWaitDuration* symbols the transmission attempt has failed. The originator shall repeat the process of transmitting the frame and waiting for the acknowledgment frame up to *aMacMaxFrameRetries* times. Before retransmitting the node shall wait for a random backoff period (see clause A.4.4.1.4.4).

If an acknowledgment frame is still not received after *aMacMaxFrameRetries* retransmissions, the MAC sublayer shall assume the transmission has failed and notify the network layer of the failure. This is done via the MD-DATA.confirm primitive with a status of NO_ACK (see clause A.4.1.1.2).

A.4.4.1.4.4 Random backoff

If a singlecast frame with its acknowledgment request subfield set to 1 or the corresponding acknowledgment frame is lost or corrupted, the singlecast frame shall be retransmitted. The MAC layer collision avoidance mechanism prevents nodes from retransmitting at the same time. The random delay is calculated as a period in the interval *aMacMinRetransmitDelay… aMacMaxRetransmitDelay;* Refer to Table A.46.

A.4.4.1.5 Idle mode

If the MLME is requested to set *macRxOnWhenIdle* to TRUE the PHY will enter RX mode and stay in RX mode when a frame has been transmitted (always listening). This request is achieved when the MLME issues the PLME-SET-TRX-STATE.request primitive with a state of RX_ON.

If the MLME is requested to set *macRxOnWhenIdle* to FALSE, the PHY will disable its receiver when a frame has been transmitted. This is achieved by the MLME issuing the PLME-SET-TRX-STATE.request primitive with a state of TRX_OFF.

A.4.4.2 Transmission scenarios

Due to the imperfect nature of the radio medium, a transmitted frame does not always reach its intended destination. Figures A.33 to A.35 illustrates three different data transmission scenarios:

– ***Successful data frame transmission.*** The originating MAC sublayer transmits the frame to the recipient via the PHY data service. In waiting for an acknowledgment frame, the originating MAC sublayer starts a timer that will expire after *aMacMinAckWaitDuration* symbols. The recipient MAC sublayer receives the frame, sends an acknowledgment frame back to the originator, and passes the frame to the next higher layer. The originating MAC sublayer receives the acknowledgment frame from the recipient before its timer expires and then disables and resets the timer. The data transfer is now complete, and the originating MAC sublayer issues a success confirmation to the next higher layer.

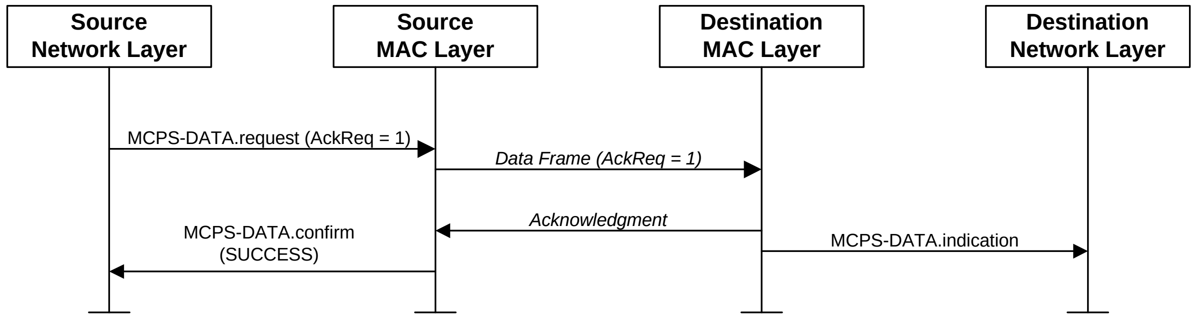

Figure A.33 – Successful frame transmission scenario

– ***Lost data frame transmission.*** The originating MAC sublayer transmits the frame to the recipient via the PHY data service. In waiting for an acknowledgment frame, the originator MAC sublayer starts a timer that will expire after *aMacMinAckWaitDuration* symbols. The recipient MAC sublayer does not receive the frame and does therefore not return an acknowledgment frame. The timer of the originator MAC sublayer expires before an acknowledgment frame is received. The data transfer has failed and the originator retransmits the frame. This sequence is repeated up to *aMacMaxFrameRetries* times. If a data transfer attempt fails a total of (1 + *aMacMaxFrameRetries*) times, the originator MAC sublayer will issue a failure confirmation to the network layer.

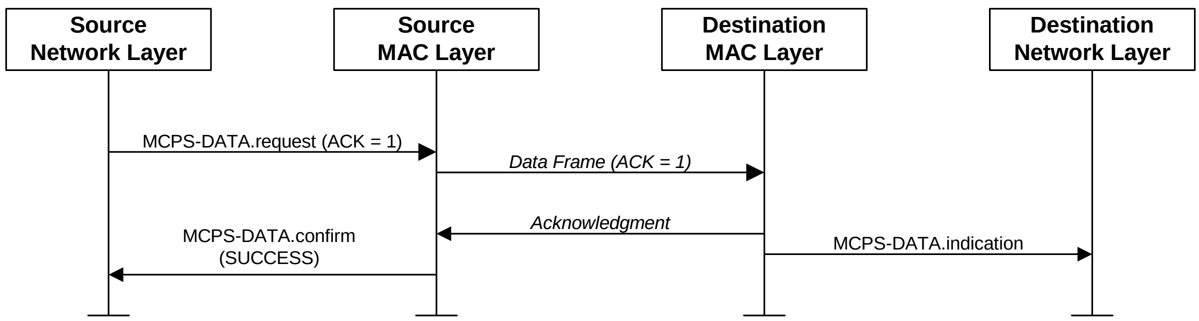

Figure A.34 – Lost frame transmission scenario

– ***Lost acknowledgment frame transmission.*** The originating MAC sublayer transmits the frame to the recipient via the PHY data service. In waiting for an acknowledgment frame, the originator MAC sublayer starts a timer that will expire after *aMacMinAckWaitDuration* symbols. The recipient MAC sublayer receives the frame, sends an acknowledgment frame back to the originator, and passes the frame to the next network layer. The originator MAC sublayer does not receive the acknowledgment frame and its timer expires. The data transfer has failed, and the originator will retransmit the data. If a data transfer attempt fails a total of (1 + *aMacMaxFrameRetries*) times, the MAC sublayer will issue a failure confirmation to the next higher layer.

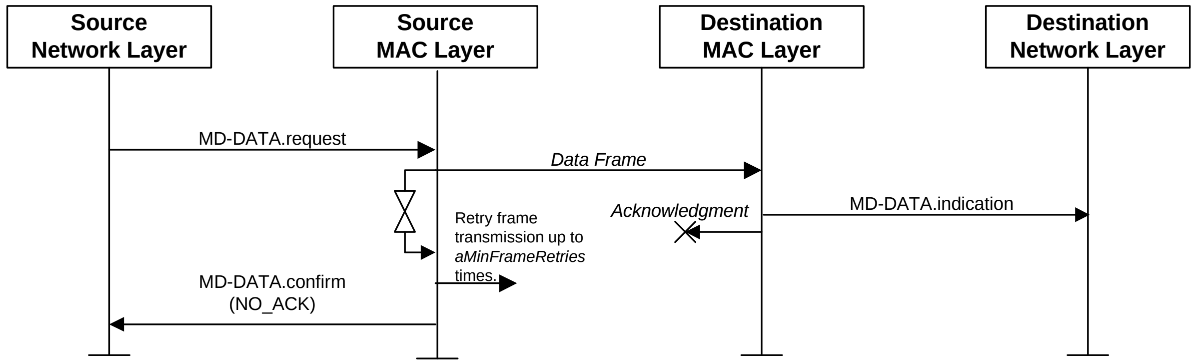

Figure A.35 – Lost acknowledgment frame transmission scenario

Bibliography

[b-ITU-T X.210] Recommendation ITU-T X.210 (1993), *Information technology – Open systems interconnection – Basic reference model: Conventions for the definitions of OSI services*.

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, 2012