IEEE P802.15

Wireless Personal Area Networks

Project	IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)
Title	IEEE Std 802.15.4e-D0.01/r5
Date Submitted	2010/02/02
Source	[Ludwig Winkel, Zafer Sahinoglu, Liang Li], and fed by the following subgroups: TSCH, FA, E-GTS, Low-Energy, Short-Header] Voice: [] E-mail: [ludwig.winkel@siemens.com,] [liang_1@yahoo.com,] [zafer@merl.com]
Re:	[]
Abstract	[1st draft IEEE Std 802.15.4e-D0.01 based on submissions of the TSCH-, FA-, E-GTS-, Low-Energy-subgroups, of TG4e. Changes r1 to r2: Integrated the DSME and Low Energy contributions Changes r2 to r3: Spelling improvements Changes r3 to r4 are based on the following contributions: 1) FA Michael Bahr 22.11.2009 n direct edit in original doc; 2) Betty Zhao 02.12.2009 2 DOC; 3) LE Wei Hong 02.12.2009 6 DOC 09_4e_828r1 + 15-10-0048-02-004e-low-energy-revision-to-604r4.doc; 4) PA Jon Simon 21.01.2010 DOC 09_4e_833r1; 5) Channel Diversity Wun-Cheol Jeong 18.12.2009 5 0850-00; Changes r4 to r5 are based on the following contributions: 1) Tim Godfrey, Metrics, 15-09-0734-03-004e-enhanced-mac-metrics.doc 2) M.Bahr, LL, 15-09-0604-05-004e-802_15_4e_D0_01_20100119_01-fa-00.zip; 3) Jonathan, TSCH improvements, 15-09-0833-00-004e-revision-to-mac-enhancement-for-tsch.doc and 15-09-0833-01-004e-revision-to-mac-enhancement-for-tschLW.doc; 4) Wei , LE, 10-0048-r2; 5) Partly as far discussed at 2010-01-21: René Struik 15-08-0848-07-004e-security-and-efficiency-enhancements-overview.doc; 6) Liang Lee DSME 15-09-0604-05-004e-802_15_4e_D0_01_20100120-Qin-Fum-Lee.doc]
Purpose	[]
Notice	This document has been prepared to assist the IEEE P802.15. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein.
Release	The contributor acknowledges and accepts that this contribution becomes the property of IEEE and may be made publicly available by P802.15.

IEEE P802.15.4e™/D0.01 Draft Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements— Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs) Amendment 1: Add MAC enhancements for industrial applications and CWPAN

Prepared by the LAN/MAN Standards Committee Working Group of the

IEEE Computer Society Committee

Abstract: First draft of 802 WG15 TG4e. Amendment to IEEE Std 802.15.4-2006
Keywords: Amendment for application domains like Process automation, Low latency, Commercial, and enhancements for Security, low energy, etc.

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

PDF: ISBN 978-0-XXXX-XXXX-X STDXXXXX
Print: ISBN 978-0-XXXX-XXXX-X STDPDXXXXX

This page is left blank intentionally.

Introduction

This introduction is not part of IEEE P802.15.4e/D0.01, Draft Standard for Information technology—Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements— Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs) Amendment 1: Add MAC enhancements for industrial applications and CWPAN.

Notice to users

Text formatted as this line means an editor note and will be removed before publication.

Laws and regulations

Users of these documents should consult all applicable laws and regulations. Compliance with the provisions of this standard does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

This document is copyrighted by the IEEE. It is made available for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making this document available for use and adoption by public authorities and private users, the IEEE does not waive any rights in copyright to this document.

Updating of IEEE documents

Users of IEEE standards should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE Standards Association web site at http://ieeexplore.ieee.org/xpl/standards.jsp, or contact the IEEE at the address listed previously.

For more information about the IEEE Standards Association or the IEEE standards development process, visit the IEEE-SA web site at http://standards.ieee.org.

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: http://standards.ieee.org/reading/ieee/updates/errata/index.html. Users are encouraged to check this URL for errata periodically.

Interpretations

Current interpretations can be accessed at the following URL: http://standards.ieee.org/reading/ieee/interp/index.html.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this draft standard was submitted to the IEEE-SA Standards Board for approval, the LAN/MAN Standards Committee Working Group had the following membership:

Bob Heile, *Chair*
Pat Kinney, *Vice Chair*

Participant1
Participant2
Participant3
Participant4
Participant5
Participant6
Participant7
Participant8
Participant9

The following members of the [individual/entity] balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

(to be supplied by IEEE)

Participant1
Participant2
Participant3
Participant4
Participant5
Participant6
Participant7
Participant8
Participant9

Contents

Draft Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements— Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs) Amendment 1: Add MAC enhancements for industrial applications and CWPAN

NOTE—The editing instructions contained in this <amendment/corrigendum> define how to merge the material contained therein into the existing base standard and its amendments to form the comprehensive standard.

The editing instructions are shown in ***bold italic***. Four editing instructions are used: change, delete, insert, and replace. ***Change*** is used to make corrections in existing text or tables. The editing instruction specifies the location of the change and describes what is being changed by using ~~strikethrough~~ (to remove old material) and underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material without disturbing the existing material. Insertions may require renumbering. If so, renumbering instructions are given in the editing instruction. ***Replace*** is used to make changes in figures or equations by removing the existing figure or equation and replacing it with a new one. Editorial notes will not be carried over into future editions because the changes will be incorporated into the base standard.

1. Overview

2. Normative references

3. Definitions

Insert in alphabetical order the following definitions.

Coordinated Sampled Listening (CSL): A low-energy mode to the MAC which allows receiving devices to periodically sample the channel(s) for incoming transmissions at low duty cycles. The receiving device and the transmitting device are coordinated to reduce transmit overhead.

CSL Period: The period in which receiving devices sample the channel(s) for incoming transmissions.

CSL Phase: The length of time between now and the next channel sample.

CSL Payload Frame: a beacon, data or command frame.

CSL Wakeup Frame: a special short frame transmitted back-to-back before the payload frame to ensure its reception by CSL receiving device.

CSL Wakeup Frame Sequence: a sequence of back-to-back wakeup frames up to the duration of the CSL Period.

CSL Rendezvous Time (RZTime): 2-octet timestamp in wakeup frame payload indicating the expected length of time in milliseconds between the end of the wakeup frame transmission and the beginning of the payload frame transmission.

CSL Channel Sample: The operation to perform ED on a channel and attempt to receive wakeup frame when energy is detected.

low latency network (LL_NW): A PAN organized as star-network with a superframe structure and using frames with a MAC header of 1 octet length (frame type b100). The PAN coordinator of a low latency network indicates the existence of such a low latency network by periodically sending beacons with a MAC header of 1 octet (frame type b100).

Receiver Initiated Transmission (RIT): An alternative low-energy mode to CSL in which receiving devices periodically broadcast data request frames and transmitting devices only transmit to a receiving device upon receiving a data request frame. RIT is suitable for the following application scenarios:

- Low data traffic rate and loose latency requirement (tens of seconds per transmission)
- Local regulations restricting the duration of continuous radio transmissions (e.g., 950MHZ band in Japan).

4. Acronyms and abbreviations

Insert in alphabetical order the following acronyms.

CM	Commercial
FA	Factory automation
LL	Low latency
LL_NW	Low Latency Network
ND	Network device
PA	Process automation
RIT	Receiver Initiated Transmission
DSME	Enhanced GTS
LE	Low Energy
CSL	Coordinated Sampled Listening
TSCH	

5. General description

5.1 Introduction

Insert before 5.2 the following text.

In addition, several behaviors are amended for

— different industrial and other application domains and

— functional improvements.

The different industrial and other application domains have quite different requirements as they are often also diametrical opposition to each other so that the resulting solutions cannot be the same (see Annex M). That is the rational for specifying more than one solution because they are more than one problem to solve. Those solutions are marked in the normative clauses with terms that are given in Annex M.

5.2 Components of the IEEE 802.15.4 WPAN

5.3 Network topologies

5.3.1 Star network formation

5.3.2 Peer-to-peer network formation

Insert before 5.4 the following subclauses.

5.3.3 LL-Star network for wireless low latency networks

5.3.3.1 General

Due to the stringent latency requirements of low latency applications, the star network becomes a topology of choice with a superframe structure that supports low latency communication between the PAN coordinator device and its sensor/actuator devices. Both to accelerate frame processing and to reduce transmission time, short MAC frames with a 1-octet MAC header (shortened frame control) are deployed.

5.3.3.2 TDMA Access

The PHY is accessed by a TDMA scheme, which is defined by a superframe of fixed length. The superframe is synchronized with a beacon transmitted periodically from the LL_NW PAN coordinator. Access within the superframe is divided into time slots. The superframe can be configured to provide the full spectrum from complete deterministic access to shared access. For deterministic access each device is assigned to a particular time slot of fixed length. Shared Group timeslots allow multiple access for a group of nodes within a duration enclosing an arbitrary number (up to the whole superframe) of dedicated time slots.

To ensure coexistence with other RF technologies in the 2.4GHz ISM band, no channel hopping is applied.

5.3.3.3 Addressing

The LL-star network supports two addressing schemes. The first addressing mode is based on the time slot assigned to a device for communication, i.e. the time slot corresponds exactly to a single device. The second mode supports the short address format.

5.3.3.4 Network Topology

The LL-sensor network requires a star topology (see Figure 1.a). Sensor/actuator devices are connected to a single PAN coordinator. The sensors send the sensor-data unidirectionally to the LL NW PAN coordinator. Actuators are configured to exchange data bidirectionally with the LL NW PAN coordinator.

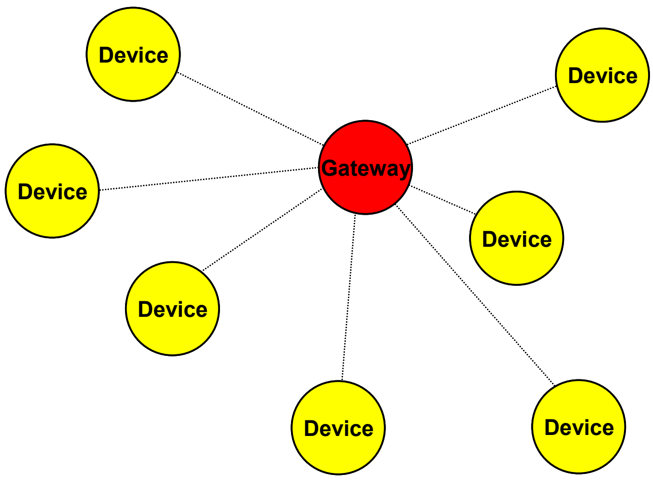


Figure 1.a—Star topology LL-MAC.

The selection of channels and time slots for communication is planned in a network management instance. The sensors and actuators are configured over the LL_NW PAN coordinator based on planning information of the network management instance.

5.4 Architecture

5.5 Functional overview

5.5.1 Superframe structure

Insert after the heading of 5.5.1 the following subclause.

5.5.1.1 General

Insert after the first sentence of 5.5.1 the following paragraph and subclauses.

There are different superframe structures possible:

- Superframe structure based on beacons of frame type Beacon as defined in 7.2.2.1. These beacons have a long MAC header.
- Superframe structure based on beacons with a 1-octet MAC header as defined in **Error! Reference source not found.**. These beacons have a short MAC header.

Comment [W1]: TBD by René.

5.5.1.2 Superframe structure based on Beacons

Insert before 5.5.2 the following subclause.

5.5.1.3 Superframe structure based on Beacons with 1-octet MAC header

If *macFAlowLatencePAN* is set to TRUE, the device is the PAN coordinator in a low latency network as described in 5.3.3.

The superframe is divided into a beacon slot and *macFAnumTimeSlots* number of time slots of equal length, see Figure 1.b.

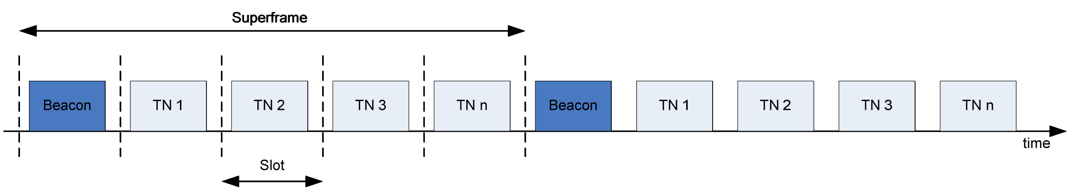


Figure 1.b—Superframe with dedicated time slots.

The first time slot of each superframe contains a beacon frame. The beacon frame is used for synchronization with the superframe structure. It is also used for re-synchronization of devices that went into power save or sleep mode.

The remaining time slots are assigned to the sensor and actuator devices in the network, so that there is no explicit addressing necessary inside the frames provided that there is exactly one device assigned to a time slot (see 7.5.1.6.6). The determination of the sender is achieved through the indexing of time slots. If there are more than one device assigned to a time slot, the time slot is referred to as shared group time slot, and a simple addressing scheme with 8-bit addresses is used as described in 7.1.1.

As shown in Figure 1.c, there is a specific order in the meaning or usage of the time slots.

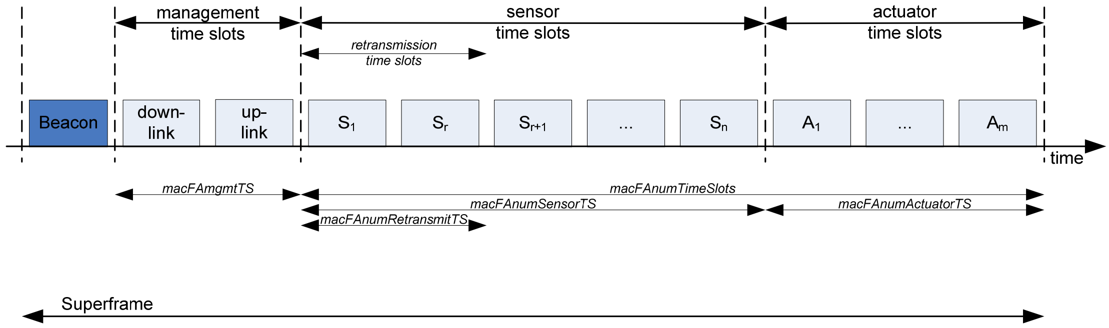


Figure 1.c—Usage and order of slots in a superframe.

- — Beacon Time Slot: always there (see 7.5.1.6.2)
- — Management Time Slots: one time slot for downlink, one time slot for uplink, existence is configurable in *macFAmgmtTS* during setup (see 7.5.1.6.3)
- — Sensor Time Slots: *macFAnumSensorTS* time slots for uplink (unidirectional communication), *macFAnumRetransmitTS* time slots at the beginning are reserved for retransmissions according to the Group Acknowledgement field contained in the beacon (see 7.5.1.6.4, **Error! Reference source not found.** and 7.5.9).
- — Actuator Time Slots: *macFAnumActuatorTS* time slots for uplink / downlink (bi-directional communication) (see 7.5.1.6.5)

Comment [W2]: TBD by René

It is also possible to use a separate Group Acknowledgement (G_{ACK}) frame (see **Error! Reference source not found.**) in order to facilitate retransmissions of the sensor transmissions within the same superframe. The use of a separate G_{ACK} is configurable during configuration mode. If the use of a separate G_{ACK} is configured, the structure of the superframe is as depicted in Figure 1.d

Comment [W3]: TBD by Ghulap

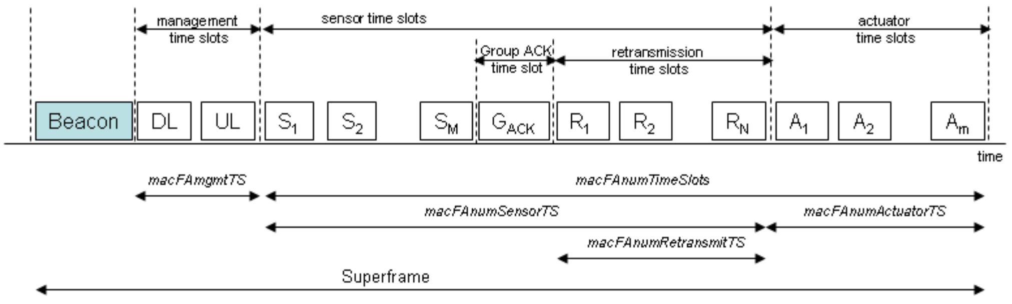


Figure 1.d—Usage and order of slots in a superframe with configured use of separate GACK

Descriptions of the configuration parameters and intervals for the superframe with a separate GACK are only different for the Sensor Time Slots:

- — Beacon Time Slot
- — Management Time Slots
- — Sensor Time Slots: macFAnumSensorTS denotes the total number of time slots available for sensors for uplink (unidirectional) communication. Typically, one time slot is allocated to each sensor. In this case, M denotes the number of sensors. The macFAnumRetransmitTS denotes the number of time slots allocated for sensors that failed their original transmissions prior to the GACK and need to retransmit their message. N denotes the number of sensors that are allowed to retransmit. One time slot is allocated for each retransmitting sensor.
- — GACK: It contains an M-bit bitmap to indicate successful and failed sensor transmissions in the same order as the sensor transmissions (see **Error! Reference source not found.**).
- — Actuator Time Slots

Comment [W4]: TBD by Ghulap)

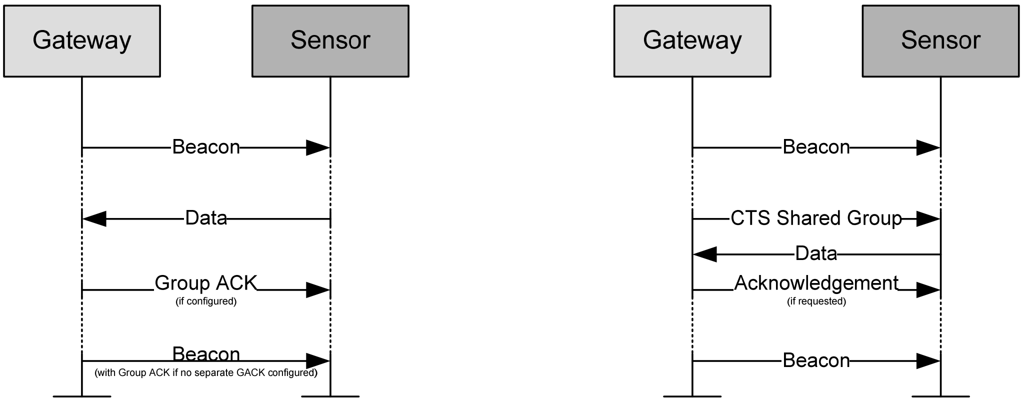
In this configuration mode, no group acknowledgment field is present in the beacon frame, because it is explicitly reported in the G_{ACK} time slot.

5.5.2 Data transfer model

5.5.2.1 Data transfer to a coordinator

Insert after Figure 6 the following paragraph and figure.

When a device wishes to transfer data to a PAN coordinator in a low latency network, it first listens for the network beacon. When the beacon is found, the device synchronizes to the superframe structure. At the appropriate time, the device transmits its data frame to the LL NW PAN coordinator. If the device transmits its data frame in a dedicated time slot or as slot owner of a shared group time slot, the data frame is transmitted without using CSMA-CA. If the device transmits its data frame in a shared group timeslot and is not the slot owner, the data frame is transmitted using slotted CSMA-CA as described in 7.5.1.6, or ALOHA, as appropriate. The LL_NW PAN coordinator may acknowledge the successful reception of the data by transmitting an optional acknowledgment frame. Successful data transmissions in dedicated time slots or by the slot owner are acknowledged by the LL_NW PAN coordinator with a Group Acknowledgement either in the next beacon or as a separate GACK frame. This sequence is summarized in Figure 6.a.



(Left: dedicated time slot. Right: shared group time slot.)

Figure 6.a—Communication to a PAN coordinator in a low latency network

5.5.2.2 Data transfer from a coordinator

Insert after Figure 8 the following paragraph and figure.

In low latency networks, a data transfer from a PAN coordinator is only possible in the macFAnumActuatorTS actuator time slots (see 5.5.1.2) and if the Actuator Direction subfield in the Flags field of the beacon indicates downlink direction (see **Error! Reference source not found.**).

Comment [W5]: TBD

When the PAN coordinator wishes to transfer data to an actuator in a low latency network, it indicates in the network beacon that the actuator direction is downlink. At the appropriate time, the LL NW PAN coordinator transmits its data frame to the device without using CSMA-CA. The device may acknowledge the successful reception of the data by transmitting an acknowledgement frame to the LL NW PAN coordinator in the same time slot of the next superframe. In order to do so, the actuator direction has to be uplink in that superframe. This sequence is summarized in Figure 8.a.

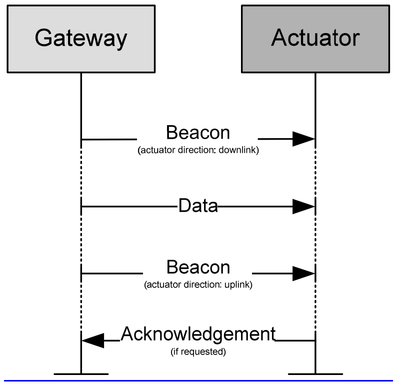


Figure 8.a—Communication from a PAN coordinator to an actuator in a low latency network

5.5.4 Improving probability of successful delivery

5.5.4.1 CSMA-CA mechanism

Insert before 5.5.4.2 the following paragraph.

Low Latency Networks use a slotted CSMA-CA channel access mechanism, where the backoff slots are aligned

— with the start of the beacon transmission in management time slots.

— with tSlotTxOwner in shared group time slots.

The backoff slots of all devices within one Low Latency Network are aligned to the PAN coordinator. Each time a device wishes to transmit data frames with CSMA-CA at the appropriate places, it locates the boundary of the next backoff slot and then waits for a random number of backoff slots. If the channel is busy, following this random backoff, the device waits for another random number of backoff slots before trying to access the channel again. If the channel is idle, the device begins transmitting on the next available backoff slot boundary. Acknowledgment and beacon frames are sent without using a CSMA-CA mechanism.

5.5.4.2 ALOHA mechanism for the UWB device

5.5.5

5.5.5.1 Additional power saving features provided by the UWB PHY

Insert before 5.5.6 the following subclause.

5.5.5.2 Low-energy mechanisms

Two low-energy mechanisms are provided to further reduce energy consumption by allowing devices to communicate while maintaining low duty cycles. They are Coordinated Sampled Listening (CSL) and Receiver Initiated Transmission (RIT).

Coordinated Sampled Listening (CSL) allows receiving devices to periodically sample the channel(s) for incoming transmissions at low duty cycles. The receiving device and the transmitting device are coordinated to reduce transmit overhead.

Receiver Initiated Transmission (RIT) allows receiving devices to periodically broadcast data request frames and transmitting devices only transmit to a receiving device upon receiving a data request frame. RIT is suitable for the following application scenarios:

- Low data traffic rate and loose latency requirement (tens of seconds per transmission)
- Local regulations restricting the duration of continuous radio transmissions (e.g., 950MHZ band in Japan).

5.5.6 Security

5.5.7

5.5.8

5.6 Concept of primitives

6. PHY specification

6.1

7. MAC sublayer specification

7.1 MAC sublayer service specification

7.1.1 MAC data service

7.1.1.1 MCPS-DATA.request

Insert before 7.1.1.1.1 the following sentence.

For TSCH, the following requirement applies in addition:

These addresses shall be specified in any of the destination addresses in DstAddr and additionalDstAddr.

7.1.1.1.1 Semantics of the service primitive

Insert after the heading of 7.1.1.1.1 the following subclause.

7.1.1.1.1.1 General

Insert before 7.1.1.1.2 the following paragraph and subclause.

The semantics of the MCPS-DATA.confirm primitive for TSCH shall have additional parameter numberOfAdditionalDstAddr and additionalDstAddr compared to 7.1.1.1.1.1, see 7.1.1.1.1.2.

7.1.1.1.1.2 TSCH-Semantics of the service primitive

The semantics of the MCPS-DATA.request primitive are as follow:

```
MCPS-DATA.request          (
          SrcAddrMode,
          DstAddrMode,
          DstPANId,
          DstAddr,
          msduLength,
          msdu,
          msduHandle,
          TxOptions,
          SecurityLevel,
          KeyIdMode,
          KeySource,
          KeyIndex,
          numberOfAdditionalDstAddr
          additionalDstAddr,
                )
```

Table 41.a specifies parameters for the MCPS-DATA.request primitive.

Table 41.a—MCPS-DATA.request parameters

Name	Type	Valid Range	Description
SrcAddrMode	Table 41	Table 41	The source addressing mode for this primitive and subsequent MPDU. This value can take one of the following values: 0x00 = no address (addressing fields omitted, see 7.2.1.1.8). 0x01 = 8-bit short address. 0x02 = 16-bit short address. 0x03 = 64-bit extended address.
DstAddrMode	Table 41	Table 41	The destination addressing mode for this primitive and subsequent MPDU. This value can take one of the following values: 0x00 = no address (addressing fields omitted, see 7.2.1.1.6). 0x01 = 8-bit short address. 0x02 = 16-bit short address. 0x03 = 64-bit extended address.
DstPANId	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
DstAddr	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
msduLength	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
msdu	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
msduHandle	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
TxOptions	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
SecurityLevel	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
KeyIdMode	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
KeySource	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
KeyIndex	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
numberOfAdditionalDstAddr	Integer	0x00-0x04	If the number of additionalDstAddr is zero, no additionalDstAddr will follow.
additionalDstAddr	List of DstAddr	0x0000-0xffff for each DstAddr	One or more alternate destination addresses. The data SPDU should be transferred to the destination specified by either DstAddr or any of destinations in additionalDstAddr.

7.1.1.1.2 When generated

Insert before 7.1.1.1.3 the following paragraph.

For TSCH, the following requirement applies in addition:

These addresses shall be specified in any of the destination addresses in DstAddr and additionalDstAddr.

7.1.1.1.3 Effect on receipt

Insert before 7.1.1.2 the following paragraph.

For TSCH, the following requirement applies in addition:

If numberOfAdditionalDstAddr is not zero and the transmission to the first transfer attempt to DestAddr fails, then MAC should transfer the data SPDU to any of the alternate destinations specified in additionalDstAddr. MAC selects a link to transmit (or retransmit if ACK is not received) in earliest possible opportunity.

7.1.1.2 MCPS-DATA.confirm

7.1.1.2.1 Semantics of the service primitive

Insert after the heading of 7.1.1.2.1 the following subclause header.

7.1.1.2.1.1 General

Insert before 7.1.1.2.2 the following paragraph and subclause.

The semantics of the MCPS-DATA.confirm primitive for TSCH shall have the parameter according to 7.1.1.2.1.2.

7.1.1.2.1.2 TSCH-Semantics of the service primitive

The TSCH-Semantics of the service primitive is as follows:

```
MCPS-DATA.confirm                                    (
          msduHandle,
          status,
          Timestamp,
          DstAddr,
          )
```

Table 42.a specifies parameters for the MCPS-DATA.request primitive.

Table 42.a—MCPS-DATA.confirm parameters

Name	Type	Valid Range	Description
msduHandle	Table 78	Table 78	See Table 78.
status	Table 78	Table 78	See Table 78.
Timestamp	Table 78	Table 78	See Table 78.
DstAddr	Device address	As specified by the DstAddrMode parameter of MCPS-DATA.request	Destination address to which the data SPDU was transferred.

7.1.1.2.2 When generated

7.1.1.2.3 Appropriate usage

Insert before 7.1.1.3 the following paragraph.

For TSCH, the following requirement applies in addition:
If the transmission attempt was successful, DstAddr is set to the address of the destination to which the data SPDU was transferred.

7.1.1.3 MCPS-DATA.indication

7.1.2 MAC management service

Insert after the heading of 7.1.2 the following subclause.

7.1.2.1 General

Insert before 7.1.3 the following subclauses.

7.1.2.2 TSCH-MAC management service

For TSCH the MAC management services shown in Table 46.a are mandatory. The primitives are discussed in the subclauses referenced in the table.

Table 46.a—Summary of the primitives accessed through the MLME-SAP for TSCH

Name	Request	Indication	Response	Confirm
MLME-SET-SLOTFRAME	7.1.18.1.1	—	—	7.1.18.1.2
MLME-SET-LINK	7.1.18.2.1	—	—	7.1.18.2.2
MLME-TSCH-MODE	7.1.18.3.1	—	—	7.1.18.3.2
MLME-LISTEN	7.1.18.4.1	—	—	7.1.18.4.2
MLME-ADVERTISE	7.1.18.5.1	7.1.18.5.2	—	7.1.18.5.3
MLME-KEEP-ALIVE	7.1.18.6.1	—	—	7.1.18.6.2
MLME-JOIN	7.1.18.7.1	7.1.18.7.2	—	7.1.18.7.3
MLME-ACTIVATE	7.1.18.8.1	7.1.18.8.2	—	7.1.18.8.3
MLME-DISCONNECT	7.1.18.9.1	7.1.18.9.2	—	7.1.18.9.3

7.1.2.3 LL-MAC management service

LL-provider Comment: Other primitives might be needed to be extended for 1-octet MHR data frames.

For LL the MAC management services shown in Table 46.b are mandatory. The primitives are discussed in the subclauses referenced in the table.

Table 46.b—Summary of the primitives accessed through the MLME-SAP for LL

Name	Request	Indication	Response	Confirm
MLME-LL_NW.discovery	7.1.19.1.2	—	—	7.1.19.1.3
MLME-LL_NW.configuration	7.1.19.1.4	—	—	7.1.19.1.5
MLME- LL_NW.online	7.1.19.1.6	7.1.19.1.7	—	—

7.1.2.4 DSME-MAC management service

For the commercial (C) applications the MAC management services shown in Table 46.c are mandatory. The primitives are discussed in the subclauses referenced in the table.

Table 46.c—Summary of the primitives accessed through the MLME-SAP for DSME

Name	Request	Indication	Response	Confirm
MLME-DSME	7.1.20.1.2	7.1.20.1.4	—	7.1.20.1.3
MLME-DSME-START	7.1.20.2.1	—	—	—
MCPS-DSME-DATA	7.1.20.2.1	7.1.1.2	—	7.1.1.3
MLME-DSMEinfo	7.1.20.4.2	—	—	7.1.20.4.3
MLME-DSME-LINKSTATUSPRT	7.1.20.5.2	7.1.20.5.4	—	7.1.20.5.3
MLME-DSME-BEACON-NOTIFY	—	7.1.20.6.2	—	—
MCPS-DSME-SCAN	7.1.20.7.2	—	—	7.1.20.7.3

7.1.3 Association primitives

C: tbd – changes & additions to be provided

7.1.3.1 MLME-ASSOCIATE.request

The MLME-ASSOCIATE.request primitive allows a device to request an association with a coordinator.

7.1.3.1.1 Semantics of the service primitive

The semantics of the MLME-ASSOCIATE.request primitive are as follows:

```
MLME-ASSOCIATE.request        (
                              LogicalChannel,
                              ChannelPage,
                              CoordAddrMode,
                              CoordPANId,
                              CoordAddress,
                              CapabilityInformation,
                              SecurityLevel,
                              KeyIdMode,
                              KeySource,
                              KeyIndex,
                              LowLatencyNetworkInfo
                              )
```

Table 83 specifies the parameters for the MLME-ASSOCIATE.request primitive.

Insert at the end of Table 83 the following row.

Name	Type	Valid Range	Description
LowLatencyNetworkInfo	Object	—	Information for association specific to low latence networks. Only available if macLLenabled is set to TRUE.

7.1.3.2 MLME-ASSOCIATE.indication

The MLME-ASSOCIATE.indication primitive is used to indicate the reception of an association request command.

7.1.3.2.1 Semantics of the service primitive

The semantics of the MLME-ASSOCIATE.indication primitive are as follows:

```
MLME-ASSOCIATE.indication      (
                                DeviceAddress,
                                CapabilityInformation,
                                SecurityLevel,
                                KeyIdMode,
                                KeySource,
                                KeyIndex,
                                LowLatencyNetworkInfo
                                )
```

Table 84 specifies the parameters for the MLME-ASSOCIATE.indication primitive.

Append at the end of table 84 (MLME-ASSOCIATE.indication parameters) the following row.

Name	Type	Valid Range	Description
LowLatencyNetworkInfo	Object	—	Information for association specific to low latence networks. Only available if macLLenabled is set to TRUE.

7.1.3.3 MLME-ASSOCIATE.response

The MLME-ASSOCIATE.response primitive is used to initiate a response to an MLMEASSOCIATE.indication primitive.

7.1.3.3.1 Semantics of the service primitive

The semantics of the MLME-ASSOCIATE.response primitive are as follows:

```
MLME-ASSOCIATE.response        (
                                DeviceAddress,
                                AssocShortAddress,
                                status,
                                SecurityLevel,
                                KeyIdMode,
                                KeySource,
                                KeyIndex,
                                LowLatencyNetworkInfo
                                )
```

Table 85 specifies the parameters for the MLME-ASSOCIATE.response primitive.

Insert at the end of Table 85 the following row.

Name	Type	Valid Range	Description
LowLatencyNetworkInfo	Object	—	Information for association specific to low latence networks. Only available if macLLenabled is set to TRUE.

7.1.3.4 MLME-ASSOCIATE.confirm

The MLME-ASSOCIATE.confirm primitive is used to inform the next higher layer of the initiating device whether its request to associate was successful or unsuccessful.

7.1.3.4.1 Semantics of the service primitive

The semantics of the MLME-ASSOCIATE.confirm primitive are as follows:

```
MLME-ASSOCIATE.confirm          (
                                AssocShortAddress,
                                status,
                                SecurityLevel,
                                KeyIdMode,
                                KeySource,
                                KeyIndex,
                                LowLatencyNetworkInfo
                                )
```

Table 86 specifies the parameters for the MLME-ASSOCIATE.confirm primitive.

Insert at the end of Table 86 the following row.

Name	Type	Valid Range	Description
LowLatencyNetworkInfo	Object	—	Information for association specific to low latence networks. Only available if macLLenabled is set to TRUE.

7.1.4 Disassociation primitives

C: tbd – changes & additions to be provided

7.1.5 Beacon notification primitive

C: tbd – changes & additions to be provided

7.1.5.1.1 Semantics of the service primitive

The semantics of the MLME-BEACON-NOTIFY.indication primitive are as follows:

```
MLME-BEACON-NOTIFY.indication   (
                                BSN,
                                PANDescriptor,
                                PendAddrSpec,
                                AddrList,
```

sduLength,
sdu,
LowLatencyNetworkInfo
)

Table 90 specifies the parameters for the MLME-BEACON-NOTIFY.indication primitive.

Table 91 describes the elements of the PANDescriptor type.

Insert at the end of Table 90 the following row.

Name	Type	Valid Range	Description
LowLatencyNetworkInfo	Object	—	Information specific to low latency networks. Only available if macLLenabled is set to TRUE.

7.1.7 GTS management primitives

7.1.7.1 MLME-GTS.request

7.1.14 MLME-START

7.1.16 Primitives for requesting data from a coordinator

7.1.17 MAC enumeration description

Insert the following row at the bottom of Table 78.

Enumeration	Value	Description
FCS_ERROR	TBD	The received data frame contains incorrect value in the FCS field in the MFR.

Insert before the heading of 7.2 the following subclauses.

In a consolidated/integrated new edition the new subclauses 7.18 ... 7.21 should be moved before 7.17.

7.1.18 TSCH-specific MAC sublayer service specification

7.1.18.1 MLME-SET-SLOTFRAME

7.1.18.1.1 MLME-SET-SLOTFRAME.request

7.1.18.1.1.1 General

The MLME-SET-SLOTFRAME.request primitive is used to add, delete, or change a slotframe at the MAC layer.

7.1.18.1.1.2 Semantics

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

```
MLME-SET-SLOTFRAME.request          (
          slotframeId,
          operation,
          size,
          channelPage,
          channelMap,
          activeFlag
                            )
```

Table 78.a specifies parameters for the MLME-SET-SLOTFRAME.request primitive.

Table 78.a—MLME-SET-SLOTFRAME.request parameters

Name	Type	Valid Range	Description
slotframeId	Integer	0x00-0xff	Unique identifier of the slotframe.
operation	Enumeration	ADD DELETE MODIFY	Operation to perform on the slotframe.
size	Integer	0x0000-0xffff	Number of timeslots in the new slotframe
channelPage	Integer	Selected from the available channel pages supported by the PHY (see 6.1.2)	Channel page supported by PHY.
channelMap	Bitmap	Array of bits	Indicating which frequency channels in the channel page are to be used for channel hopping. 27-bit bit field for Channel Page 0, 1, and 2
activeFlag	Enumeration	TRUE FALSE	Slotframe is active. Slotframe is not active.

7.1.18.1.1.3 When generated

An MLME-SET-SLOTFRAME.request is generated by the device management layer and issued to the MLME to create, delete, or update a slotframe on the MAC layer.

7.1.18.1.1.4 Effect on receipt

On receipt of an MLME-SET-SLOTFRAME.request, the MLME shall verify the parameters passed with the primitive. If the requested operation is ADD, the MLME shall attempt to add an entry into the macSlotframeTable. If the operation is MODIFY, it shall attempt to update an existing slotframe record in the table. If the operation is DELETE, all parameters except slotframeId and operation shall be ignored, and the slotframe record must be deleted from the macSlotFrameTable. If there are links in the slotframe that is being deleted, the links shall be deleted from the MAC layer. If the device is in the middle of using a link in the slotframe that is being updated or deleted, the update should be postponed until after the link operation completes.

7.1.18.1.2 MLME-SET-SLOTFRAME.confirm

7.1.18.1.2.1 General

The MLME-SET-SLOTFRAME.confirm primitive reports the results of the MLME-SET-SLOTFRAME.request command.

7.1.18.1.2.2 Semantics

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

```
MLME-SET-SLOTFRAME.confirm          (
          slotframeId,
          operation,
          status
                                    )
```

Table 78.b specifies parameters for the MLME-SET-SLOTFRAME.confirm primitive.

Table 78.b—MLME-SET-SLOTFRAME.confirm parameters

Name	Type	Valid Range	Description
slotframeId	Integer	0x00-0xff	Unique identifier of the slotframe to be added, modified, or deleted.
operation	Enumeration	ADD DELETE MODIFY	Operation to perform on the slotframe.
status	Enumeration	SUCCESS INVALID_PARAMETER SLOTFRAME_NOT_FOUND MAX_SLOTFRAMES_EXCEEDE D	Results of the MLME-SET-SLOTFRAME.request command.

7.1.18.1.2.3 When generated

The MLME-SET-SLOTFRAME.confirm primitive is generated by the MLME when the MLME-SET-SLOTFRAME.request is completed.

If any of the arguments fail a range check, the status shall be INVALID_PARAMETER. If a new slotframe is being added and the macSlotFrameTable is already full, the status shall be MAX_SLOTFRAMES_EXCEEDED. If an update or deletion is being requested and the corresponding slotframe cannot be found, the status shall be SLOTFRAME_NOT_FOUND. If an add is being requested with a slotframeID corresponding to an existing slotframe, the status shall be INVALID PARAMETER.

7.1.18.1.2.4 Effect on receipt

On receipt of a MLME-SET-SLOTFRAME.confirm primitive, the device management application is notified of the status of its corresponding MLME-SET-SLOTFRAME.request.

7.1.18.2 MLME-SET-LINK

7.1.18.2.1 MLME-SET-LINK.request

7.1.18.2.1.1 General

The MLME-SET-LINK.request primitive requests to add a new link, modify or delete an existing link at the MAC layer. The operationType parameter indicates whether the MLME-SET-LINK operation is to add or to delete a link.

7.1.18.2.1.2 Semantics

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

```
MLME-SET-LINK.request    to add a link (
          operationType (ADD_LINK or MODIFY_LINK),
          linkHandle,
          slotframeId,
          timeslot,
          chanOffset,
          linkOptions,
          linkType,
          nodeAddr
                              )
MLME-SET-LINK.request    to delete a link (
          operationType (DELETE_LINK),
          linkHandle,
                              )
```

Table 78.c specifies parameters for the MLME-SET-LINK.request primitive with the ADD_LINK or MODIFY_LINK operationType.

Table 78.c– MLME-SET-LINK.request parameters

Name	Type	Valid Range	Description
operation	Enumeration	ADD_LINK, MODIFY_LINK, DELETE_LINK	Type of link management operation to be performed.
linkHandle	Integer	0x00–0xFF	Unique identifier (local to specified slotframe) for the link.
slotframeId	Integer	0x00-0xFF	Slotframe ID of the link to be added.
timeslot	Integer	0x0000-0xFFFF	Timeslot of the link to be added.
chanOffset	Integer	0x00-0xnn	nn = number of active channels in the channel map for the slotframe used0x01-0xFF
linkOptions	Bitmap	b000 – b111	b001 = Transmit. b010 = Receive. b100 = Shared.
linkType	Enumeration	NORMAL ADVERTISING	Type of link. Links marked advertising are to be included in the advertisement frame generated in response to a MLME-ADVERTISE.request.
nodeAddr	Integer	0x0000-0xffff	Address list of neighbor devices connected to the link. 0xffff means the broadcasting to every node.

7.1.18.2.1.3 When generated

When operationType=ADD_LINK or MODIFY_LINK:

> MLME-SET-LINK.request primitive is generated by the device management layer to add a link or to modify an existing link in a slotframe.

When operationType=DELETE_LINK:

> MLME-SET-LINK.request primitive is generated by the device management layer to delete an existing link at the MAC layer.

7.1.18.2.1.4 Effect on receipt

When operationType=ADD_LINK or MODIFY_LINK:

On receipt of the MLME-SET-LINK.request, the MAC layer shall attempt to add the indicated link to the macLinkTable and add the new neighbor to its neighbor table, if needed. Upon completion, the result of the operation must be reported through the corresponding MLME-SET-LINK.confirm primitive. The use of the Shared bit in the linkOptions bitmap indicates that if the link is also a transmit link that the device must back off according to the method described in 7.5.5. Its behavior is not defined for receive links. Resolution between the short form nodeAddr and its long form address (8 octets) may be needed for security purposes. This is determined by NHL (next higher layer).

When operationType=DELETE_LINK:

On receipt of the MLME-SET-LINK request the device shall attempt to remove the link from the macLinkTable. If the link is currently in use, the deletion shall be postponed until after the link operation completes.

7.1.18.2.2 MLME-SET-LINK.confirm

7.1.18.2.2.1 General

The SET-LINK.confirm primitive indicates the result of add, modify or delete link operation.

7.1.18.2.2.2 Semantics

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

```
MLME-SET-LINK.confirm    (
          status,
          linkHandle
                         )
```

Table 78.d specifies parameters for the MLME-SET-LINK.confirm primitive.

Table 78.d—MLME-SET-LINK.confirm parameters

Name	Type	Valid Range	Description
status	Enumeration	SUCCESS INVALID_PARAMETER UNKNOWN_SLOTFRAME MAX_LINKS_EXCEEDED MAX_NEIGHBORS_EXCEEDED	Result of the add or modify link operation.
linkHandle	Integer	0x00 – 0xFF	Unique (local to specified slotframe) identifier for the link.

7.1.18.2.2.3 When generated

The MLME-SET-LINK.confirm is generated as a result of the MLME-SET-LINK.request operation.

If any of the arguments fail a range check, the status shall be INVALID_PARAMETER. If a new slotframe is being added and the macSlotFrameTable is already full, the status shall be MAX_SLOTFRAMES_EXCEEDED. If an update or deletion is being requested and the corresponding slotframe cannot be found, the status shall be SLOTFRAME_NOT_FOUND. If an add is being requested with a slotframeID corresponding to an existing slotframe, the status shall be INVALID PARAMETER.

7.1.18.2.2.4 Effect on receipt

The layer that issued the MLME-SET-LINK.request to the MAC may process the result of the operation. The status of the primitive shall indicate SUCCESS if the operation completed successfully. Otherwise, the status indicates the cause of the failure. If the operationType=ADD_LINK of the MLME-SET-

LINK.request and the linkHandle already exists, the status of the primitive shall indicate INVALID PARAMETER.

7.1.18.3 MLME-TSCH-MODE

7.1.18.3.1 MLME-TSCH-MODE.request

The MLME-TSCH-MODE.request puts the MAC into TSCH-mode, or out of TSCH-mode.

7.1.18.3.1.1 Semantics

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

```
MLME-TSCH-MODE.request          (
        modeSwtich
        )
```

Table 78.e specifies parameters for the MLME-TSCH-MODE.request primitive.

Table 78.e—MLME-TSCH-MODE.request parameters

Name	Type	Valid Range	Description
modeSwitch	Enumeration	ON, OFF	Target mode. This mode indicates whether TSCH-mode should be started or stopped.

7.1.18.3.1.2 When generated

The MLME-TSCH-MODE.request may be generated by the higher layer after the device has received advertisements from the network and is synchronized to a network (i.e. in response to an MLME-ADVERTISE.indication).

7.1.18.3.1.3 Effect on receipt

Upon receipt of the request, the MAC shall start operating its TSCH- state machine using slotframes and links already contained in its database. To successfully complete this request the device must already be synchronized to a network. Once in TSCH-mode, non-TSCH- frames are ignored by the device until it is taken out of TSCH-mode or the MAC is reset by a higher layer.

7.1.18.3.2 MLME-TSCH-MODE.confirm

The MLME-TSCH-MODE.confirm primitive reports the result of the MLME-TSCH-MODE.request primitive.

7.1.18.3.2.1 Semantics

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

```
MLME-TSCH-MODE.confirm          (
    modeSwitch,
    status
        )
```

Table 78.f specifies parameters for the MLME-TSCH-MODE.confirm primitive.

Table 78.f—MLME-TSCH-MODE.confirm parameters

Name	Type	Valid Range	Description
modeSwitch	Enumeration	ON, OFF	Target mode. This mode indicates whether this confirmation is due to TSCH-mode ON request or OFF request.
status	Enumeration	SUCCESS NO_SYNC	

7.1.18.3.2.2 When generated

The MLME-TSCH-MODE.confirm is generated by the MAC layer to indicate completion of the corresponding request. If the corresponding request was to turn on the TSCH-MODE, but the MAC layer has not been synchronized to a network, the status shall be NO_SYNC. Otherwise, the status shall be SUCCESS.

If the corresponding request was to turn off the TSCH-MODE, the status shall be SUCCESS, and the MAC layer will stop the TSCH-MODE operation.

7.1.18.3.2.3 Effect on receipt

The higher layer may use the confirmation to process the result of MLME-TSCH-MODE.request.

7.1.18.4 MLME-LISTEN

7.1.18.4.1 MLME-LISTEN.request

7.1.18.4.1.1 Semantics

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

```
MLME-LISTEN.request      (
          time,
          numPageChannel,
          pageChannelsDes[]
                         )
```

Table 78.g specifies parameters for the MLME-LISTEN.request primitive.

Table 78.g—MLME-LISTEN.request parameters

Name	Type	Valid Range	Description
onTime	Integer	0x0000 – 0xFFFF	The amount of time (10 ms units) to stay on each channel. 0x0000 indicates that the MAC stops listening
offTime	Integer	0x0000 – 0xFFFF	The amount of time (10 ms units) to wait between channel changes.
numPageChannel	Integer	0x01-0xFF	The number of page channel descriptors in the page channels array.
pageChannelsDes[]	Table 78.h	Table 78.h	Array of page channel descriptor. See Table 78.h for the format of page channel descriptor.

Table 78.h—MLME-LISTEN.request pageChannelDesc parameters

Name	Type	Valid Range	Description
channelPageId	Integer	Selected from the available channel pages supported by the PHY (see 6.1.2)	Channel page ID.
numChannel	Integer	0x01-0xFF	The number of channels in this channel page to be included in listening.
Channels[]	Array of Channel	Table 4.	The array of channels on which to listen. See Table 4 for the valid range of channels in each channel page.

7.1.18.4.1.2 When generated

The MLME-LISTEN.request shall be generated by the next higher layer to initiate the search for a TSCH-network.

7.1.18.4.1.3 Effect on receipt

Upon receipt of the request the MAC layer shall activate the radio on the indicated channel and wait for an Advertisement command. The MAC shall listen on Channel[0] for onTime, inactivate the radio for offTime, then repeat with Channels[1], etc. After listening to the last channel in Channels[], the MAC returns to Channel[0]. Valid Advertisement command frames received in this state shall result in the generation of MLME-ADVERTISE.indication. All other frames shall be dropped. The MAC shall stay in the listening state until it receives a MLML-LISTEN.request with an onTime of 0x0000, or a MLME-TSCH-MODE.request is received. The higher layer selects the advertiser and the network before setting the slotframe, link(s), and TSCH-mode. Advertisements will continue to be received, and passed on to the higher layer until leaving the listen state.

7.1.18.4.2 MLME-LISTEN.confirm

7.1.18.4.2.1 Semantics

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

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

Table 78.i specifies parameters for the MLME-LISTEN.confirm primitive.

Table 78.i—MLME-LISTEN.confirm parameters

Name	Type	Valid Range	Description
status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.18.4.2.2 When generated

The MAC layer shall generate MLME-LISTEN.confirm when it completes the listen operation started by MLME-LISTEN.request. If any of the fields of the MLME-LISTEN.request are not valid, the status of the primitive shall indicate INVALID PARAMETER.

7.1.18.4.2.3 Effect on receipt

On receipt of the primitive, the higher layer may continue with its joining state machine.

7.1.18.5 MLME-ADVERTISE

7.1.18.5.1 MLME-ADVERTISE.request

7.1.18.5.1.1 Semantics

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

```
MLME-ADVERTISE.request          (
          advertiseInterval,
          channelPage,
          channelMap,
          hoppingSequenceId,
          timeslotTemplateId,
          securityLevel,
          joinPriority,
          numSlotframe,
          slotframes[]
                                )
```

Table 78.j specifies parameters for the MLME-ADVERTISE.request primitive.

Table 78.j—MLME-ADVERTISE.request parameters

Name	Type	Valid Range	Description
advertiseInterval	Integer	0x0000 – 0xFFFF	Interval specifying the transmission of the Advertisement command (in 10 ms units)
channelPage	Integer	Selected from the available channel pages supported by the PHY (see 6.1.2)	Channel page supported by PHY.
channelMap	Bitmap	Array of bits	Map of channels to be included in the Advertisement command.
hoppingSequenceId	Integer	0x0 – 0xF	ID of hopping sequence used.
timeslotTemplateId	Integer	0x0 – 0xF	ID of timeslot template used.
securityLevel	Enumeration	Table 136	Security level in the Advertisement command. See Table 136.
joinPriority	Integer	0x00 – 0xFF	Join priority to be indicated in the Advertisement command.
numSlotframe	Integer	0x0 – 0xF	Number of slotframes to be indicated in the Advertisement command.
Slotframes[]	See Table 78.k	See Table 78.k	See Table 78.k.

Table 78.k—MLME-ADVERTISE.request Slotframe parameters (per slotframe)

Name	Type	Valid Range	Description
slotframeId	Integer	0x00 – 0xFF	Slotframe ID.

7.1.18.5.1.2 When generated

The next higher layer requests the MAC layer to start sending Advertisement command frames using MLME-ADVERTISE.request so that new nodes can find the network and this device.

7.1.18.5.1.3 Effect on receipt

Upon receipt of the request the MAC layer shall send the Advertisement command frame on the first available TX link. Whenever the time specified in AdvertiseInterval lapses from the previous transmission of Advertisement command frame, the MAC layer shall repeat the Advertisement command frame on next TX link available. The remaining parameters specify the slotframes to be included in the Advertisement command frames. Links in the specified slotframes with an Advertising linkType are to be included in the Advertisement command.

7.1.18.5.2 ADVERTISE.indication

The MLME-ADVERTISE.indication indicates that a device received an Advertisement command frame.

7.1.18.5.2.1 Semantics

The semantics of the MLME-ADVERTISE.indication primitive is as follows:

```
MLME-ADVERTISE.indication          (
          PANId,
          timingInformation,
          channelPage,
          channelMap,
          hoppingSequenceId,
          timeslotTemplateId,
          securityLevel,
          joinPriority,
          linkQuality,
          numSlotframes,
          slotframes[]
                              )
```

Table 78.l specifies parameters for the MLME-ADVERTISE.indication primitive.

Table 78.l—MLME-ADVERTISE.indication parameters

Name	Type	Valid Range	Description
PANId	Integer	0x0000 – 0xFFFF	The PAN identifier indicated in the Advertisement command.
timingInformation			The time information (absolute slot number) of the timeslot in which the Advertisement command was received.
channelPage	Integer	Selected from the available channel pages supported by the PHY (see 6.1.2)	Channel page.
channelMap	Bitmap	Array of bits	Bit map of channels.
hoppingSequenceId	Integer	0x0 – 0xF	ID of hopping sequence used.
timeslotTemplateId	Integer	0x0 – 0xF	ID of timeslot template used.
securityLevel	Enumeration	Table 136	Security level in advertisement packet See Table 136.
joinPriority	Integer	0x00 – 0xFF	Join priority indicated in advertisement.
linkQuality	Integer	0x00 – 0xFF	Link quality indicated in the frame by the PHY layer.
numSlotframes	Integer	0x0 – 0xF	Number of slotframes indicated in the Advertisement command received.
slotframes[]	See Table 78.m	See Table 78.m	See Table 78.m. Slotframes and links are from received Advertisement command frame.

Table 78.m—MLME-ADVERTISE.indication parameters (per slotframe)

Name	Type	Valid Range	Description
slotframeId	Integer	0x00 – 0xFF	Slotframe ID.
slotframeSize	Integer	0x0000 – 0xFFFF	Slotframe size.
numLink	Integer	0x0 – 0xF	Number of links for the specified slotframe
links	Table 78.n	Table 78.n	Table 78.n for parameters (per link)

Table 78.n—MLME-ADVERTISE.indication parameters (per link)

Name	Type	Valid Range	Description
timeslot	Integer	0x0000 – 0xFFFF	Timeslot.
chanOffset	Integer	0x00 – 0xFF	Channel offset.
linkOption	Enumeration	TX RX SHARED_TX	Option of the link.
timeslot	Integer	0x0000 – 0xFFFF	Timeslot.

7.1.18.5.2.2 When generated

The MLME-ADVERTISE.indication shall be generated when an Advertisement command frame has been received by the device. Upon receiving a valid Advertisement command, the device shall be synchronized to the network and ready to enable the TSCH-MODE if requested by the higher layer.

7.1.18.5.2.3 Effect on receipt

The higher layer may wait and record more than one advertisement and then select the desired advertising device before configuring the superframe(s) and link(s) and before enabling TSCH-MODE. After joining a TSCH-network, the high layer uses the indication to collect the list of neighbors and information about neighbors.

7.1.18.5.3 MLME-ADVERTISE.confirm

7.1.18.5.3.1 Semantics

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

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

Table 78.o specifies parameters for the MLME-ADVERTISE.confirm primitive.

Table 78.o—MLME-ADVERTISE.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.18.5.3.2 When generated

The MAC layer shall generate MLME-ADVERTISE.confirm when it starts sending the Advertisement command. If the any of the fields of the MLME-ADVERTISE.request are not valid, the status of the primitive shall indicate INVALID PARAMETER.

7.1.18.5.3.3 Effect on receipt

On receipt of the primitive, the higher layer may expect that it will receive the Join command on any of the links provided in the Advertisement command.

7.1.18.6 MLME-KEEP-ALIVE

7.1.18.6.1 MLME-KEEP-ALIVE.request

7.1.18.6.1.1 Semantics

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

```
MLME-KEEP-ALIVE.request         (
        dstAddr,
        period
                                )
```

Table 78.p specifies parameters for the MLME-KEEP-ALIVE.request primitive.

Table 78.p—MLME-KEEP-ALIVE.request parameters

Name	Type	Valid Range	Description
dstAddr	Integer	0x0000 - 0xFFFF	Address of neighbor device to maintain the timing. Keepalives with dstAddr of 0xFFFF do not expect to be acknowledged.
period	Integer	0x0001 – 0xFFFF	Duration of quiet time in seconds that a Keep-Alive command frame should be sent if no traffic is present.

7.1.18.6.1.2 When generated

7.1.18.6.1.3 Effect on receipt

Upon receipt of the request, the MAC layer shall monitor the frame sent to the destination node specified in the dstAddr parameter. If no frame is sent to the destination node for any duration defined by the period parameter, the MAC shall send an empty (no MAC payload) frame to the node dstAddr. The Sequence Number subfield of the MHR of the frame shall be set to the least significant byte of the absolute slot number. Resolution between the short form dstAddr and its long form address (8 octets) may be needed for security purposes. This is determined by NHL (next higher layer).

7.1.18.6.2 MLME-KEEP-ALIVE.confirm

7.1.18.6.2.1 Semantics

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

```
MLME-KEEP-ALIVE.confirm            (
          status
               )
```

Table 78.q specifies parameters for the MLME-KEEP-ALIVE.confirm primitive.

Table 78.q—MLME-KEEP-ALIVE.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.18.6.2.2 When generated

The MAC layer shall generate MLME-KEEP-ALIVE.confirm to acknowledge that it received MLME-KEEP-ALIVE request. If the dstAddr of the MLME-KEEP-ALIVE.request is not 0xFFFF and the dstAddr does not exist in the devices neighbor table, the status of the primitive shall indicate INVALID PARAMETER.

7.1.18.6.2.3 Effect on receipt

None.

7.1.18.7 MLME-JOIN

7.1.18.7.1 MLME-JOIN.request

7.1.18.7.1.1 Semantics

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

```
MLME-JOIN.request          (
          dstAddr,
          securityInformation,
          numNeighbors,
          neighbors[]
                           )
```

Table 78.r specifies parameters for the MLME-JOIN.request primitive.

Table 78.r—MLME-JOIN.request parameters

Name	Type	Valid Range	Description
dstAddr	Integer	0x0000 - 0xFFFF	Address of neighbor device to send Join command.
securityInformation	Table 78.s	Table 78.s	See Table 78.s for the detail.
numNeighbors	Integer	0x0 – 0xF	Number of neighbors found by the joining device.
neighbors	Table 78.t	Table 78.t	Neighbor information for the number of neighbors specified in numNeighbors. See Table 78.t for the definition of a neighbor.

Table 78.s—MLME-JOIN.request securityInformation parameters

Name	Type	Valid Range	Description
TBD	TBD	TBD	The securityInformation definition will be defined with Security sub-group.

Table 78.t—MLME-JOIN.request neighbors parameters

Name	Type	Valid Range	Description
neighborId	Integer	0x0000 – 0xFFFF	16 bit address of neighbor.
RSSI	Integer	-128 to 127	Received signal strength (in dBm) of frames received from the neighbor.

7.1.18.7.1.2 When generated

Device management of a new device (or device who lost connection with the TSCH-network) will invoke this service primitive to join the TSCH-network.

7.1.18.7.1.3 Effect on receipt

Upon receipt of the request, the MAC layer shall send either a Join command frame or data frame containing a higher layer management packet requesting to join the network, using any link to the dstAddr. The content of the Join command frame will be formatted using the other parameters and the format of Join command frame is specified in 7.3.10.2.9. If a data frame with the higher layer management packet is used instead of a Join command frame, the content of the higher layer payload of the data frame containing the request to join the network is constructed using the other parameters. The explicit format of the higher layer payload is out of scope of this document. Resolution between the short form dstAddr and its long form address (8 octets) may be needed for security purposes. This is determined by NHL (next higher layer).

7.1.18.7.2 MLME-JOIN.indication

7.1.18.7.2.1 Semantics

The semantics of the MLME-JOIN.indication primitive is as follows:

```
MLME-JOIN.indication       (
          newNodeAddr,
          securityInformation,
          numNeighbors,
          neighbors[]
               )
```

Table 78.u specifies parameters for the MLME-JOIN.indication primitive.

Table 78.u—MLME-JOIN.indication parameters

Name	Type	Valid Range	Description
newNodeAddr	Array of octets	64-bit binary string	64-bit long address of new device sending the Join command.
securityInformation	Table 78.s	Table 78.s	See Table 78.s.
numNeighbors	Integer	0x0 – 0xF	Number of neighbors reported by the joining device.
Neighbors	Table 78.t	Table 78.t	Neighbor information for the number of neighbors specified in numNeighbors. See Table 78.t for the definition of a neighbor in neighbors.

7.1.18.7.2.2 When generated

MLME-JOIN.indication indicates the Device Management layer that the MAC layer has received a Join command frame from a new device attempting to join the TSCH-network.

7.1.18.7.2.3 Effect on receipt

Upon receipt of the MLME-JOIN.indication, the Device Management layer shall invoke the device management procedure to transfer the join attempt of the new device to the Device Manager.

7.1.18.7.3 MLME-JOIN.confirm

7.1.18.7.3.1 Semantics

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

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

Table 78.v specifies parameters for the MLME-JOIN.confirm primitive.

Table 78.v—MLME-JOIN.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.18.7.3.2 When generated

The MAC layer shall generate MLME-JOIN.confirm to acknowledge that it received the MLME-JOIN.request primitive. If the any of the fields of the MLME-JOIN.request are not valid, the status of the primitive shall indicate INVALID PARAMETER.

7.1.18.7.3.3 Effect on receipt

None

7.1.18.8 MLME-ACTIVATE

7.1.18.8.1 MLME-ACTIVATE.request

7.1.18.8.1.1 Semantics

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

```
MLME-ACTIVATE.request   (
          dstAddr,
          securityInformation,
          slotframes[]
              )
```

Table 78.w and Table 78.y specify parameters for the MLME-ACTIVATE.request primitive.

Table 78.w—MLME-ACTIVATE.request parameters

Name	Type	Valid Range	Description
dstAddr	Integer	0x0000 - 0xFFFF	Address of neighbor device to send Activate command.
securityInformation	Table 78.x	Table 78.x	See Table 78.x for details.
slotframes[]	Table 78.k	Table 78.k	See Table 78.k.

Table 78.x—MLME-ACTIVATE.request securityInformation parameters

Name	Type	Valid Range	Description
TBD	TBD	TBD	The securityInformation definition will be defined with the Security sub-group.

Table 78.y—MLME-ACTIVATE.request slotframe parameters (per slotframe)

Name	Type	Valid Range	Description
slotframeId	Integer	0x00 – 0xFF	Slotframe ID.
slotframeSize	Integer	0x00 – 0xFFFF	Slotframe size.
numLink	Integer	0x0 – 0xF	Number of links for the specified slotframe to be indicated in the Advertisement command.
links	Table 78.z	Table 78.z	See Table 78.z for parameters (per link)

Table 78.z—MLME-ACTIVATE.request Link parameters (per link)

Name	Type	Valid Range	Description
timeslot	Integer	0x0000 – 0xFFFF	Timeslot.
chanOffset	Integer	0x00-0xnn	nn = number of active channels in the channel map for the slotframe used.
linkOption	Enumeration	TX RX SHARED_TX	Option of the link.

7.1.18.8.1.2 When generated

An Activate command is generated by a higher layer in response to a Join command or a Join data frame.

7.1.18.8.1.3 Effect on receipt

Upon receipt of the request or a data frame containing a higher layer management packet to activate the new joining device. The MAC shall send the Activate command frame to the node using the linkHandle parameter. The content of the Activate command is formatted using the other parameters. If a data frame with a higher layer management packet is used instead of Activate command frame, the content of the higher layer payload to activate the network is constructed using the other parameters. The explicit format of the higher layer payload is out of scope of this document. Resolution between the short form dstAddr and its long form address (8 octets) may be needed for security purposes. This is determined by NHL (next higher layer).

7.1.18.8.2 MLME-ACTIVATE.indication

7.1.18.8.2.1 Semantics

The semantics of the MLME-ACTIVATE.indication primitive is as follows:

```
MLME-ACTIVATE.indication          (
          srcAddr,
          securityInformation,
          slotframes[]
      )
```

Table 78.aa specifies parameters for the MLME-ACTIVATE.indication primitive.

Table 78.aa—MLME-ACTIVATE.indication parameters

Name	Type	Valid Range	Description
srcAddr	Integer	0x0000 - 0xFFFF	Address of neighbor from whom the Activate command was received.
securityInformation	Table 78.x	Table 78.x	Table 78.x.
Slotframes[]	Table 78.bb	Table 78.bb	See Table 78.bb. Slotframes and links are from received Activate command frame.

Table 78.bb—MLME-ACTIVATE.indication slotframe parameters (per slotframe)

Name	Type	Valid Range	Description
slotframeId	Integer	0x00 – 0xFF	Slotframe ID.
slotframeSize	Integer	0x0000 – 0xFFFF	Slotframe size.
numLink	Integer	0x0 – 0xF	Number of links for the specified slotframe
links	Table 78.cc	Table 78.cc	Table 78.cc for parameters (per link).

Table 78.cc—MLME-ACTIVATE.indication Link parameters (per link)

Name	Type	Valid Range	Description
timeslot	Integer	0x0000 – 0xFFFF	Timeslot.
chanOffset	Integer	0x00 – 0xFF	Channel offset.
linkOption	Enumeration	TX RX SHARED_TX	Option of the link.

7.1.18.8.2.2 When generated

MLME-ACTIVATE.indication indicates the device management layer that the MAC layer has received an Activate command frame from the neighbor indentified in srcAddr.

7.1.18.8.2.3 Effect on receipt

Upon receipt of the MLME-ACTIVATE.indication, the device management layer shall process the securityInformation received to set up secure connections. The device management layer should use the information in the MLME-ACTIVATE.indication to add slotframes and links using the MLME-SET-SLOTFRAME.request and MLME-SET-LINK.request primitives. The device management layer should delete slotframes and links obtained from previous advertisements if not explicitly added in the activate command frame. Resolution between the short form srcAddr and its long form address (8 octets) may be needed for security purposes. This is determined by NHL (next higher layer).

7.1.18.8.3 MLME-ACTIVATE.confirm

7.1.18.8.3.1 Semantics

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

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

Table 78.dd specifies parameters for the MLME-ACTIVATE.confirm primitive.

Table 78.dd—MLME-ACTIVATE.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.18.8.3.2 When generated

The MAC layer shall generate MLME-ACTIVATE.confirm to acknowledge that it received MLME-ACTIVATE.request. If any of the fields of the MLME-ACTIVATE.request are not valid, the status of the primitive shall indicate INVALID PARAMETER.

7.1.18.8.3.3 Effect on receipt

7.1.18.9 MLME-DISASSOCIATE

7.1.18.9.1 MLME-DISASSOCIATE.request

7.1.18.9.1.1 Semantics

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

```
MLME-DISASSOCIATE.request          (
                  )
```

7.1.18.9.1.2 When generated

MLME-DISASSOCIATE.request primitive is used to initiate the graceful disassociation from TSCH-network.

7.1.18.9.1.3 Effect on receipt

Upon receipt of the request, the MAC layer shall send a disassociation notification command frame or a data frame containing a higher layer management packet to indicate that it is about to leave the TSCH-network on all unicast transmit links. The Sequence Number subfield of the MHR of the frame shall be set to the least significant byte of the absolute slot number.

After the MAC sends the disassociation notification command frame for *macDisconnectTime*, it shall release all slotframe and link resources.

7.1.18.9.2 MLME-DISASSOCIATE.indication

7.1.18.9.2.1 Semantics

The semantics of the MLME-DISASSOCIATE.indication primitive is as follows:

```
MLME-DISASSOCIATE.indication          (
          srcAddress,
              )
```

Table 78.ee specifies parameters for the MLME-DISASSOCIATE.indication primitive.

Table 78.ee—MLME-DISASSOCIATE.indication parameters

Name	Type	Valid Range	Description
srcAddr	Integer	0x0000 – 0xFFFF	16-bit short address of the neighbor node from which the DISASSOCIATE command frame was received.

7.1.18.9.2.2 When generated

MLME-DISASSOCIATE.indication indicates to the device management layer that the MAC layer has received a Disassociate command frame from a neighbor node, the address of which is indicated by srcAddress.

7.1.18.9.2.3 Effect on receipt

Upon receipt of the MLME-DISASSOCIATE.indication, the device management layer shall process the disassociation of the neighbor from which the Disassociate command frame is received. Resolution between the short form srcAddr and its long form address (8 octets) may be needed for security purposes. This is determined by NHL (next higher layer).

A device should only leave the TSCH network if it receives a disassociate command frame from either the PAN coordinator or all of its clock source neighbors. This command should only be accepted when addressed directly to the device using appropriate peer level security.

7.1.18.9.3 MLME-DISASSOCIATE.confirm

7.1.18.9.3.1 Semantics

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

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

Table 78.ff specifies parameters for the MLME-DISASSOCIATE.confirm primitive.

Table 78.ff—MLME-DISASSOCIATE.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS	

7.1.18.9.3.2 When generated

The MAC layer shall generate MLME-DISASSOCIATE.confirm to acknowledge that it received MLME-DISASSOCIATE request.

7.1.18.9.3.3 Effect on receipt

None.

7.1.19 LL-specific MAC sublayer service specification

7.1.19.1 Primitives for Superframe Configuration of low latency networks

7.1.19.1.1 General

These primitives control the different modes for the configuration and operation of the superframe in a low latency network.

7.1.19.1.2 MLME-LL_NW.discovery

7.1.19.1.2.1 General

This primitive switches the LL-network into discover mode.

7.1.19.1.2.2 Semantics of the Service Primitive

The semantics of the MLME-LL_NW.discovery primitive is as follows:

```
MLME-LL_NW.discovery (
            LowLatencyNetworkConfiguration
            )
```

Table 78.gg specifies the parameters for the MLME-LL_NW.discovery primitive.

Table 78.gg—MLME-LL_NW.discovery parameters

Name	Type	Valid Range	Description
LowLatencyNetwork Configuration	Object		Contains the necessary configuration parameters for the low latency network in discovery mode

7.1.19.1.2.3 Appropriate usage

The MLME-LLNW.discovery primitive is generated by the next higher layer of an LLNW coordinator and issued to its MLME to switch the low latency network into Discovery mode (7.5.9.2).

7.1.19.1.2.4 Effect on receipt

When the MLME of an LLNW coordinator receives the MLME-LLNW.discovery primitive, it sets the Transmission Mode subfield in the Flags field of the payload of the 1-octet MHR Beacons to the value for Discovery Mode as indicated in Table 122.b(Transmission Mode settings) and follows the procedures as defined for Discovery Mode in 7.5.9.2.

7.1.19.1.3 MLME-LL_NW.discovery_confirm

7.1.19.1.3.1 General

This primitive indicates the end of the discover mode and gives the status of the discover mode to the next higher layer.

7.1.19.1.3.2 Semantics of the Service Primitive

The semantics of the MLME-LL_NW.discovery_confirm primitive is as follows:

```
MLME-LL_NW.discovery_confirm (
                LowLatencyNetworkConfiguration
                DiscoveryModeStatus
                )
```

Table 78.hh specifies the parameters for the MLME-LL_NW.discovery_confirm primitive.

Table 78.hh—MLME-LLNW.discovery_confirm parameters

Name	Type	Valid Range	Description
LowLatencyNetwork Configuration	Object		Contains the necessary configuration parameters for the low latency network in discovery mode
DiscoveryModeStatu s	Object		Contains the collected information about discovered devices in the low latency network

7.1.19.1.3.3 When generated

The MLME-LLNW.discovery_confirm primitive is generated by the MLME of the LLNW coordinator and issued to its next higher layer to indicate the end of the discovery mode in the low latency network. It returns the collected information about the discovered devices in the low latency network to the higher layer.

7.1.19.1.3.4 Appropriate usage

When the next higher layer of an LLNW coordinator receives the MLME-LLNW.discovery_confirm primitive, the LLNW coordinator determines a configuration of the low latency network based on the information about the discovery devices received in DiscoveryModeStatus. It uses an algorithm outside the scope of this standard. The next higher layer of the LLNW coordinator then issues the MLME-LLNW.configuration primitive to its MLME.

7.1.19.1.4 MLME-LL_NW.configuration

7.1.19.1.4.1 General

This primitive switches the LL-network into configuration mode.

7.1.19.1.4.2 Semantics of the Service Primitive

The semantics of the MLME-LL_NW.configuration primitive is as follows:

```
MLME-LL_NW.configuration (
                LowLatencyNetworkConfiguration
                )
```

Table 78.ii specifies the parameters for the MLME-LL_NW.configuration primitive.

Table 78.ii—MLME-LLNW.configuration parameters

Name	Type	Valid Range	Description
LowLatencyNetwork Configuration	Object		Contains the necessary configuration parameters for the low latency network in configuration mode

7.1.19.1.4.3 Appropriate usage

The MLME-LLNW.configuration primitive is generated by the next higher layer of an LLNW coordinator and issued to its MLME to switch the low latency network into Configuration mode (7.5.9.3).

7.1.19.1.4.4 Effect on receipt

When the MLME of an LLNW coordinator receives the MLME-LLNW.configuration primitive, it sets the Transmission Mode subfield in the Flags field of the payload of the 1-octet MHR Beacons to the value for Configuration Mode as indicated in Table 122.b (Transmission Mode settings) and follows the procedures as defined for Configuration Mode in 7.5.9.3.

7.1.19.1.5 MLME-LL_NW.configuration_confirm

7.1.19.1.5.1 General

This primitive indicates the end of the configuration mode and gives the status of the configuration mode to the next higher layer.

7.1.19.1.5.2 Semantics of the Service Primitive

The semantics of the MLME-LL_NW.configuration_confirm primitive is as follows:

```
MLME-LL_NW.configuration_confirm (
                LowLatencyNetworkConfiguration
```

```
              ConfigurationModeStatus
              )
```

Table 78.jj specifies the parameters for the MLME-LL_NW.configuration_confirm primitive.

Table 78.jj—MLME-LLNW.configuration_confirm parameters

Name	Type	Valid Range	Description
LowLatencyNetwork Configuration	Object		Contains the necessary configuration parameters for the low latency network in configuration mode
ConfigurationModeS tatus	Object		Contains the return status of the configuration of the discovered devices in the low latency network

7.1.19.1.5.3 When generated

The MLME-LLNW.configuration_confirm primitive is generated by the MLME of the LLNW coordinator and issued to its next higher layer to indicate the end of the configuration mode in the low latency network. It returns the configuration status about the discovered devices in the low latency network to the higher layer.

7.1.19.1.5.4 Appropriate usage

When the next higher layer of an LLNW coordinator receives the MLME-LLNW.confirmation_confirm primitive, the next higher layer of the LLNW coordinator issues the MLME-LLNW.online, the MLME-LLNW.configuration, or the MLME-LLNW.discovery primitive to its MLME based on the value of ConfigurationModeStatus.

7.1.19.1.6 MLME-LL_NW.online

7.1.19.1.6.1 General

This primitive switches the LL-network into online mode.

7.1.19.1.6.2 Semantics of the Service Primitive

The semantics of the MLME-LL_NW.online primitive is as follows:

```
    MLME-LL_NW.online (
              )
```

Table 78.kk specifies the parameters for the MLME-LL_NW.online primitive.

Table 78.kk—MLME-LLNW.online parameters

Name	Type	Valid Range	Description
none			

7.1.19.1.6.3 Appropriate usage

The MLME-LLNW.online primitive is generated by the next higher layer of an LLNW coordinator and issued to its MLME to switch the low latency network into Online mode (7.5.9.4).

7.1.19.1.6.4 Effect on receipt

When the MLME of an LLNW coordinator receives the MLME-LLNW.online primitive, it sets the Transmission Mode subfield in the Flags field of the payload of the 1-octet MHR Beacons to the value for Online Mode as indicated in Table 122.b (Transmission Mode settings) and follows the procedures as defined for Online Mode in 7.5.9.4.

7.1.19.1.7 MLME-LL_NW.online_indication

7.1.19.1.7.1 General

This primitive indicates any problems during the online mode to the next higher layer.

7.1.19.1.7.2 Semantics of the Service Primitive

The semantics of the MLME-LL_NW.online_indication primitive is as follows:

```
MLME-LL_NW.online_indication (
                OnlineModeStatus
                )
```

Table 78.ll specifies the parameters for the MLME-LL_NW.online_indication primitive.

Table 78.ll—MLME-LLNW.online_indication parameters

Name	Type	Valid Range	Description
OnlineModeStatus	Object		Contains the statusin the low latency network including any discovered problems.

7.1.19.1.7.3 When generated

The MLME-LLNW.online_indication primitive is generated by the MLME of any LLNW device and issued to its next higher layer to indicate the status and any problems that occurred in the low latency network during the operation in online mode. It returns the indication of the problem and the additional supporting information to the higher layer.

7.1.19.1.7.4 Appropriate usage

When the next higher layer of an LLNW device receives the MLME-LLNW.online_indication primitive, the LLNW device determines appropriate countermeasures using an algorithm outside the scope of this standard.

7.1.20 DSME-specific MAC sublayer service specification

7.1.20.1 MLME-DSME

7.1.20.1.1 General

These DSME management primitives are optional and extent the MLME-SAP GTS management primitives specified in 7.1.7. A device wishing to use these DSME primitives and GTSs in general will already be tracking the beacons of its PAN coordinator.

7.1.20.1.2 MLME-DSME.request

7.1.20.1.2.1 General

The MLME-GTS.request (see 7.1.7.1) and MLME-DSME.request primitive allow a device to send a request to the PAN coordinator to allocate a new GTS slot or to the Destination device to allocate a new GTS or DSME slot. This primitive is also used to deallocate GTS or DSME.

If the value of the DSMEFlag in this primitive is set to ‘1’, the MLME-GTS.request primitive allows a Source device to send a request to a Destination device to allocate a new DSME, or to deallocate / reallocate / change an existing DSME. This primitive is also used by a Destination device to initiate an DSME deallocation, reallocation, or change (reduce or restart).

7.1.20.1.2.2 Semantics

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

```
MLME-DSME.request        (
            GTSCharacteristics,
            SecurityLevel,
            KeyIdMode,
            KeySource,
            KeyIndex,
            DSMECharacteristics,
            DSMEFlag
            )
```

Table 78.mm specifies the parameters for the MLME-DSME.request primitive.

Table 78.mm— MLME-DSME.request parameters

Name	Type	Valid Range	Description
GTSCharacteristics,	Table 94	Table 94	See Table 94
SecurityLevel,	Table 94	Table 94	See Table 94
KeyIdMode,	Table 94	Table 94	See Table 94
KeySource,	Table 94	Table 94	See Table 94
KeyIndex	Table 94	Table 94	See Table 94
DSMEFlag	Boolean	TRUE or FALSE	If this value is FALSE, the operation of this primitive is the same way defined in 7.1.7.1. If this value is TRUE, the operation of this primitive is for the DSME mode specified in 7.1.20.1.2.
DSMECharacteristics	DSMECharacteristics	See 7.3.10.2	The characteristics of the DSME request, including whether the request is for the allocation of a new DSME or the deallocation / reallocation / change of an existing GTS.

7.1.20.1.2.3 When generated

The MLME-DSME.request primitive can also be generated by the next higher layer of a Source device and issued to its MLME to request the allocation of a new DSME or to request the deallocation / reallocation / change of an existing DSME. It is also generated by the next higher layer of the Destination device and issued to its MLME to request the deallocation, reallocation, or change of an existing DSME.

7.1.20.1.2.4 Effect on receipt

If the value of the DSMEFlag in this primitive equals to ‘0’, the effect of MLME-DSME.request is the same as it is described in 7.1.7.1. If the value of the DSMEFlag is set to ‘1’, the effect of MLME-DSME.request is as follows.

On receipt of the MLME-DSME.request primitive for DSME allocation, the MLME of the Source device attempts to generate an DSME handshake command frame(see 7.3.10.2) with the Characteristics Type subfield of the DSME Characteristics field set to one (DSME allocation), the DSME Handshake Type subfield set to zero (DSME request). Then the MLME of the Source device will send it to the Destination device.

If macShortAddress is equal to 0xfffe or 0xffff, the Source device is not permitted to request an DSME allocation. In this case, the MLME issues the MLME-DSME.confirm primitive containing a status of NO_SHORT_ADDRESS.

If the SecurityLevel parameter is set to a valid value other than 0x00, indicating that security is required for this frame, the MLME will set the Security Enabled subfield of the Frame Control field to one. The MAC sublayer will perform outgoing processing on the frame based on the DstAddress, SecurityLevel, KeyIdMode, KeySource, and KeyIndex parameters, as described in 7.5.8.2.1. If any error occurs during outgoing frame processing, the MLME will discard the frame and issue the MLME-GTS.confirm primitive with the error status returned by outgoing frame processing.

If the DSME handshake request command frame cannot be sent due to the channel condition, the MLME will issue the MLME-DSME.confirm primitive with a status of CHANNEL_ACCESS_FAILURE.

If the MLME successfully transmits an DSME handshake command, the MLME will expect an acknowledgment in return. If an acknowledgment is not received, the MLME will issue the MLME-GTS.confirm primitive with a status of NO_ACK (see 7.5.6.4).

If the DSME request command frame is being sent (see 7.5.10.1), the source device will wait for at most an DSMERequestWaitingTime symbols, if no DSME handshake reply command frame from the destination device appears within this time, the MLME of the source device shall notify the next higher layer of the failure by the MLME-DSME.confirm primitive with a status of NO_DATA.

On receipt of an DSME handshake command frame indicating an DSME allocation request, the Destination device shall first check if there is available capacity in the current multi-superframe. When the Destination device determines whether capacity is available for the requested DSME, it shall generate an DSME descriptor (see 7.3.10.2) with the requested specifications and the 16-bit short address of the requesting source device.

If the MLME of the Destination device can allocate the requested DSME, it will set the DSME Slot Identifier subfield in the DSME descriptor to the multi-superframe slot at which the allocated DSME begins from, the length in the DSME descriptor to the length of the DSME and the Device short address to the address of the source device. In addition, the destination device shall issue the MLME-DSME.indication primitive with the characteristics of the allocated DSME and the DSMEFlag set to TRUE to notify the next higher layer of the newly allocated DSME. If the MLME of the Destination device cannot allocate the requested DSME, the DSME Slot Identifier shall be set to zero and the DSME length set to the largest DSME length that can currently be supported.

The Destination device shall then include the DSME descriptor in its DSME handshake command frame and broadcast it to its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field shall be set to one (DSME allocation) and the Handshake Type subfield shall be set to one (DSME reply).

On receipt of the DSME handshake command frame indicating an DSME allocation reply, the device shall process the DSME descriptor.

If the address in the Device Short Address subfield of the DSME descriptor does not correspond to macShortAddress of the device, the device updates its ABT to reflect the neighbor’s newly allocated DSME. If the newly allocated DSME is conflicting with the device’s known DSME, the device shall send an DSME handshake command frame to the origin device of the DSME handshake reply command frame. The Characteristics Type subfield of the DSME Characteristics field set to three (DSME duplicate allocation notification) and the Handshake Type subfield set to two (DSME notify), with the DSME Slot Identifier subfield in the DSME descriptor set to the multi-superframe slot at which the DSME duplicate allocated, the length in the DSME descriptor to the length of the duplicate allocated DSME and the Device short address to the address of the device for which the DSME allocation replied.

If the address in the Device Short Address subfield of the DSME descriptor corresponds to macShortAddress of the device, the MLME of the device shall then notify the next higher layer of whether the DSME allocation request was successful by the primitive MLME-GTS.confirm, with a status of SUCCESS (if the DSME Slot Identifier in the DSME descriptor was greater than zero) or DENIED(if the DSME Slot Identifier in the DSME descriptor was equal to zero or if the length did not match the requested length).

After that, the Source device shall broadcast an DSME handshake command frame to all its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field shall be set to one (DSME allocation) and the Handshake Type subfield shall be set to two (DSME notify).

On receipt of an DSME handshake command frame indicating an DSME allocation notify, the device shall process the DSME descriptor. The device updates its ABT to reflect the neighbor's newly allocated DSME. If the newly allocated DSME conflicts with the device's known DSME, the device shall send an DSME handshake command frame to the origin device of the DSME handshake notify command frame. The Characteristics Type subfield of the DSME Characteristics field shall be set to three (DSME duplicate allocation notification) and the Handshake Type subfield shall be set to two (DSME notify), with the Device short address to the address of the device which sent the DSME allocation notify.

On receipt of an DSME handshake command frame indicating an DSME duplicate allocation notification, the MLME shall notify the next higher layer of the conflicts by the MLME-GTS.indication primitive with the DSMEFlag set to TRUE, the Characteristics Type subfield set to three and the DSMECharacteristics set to the characteristics of the duplicate allocation DSME.

If the DSME request is to deallocate an existing DSME (see 7.5.7.4), on receipt of the MLME-DSME.request primitive, the MLME shall send the DSME handshake command (see 7.3.10) to the corresponding device (the Source or Destination of which the DSME to be deallocated), with the Characteristics Type subfield of the DSME Characteristics field set to zero (DSME deallocation), the Handshake Type subfield shall be set to zero (DSME request), and the DSME Length of the DSMEDescriptor subfields shall be set according to the characteristics of the DSME to deallocate.

On receipt of an DSME handshake command frame indicating an DSME deallocation request, the device shall attempt to deallocate the DSME.

If the DSME characteristics contained in the command do not match the characteristics of a known DSME, the device shall ignore the request.

If the DSME characteristics contained in the DSME request command match the characteristics of a known DSME, the MLME of the device shall deallocate the specified DSME, update its ABT and notify the next higher layer of the change by the primitive MLME-GTS.indication with the DSMEFlag set to TRUE, the DSME Characteristics parameter containing the characteristics of the deallocated DSME and the Characteristics Type subfield set to zero (DSME deallocation).

Then, the device shall broadcast an DSME handshake command to its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field of the DSME handshake command shall be set to zero (DSME deallocation), and the Handshake Type subfield shall be set to one (DSME reply).

On receipt of an DSME handshake command indicating an DSME deallocation reply, the device shall process the DSME descriptor. If the address in the Device Short Address subfield of the DSME descriptor does not correspond to macShortAddress of the device, the device updates its ABT to reflect all the neighbor's deallocated DSME. If the address in the Device Short Address subfield of the DSME descriptor corresponds to macShortAddress of the device, the MLME of the device shall then notify the next higher layer of whether the DSME deallocation request was successful by the primitive MLME-GTS.confirm primitive with a status of SUCCESS (if the length in the DSME descriptor matched the requested deallocation length) or DENIED (if the length in the DSME descriptor did not match the requested deallocation length).

Then, the device shall broadcast an DSME handshake command to all its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field shall be set to zero (DSME deallocation) and the Handshake Type subfield shall be set to two (Notify).

On receipt of an DSME handshake command indicating an DSME deallocation notify, the device shall process the DSME descriptor and update its ABT to reflect the neighbor's deallocated DSME.

If Destination device request to change an allocated DSME of the Source device and the request has been acknowledged by the Source device, the Destination device will issue the MLME-GTS.confirm primitive with a status of SUCCESS, the DSMEFlag set to TRUE, the DSMECharacteristics Type subfield of the DSMECharacteristics parameter set to 101 for DSME reduce or 110 for DSME restart accordingly, and other subfields set according to the characteristics of the DSME which the Destination device requests the Source device to change its original DSME to.

On receipt of an DSME handshake request command for DSME change from the destination device, the Source device will acknowledge the frame and immediately change the DSME. Then the MLME of the Source device will issue the MLME-GTS.indication primitive with an DSMECharacteristics parameter set according to the characteristics of the DSME which the Destination device requests the Source device to change its original DSME to.

If any parameter in the MLME-DSME.request primitive is not supported or is out of range, the MLME will issue the MLME-GTS.confirm primitive with a status of INVALID_PARAMETER.

When DSMEFlag is TRUE, two types of channel diversity, channel adaptation and channel hopping, can be employed to allocate DSME slots. Both channel diversity modes allocate DSME slots via DSME handshake command. To exchange channel and DSME slot usage between devices, channel adaptation uses DSME allocation bitmap table (ABT), while channel hopping uses DSME timeslot allocation bitmap (TAB). Detailed procedures for allocating and deallocating DSME slots in two channel diversity modes are explained in 7.5.10.1 and 7.5.10.7.

7.1.20.1.3 MLME-DSME.confirm

7.1.20.1.3.1 General

If the value of the DSMEFlag equals to ‘1’, the MLME-DSME.confirm primitive reports the results of a request to allocate a new DSME or to deallocate / reallocate / change an existing GTS.

7.1.20.1.3.2 Semantics

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

```
MLME-GTS.confirm          (
          GTSCharacteristics,
          status,
          DSMEFlag,
          DSMECharacteristics
          )
```

Table 78.nn specifies the parameters for the MLME-DSME.request primitive.

Table 78.nn— MLME-DSME.confirm parameters

Name	Type	Valid Range	Description
GTSCharacteristics,	Table 94	Table 94	See Table 94
status	Table 94	Table 94	See Table 94
DSMEFlag	Boolean	TRUE or FALSE	If this value is FALSE, the operation of this primitive is the same way defined in 7.1.7.1. If this value is TRUE, the operation of this primitive is for the DSME mode specified in 7.1.20.1.2.
DSMECharacteristics	DSMECharacteristics	See 7.3.10.2	The characteristics of the DSME confirm, including whether the confirm is for the allocation of a new DSME or the deallocation / reallocation / change of an existing GTS.

7.1.20.1.3.3 When generated

If the request to allocate, deallocate, reallocate, or change (reduce or restart) an DSME was successful, this primitive will return a status of SUCCESS with the DSMEFlag set to '1' and the DSMECharacteristics Type subfield of the DSMECharacteristics parameter set accordingly. Otherwise, the status parameter will indicate the appropriate error code. The reasons for these status values are fully described in 7.1.7.1.3 and subclauses referenced by 7.1.7.1.3.

7.1.20.1.3.4 Effect on receipt

On receipt of the MLME-GTS.confirm primitive with the value of the DSMEFlag set to '1', the next higher layer is notified of the result of its request to allocate, deallocate, or change an DSME. If the request was successful, the status parameter will indicate a successful DSME operation, and the MLME of the device will generate an DSME handshake command frame with the Handshake Type subfield set to two (DSME notify) and the information contained in the DSMECharacteristics parameter in this primitive. Otherwise, the status parameter will indicate the error.

7.1.20.1.4 MLME-DSME.indication

7.1.20.1.4.1 General

If the value of the DSMEFlag equals to '1', the MLME-DSME.indication primitive reports the results of a request to allocate a new DSME or to deallocate / reallocate / change an existing GTS.

7.1.20.1.4.2 Semantics

The semantics of the MLME-DSME.indication primitive is as follows:

```
MLME-GTS.indication        (
          DeviceAddress,
          GTSCharacteristics,
          SecurityLevel,
          KeyIdMode,
          KeySource,
          KeyIndex,
          DSMEFlag,
          DSMECharacteristics
          )
```

Table 78.nn specifies the parameters for the MLME-DSME.request primitive.

Table 78.oo— MLME-DSME.indication parameters

Name	Type	Valid Range	Description
GTSCharacteristics,	Table 94	Table 94	See Table 94
status	Table 94	Table 94	See Table 94
DSMEFlag	Boolean	TRUE or FALSE	If this value is FALSE, the operation of this primitive is the same way defined in 7.1.7.1. If this value is TRUE, the operation of this primitive is for the DSME mode specified in 7.1.20.1.2.
DSMECharacteristics	DSMECharacteristics	See 7.3.10.2	The characteristics of the DSME indication, including whether the indication is for the allocation of a new DSME or the deallocation / reallocation / change of an existing GTS.

7.1.20.1.4.3 When generated

This primitive is generated by the MLME of a Source or Destination device and issued to its next higher layer when an DSME is allocated, deallocated, reallocated, or changed following the reception of an DSME handshake command (see 7.3.10). The DSME Characteristics type subfield of the DSMECharacteristics parameter is set accordingly.

7.1.20.1.4.4 Effect on receipt

On receipt of the MLME-GTS.indication primitive with the value of the DSME flag in the DSMECharacteristics field set to ‘1’, the next higher layer is notified of the allocation, deallocation, reallocation, or change of an DSME.

7.1.20.1.5 DSME management message sequence charts

Figure 41.a and Figure 41.b illustrate the sequence of messages necessary for successful DSME management. Figure 41.a depicts the message flow for the case in which a Source device initiates the DSME allocation, deallocation, reallocation, or change. Figure 41.b depicts the message flow for the case in which a Destination device initiates the DSME deallocation, reallocation, or change.

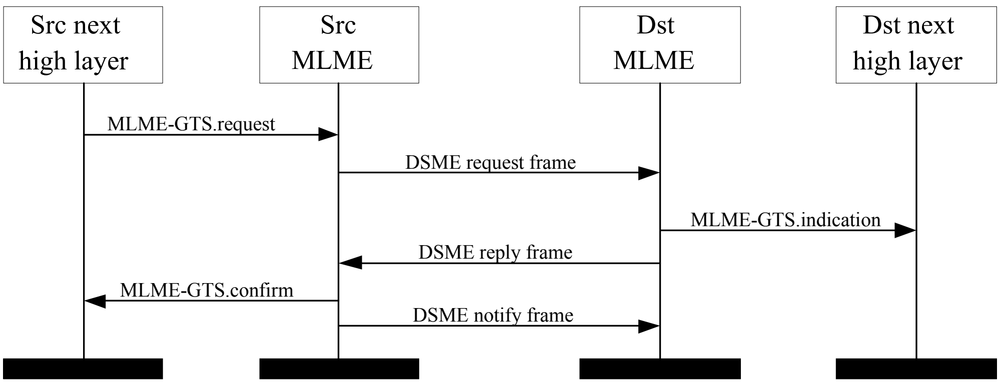


Figure 41.a—Message sequence chart for DSME allocation initiated by a Source device

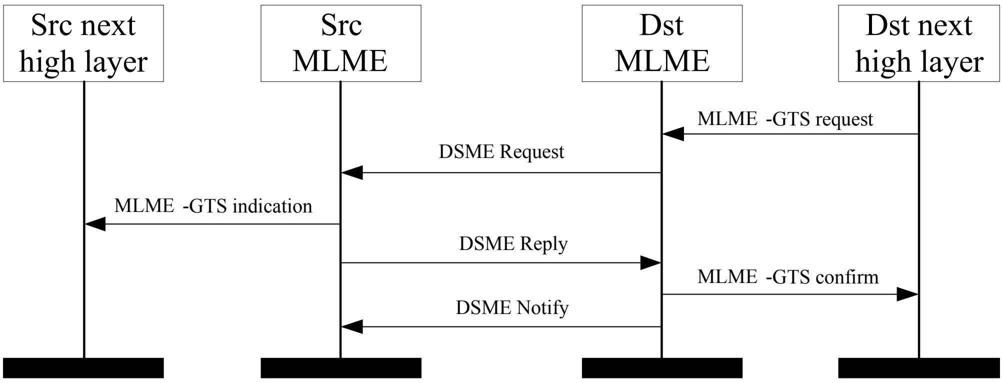


Figure 41.b—Message sequence chart for DSME allocation initiated by a Destination device

7.1.20.2 MLME-DSME-START

7.1.20.2.1 General

The MLME-DSME.request primitive (see also 7.1.7.1) allows a device to send a request to the PAN coordinator to allocate a new DSME slot or to the Destination DSME-device to allocate a new DSME slot. This primitive shall also be used to deallocate GTS or DSME.

7.1.20.2.2 MLME-DSME-START.request

7.1.20.2.2.1 General

If the value of the DSMEFlag in this primitive is set to ‘1’, the MLME-GTS.request primitive allows a Source device to send a request to a Destination device to allocate a new DSME, or to deallocate / reallocate / change an existing DSME. This primitive is also used by a Destination device to initiate an DSME deallocation, reallocation, or change (suspend, or restart).

7.1.20.2.2.2 Semantics

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

```
MLME-DSME-START.request          (
        PANId,
        LogicalChannel,
        ChannelPage,
        SuperframeStartBank,
        BeaconOrder,
        SuperframeOrder,
        PANCoordinator,
        BatteryLifeExtension,
        CoorRealignment,
        CoorRealignSecurityLevel,
        CoorRealignKeyIdMode,
        CoorRealignKeySource,
        CoorRealignKeyIndex,
```

BeaconSecurityLevel,
BeaconKeyIdMode,
BeaconKeySource,
BeaconKeyIndex,
DSMEFlag,
DSMESuperframeSpecification,
DCHDescriptor
)

Table 78.pp specifies the parameters for the MLME-DSME-START.request primitive.

Table 78.pp—MLME-DSME-START.request parameters

Name	Type	Valid Range	Description
PANId,	Table 108	Table 108	See Table 108
LogicalChannel,	Table 108	Table 108	See Table 108
ChannelPage,	Table 108	Table 108	See Table 108
SuperframeStartBank,	Integer	0x00 – 0x1F	In superframe bank , the coordinator transmits its own beacon frame. If this parameter is set to 0x00, beacon transmitting will begin immediately. Otherwise, the specified bank number is relative to the bank index of the received beacon with which the device synchronizes. This parameter shall be ignored if either the BeaconOrder parameter has a value of 15 or the PANCoordinator parameter is TRUE.
BeaconOrder,	Table 108	Table 108	See Table 108
SuperframeOrder,	Table 108	Table 108	See Table 108
PANCoordinator,	Table 108	Table 108	See Table 108
BatteryLifeExtension,	Table 108	Table 108	See Table 108
CoorRealignment,	Table 108	Table 108	See Table 108
CoorRealignSecurityLevel,	Table 108	Table 108	See Table 108
CoorRealignKeyIdMode,	Table 108	Table 108	See Table 108
CoorRealignKeySource,	Table 108	Table 108	See Table 108
CoorRealignKeyIndex,	Table 108	Table 108	See Table 108
BeaconSecurityLevel,	Table 108	Table 108	See Table 108
BeaconKeyIdMode,	Table 108	Table 108	See Table 108
BeaconKeySource,	Table 108	Table 108	See Table 108
BeaconKeyIndex,	Table 108	Table 108	See Table 108
DSMEFlag,	Boolean	TRUE or FALSE	If this value is FALSE, the operation of this primitive is the same way defined in 7.1.7.1. If this value is TRUE, the operation of this primitive is for the DSME mode specified in 7.1.20.1.2.
DSMESuperframeSpecification,	DSMESuperframeSpecification Value	See 7.2.2.1.9	The DSME superframe specification.
DCHDescriptor	DCHDescriptor Value	See Table 78.qq	The DCHDescriptor for the received beacon.

Table 78.qq—Elements of DCHDescriptor

Name	Type	Valid Range	Description
ChannelHoppingSequenceLength	Integer	0-255	Specifies the length of ChannelHoppingSequence.
ChannelHoppingSequence	Set of Octets	-	Specifies the sequence of logical channel numbers, which is set by the next higher layer. PAN coordinator shall select the sequence to use when it establishes a PAN.
ChannelOffset	Integer	0-255	Specifies the offset value of ChannelHoppingSequence.
ChannelOffsetBitmapLength	Integer	0-255	Specifies the length of ChannelOffsetBitmap.
ChannelOffsetBitmap	Set of Octets	-	Bit value of ChannelOffsetBitmap sequence represents whether the corresponding channel offset is used. If the corresponding channel offset is used, the bit value shall be set to '1'. Otherwise, it shall be set to '0'. For instance, if the 1st, 2nd, 4th channels offset are used with ChannelOffsetBitmapLength of 16, ChannelOffsetBitmap shall be 0110100000000000.

7.1.20.2.2.3 Appropriate usage

See 7.1.14.1.2.

7.1.20.2.2.4 Effect on receipt

If the SuperframeStartBank parameter is non-zero and the MLME is not currently tracking the beacon of the coordinator through which it is associated, the MLME will issue the MLME-START.confirm primitive with a status of TRACKING_OFF.

7.1.20.3 MAC DSME-data service

7.1.20.3.1 General

The MCPS-SAP supports the transport of SSCS protocol data units (SPDUs) between peer SSCS entities. Table 78.rr lists the primitives supported by the DSME-MCPS-SAP. Primitives marked with a diamond (♦) are optional for an RFD. These primitives are discussed in the subclauses referenced in the table.

Table 78.rr—MCPS-DSME-SAP primitives

Name	Request	Confirm	Indication
MCPS-DATA	7.1.1.1	7.1.1.2	7.1.1.3
MCPS-DSME-DATA	7.1.20.2.1	7.1.1.2	7.1.1.3
MCPS-DSME-PURGE	7.1.1.4♦	7.1.1.5♦	—

7.1.20.3.2 MCPS-DSME-DATA.request

Subclause 7.1.1.1 applies. Subclause 7.1.1.1.1 applies except the definition of TxOptions in Table 77 shall be replaced by Table 78.ss.

Table 78.ss—MCPS-DSME-DATA.request parameter TxOptions

Name	Type	Valid Range	Description
TxOptions	Bitmap	5-bit field	The 5 bits (b0, b1, b2, b3, b4) indicate the transmission options for this MSDU. For b0, 1 = acknowledged transmission, 0 = unacknowledged transmission. For b1, 1 = GTS transmission, 0 = CAP transmission for a beacon-enabled PAN. For a non-beacon-enabled PAN, bit b1 should always be set to 0. For b2, 1 = indirect transmission, 0 = direct transmission. For b3, 1 = CAP/DSME transmission, 0 = CAP/GTS transmission. For b4, 1 = High Priority transmission, 0 = Low Priority transmission.

7.1.20.3.2.1 Appropriate usage

Subclause 7.1.1.1.2 applies.

7.1.20.3.2.2 Effect on receipt

If the TxOptions parameter specifies that an DSME transmission is required, the MAC sublayer will set the DSME flag in the DSME Characteristics field to one, indicating the DSME transmission, and determine whether it has a valid DSME (for DSME usage rules, see 7.5.10). If a valid DSME could not be found, the MAC sublayer will issue the MCPS-DATA.confirm primitive with a status of INVALID_GTS. If a valid DSME was found, the MAC sublayer will defer, if necessary, until the DSME.

7.1.20.4 MLME-DSMEinfo

7.1.20.4.1 DSME-Primitives for requesting DSME information

MLME-DSMEinfo defines how a device can request DSME information. All DSME-devices shall provide an interface for these DSME information request primitives.

7.1.20.4.2 MLME-DSMEinfo.request

7.1.20.4.2.1 General

The MLME-DSMEinfo.request primitive allows a Source device to request the timestamp and the parameters of its DSME from the Destination device.

7.1.20.4.2.2 Semantics

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

```
MLME-DSMEinfo.request   (
          DstAddrMode,
          DstAddr,
          SecurityLevel,
          KeyIdMode,
          KeySource,
          KeyIndex
```

)

Table 78.tt specifies the parameters for the MLME-DSME-START.request primitive.

Table 78.tt—MLME-DSMEinfo.request parameters

Name	Type	Valid Range	Description
DstAddrMode	Integer	0x02–0x03	The addressing mode of the Destination device to which the request is intended. This parameter can take one of the following values: 0x02 = 16-bit short address, 0x03 = 64-bit extended address.
DstAddr	DeviceAddress	As specified by the DstAddrMode parameter	The address of the Destination device to which the request is intended.
SecurityLevel	Integer	0x00-0x07	The security level to be used (see Table 136 in 7.6.2.2.1).
KeyIdMode	Integer	0x00–0x03	The mode used to identify the key to be used (see Table 96 in 7.6.2.2.2). This parameter is ignored if the SecurityLevel parameter is set to 0x00.
KeySource	Set of 0, 4, or 8 octets	As specified by the KeyIdMode parameter	The originator of the key to be used (see 7.6.2.4.1). This parameter is ignored if the KeyIdMode parameter is ignored or set to 0x00.
KeyIndex	Integer	0x01–0xff	The index of the key to be used (see 7.6.2.4.2). This parameter is ignored if the KeyIdMode parameter is ignored or set to 0x00.

7.1.20.4.2.3 Appropriate usage

The MLME-DSMEinfo.request primitive is generated by the next higher layer of a Source device and issued to its MLME when the timestamp and the parameters of its DSME are to be requested from the Destination device.

7.1.20.4.2.4 Effect on receipt

On receipt of the MLME-DSMEinfo.request primitive by a device, the MLME of the device generates and sends an DSME information request command (see 7.3.11). The DSME information request command is generated with the destination address information in the DstAddress parameter.

If the SecurityLevel parameter is set to a valid value other than 0x00, indicating that security is required for this frame, the MLME will set the Security Enabled subfield of the Frame Control field to one. The MAC sublayer will perform outgoing processing on the frame based on the DstAddress, SecurityLevel, KeyIdMode, KeySource, and KeyIndex parameters, as described in 7.5.8.2.1. If any error occurs during outgoing frame processing, the MLME will discard the frame and issue the MLME-DSMEinfo.confirm primitive with the error status returned by outgoing frame processing.

If the DSME information request command cannot be sent due to a CSMA-CA algorithm failure, the MLME will issue the MLME-DSMEinfo.confirm primitive with a status of CHANNEL_ACCESS_FAILURE.

If the MLME successfully transmits an DSME information request command, the MLME will expect an acknowledgment in return. If an acknowledgment is not received, the MLME will issue the MLME-DSMEinfo.confirm primitive with a status of NO_ACK (see 7.5.6.4). If an acknowledgment is received, the MLME will wait for the DSME information reply command.

If an DSME information reply command is received from the Destination device, the MLME of the source device will issue the MLME-DSMEinfo.confirm primitive with a status of SUCCESS.

And if an DSME information reply command is not received within macMaxFrameTotalWaitTime CAP symbols in a beacon-enabled PAN, or symbols in a non-beacon-enabled PAN, the MLME of the source device will issue the MLME-DSMEinfo.confirm primitive with a status of NO_DATA.

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

7.1.20.4.3 MLME-DSMEinfo.confirm

7.1.20.4.3.1 General

The MLME-DSMEinfo.confirm primitive reports the results of a request for the timestamp and the DSME parameters.

7.1.20.4.3.2 Semantics

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

```
MLME-GTS.confirm          (
          DSMECharacteristics,
          Timestamp,
          status
          )
```

Table 78.nn specifies the parameters for the MLME-DSMEinfo.confirm primitive.

Table 78.uu— MLME-DSMEinfo.confirm parameters

Name	Type	Valid Range	Description
DSMECharacteristics	DSME Characteristics	See 7.3.10.2	The characteristics of the DSME.
Timestamp	Integer	0x000000–0xffffff	The time, in symbols, at which the DSME information reply command (see 7.3.11) was transmitted. This parameter will be considered valid only if the value of the status parameter is SUCCESS. The symbol boundary is described by macSyncSymbolOffset (see Table 127 in 7.4.2). This is a 24-bit value, and the precision of this value shall be a minimum of 20 bits, with the lowest 4 bits being the least significant.
Status	Enumeration	SUCCESS, CHANNEL_ACCESS_FAILURE, NO_ACK, NO_DATA, COUNTER_ERROR, FRAME_TOO_LONG, UNAVAILABLE_KEY, UNSUPPORTED_SECURITY or INVALID_PARAMETER.	The status of the DSME information request.

7.1.20.4.3.3 When generated

The MLME-DSMEinfo.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-DSMEinfo.request primitive. If the request was successful, the status parameter will be equal to SUCCESS and the DSME Characteristics Type field of the DSMECharacteristics parameter will be set to Restart (see Table 3). Otherwise, the status parameter indicates the appropriate error code. The status values are fully described in 7.1.18.1.3.

7.1.20.4.3.4 Appropriate usage

On receipt of the MLME-DSMEinfo.confirm primitive the next higher layer is notified of the result of the procedure to request the timestamp and the DSME parameters from the Destination device.

7.1.20.4.4 DSME information sequence chart

Figure 41.c illustrates the sequence of messages necessary for successful DSME information request. Figure 3 depicts the messages flow for the case in which the Source device requests the timestamp and the DSME parameters from the Destination device.

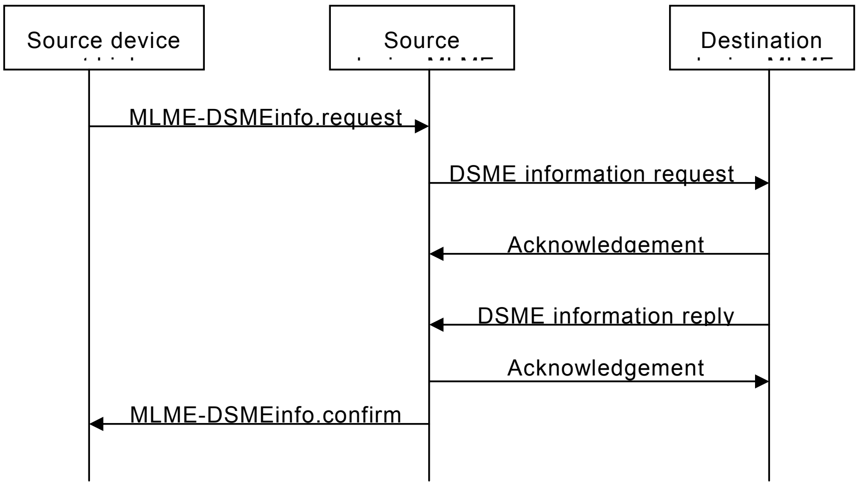


Figure 41.c—Message sequence chart for DSME information request

7.1.20.5 MLME-DSME-LINKSTATUSRPT

7.1.20.5.1.1 General

The MLME-DSME-LINKSTATUSRPT primitives define how a source device reports the communication status between the source device and the destination device.

All DSME-devices shall provide an interface for these link status report primitives.

7.1.20.5.2 MLME-DSME-LINKSTATUSRPT.request

7.1.20.5.2.1 General

The MLME-DSME-LINKSTATUSRPT.request primitive is generated by the higher layer of a source device, and is issued to its MLME to request a device start a link quality statistic and periodically report the statistic results to the destination device.

7.1.20.5.2.2 Semantics

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

```
MLME-DSME-LINKSTATUSRPT.request   (
                                   DstAddr,
                                   ReportPeriod
                                   )
```

Table 78.vv specifies the parameters for the MLME-DSME-LINKSTATUSRPT.request primitive.

Table 78.vv—MLME-DSME-LINKSTATUSRPT.request parameters

Name	Type	Valid Range	Description
DstAddr	Integer	0-0xffff	16bit address of the Destination device to which the link status report request is intended.
ReportPeriod	Integer	0-0xffffff	The time interval between two link status report command frames is defined as ReportPeriod * aBaseSuperframeDuration * 2MO symbols. If the parameter equals to 0x000000, link status reposrt command frame is not allowed to be sent.

7.1.20.5.2.3 Appropriate usage

The MLME-DSME-LINKSTATUSRPT.request primitive is generated by the higher layer of a device, and issued to its MLME to initiate a link status statistic.

7.1.20.5.2.4 Effect on receipt

On receipt of MLME-DSME-LINKSTATUSRPT.request primitive by a device, the MLME of the device attempts to generate a link status report command (see 7.3.14) with the information contained in this primitive, and if successful, sends it to the destination device according to the DstAddress parameter.

If the link status report command frame cannot be sent due to a CSMA-CA algorithm failure, the MLME will issue the MLME-DSME-LINKSTATUSRPT.confirm primitive with a status of CHANNEL_ACCESS_FAILURE.

If the MLME successfully transmits a link status report command frame, the MLME will expect an acknowledgement in return. If an acknowledgement is not received, the MLME will issue the MLME-DSME-LINKSTATUSRPT.confirm primitive with a status of NO_ACK.

If the link status report command frame has been acknowledged, the device will send another link status report command frame again in the interval defined in the parameter ReportPeriod.

If a device received a link status report command frame from another device in the PAN, the destination device will get the link status, and notify the result to its higher layer by the primitive MLME-DSME-LINKSTATUSRPT.indication.

7.1.20.5.3 MLME-DSME-LINKSTATUSRPT.confirm

7.1.20.5.3.1 General

The MLME-DSME-LINKSTATUSRPT.confirm primitive reports the results to start a link status report process.

7.1.20.5.3.2 Semantics

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

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

Table 78.ww specifies the parameters for the MLME-DSME-LINKSTATUSRPT.confirm primitive.

Table 78.ww— MLME-DSME-LINKSTATUSRPT.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	CHANNEL_ACCESS_ FAILURE, NO_ACK or SUCCESS	The status of starting link status report.

7.1.20.5.3.3 When generated

The MLME-DSME-LINKSTATUSRPT.confirm primitive is generated by the MLME and issued to its next higher layer in response to an MLME-LINKSATUSRPT.request primitive.

The MLME-DSME-LINKSTATUSRPT.confirm primitive returns a status of either SUCCESS, indicating the MAC sublayer has started reporting its statistic results periodically, or the appropriate error code.

7.1.20.5.3.4 Effect on receipt

On receipt of the MLME-DSME-LINKSTATUSRPT.confirm primitive by a device, the next higher layer is notified of the result of its request to start reporting link status in the PAN. If the request was successful, the status parameter will indicate a successful link status report operation. Otherwise, the status parameter will indicate the error.

7.1.20.5.4 MLME-DSME-LINKSTATUSRPT.indication

7.1.20.5.4.1 General

The MLME-DSME-LINKSTATUSRPT.indication primitive indicates the transfer of a link status report of a device from the MAC sublayer to the local next higher layer.

7.1.20.5.4.2 Semantics

The semantics of the MLME-DSME-LINKSTATUSRPT.indication primitive is as follows:

```
MLME-GTS.indication      (
          DstAddr,
          LinkStatusSpecification
          )
```

Table 78.xx specifies the parameters for the MLME-DSME-LINKSTATUSRPT.indication request primitive.

Table 78.xx—MLME-DSME-LINKSTATUSRPT.indication parameters

Name	Type	Valid Range	Description
DstAddr	Integer	0-0xffff	16bit address of the Destination device to which the link status report request is intended.
LinkStatusSpecification	Link Status Specification	See7.3.14	The link status specification.

7.1.20.5.4.3 When generated

The MLME-DSME-LINKSTATUSRPT.indication primitive is generated by the MAC sublayer and issued to the next higher layer on receipt of a link status report command.

7.1.20.5.4.4 Effect on receipt

On receipt of the MLME-DSME-LINKSTATUSRPT.indication primitive, the next higher layer is notified of the arrival of a link status report command frame from a DSME-device. The usage of the link status report by the next higher layer is beyond the scope of this document.

7.1.20.5.5 MLME-DSME-LINKSTATUSRPT message sequence charts

Figure 41.d illustrates the sequence of messages necessary for link status report initialized by a source device and Figure 41.e for destination device.

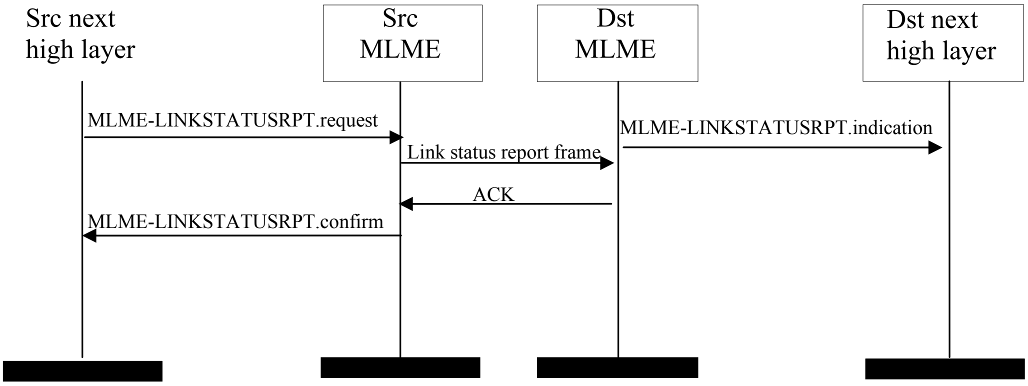


Figure 41.d—Message sequence chart for link status report

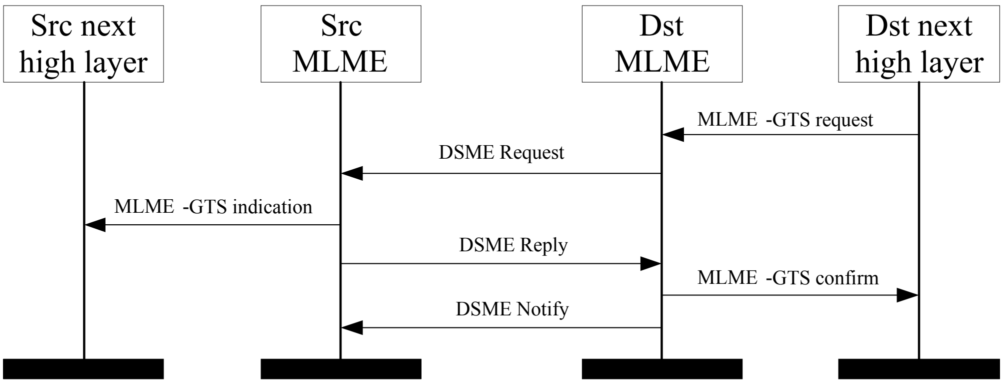


Figure 41.e—Message sequence chart for DSME allocation initiated by a Destination device

7.1.20.6 DSME-Beacon notification primitive

7.1.20.6.1 General

The MLME-SAP DSME-Beacon Notification primitive defines how a device may be notified when a beacon is received during normal operating conditions.

All DSME-devices shall provide an interface for the beacon notification primitive.

7.1.20.6.2 MLME-DSME-BEACON-NOTIFY.indication

7.1.5.1 applies except Table 91 shall be amended by Table 78.yy.

Table 78.yy—Additional elements of DSME-PANDescriptor

Name	Type	Valid Range	Description
DSMESuperframeSpec	Bitmap	See 7.2.2.1.2	The DSME superframe specification as specified in the received beacon frame.
BeaconBitmap	Bitmap	See 7.2.2.1.2	Indicates the beacon frame allocation information of neighbor nodes.

7.1.20.7 DSME-Primitives for channel scanning

7.1.20.7.1 General

MLME-SAP DSME-scan primitives define how a DSME-device can determine the energy usage or the presence or absence of PANs in a communications channel.

All DSME-devices shall provide an interface for these scan primitives.

7.1.20.7.2 MLME-DSME-SCAN.request

7.1.20.7.2.1 General

7.1.11.1 applies.

7.1.20.7.2.2 Semantics of the service primitive

7.1.11.1.1 applies except the parameter ScanType in Table 103 shall be amended for the coding 0x04 and 0x05, see Table 78.zz.

Table 78.zz—Additional elements of MLME-DSME-SCAN.request parameters

Name	Type	Valid Range	Description
ScanType	Integer	0x00–0x05	Indicates the type of scan performed: 0x00 = see Table 103. 0x01 = see Table 103 0x02 = see Table 103. 0x03 = see Table 103. 0x04 = asymmetric multi-channel active scan. 0x05 = channel probe.

7.1.20.7.2.3 Appropriate usage

This primitive can be used to perform an ED scan to determine channel usage, an active or passive scan to locate beacon frames containing any PAN identifier, or an orphan scan to locate a PAN to which the device is currently associated, or an asymmetric multi-channel active scan to detect the best designated channel for the device, or a channel probe scan to probe other channels and choose a better channel to switch to.

7.1.20.7.2.4 Effect on receipt

The asymmetric multi-channel active scan is performed on each channel by the MLME first sending a multi-channel beacon request command (see 7.3.11). The MLME then enables the receiver and records the information contained in the received beacon in a PAN descriptor structure (see Table 91 and Table 78.yy). If the LinkQualityScan flag is FALSE, the asymmetric multi-channel active scan terminates when the device receives a beacon and then choose the current channel as its designated channel. Otherwise, if the LinkQualityScan flag is TRUE, the asymmetric multi-channel active scan on a particular channel terminates when [aBaseSuperframeDuration * (2n + 1)] symbols, where n is the value of the ScanDuration parameter have elapsed after successful transmission of the multi-channel beacon request command, then switch to the

Comment [W6]: added this paragraph according to the description in 7.5.11.2, but I can't find the definition of "linkqualityscan", I think we should add it, in the PIB or just in the primitive.

next channel and repeat the same procedure. In this case, the whole asymmetric multi-channel active scan terminates when the device have scanned every channel twice, and the device will choose its designated channel according to the LQI or RSSI of the received beacons. See 7.5.11.2 for more detailed information on the asymmetric multi-channel active scan.

The channel probe scan is performed on the channel specified by the ScanChannels parameter by the MLME first sending a channel probe request frame (see 7.3.11) to one of its neighbors on the designated channel of the neighbor. The MLME then switch to the device own designated channel and enables the receiver and waits the channel probe reply frame. The channel probe scan terminates when [aBaseSuperframeDuration * (2n + 1)] symbols, where n is the value of the ScanDuration parameter have elapsed after successful transmission of the channel probe request frame. The device shall check the LQI or RSSI of the channel probe reply frame upon receiving it. See 7.5.11.4 for more detailed information on the channel probe scan.

The results of an asymmetric multi-channel active scan are reported to the next higher layer through the MLME-SCAN.confirm primitive. If the scan is successful and macAutoRequest is set to TRUE, the primitive results will include a set of PAN descriptor values. If the scan is successful and macAutoRequest is set to FALSE, the primitive results will contain a null set of PAN descriptor values, and each PAN descriptor value will be sent individually to the next higher layer using separate MLME-BEACON-NOTIFY (see 7.1.5.1) primitives. In both cases, the MLME-SCAN.confirm primitive will contain a list of unscanned channels and a status of SUCCESS.

If, during an asymmetric multi-channel active scan, the MLME is unable to transmit a multi-channel beacon request command on a channel specified by the ScanChannels parameter due to a channel access failure, the channel will appear in the list of unscanned channels returned by the MLME-SCAN.confirm primitive. If the MLME was able to send a multi-channel beacon request command on at least one of the channels but no beacons were found, the MLME-SCAN.confirm primitive will contain a null set of PAN descriptor values, regardless of the value of macAutoRequest, and a status of NO_BEACON.

The results of a channel probe scan are reported to the next higher layer through the MLME-SCAN.confirm primitive. If the scan is successful the primitive results will include a status of SUCCESS.

If, during a channel probe scan, the MLME is unable to transmit a channel probe request frame on a channel specified by the ScanChannels parameter due to a channel access failure, the channel will appear in the list of unscanned channels returned by the MLME-SCAN.confirm primitive. If the MLME was able to send a channel probe request frame but no channel probe reply was found, the MLME-SCAN.confirm primitive will contain a status of BAD_CHANNEL.

7.1.20.7.3 MLME-DSME-SCAN.confirm

7.1.20.7.3.1 General

7.1.11.2 applies.

7.1.20.7.3.2 Semantics of the service primitive

7.1.11.2.1 applies except the parameter status in Table 104 shall be amended with BAD_CHANNEL, see Table 78.aaa.

Table 78.aaa—Additional element of MLME-DSME-SCAN.confirm parameters

Name	Type	Valid Range	Description
status	Enumeration	SUCCESS, LIMIT_REACHED, NO_BEACON, SCAN_IN_PROGRESS, COUNTER_ERROR, FRAM_TOO_LONG, UNAVAILABLE_KEY, UNSUPPORTED_SECURITY, BAD_CHANNEL or INVALID_PARAMETER	The status of the scan request.

DSME-technology provider: More description should be added here.

7.1.21 LE-specific MAC sublayer service specification

7.1.21.1 General

MLME-SAP FCS error primitive defines how a device may be notified when data frame is received with error in the RIT mode.

This primitive is optional for a LE-device supporting RIT mode.

7.1.21.2 MLME-FRAME-ERROR.indication

7.1.21.2.1 General

The MLME-FRAME-ERROR.indication primitive is used to notify the reception of an erroneous data frame by the MAC sublayer to the next higher layer.

7.1.21.2.2 Semantics of the service primitive

The semantics of the MLME-FRAME-ERROR.indication primitive are as follows:

```
MLME-FRAME-ERROR.indication(
                                status,
                                )
```

Table 78.bbb specifies the parameters for the MLME-FRAME-ERROR.indication primitive.

Table 78.bbb— MLME-FRAME-ERROR.indication parameters

Name	Type	Valid Range	Description
status	Enumeration	FCS_ERROR, SECURITY_ERROR	The status of the erroneous reception of data frame in RIT mode

7.1.21.2.3 When generated

The MLME-FRAME-ERROR.indication primitive is generated by MLME and issued to its next higher layer upon reception of an erroneous data frame in RIT mode.

7.1.21.2.4 Appropriate usage

On receipt of the MLME-FRAME-ERROR.indication primitive, the next higher layer is notified of the reception of an erroneous data frame at the MAC sublayer.

7.2 MAC frame formats

7.2.1 General MAC frame format

Change the first paragraph 7.2.1.

The MAC frame format is composed of a MHR, a MAC payload, and a MFR. The fields of the MHR appear in a fixed order; however, the addressing fields may not be included in all frames. Furthermore, some frame types use a MHR of only 1 octet length with a shortened Frame Control field. The general MAC frame shall be formatted as illustrated in Figure 41.

Replace Figure 41 by the following figure.

octets: 1/2	0/1	0/2	0/2/8	0/2	0/2/8	0/5/6/10/14	0/1	variable	0/2
Frame control	Sequence number	Destination PAN Identifier	Destination Address	Source PAN Identifier	Source Address	Auxiliary Security Header	Frame Payload Header	Frame Payload	FCS
		Addressing fields							
MHR							MAC payload		MFR

7.2.1.1 Frame Control field

Change the paragraph as follows.

The frame control field is 1 or 2 octets in length and contains information defining the frame type, addressing fields, and other control flags. For ease of reference, the 1-octet frame control field shall be referred to as the short frame control field in this clause, whereas the 2-octet frame control field may be referred to as the full frame control field.

Replace Figure 42 by the following figure.

bits: 0-1	2	3	4	5	6	7-8	9	10-11	12-13	14-15
Frame Type	sFCF=0	Security	Frame Pending	ACK request	PANId Compression	Reserved	Split Payload Field	Dest. Addressing Mode	Frame Version	Source Addressing Mode

Add the following text and figures before 7.2.1.1.1:

Full frame control field

The frame control field shall be formatted as illustrated in Figure 42.a or Figure 42.b, depending on the value of the frame type subfield.

Short frame control field

The short frame control field shall be formatted as illustrated in Figure 42.a or Figure 42.b, depending on the value of the frame class subfield.

bits: 0-1	2	3	4	5	6	7
Frame Type	sFCF=1	Security	Frame Pending	ACK request	Ext Frame Type	Frame version

Figure 42.a—Frame control field when Frame type subfield is set to ???

Comment [W7]: René to check title

bits: 0-1	2	3	4	5	6	7
Frame Type	sFCF=1	Security	Ext. Frame Type	Ext. Frame Type	Ext Frame Type	Frame version

Figure 42.b—Frame control field when Frame type subfield is set to ???

Comment [W8]: René to check title

7.2.1.1.1 Frame Type subfield

Change Table 79 as follows.

Frame Type b1b0	Ext. b6	Ext. b5	Ext b4	Description
01	1		0	LL-Beacon
11	1		0	LL-Data
11	1		1	LL-Command
01	1	0	1	LL-ACK
01	1	1	1	CSL Wake-up
10	1	0	0	Blink
10	1	0	1	Blink w/ source
10	1	1	0	Blink w/Dst PAN
10	1	1	1	Blink w/ source & Dst PAN
00				Reserved
01	0			Short Data
11	0			Short Command
10	0			Short ACK
Non-specified sub-fields are frame type specific and specified in the respective subclause for the frame type				

Comment [W9]: LW: René will provide revised text according discussion in LA at 2010-01-21.

Insert before 7.2.1.1.2 the following subclauses.

7.2.1.1.1.1 Frame type identifier for full frame control field

The frame type identifier is 2 bits in length and shall be set to one of the nonreserved values listed in Table 79.

7.2.1.1.1.2 Frame type identifier for short frame control field

The frame type subfield is 3-5 bits in length and shall be set to one of the nonreserved values listed in Table 79.

7.2.1.1.2 Security Enabled subfield

7.2.1.1.5

7.2.1.1.6 Destination Addressing Mode subfield

Change Table 80 as follows.

Addressing mode value b1 b0	Description
00	PAN identifier and address fields are not present.
01	Address field contains an 8-bit simple address.
10	Address field contains a 16-bit short address.
11	Address field contains a 64-bit extended address.

Change the text as follows:

If this subfield is equal to zero, and the Frame Type subfield does not specify that this frame is an acknowledgment or beacon, the Source Addressing Mode subfield shall be non-zero, implying that the frame is directed to the PAN coordinator with the PAN identifier as specified in the Source PAN identifier field.

7.2.1.1.7 Frame Version subfield

Insert below 7.2.1.1.7 a new subclause.

7.2.1.1.7.1 Frame version subfield for long frame control field

Insert before 7.2.1.1.8 the following subclause:

7.2.1.1.7.2 Frame version subfield for short frame control field

The frame version subfield of the short frame control field is 1 bit in length and specifies the version number corresponding to the frame.

This subfield shall be set to 0 to indicate a frame is compatible with IEEE Std 802.15.4-2006. All other subfield values shall be reserved for future use. See 7.2.3 for details on frame compatibility.

7.2.1.1.8 Source Addressing Mode subfield

Insert the following subclauses before 7.2.1.2 (in a consolidated version the order should be revised so that the occurrence of fields are sequential):

Comment [W10]: RS to provide revised text

7.2.1.1.9 Short Frame Control Field Subfield

The Short Frame Control Field (sFCF) subfield is 1 bit in length and shall be set to one if the frame control field is a short frame control field and shall be set to zero otherwise.

7.2.1.2 Sequence Number field

Insert the following text at the end of subclause7.2.1.2:

This field shall not be present if

- the frame type indicates an LL-frame type and
- the security enabled subfield is set to zero.

Editorial note RS:
remove of §7.2.5.1 of 09/604r3, since equivalent functionality.
LW: Done, but a substitution is needed.

Comment [W11]: René to provide a revised 7.2.5.1 and other subclauses related to the one octed FCF.

7.2.1.3 Destination PAN Identifier field

7.2.1.4 Destination Address field

7.2.1.5 Source PAN Identifier field

7.2.1.6 Source Address field

7.2.1.7 Auxiliary Security Header field

7.2.1.8 Frame Payload field

7.2.1.9 FCS field

Insert at the end of the 1st paragraph the following text:

This field shall be present only if the security enabled subfield is set to zero or if frame protection does not result in data expansion of the frame payload field (see Figure 41).

Insert the following subclauses before 7.2.2 (in a consolidated version the order should be revised so that the occurrence of fields are sequential):

7.2.1.10 Split payload subfield

The Split Payload subfield is 1 bit in length and shall be set to one if the frame payload consists of at least two distinct subfields and shall be set to zero otherwise. This subfield shall be ignored for the acknowledgement frame type. If this subfield is set, the frame version shall be set to 0x02.

NOTE—Indicating a TG4e frame (and the full frame control field shall be used).

Comment [W12]: RS to provide revised text

7.2.1.11 Frame payload header field

The frame payload header field is a 1-octet field and shall be set to one of the non-reserved values in Table 42.a.

Table 42.a—Values of the Frame payload header field

Frame payload header	Description
0x00	CSL sync field
0x01- 0xff	Reserved

7.2.2 Format of individual frame types

7.2.2.1 Beacon frame format

Update Fig. 44 to align octet sizes of frame subfield with those in §7.2.1.

Comment [W13]: René provide Table with header

Editorial note RS:

Make similar editorial changes for the data frame (§7.2.2.2 – Fig. 52) and command frame (§7.2.4 – Fig. 54).

Comment [W14]: René provide Table with header

7.2.2.1.1 Beacon frame MHR fields

Change the second paragraph as follows:

In the Frame Control field, the Frame Type subfield shall contain the value that indicates a beacon frame, as shown in Table 79, and the Source Addressing Mode subfield shall be set as appropriate for the address of the coordinator transmitting the beacon frame. If protection is used for the beacon, the Security Enabled subfield shall be set to one. The Frame Version subfield shall be set to a value unequal to 0x00 only if the Security Enabled subfield is set to one. If a broadcast data or command frame is pending, the Frame Pending subfield shall be set to one. All other subfields shall be set to zero and ignored on reception.

7.2.2.2 Data frame format

7.2.2.3 Acknowledgment frame format

Replace Fig.53 by the following figure (i.e., allow security, addressing, etc.).

octets: 1/2	0/1	0/2	0/2/8	0/2	1/2/2008	0/5/6/10/14	variable	0/2
Frame control	Sequence number	Destination PAN Identifier	Destination Address	Source PAN Identifier	Source Address	Auxiliary Security Header	Frame Payload	FCS
		Adddressing fields						
MHR							MAC payload	MFR

7.2.2.3.1 Acknowledgment frame MHR fields

Editorial note RS:

The acknowledgement frames introduced with various proposals, such as TSCH, low-energy subgroup, and group acknowledgement, should be treated similarly as the various command frames and warrant a separate section. General format follows the following principles:

- **legacy 802.15.4-2006 acknowledgement frame is supported as special case (no payload, no security, etc.);**
- **if frame is sent with cryptographic protection, corresponding acknowledgement is sent with the same key, frame counter, and security level as indicated in security policy. Using the same key and frame counter allows compressing the auxiliary security header field and associated frame overhead entirely.**
- **if acknowledgement is sent with much delay, security processing which depends on correlation of frame counter with on-device time notion will fail or cause unacceptable ambiguities. This can be remedied, but may require a slight change of the incoming and outgoing frame security procedure (which may not entirely be an extension of 802.15.4-2006).**

Coordination between various proposers required.

Comment [W15]: LW: TBD by René.

Insert the following Clause:

Comment [W16]: LW: This subclause exists in 15.4:2006! Is that a replacement or an amendment?

§7.2.2.3.1 Acknowledgement frame MHR fields

Insert at the end of 7.2.2.3.1 the folowing text and new subclauses.

The MHR for an acknowledgment frame shall contain the short Frame Control field, the Sequence Number field, and may contain addressing fields of the originator of the frame.

In the Short Frame Control field, the subfields identifying the frame type shall contain the value that indicates an acknowledgment frame, as shown in Table 79. If protection is used for the acknowledgement frame, the Security Enabled subfield shall be set to one. The (virtual) Source Addressing Mode subfield shall be set to the value that indicates an extended address if the Addressing Mode Subfield of the Short Frame Control Field is set to one and to the value indicating that a source address is not present otherwise. The Frame Version subfield shall be set to the value indicating a TG4e frame version, as shown in Table {xxx}.

The Sequence Number field shall contain the value of the sequence number received in the frame for which the acknowledgment is to be sent.

The addressing fields shall comprise only the source addressing fields. The Source Address field shall contain the address of the device originating the short acknowledgement frame according to the value of the virtual Source Addressing Mode subfield.

If protection is used for the acknowledgement frame, the (virtual) Auxiliary Security Header field shall be set to the same value as the corresponding field of the frame that is being acknowledged and shall not be included in the acknowledgement frame to be sent.

7.2.2.3.2 Acknowledgement frame payload fields

7.2.2.3.2.1 General

The acknowledgement frame payload field has a variable length and specifies information useful for synchronizing communications between sender and recipient, including loosely synchronizing timing information and capability information.

***Editorial note RS:* the so-called "piggy-backing".**

NOTE—Inclusion of this field is determined via inspection of the Frame Length subfield of the PHY header field of the PPDU and the other frame fields.

The acknowledgement payload fields shall be formatted as illustrated in Figure 54.a:

0/1	variable
ACK control field	Acknowledgement Payload
MAC payload	

Figure 54.a—Acknowledgement payload fields???

Comment [W17]: René to check title

7.2.2.3.2.2 Acknowledgement control subfield

7.2.2.3.2.2.1 General

The acknowledgement control subfield, when present, is 1 octet in length and specifies which synchronization information, if any, is communicated back to the originator of the frame that is being acknowledged, see Figure 54.b.

bits: 0-2	3	4	5	6	7
ACK Identifier	Frame pending	Reserved	Time sync	Time offset	Security sync

Figure 54.b—Acknowledgement control field???

Comment [W18]: René to check title

7.2.2.3.2.2.2 ACK identifier subfield

***TBD: Editorial note RS:* include table with ACK types with LE, LL, PA.**

Comment [W19]: TBD by René.

7.2.2.3.2.2.3 Frame pending subfield

***Editorial note RS:* the frame pending subfield is the same as used within 802.15.4-2006, but now grouped with other synchronization information.**

The Frame Pending subfield is 1 bit in length and shall be set to one if the device sending the frame has more data for the recipient and shall be set to zero otherwise (see 7.5.6.3).

If the acknowledgment frame is being sent in response to a received data request command, the device sending the acknowledgment frame shall determine whether it has data pending for the recipient. If the device can determine this before sending the acknowledgment frame (see 7.5.6.4.2), it shall set the Frame Pending subfield according to whether there is pending data. Otherwise, the Frame Pending subfield shall be set to one. If the acknowledgment frame is being sent in response to either a data frame or another type of MAC command frame, the device shall set the Frame Pending subfield to zero.

7.2.2.3.2.2.4 Time sync subfield

The Time sync subfield is 1 bit in length and shall be set to one if the acknowledgement contains time synchronization information.

7.2.2.3.2.2.5 Time offset subfield

The Time sync subfield is 1 bit in length and shall be set to one if the acknowledgement is sent with time delay.

7.2.2.3.2.2.6 Security sync subfield

The Time sync subfield is 1 bit in length and shall be set to one if the acknowledgement contains security synchronization information.

***TBD: Editorial note RS:* to be finalized after receipt of offline feedback from different subgroups.**

Comment [W20]: TBD by René.

7.2.3 Frame compatibility

7.2.4 LL-Frame Formats

(Editorial note RS: this refers to LL-frame types)

§7.2.4 (of 09/604r3) PA-frame formats (Editorial note RS: this refers to Secured ACK)

Editorial note RS:

This entire clause §7.2.4 can be removed, since secured ACKs, both with and without payload fields, are handled elsewhere in the TG4e draft specification (see §7.2.2.3 as described in this document).

LW: Done

7.2.5 DSME-Frame Formats

(Editorial note RS: this refers to LL-frame types)

7.2.5.1 General MAC frame format

Subclause 7.2.1 applies.

Editorial note RS:

This entire clause §7.2.5.1 can be removed, since the functionality is provided by particular instantiations of the general frame format (see §7.2.1, as described in this document).

7.2.5.2 Format of individual frame types

7.2.5.2.1 General

(Editorial note RS: this refers to LL-frame types)

Four frame types are defined: beacon, data, acknowledgment, and MAC command. These frame types are discussed in 7.2.6.2 through 7.2.6.5.

7.2.5.2.2 Beacon frame format

7.2.5.2.2.1 General

The beacon frame shall be formatted as illustrated in Figure 82.

octets: 2	1	4/10	0/5/6/10/14	2	variable	variable	4	variable	4	variable	variable	2
Frame Control	Sequence Number	Addressing Fields	Auxiliary Security Header	Superframe Specification	GTS (Figure 45)	Pending address fields (Figure 46)	DSME Superframe Specification (Figure 51b)	ChannelHoppingSpecification (Figure 44a)	Time synchronization Specification (Figure 54)	Beacon Bitmap (Figure 53)	Beacon Payload	FCS
MHR				MAC Payload								MFR

Figure 54.c—Beacon frame format

All frame formats in this clause shall use the short frame control field (see 7.2.1.1.9).

Editorial note RS:

This allows lots of editorial clean-up in §7.2.5.2 of 09/604r3, All editorial changes are aimed at replacing explicit cross-references to the to bit positions indicating low latency frame type to corresponding name for this frame type (this is similar to referring to, e.g., security subfield of the FCF, rather than to bit b3 of the FCF ["replacement of literal value by variable indicating this value"]).

We illustrate this for the LL-beacon frame below:

§7.2.5.2.2 (of 09/604r3) Beacon frame

§7.2.5.2.2.1:

l. 11: Replace “The Beacon frame with shortened frame control (1 octet MAC header)” by “The LL-Beacon frame”.

l. 17-18: Replace “The beacon frame does have a very short MAC header (MHR) of one octet containing the frame type and sub frame type, followed by the beacon payload and the MAC footer (MFR)” by “The beacon frame consists of a short message header, followed by the beacon payload and the MAC footer (MFR)”.

§7.2.5.2.2.2: Replace this clause by the following text:

“In the frame control field, the frame type subfield shall contain the value that indicates an LL- beacon frame.

Editorial note RS:

Text below on blink frame based on email received from Dalibor Pokrajac as of Monday November 30, 2009, 3:44pm EST.

7.2.5.2.2.2 Beacon frame MHR fields

The Frame Version subfield is 2 bits in length and specifies the version number corresponding to the frame.

This subfield shall be set to 0x00 to indicate a frame compatible with IEEE Std 802.15.4-2003, 0x01 to indicate an IEEE Std 802.15.4-2006 frame and 0x10 to indicate an DSME-frame. All other subfield values shall be reserved for future use. See 7.2.3 for details on frame compatibility.

7.2.5.2.2.3 Superframe Specification field

7.2.2.1.2 applies.

7.2.5.2.2.4 GTS Specification field

7.2.2.1.3 applies.

7.2.5.2.2.5 GTS Directions field

7.2.2.1.4 applies.

7.2.5.2.2.6 GTS List field

7.2.2.1.5 applies.

7.2.5.2.2.7 Pending Address Specification field

7.2.2.1.6 applies.

7.2.5.2.2.8 Address List field

7.2.2.1.7 applies.

7.2.5.2.2.9 Beacon Payload field

7.2.2.1.8 applies.

7.2.5.2.2.10 DSME Superframe Specification field

The DSME Superframe Specification field shall be formatted as illustrated in Figure 54.d.

bits: 0-3	4	5	6	7	8-23	24-31	32	33-35	variable
Multi-superframe Order (MO)	DSME Flag	CAP Reduction Flag	Embedded CAP/CFP Flag	Channel Diversity Mode	CAP Index	Number of Subslots	GACK Flag	ECFP Start Slot Length	ECFP Start Slot

Comment [youcy21]: In 15.4, all the subfields indicating the timeslot in different frame are 4 bits in length, but why the length of the ECFP Start Slot subfield is variable?

Figure 54.d—Format of the DSME Superframe Specification field

The Multi-superframe Order subfield is 4 bits in length and shall specify the length of time during which a group of superframes that is considered as one multi-superframe is active (i.e.. receiver enabled), including the beacon frame transmission time. See 7.5.1.1 for an explanation of the relationship between the Multi-superframe Order and the multi-superframe duration.

The DSME Flag subfield is 1 bit in length, and it shall be set to zero if the CFP of a superframe is operated the same way as defined in IEEE 802.15.4-2006, and the other subfields in the DSME Superframe Specification field shall all be ignored. The DSME Flag bit shall be set to one if the CFP of a superframe is operated as the DSME mode (defined in 7.5.9 and 7.5.10).

The CAP Reduction Flag subfield is 1 bit in length and shall be set to one if the CAP reduction is enabled. Otherwise, the CAP Reduction Flag subfield shall be set to zero.

The Embedded CAP/CFP Flag subfield is 1 bit in length, and shall be set to zero if the Embedded CAP is used (see 7.5.9). The Embedded CAP/CFP Flag bit shall be set to zero if the Embedded CFP is used (see 7.5.9).

The CAP Index subfield is 2 octets in length and shall specify the number of superframes before the next CAP begins. This subfield is valid only if the CAP Reduction Flag subfield is set to one.

The Number of Sub-slots subfield is 8 bits in length and shall specify the number of sub-slots which are divided within a slot. This subfield is valid only if the Embedded CAP/CFP Flag subfield is set to zero.

The Channel Diversity mode subfield is 1 bit in length and shall indicate the type of channel diversity. If this value is '0', DSME runs on channel adaptation mode. If this value is '1', DSME runs on channel hopping mode. If this subfield is '0', the following Channel Hopping Specification field is not present.

The GACK Flag subfield is 1 bit in length and shall indicate whether the transmitting device is using DSME multi-frame structure with group acknowledgement mechanism. If the GACK Flag subfield is set to '1', the superframe of the transmitting device shall be using group acknowledge mechanism, and have a structure as shown in Figure 8. If the GACK Flag subfield is set to '0', the transmitting FFD can not support group acknowledgement mechanism, the superframe structure will be shown as Figure 7, and the following ECFP Start subfield is not present.

The ECFP Start Length subfield is 3 bits in length and shall specify the length of the ECFP Start subfield.

The ECFP Start subfield shall specify the timeslot number of GACK frame transmitting (see 7.5.9.3), as the end of the CFP and start of the ECFP in the superframe structure as shown in Figure 8. The length of the ECFP Start subfield is variable and specified by the ECFP Start Length subfield.

7.2.5.2.2.11 Channel Hopping Specification field

The Channel Hopping Specification field shall be formatted as illustrated in Figure 54.e.

octets: 1	1	variable
Channel Offset	ChannelOffset Bitmap Length	ChannelOffset Bitmap

Figure 54.e— Format of the ChannelHoppingSpecification field

The Channel Hopping Specification field shall be present, if Channel Diversity Mode subfield in the DSME Superframe Specification field is set to '1'.

The ChannelOffset subfield is 1 octet in length and shall specify the channel hopping offset value of the device.

The ChannelOffsetBitmapLength subfield is 1 octet in length and shall specify the length of ChannelOffsetBitmap subfield.

The ChannelOffsetBitmap subfield shall indicate the occupancy of channel hopping offset values among neighbor devices and be represented in bitmap. Each bit shall be set to '1', if the corresponding channel hopping offset value is already occupied by the neighbor devices, otherwise it shall be set to '0' if the corresponding channel hopping value is not occupied. For instance, ChannelOffsetBitmap of 1100100..0 indicates that channel hopping offset values of 0, 1, and 4 are being used by neighbor devices. Note that the (i)th bit in the ChannelOffsetBitmap corresponds to (i-1)th channel offset value. The length of ChannelOffsetBitmap subfield is variable, which is defined by the values specified in ChannelOffsetBitmapLength subfield.

7.2.5.2.2.12 Time Synchronization Specification field

The Time Synchronization Specification field is 4 octets in length and shall be formatted as illustrated in Figure 54.f.

bits: 0	1-4	5-7	8-31
Deferred Beacon Flag	Deferred Beacon Time	Reserved	Beacon Timestamp

Figure 54.f—Format of the Time Synchronization Specification field

The Deferred Beacon Flag subfield is 1 bit in length and shall be set to one if the device uses CCA before transmitting beacon frame, otherwise the bit shall be set to zero if the device shall not use CCA before transmitting beacon.

The Deferred Beacon Time subfield is 4 bits in length and shall specify the number of backoff period for CCA. If the Deferred Beacon Flag bit is set to zero, this subfield shall be ignored.

The Beacon Timestamp subfield is 3 octets in length and shall specify the time of beacon transmission for time synchronization in symbol periods.

7.2.5.2.2.13 Beacon Bitmap field

The Beacon Bitmap field is 8 bits in length and shall be formatted as illustrated in Figure 51D.

octets: 2	variable
SD Index	SD Bitmap

Figure 54.g—Format of the Beacon Bitmap field

The SD Index subfield is 2 octets in length and specifies the Superframe Duration (SD) bank number that is allocated to the Source device of the beacon.

The SD Bitmap subfield is 2(BO-SO) bits in length and shall indicate the beacon frame allocation information of neighbor nodes. This subfield is expressed in bitmap format which orderly represents the schedule of beacons, with corresponding bit shall be set to one if a beacon of neighbor nodes is allocated in that SD.

7.2.5.2.3 Data frame format

Subclause 7.2.2.2 applies.

7.2.5.2.4 Acknowledgment frame format

Subclause 7.2.2.3 applies.

7.2.5.2.5 Data Group ACK (GACK)

As opposed to the acknowledgement to aindividual frames, the structure of a group ACK frame is more complex. A group acknowledgement not only indicates the reception status for a group of GTS data frames but it also spacifies new slot allocations, if any, for retransmission of failed GTS transmissions. The structure of the Acknowledgement Payload field of the group acknowledgement frame is shown in Figure 54.h.

Octets: 2	1								
Frame Control	Sequence Number	PAN ID	Source ID	Group ACK Flags	Channel Index	EGTS Device List	EGTS Index	EGTS Directions	FCS
MHR		MAC Payload							MFR

Figure 54.h—Format of the GACK Frame

PAN ID: This field shall identify the PAN of the transmitting coordinator.

Source ID: This field shall identify the transmitting LL_NW PAN coordinator.

Group Ack Flags: It is a bitmap that indicates the state of transmission in each GTS in previous SCFP or ECFP. A bit having ‘1’ indicates the fact that the coordinator received the data frame successfully in the corresponding GTS. A ‘0’ means that the coordinator failed in receiving a data frame in the corresponding slot.

Channel Index: This field specifies the channel sequence to be followed in the ECFP or a tailing CAP, if allowed in a system, in a channel hopping system. This field shall be non-existent in the GACK 2 frame.

DSME Device List: This list identifies the sensor nodes that are being allocated the time slots in ECFP portion of the superframe. This field shall be non-existent in the GACK 2 frame.

DSME Index: It is a list that specifies the start of each GTS for the allocated nodes in the same order as in DSME device list. This field is applicable only in those systems that allow a GTS to consist of multiple time slots. This field shall be non-existent in the GACK 2 frame.

DSME Directions: This list specifies the direction of transmission (uplink or downlink) for each GTS. This is applicable only in the systems that allow the coordinator to transmit a frame to its sensor nodes by using a GTS. This field shall be non-existent in the GACK 2 frame.

The Group Ack Flags field is a bitmap that indicates the states of transmissions of the sensors in the sensor time slots of the current superframe. A bit set to 1 indicates the fact that the coordinator received the data frame successfully in the corresponding time slot. A value of 0 means that the coordinator failed in receiving a data frame in the corresponding slot from the sensor.

7.2.5.2.6 MAC command frame format

Subclause 7.2.2.4 applies.

7.2.6 Extensibility Frame

7.2.6.1 General

Frames with the Frame Type subfield in the Frame Control field set to the value for Extensibility according to Table 120 allow to extend the range of IEEE 802.15.4 frame types beyond the existing ones.

The general structure of an extensibility frame is shown in Figure 54.i.

Octets: 2 + variable		variable	2
Frame Control	*other MHR fields depending on Subframe Type*	Frame Payload	FCS
MHR		MAC Payload	MFR

Figure 54.i—Format of Extensibility Frame

The Extensibility MAC frame does have a MAC header (MHR) of variable size, but of at least 2 octets. The MHR contains the Frame Control with the frame type and subframe type, and other MHR fields depending on the subframe type. The MHR is followed by the MAC payload and the MAC footer (MFR).

7.2.6.2 Frame Control field

7.2.6.2.1 General

The Frame Control field is 2 octets in length and contains information defining the frame type and the sub frame type. The Frame Control field shall be formatted as illustrated in Figure 54.j.

Bits: 0 - 2	3	4 - 6	7 - 9	10 - 11	12 - 13	14 - 15
Frame Type	Security Enabled	*same as IEEE 802.15.4, ignored, or redefined based on Sub Frame Type*	Sub Frame Type	*same as IEEE 802.15.4, ignored, or redefined based on Sub Frame Type*	Frame Version	*same as IEEE 802.15.4, ignored, or redefined based on Sub Frame Type*

Figure 54.j—Format of Extensibility Frame

7.2.6.2.2 Frame Type subfield

The Frame Type subfield is 3 bits in length and shall be set to b111 indicating this type of Extensibility MAC frame (see Table 120). The Frame Type subfield corresponds to the Frame Type subfield of the general MAC frame format in 7.2.1 in meaning and position. The new type b111 allows efficient recognition of frames with the extensible Subframe Type subfield (extensibility feature), but allows the usage of all other MAC frames as well.

7.2.6.2.3 Security Enabled subfield

The Security Enabled subfield is 1 bit in length, and it shall be set to one if the frame is protected by the MAC sublayer and shall be set to zero otherwise.

7.2.6.2.4 Sub Frame Type subfield

The Sub Frame Type subfield is 3 bits in length. It is located at bits 7 to 9, which are reserved in IEEE 802.15.4-2006 MAC frames. The Sub Frame Type field allows the specification of at least additional 8 frame types, One of these shall be the next extensible frame type.

7.2.6.2.5 Frame Version subfield

The Frame Version subfield is 2 bits in length and specifies the version number corresponding to the frame.

7.2.6.2.6 Other subfields

All other subfields may have the same meaning as in IEEE 802.15.4-2006 MAC frames as described in 7.2.1.1, may be ignored / reserved, or may be redefined depending on the value of the Sub Frame Type subfield.

7.2.7 Blink frame format

7.2.7.1 General

***Editorial note RS:* descriptive text for blink frames, as suggested by Dalibor Pokrajac, Monday November 30, 2009, 3:44pm EST:**

The blink frame shall be formatted as illustrated in Figure 54.k.

octets: 1/2	**1**	**0/2**	**0/8**	**0/5/6/10/14**	**variable**	**0/2**
Frame control	Sequence number	Destination PAN Identifier	Source Address	Auxiliary Security Header	Frame Payload	FCS
		Adddressing fields				
MHR					MAC payload	MFR

Figure 54.k—Blink frame format

The order of the fields of the blink frame shall conform to the order of the general MAC frame as illustrated in Figure 43.

7.2.7.2 Blink frame MHR fields

The MHR for a blink frame shall contain the Short Frame Control Field, the Sequence Number Field, and optionally the Destination PAN Id and/or the Source Address field.

In the Short Frame Control Field, the Frame Type shall contain the value that indicates a blink frame, as shown in Table 81.a.

Table 81.a—Blink frame format

Bit						
0, 1	**2**	**3**	**4**	**5**	**6**	**7**
A-Class	sFCF=1	Security	Source addressing mode	Destination PAN Identifier	1	Frame version

If protection is used for the wake-up frame, the Security Enabled subfield shall be set to one.

The Frame Version subfield of the Short Frame Control Field shall be set to the value zero, indicating a TG4e frame version, as shown in Table 81.a.

Comment [W22]: LW: As decided in LA the term TG4e shall be revised. René TBD.

The Sequence Number field shall be set to the current value of *macDSN*.

The Destination PAN Identifier field, if present, shall contain the PAN Identifier of the device receiving the blink frame. The Source Address field, if present, shall contain the extended address of the device originating the blink frame. All other addressing fields shall be omitted. The presence of these addressing fields shall be indicated by the Destination PAN Identifier subfield and the Source Addressing subfield of the Short Frame Control field, respectively (present if set; absent otherwise).

7.2.7.3 Blink frame payload field

The blink frame payload field is an optional sequence specified to be transmitted by the next higher layer. The set of octets contained in *macBlinkPayload* shall be copied into this field.

Editorial note RS:

The payload field, if present, is expected to encode and alternative identifier for the originating device. The details of this encoding are outside scope of this specification.

7.3 MAC command frames

Change the first two paragraphs according to the text below.

The command frames defined by the MAC sub-layer are listed in Table 123. An FFD shall be capable of transmitting and receiving all command frame types, with the exception of the GTS request command, while the requirements for an RFD are indicated in the table. MAC commands shall only be transmitted in the CAP for beacon-enabled PANs, in management time slots of Low Latency networks, or at any time for non-beacon-enabled PANs.

How the MLME shall construct the individual commands for transmission is detailed in 7.3.1 through 7.3.12~~9~~. MAC command reception shall abide by the procedure described in 7.5.6.2.

7.3.1 Association request command

Change the Table 55 according to the Table below.

1 **Table 55—MAC command frames**

Command frame identifier	Command name	RFD		Subclause
		Tx	Rx	
0x01	Association request	X		7.3.1
0x02	Association response		X	7.3.2
0x03	Disassociation notification	X	X	7.3.3
0x04	Data request	X		7.3.4
0x05	PAN ID conflict notification	X		7.3.5
0x06	Orphan notification	X		7.3.6
0x07	Beacon request			7.3.7
0x08	Coordinator realignment		X	7.3.8
0x09	GTS request			7.3.9
~~0x0a–0xff~~	~~Reserved~~			—
0x0a	Advertisement		X	7.3.10.1
0x0b	Join	X		7.3.10.2
0x0c	Activate		X	7.3.10.3
0x0d	Discover Response	X		7.3.11.1
0x0e	Configuration Response	X		7.3.11.2
0x0f	Configuration Request		X	7.3.11.3
0x10	CTS Shared Group		X	7.3.11.4
0x11	Request to send (RTS)	X	X	7.3.11.5
0x12	Clear to Send (CTS)		X	7.3.11.6
0x13	DSME handshake	X	X	7.3.10
0x14	DSME information request	X		7.3.11
0x15	DSME information reply		X	7.3.12
0x16	Beacon allocation notification			7.3.13
0x17	Beacon conflict notification	X		7.3.14
0x18	Link status report	X	X	7.3.15
0x19	Asymmetric multi-channel beacon request	X	X	7.3.16
0x1a	Multi-channel hello	X	X	7.3.17
0x1b	Multi-channel hello reply	X	X	7.3.18
0x1c	Channel probe	X	X	7.3.19
0x1d	RIT data request	X	X	7.3.14.1
0x1e	CSL wakeup	X	X	7.3.14.1
0x1f–0x3f	Reserved			—
0x40	RFID Blink	X		7.3.15.1
0x41-0x42	Reserved			—
0x43	RFID Blink with 64-bit source MAC address	X		7.3.15.1
0x44-0x5f	Reserved			—
0x60	RFID Blink with	X		7.3.15.1

Command frame identifier	Command name	RFD		Subclause
		Tx	Rx	
	Destination PAN ID			
0x61-0x62	**Reserved**			**—**
0x63	**RFID Blink with Destination PAN ID and 64-bit source MAC address**	**X**		**7.3.15.1**
0x64-0xff	**Reserved**			**—**

7.3.2 Association response command

7.3.2.1 MHR fields

7.3.2.2 Short Address field

7.3.2.3 Association Status field

Change the Table 83 according to the Table below.

Association status	Description
0x00	Association successful.
0x01	PAN at capacity.
0x02	PAN access denied.
0x03	Channel hopping sequence offset duplication (see 7.3.12.3.6)
0x04~~3~~-0x7f	Reserved.
0x80-0xff	Reserved for MAC primitives enumeration values.

7.3.8 Coordinator realignment command

7.3.9 GTS request command

Insert before 7.4 the following subclauses.

7.3.10 TSCH-commands

7.3.10.1 Advertisement command

7.3.10.1.1 General

The Advertisement command is used by FFDs to invite new devices into the network. When a device wishes to join a network, it shall use the information in Advertisement command frames to synchronize to the network and request an association. Figure 65.a shows the format of the Advertisement command.

octets: variable (see 7.2.2.4)	1	6	1	1	1	22	1	1	variable	1	variable	0/4/8/16	2
MHR	**Command Frame Identifier (see table 82)**	**Timing Information**	**Security Control Field**	**Join Control**	**Timeslot Template and Hopping Sequence ID**	**Timeslot Template (without Timeslot Template ID)**	**Channel Page/Map Length**	**Channel Page**	**Channel Map**	**Number of Slot-frames**	**Slotframe Info. and Links (for Each Slotframe)**	**MIC**	**MFR**

Figure 65.a—Advertisement command format

7.3.10.1.2 MHR field

The Source Addressing Mode subfield of the Frame Control field shall be set to three (64-bit extended addressing). The Destination Addressing Mode subfield shall be set to the broadcast address, i.e. 0xFFFF.

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon receipt, and the Acknowledgment Request subfield shall be set to zero. The Source PAN Identifier field shall contain the PAN identifier of the node. The Source Address field shall contain the value of aExtendedAddress.

The Sequence Number subfield shall be set to the least significant byte of the absolute slot number.

7.3.10.1.3 Command Frame Identifier field

The Type field shall be set to Advertisement (0x0a).

7.3.10.1.4 Timing Information field

The Timing Information field shall be set to the time information (i.e. Absolute Slot Number) of the timeslot being used for transmission of this command frame.

7.3.10.1.5 Security Control field

Figure 65.b shows the Security Control field.

Bit 4–7	Bit 0–3
Reserved	**Security Level**

Figure 65.b—Security Control field

The Security Level subfield should be set to the security level supported. The definition of the Security Level subfield can be found in Table 95.

7.3.10.1.6 Join Control field

Figure 65.c shows the Join Control field.

Bit 4–7	0–3
Reserved	**Join Priority**

Figure 65.c—Join Control field

The Join Priority subfield can be used by a joining device to decide which Network Devices to include in its Association Request if it hears advertisements from more than one device.

A lower value of join priority indicates that the device is a preferred one to connect to.

7.3.10.1.7 Timeslot Template and Hopping Sequence ID field

The Timeslot Template and Hopping Sequence ID field shall be set to the ID of the timeslot template and the ID of the hopping sequence used by the MAC. The timeslot templates and hopping sequences are defined in MAC PIB.

Figure 65.d shows the Timeslot Template and Hopping Sequence ID field.

Bit 4–7	0–3
Hopping Sequence ID	**Timeslot Template ID**

Figure 65.d—Timeslot Template and Hopping Sequence ID field

7.3.10.1.8 Timeslot Template

The Timeslot Template field shall be set to the macTimeslotTemplate from the MAC PIB (see Table 86.e) corresponding to the template ID in the previous field, minus the timeslot template ID.

7.3.10.1.9 Channel Page/Map Length field

The Channel Page/Map field shall be set to the combined length of following channel page and channel map fields.

7.3.10.1.10 Channel Page field

The Channel Page field shall be set to the channel page of channels that the joining device shall use for its hopping sequence.

7.3.10.1.11 Number of Slotframes field

The Number of Slotframes field is set to the total number of slotframes for which information is being advertised in this command frame.

7.3.10.1.12 Slotframe Information and Links (for each slotframe) field

7.3.10.1.13 General

Slotframe Information and Links field is included for each slotframe. The format of Slotframe Information and Links field is depicted as shown in Figure 65.e.

Octets: 1	2	1	Variable
Slotframe ID	**Slotframe Size**	**Number of Links**	**Link Info. for each Link**

Figure 65.e—Slotframe and Links field

7.3.10.1.14 Slotframe ID subfield

Slotframe ID shall be set to the ID that uniquely identifies the slotframe.

7.3.10.1.15 Slotframe Size subfield

Slotframe Size shall be set to the size of the slotframe in number of timeslots.

7.3.10.1.16 Number of Links subfield

The Number of Links subfield shall be set to the number of links that belong to the specific slotframe indicated in preceding slotframe ID.

7.3.10.1.17 Link Information (for each link) subfield

The Link Information subfield describes the attributes of each link. The format of Link Information subfield is depicted as shown in Figure 65.f.

Octets: 2	1	1
Timeslot	**Channel Offset**	**Link Option**

Figure 65.f—Link Information field

7.3.10.1.18 Timeslot subfield

The Timeslot subfield shall be set to the timeslot of this link.

7.3.10.1.19 Channel Offset Information subfield

The Channel Offset Information subfield shall be set to the channel offset of this link.

7.3.10.1.20 Link Option subfield

The Link Option subfield indicates whether this link is a TX link, an RX link, or a SHARED TX link. SHARED TX links can be used for a joining device to send its Join command. RX links are used for a new device to receive Advertisement commands. RX links can also be used for a joining device to receive its Activate command from the network. It is possible for one link to be used as both SHARED_TX and RX link.

7.3.10.1.21 MIC

The message integrity check of the Advertisement command frame.

7.3.10.2 Join command

Editorial note RS:

I am not sure whether the join command technically fits in the MAC layer, since it concerns an end-to-end communication between a joining device and a device that arbitrages access to the network (PAN coordinator, security manager, network manager, and the-like). Since most of these entities are unknown to the MAC layer (PAN coordinator aside), it seems this functionality needs to be provided at a higher layer – and, thereby, are outside scope of this specification. As an example, with ISA SP100.11a, this is application layer traffic. This requires more explanation.

7.3.10.2.1 General

The Join command is used by a device to join the TSCH-network through the advertiser. This command shall only be sent by a new device that wishes to join the TSCH-network or a device that lost connection with the TSCH-network.

All devices shall be capable of transmitting this command, although an RFD is not required to be capable of receiving it.

The Join command shall be formatted as illustrated in Figure 65.g.

octets: variable (see 7.2.2.4)	1	1	1	1	0/3	variable	0/3	1	0/4/8/16
MHR	Command Frame Identifier (see Table 82)	Capability Information (see Figure 56)	Clock Accuracy Capability	Number of Neighbors	Neighbor 1	…	Neighbor n	Join Security Information	MIC

Figure 65.g—Join command format

7.3.10.2.2 MHR fields

The Source Addressing Mode subfield of the Frame Control field shall be set to three (64-bit extended addressing). The Destination Addressing Mode subfield shall be set to the same mode as indicated in Advertisement command frame to which the Join command refers.

The Join command is analogous to and is used in place of an Association Request when joining a TSCH network.

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon receipt, and the Acknowledgment Request subfield shall be set to one.

The Destination PAN Identifier field shall contain the identifier of the PAN to which to join. The Destination Address field shall contain the address from the Advertisement frame that was transmitted by the coordinator to which the Join command is being sent. The PAN ID Compression subfield may be set to one and the Source PAN Identifier may be omitted. The Source Address field shall contain the value of aExtendedAddress.

The Sequence Number subfield shall be set to the least significant byte of the absolute timeslot number.

7.3.10.2.3 Command Frame Identifier field

The Type field shall be set to *Join (0x0b)*.

7.3.10.2.4 Capability Information field

The Capability Information field shall be formatted as illustrated in Figure 94.

7.3.10.2.5 Clock Accuracy Capability field

The Clock Accuracy Capability field shall be formatted as illustrated in Figure 65.h.

Bit: 0	1 - 7
10ppm capable	**reserved**

Figure 65.h—Clock Accuracy Capability

Bit 0 is to be set if the joining device has a timeslot clock that is at least 10 ppm accurate, i.e. the device will drift no more than 10 µs for every second between synchronization messages as described in 7.5.4.4.2.

7.3.10.2.6 Join Security Information field

Editorial note RS:

See editorial remark in §7.3.10.2.1.

The Join Security field is 1 octet bitmap where b0 = pre-shared symmetric key (as configured by a higher layer), b1=public key. If no bit is set, security is off. All other bits are reserved.

7.3.10.2.7 Number of Neighbor field

The Number of Neighbor field indicates the number of neighbors included in this command frame.

7.3.10.2.8 Neighbor field

The Neighbor field shall contain the information about the neighbors of the new device. The Neighbor field shall be formatted as illustrated in Figure 65.h.

Octets: 2	1
16 bit address of the neighbor	**RSSI**

Figure 65.i—Neighbor

7.3.10.2.9 MIC

This field contains the message integrity check of the Join command frame.

7.3.10.3 Activate command

7.3.10.3.1 General

Editorial note RS:

I am not sure whether the activate command technically fits witin the MAC layer, since it concerns an end-to-end communication to a joining device from a device that arbitrages access to the network or its resources (PAN coordinator, security manager, network manager, and the-like). Since most of these entities are unknown to the MAC layer (PAN coordinator aside), it seems this functionality needs to be provided at a higher layer – and, thereby, are outside scope of this specification. As an example, with ISA SP100.11a, this is application layer traffic. As an aside, this command may include the distribution of keying material (e.g., network wide keys and the-like), so embedding this with the MAC layer implies that key distribution functionality, end-to-end traffic pur sang, would now be single hop traffic. Multi-hop behavior aside, this would also raise a number of other issues, including definition of structure of keying material, including key usage policy fields and, e.g., validity period of keys. This requires more explanation.

The Activate command allows the advertiser to communicate the results of a Join attempt back to the device requested joining. The Activate command can also include the description of slotframe and links for the joining device to communicate with the TSCH-network.

The Activate command is analogous to and is used in place of an Association Response when joining a TSCH network. If links are included in the Activate command frame, those neighbors are used as clock sources as described in 7.5.4.4.2.4.

This command shall only be sent by the advertiser to the device that is currently trying to join.

All devices shall be capable of receiving this command, although an RFD is not required to be capable of transmitting it.

The Activate command shall be formatted as illustrated in Figure 65.j.

Octets: (see 7.2.2.4)	1	2	1	variable	1	0/4/8/16
MHR fields	Command frame Identifier (see Table 123)	Short Address	Number of Slot-frames	Slotframe Info. and Links (for Each Slotframe)	Activate Security Information	MIC

Figure 65.j— Activate command format

7.3.10.3.2 MHR

The Destination Addressing Mode subfield of the Frame Control field shall be set to three (i.e., 64-bit extended addressing). The Source Addressing Mode subfield of the Frame Control field shall be set to two (i.e., 16-bit addressing).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon receipt, and the Acknowledgment Request subfield shall be set to one.

The Source PAN Identifier field shall contain the value of *macPANId*. The Source Address field shall contain the value of *macCoordShortAddress*.

The Destination PAN Identifier field should be set to 0xFFFF. Destination Address field shall contain the extended address of the device requesting to join the network.

The Sequence Number subfield shall be set to the least significant byte of the absolute timeslot number.

7.3.10.3.3 Command Frame Identifier field

The Type field shall be set to Activate (0x0c).

7.3.10.3.4 Short Address field

If the advertiser was not able to join this device to its PAN, the Short Address field shall be set to 0xffff, and the Join Status field shall contain the reason for the failure. If the was advertiser is able to Join the device to its PAN, this field shall contain the short address that the device shall use in its communications on the PAN until it is disconnected.

The device shall use the source PANID of the Activate command as its PANID.

7.3.10.3.5 Number of Links field

The Number of Links field shall be set to the total number of links assigned to new device being activated.

7.3.10.3.6 Link field

Link field shall have the description of link allocated to new device being activated. The format of Link field shall be according to 7.3.10.1.11.

7.3.10.3.7 Activate Security Information field

Editorial note RS:

See previous editorial remark (see §7.3.10.3.1).

The Activate Security field is 1 octet bitmap where b0 = pre-shared symmetric key (as configured by a higher layer), b1=public key. If no bit is set, security is off. All other bits are reserved.

7.3.10.3.8 MIC

This field contains the message integrity check of the Activate command frame.

7.3.11 LL-commands

Editorial note RS:

This clause misses details so as to allow secure operation (see §7.5.6.2, §7.5.8 and §7.6 for details on I/O parameters and other MAC parameters required to properly process outgoing frames and incoming frames – note that all incoming frames pass the incoming frame security processing procedure, no matter whether security is enabled or not).

7.3.11.1 Discover Response command

7.3.11.1.1 General

The Discover Response command contains the configuration parameters that have to be transmitted to the LL_NW PAN coordinator as input for the configuration process in a Low Latency network.

This command shall only be sent by a device that has received a beacon with shortened frame control (see 7.2.2.1) indicating discovery mode as determined through the procedures of the discovery mode (see 7.5.9.2).

All devices shall be capable of transmitting this command, although an RFD is not required to be capable of receiving it.

The command payload of the discover response frame shall be formatted as illustrated in Figure 65.k.

Octets: 1	variable
Command Frame Identifier (see Table 82)	Discovery Parameters

Figure 65.k—Discover response command MAC payload

7.3.11.1.2 7.3.10.1 MHR fields

7.3.11.1.2.1 General

The discover response command can be sent using both MAC command frames (7.2.2.4) or MAC command frames with shortened frame control (**Error! Reference source not found.**).

Comment [W23]: TBD by René

7.3.11.1.2.2 Using MAC command frames

The Frame Type subfield of the Frame Control field shall contain the value that indicates a MAC command frame, as shown in Table 120.

The Source Addressing Mode subfield of the Frame Control field shall be set to 3 (64-bit extended addressing).

The Source Address field shall contain the value of aExtendedAddress.

7.3.11.1.2.3 Using MAC command frames with shortened frame control

In the Shortened Frame Control field, the Frame Type subfield shall contain the value that indicates a MAC frame with a shortened frame control, as shown in Table 120, and the Sub Frame Type subfield shall contain the value that indicates a MAC command frame, as shown in **Error! Reference source not found.**.

Comment [W24]: TBD by René

7.3.11.1.3 7.3.10.2 Command Frame Identifier field

The Command Frame Identifier field contains the value for the discover response command frame as defined in Table 123.

7.3.11.1.4 7.3.10.3 Discovery Parameters field

The Discovery Parameters field contains the configuration parameters that have to be transmitted to the LL_NW PAN coordinator as input for the configuration process. The discovery parameters consist of:

- full MAC address
- required time slot duration, this is defined by the application of the device (e.g. size of sensor data)
- sensor / actuator type indicator

7.3.11.2 Configuration Response Frame

7.3.11.2.1 General

The Configuration Response command contains the configuration parameters that are currently configured at the device as input for the configuration process in a Wireless Factory Automation network.

This command shall only be sent by a device that has received a beacon with shortened frame control (see **Error! Reference source not found.**) indicating configuration mode as determined through the procedures of the configuration mode (see 7.5.9.3).

Comment [W25]: TBD by René

All devices shall be capable of transmitting this command, although an RFD is not required to be capable of receiving it.

The command payload of the Configuration Response Frame shall be formatted as illustrated in Figure 65.l.

Octets : 1	variable
Command Frame Identifier (see Table 123)	Configuration Parameters

Figure 65.l—Configuration response command MAC payload

7.3.11.2.2 MHR fields

The configuration response command can be sent using both MAC command frames (7.2.2.4) or MAC command frames with shortened frame control (see **Error! Reference source not found.**).

Comment [W26]: TBD by René

7.3.11.2.3 Using MAC command frames

7.3.11.2.3.1 General

The Frame Type subfield of the Frame Control field shall contain the value that indicates a MAC command frame, as shown in Table 120.

The Source Addressing Mode subfield of the Frame Control field shall be set to 1 (8-bit short addressing) or 3 (64-bit extended addressing).

The Source Address field shall contain the value of aVeryShortAddress or aExtendedAddress respectively.

7.3.11.2.3.2 Using MAC command frames with shortened frame control

In the Shortened Frame Control field, the Frame Type subfield shall contain the value that indicates a MAC frame with a shortened frame control, as shown in Table 120, and the Sub Frame Type subfield shall contain the value that indicates a MAC command frame, as shown in **Error! Reference source not found.**

Comment [W27]: TBD by René

7.3.11.2.4 Command Frame Identifier field

The Command Frame Identifier field contains the value for the configuration response frame as defined in Table 123.

7.3.11.2.5 Configuration Parameters field

The Configuration Parameters field contains the configuration parameters that are currently configured at the device. The configuration parameters consist of:

- full MAC address
- short MAC address
- required time slot duration, this is defined by the application of the device (e.g. size of sensor data)
- sensor / actuator
- assigned time slots

7.3.11.3 Configuration Request Frame

7.3.11.3.1 General

The Configuration Request command contains the configuration parameters that the receiving device is requested to use during online mode in a Wireless LL-network.

This command shall only be sent by a LL_NW PAN coordinator in response to a received Configuration Response frame of a device during configuration mode.

Only LL_NW PAN coordinators are requested to be capable of transmitting this command, RFD are required to be capable of receiving it.

The command payload of the Configuration Request Frame shall be formatted as illustrated in Figure 65.m.

Octets : 1	variable
Command Frame Identifier (see Table 123)	Configuration Parameters

Figure 65.m—Configuration request command MAC payload

7.3.11.3.2 MHR fields

7.3.11.3.2.1 General

The configuration request command can be sent using both MAC command frames (7.2.2.4) or MAC command frames with shortened frame control (**Error! Reference source not found.**).

Comment [W28]: TBD by René

7.3.11.3.2.2 Using MAC command frames

The Frame Type subfield of the Frame Control field shall contain the value that indicates a MAC command frame, as shown in Table 120.

The Source Addressing Mode subfield of the Frame Control field shall be set to 1 (8-bit short addressing) or 3 (64-bit extended addressing).

The Destination Address field shall contain the value of source address of the corresponding Configuration Response frame.

7.3.11.3.2.3 Using MAC command frames with shortened frame control

In the Shortened Frame Control field, the Frame Type subfield shall contain the value that indicates a MAC frame with a shortened frame control, as shown in Table 120, and the Sub Frame Type subfield shall contain the value that indicates a MAC command frame, as shown in **Error! Reference source not found.**.

Comment [W29]: TBD by René

7.3.11.3.3 Command Frame Identifier field

The Command Frame Identifier field contains the value for the configuration response frame as defined in Table 123.

7.3.11.3.4 Configuration Parameters field

The Configuration Parameters field contains the new configuration parameters that are sent to the device in order to (re-)configure it. The configuration parameters consist of:

- full MAC address
- short MAC address
- transmission channel
- existence of management frames
- time slot duration
- assigned time slots

7.3.11.4 Clear to Send (CTS) Shared Group Frame

7.3.11.4.1 General

The Clear to Send Shared Group command indicates to the devices of the star network that they now may use the time slot for transmitting their own data with a simplified CSMA/CA.

This command shall only be sent by a LL_NW PAN coordinator in a time slot after tSlotTxOwner has been elapsed and the slot owner is not transmitting.

Only LL_NW PAN coordinators are requested to be capable of transmitting this command, devices are required to be capable of receiving it.

The command payload of the Clear to Send Shared Group frame shall be formatted as illustrated in Figure 65.n.

Octets : 1	1
Command Frame Identifier (see Table 123)	Network ID

Figure 65.n—Clear to send shared group command MAC payload

7.3.11.4.2 MHR fields

The clear to send shared group command can be sent using MAC command frames with shortened frame control (**Error! Reference source not found.**).

Comment [W30]: TBD by René

In the Shortened Frame Control field, the Frame Type subfield shall contain the value that indicates a MAC frame with a shortened frame control, as shown in Table 120, and the Sub Frame Type subfield shall contain the value that indicates a MAC command frame, as shown in **Error! Reference source not found.**.

Comment [W31]: TBD by René

7.3.11.4.3 Command Frame Identifier field

The Command Frame Identifier field contains the value for the clear to send shared group frame as defined in Table 123.

7.3.11.4.4 Network ID field

The Network ID field contains an identifier specific to the LL_NW PAN coordinator.

7.3.11.5 Request to Send (RTS) Frame

7.3.11.5.1 General

The Request to Send command may be used by devices to indicates to the LL_NW PAN coordinator and to the other devices of the star network that it wants to transmit data with a simplified CSMA/CA. The request to send frame is transmitted using a simplified CSMA/CA.

This command shall only be sent by a device in a time slot after tSlotOwner has been elapsed and a clear to send shared group frame has been received from the LL_NW PAN coordinator.

Devices are requested to be capable of transmitting and receiving this command.

The command payload of the Request to Send frame shall be formatted as illustrated in Figure 65.o.

Octets : 1	1	1
Command Frame Identifier (see Table123	Short Originator Address	Network ID

Figure 65.o—Request to send command MAC payload

7.3.11.5.2 MHR fields

The request to send command can be sent using MAC command frames with shortened frame control (see **Error! Reference source not found.**).

Comment [W32]: TBD by René

In the Shortened Frame Control field, the Frame Type subfield shall contain the value that indicates a MAC frame with a shortened frame control, as shown in Table 120, and the Sub Frame Type subfield shall contain the value that indicates a MAC command frame, as shown in **Error! Reference source not found.**.

Comment [W33]: TBD by René

7.3.11.5.3 Command Frame Identifier field

The Command Frame Identifier field contains the value for the request to send frame as defined in Table 123.

7.3.11.5.4 Short Originator Address

The Short Originator Address field contains the 1-octet short address of the device sending this request to send frame.

7.3.11.5.5 Network ID field

The Network ID field contains an identifier specific to the LL_NW PAN coordinator. It has to be identical to the Network ID of the corresponding received CTS shared group frame.

7.3.11.6 Clear to Send (CTS) Frame

7.3.11.6.1 General

The Clear to Send command indicates to a specific device of the star network that it may now use the time slot for transmitting its own data with a simplified CSMA/CA.

This command shall only be sent by a LL_NW PAN coordinator in a time slot after tSlotOwner has been elapsed and the slot owner is not transmitting.

Only LL_NW PAN coordinators are requested to be capable of transmitting this command, devices are required to be capable of receiving it.

The command payload of the Clear to Send Shared Group frame shall be formatted as illustrated in Figure 65.p.

Octets : 1	1	1
Command Frame Identifier (see Table 123	Short Destination Address	Network ID

Figure 65.p—Clear to send command MAC payload

7.3.11.6.2 MHR fields

The clear to send command can be sent using MAC command frames with shortened frame control (see **Error! Reference source not found.**).

Comment [W34]: TBD by René

In the Shortened Frame Control field, the Frame Type subfield shall contain the value that indicates a MAC frame with a shortened frame control, as shown in Table 120, and the Sub Frame Type subfield shall contain the value that indicates a MAC command frame, as shown in **Error! Reference source not found.**.

Comment [W35]: TBD by René

7.3.11.6.3 Command Frame Identifier field

The Command Frame Identifier field contains the value for the clear to send shared group frame as defined in Table 123.

7.3.11.6.4 Short Destination Address

The Short Destination Address field contains the 1-octet short address of the device to which this clear to send frame is directed.

7.3.11.6.5 Network ID field

The Network ID field contains an identifier specific to the LL_NW PAN coordinator. It has to be identical to the Network ID of the corresponding received RTS frame.

7.3.12 DSME-commands

7.3.12.1 General

Subclause 7.3 applies in addition.

7.3.12.2 DSME-Association request command

7.3.12.2.1 General

Subclause 7.3.1 applies in addition but the association request command shall be formatted as illustrated in Figure 65.q.

octets: (see 7.2.2.4)	1	1	1
MHR fields	Command Frame Identifier (see Table 123)	Capability Information	Channel Offset

Figure 65.q—Association request command format

7.3.12.2.2 MHR fields

Subclause 7.3.1.1 applies.

7.3.12.2.3 Capability Information field

Subclause 7.3.1.2 applies but the Capability Information field shall be formatted as illustrated in Figure 65.r.

bits: 0	1	2	3	4	5	6	7
Alternate PAN Coordinator	Device Type	Power Source	Receiver On When Idle	Channel Sequence Request	Reserved	Security Capability	Allocate Address

Figure 65.r—DSME-Capability Information field format

The Channel Sequence Request subfield is 1 bit in length and shall be set to one if the PAN runs in both beacon-enabled mode and Channel Hopping mode.

7.3.12.2.4 Channel Offset field

The Channel Offset field is 8 bits in length and shall be set to the offset value of the unassociated device that wished to associate with a PAN, this value is specified by the next higher layer.

7.3.12.3 DSME-Association respond command

7.3.12.3.1 General

Subclause 7.3.2 applies in addition but the DSME-Association respond command shall be formatted as illustrated in Figure 65.s.

octets: (see 7.2.2.4)	1	2	1	1	Variable
MHR fields	Command Frame Identifier (see Table 123)	Short Address	Association Status	Channel Hopping Sequence Length	Channel Hopping Sequence

Figure 65.s—DSME-Association response command format

7.3.12.3.2 MHR fields

Subclause 7.3.2.1 applies.

7.3.12.3.3 Short Address field

Subclause 7.3.2.2 applies.

7.3.12.3.4 Association Status field

Subclause 7.3.2.3 applies.

7.3.12.3.5 Channel Hopping Sequence Length field

The Channel Hopping Sequence Length field is 8 bits in length and shall specify the length of the channel hopping sequence used in the PAN, if the PAN runs in both beacon-enabled mode and Channel Hopping mode.

7.3.12.3.6 Channel Hopping Sequence field

The size of the Channel Hopping Sequence subfield is defined by the Channel Hopping Sequence Length subfield and the Channel Hopping Sequence field specifies the channel hopping sequence used in the PAN, if the PAN runs in both beacon-enabled mode and Channel Hopping mode.

7.3.12.4 DSME handshake command

7.3.12.4.1 General

The DSME handshake command is used by an associated device to request the allocation of a new DSME or the deallocation, reallocation, or change of an existing DSME from the corresponding device. Only devices that have a 16-bit short address less than 0xfffe shall send this command.

This command is mandatory for DSME-devices.

The DSME request command shall be formatted as illustrated in Figure 65.t.

octets: (see 7.2.2.4)	1	Variable
MHR fields	Command Frame Identifier (see Table 82)	DSME Characteristics

Figure 65.t—DSME handshake command format

7.3.12.4.2 DSME-MHR fields

The Destination Addressing Mode and the Source Addressing Mode subfields of the Frame Control field shall both be set to two (i.e., 16-bit short addressing).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledge Request subfield shall be set to one.

The Source PAN Identifier field shall contain the value of macPANId, and the Source Address field shall contain the value of macShortAddress.

For the DSME handshake request command frame, the Destination PAN Identifier field shall contain the identifier of the PAN to which to request for DSME, and the Destination Address field shall contain the address of the Destination device to which the DSME request command frame is being sent. For the DSME handshake reply/notify command frame, the Destination PAN Identifier field shall be set to 0xffff (e.g., broadcast PAN identifier), and the Destination Address field shall be set to 0xffff.

7.3.12.4.3 DSME Characteristics fields

The DSME Characteristics field shall be formatted as illustrated in Figure 65.u.

bit: 0	1-8	9	10-12	13-14	15	16-55	variable
Channel Diversity Mode	DSME Length	DSME Direction	DSME Characteristics Type	DSME Handshake Type	Prioritized Channel Access	DSME Descriptor	DSME ABT Specification

Figure 65.u—DSME Characteristics field format

The ChannelDiversityMode subfield is 1 bit in length and shall be set to one of the non-reserved values listed in Table 84.a.

Table 84.a—Values of the Channel Diversity Mode subfield

Channel Diversity Mode value b_0	Description
0	Channel Adaptation mode
1	Channel Hopping mode

The DSME Length subfield is 8 bits in length and shall contain the number of superframe slots being requested for the DSME.

The DSME Direction subfield is 1 bit in length and shall indicate the direction of the DSME handshake command.

The DSME Characteristics Type subfield is 3 bits in length and shall be set to one of the non-reserved values listed in Table 84.b.

Table 84.b—Values of the DSME Characteristics Type subfield

DSME Characteristics Type value $b_2b_1b_0$	Description
000	Deallocation
001	Allocation
010	Reallocation
011	Duplicated Allocation Notification
100	Robust DSME Allocation
101	Reduce
110	Restart
111	Reserved

The DSME Handshake Type subfield is 2 bits in length and shall be set to one of the non-reserved values listed in Table 84.c.

Table 84.c—Values of the DSME Handshake Type subfield

DSME Handshake Type value b_1b_0	Description
00	Request
01	Reply
10	Notify
11	Reserved

The Prioritized Channel Access subfield is 1 bit in length and shall be set to one if DSME should be reserved as high priority, or set to zero if DSME should be reserved as low priority. When the DSME request command is used in the DSME change procedure, the Prioritized Channel Access shall be set according to the original DSME.

7.3.12.4.4 DSME Descriptor field

The DSME Descriptor field is 5 octets in length and shall be formatted as illustrated in Figure 65.v.

bit: 0-15	16-31	32-39
Destination Address	DSME slot identifier	DSME Length

Figure 65.v—Format of the DSME Descriptor field

The Device Short Address subfield is 2 octets in length and shall contain the short address of the device for which the DSME allocate/ deallocate/ reallocate or change.

The DSME slot identifier subfield is 2 octets in length and shall contain the channel number (1 octet in length) and the beginning time slot number (1 octet in length) of the DSME that is to be allocated or deallocated.

The DSME Length subfield is 1 octet in length and shall contain the number of superframe slots being requested for the DSME.

7.3.12.4.5 DSME ABT Specification field

The DSME ABT Specification field shall be formatted as illustrated in Figure 65.w.

bit: 0-3	4-19	variable
DSME ABT sub-block length	DSME ABT sub-block index	DSME ABT sub-block

Figure 65.w—Format of the DSME ABT Specification field

The DSME ABT sub-block length subfield is 4 bits in length and shall contain the length of the DSME ABT sub-block in units defined in Figure 65.x.

The DSME ABT sub-block index subfield is 2 octets in length and shall indicate the beginning of the ABT Sub-block in the entire ABT as illustrated in Figure 65.x.

The DSME ABT sub-block shall contain the sub-block of the Allocation Bitmap Table as illustrated in Figure 65.x.

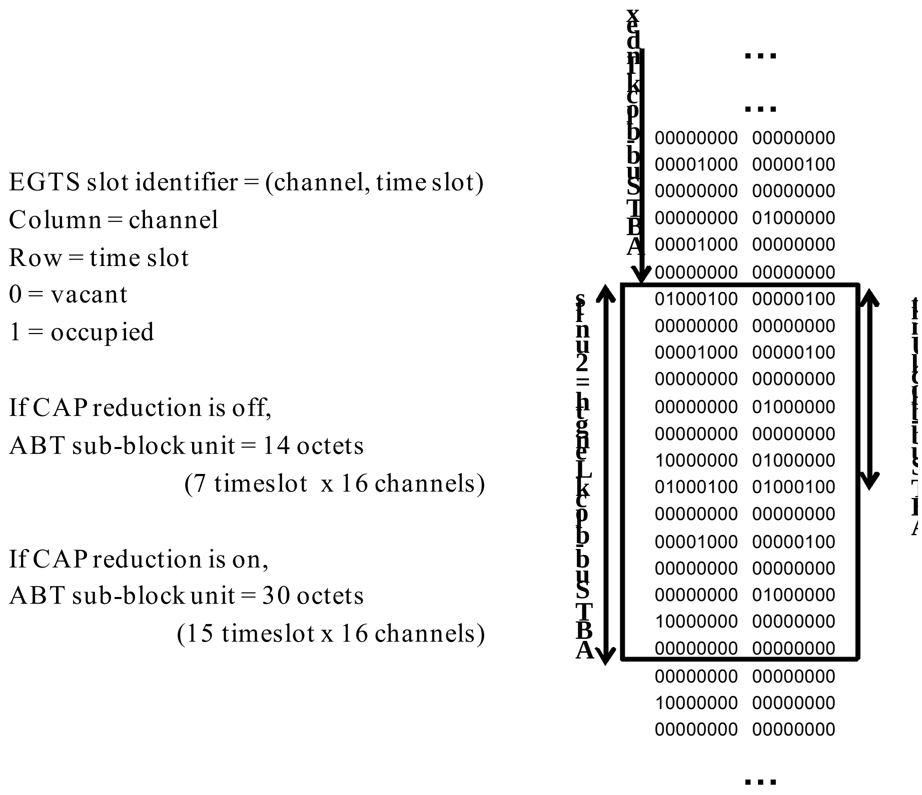


Figure 65.x—ABT Sub-block

When the channel hopping mode is used to obtain channel diversity gain, i.e. ChannelDiversityMode bit shall be set to ‘1’, Timeslot Allocation Bitmap (TAB) is employed instead of DSME ABT for handshaking. TAB represents the usage of corresponding DSME slots, a bit shall be set to '1' if the corresponding slot is allocated to transmit or receive, or '0' if the slot is available. Similarly in channel adaptation, DSME ABT sub-block Index and DSME ABT Sub-block length shall indicate the start position and the length of TAB Sub-block. Thus, only sub block of whole TAB is exchanged for scheduling. This is illustrated in Figure 65.y.

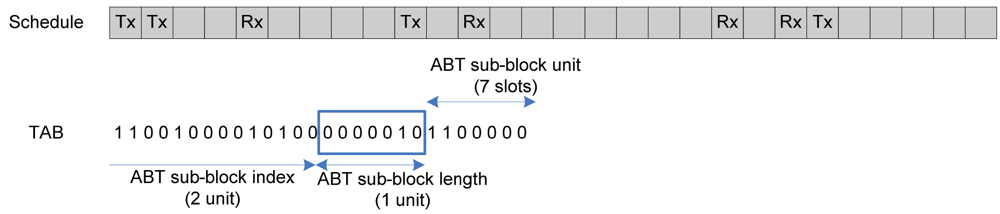


Figure 65.y—TAB Sub-block

7.3.12.5 DSME information request command

The DSME information request command is used by a source device that is requesting the timestamp and the DSME parameters from the destination device.

The DSME information request command shall be formatted as illustrated in Figure 65.z.

This command is mandatory for DSME-devices.

octets: (see 7.2.2.4)	1
MHR fields	Command Frame Identifier (see Table 82)

Figure 65.z— DSME information request command format

The Destination Addressing Mode and the Source Addressing Mode subfields of the Frame Control field shall both be set to two (e.g., 16-bit short addressing).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledgment Request subfield of the Frame Control field shall be set to one.

The Source PAN Identifier subfield shall contain the value of macPANId, and the Source Address subfield shall contain the value of macShortAddress.

The Destination PAN Identifier subfield shall contain the identifier of the PAN to which to request for DSME information, and the Destination Address subfield shall contain the address of the Destination device to which the DSME information request command frame is being sent.

7.3.12.6 DSME information reply command

The DSME information reply command frame is used by a destination device that is replying the timestamp and the DSME information to the source device.

The DSME information reply command frame shall be formatted as illustrated in Figure 65.aa.

This command is mandatory for DSME-devices.

octets: (see 7.2.2.4)	1	3	variable
MHR fields	Command Frame Identifier (see	Timestamp	DSME Characteristics

octets: (see 7.2.2.4)	1	3	variable
	Table 123)		(see 7.3.12.4.3)

Figure 65.aa—DSME information reply command format

The Destination Addressing Mode and the Source Addressing Mode subfields of the Frame Control field shall both be set to two (e.g., 16-bit short addressing).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledgment Request subfield of the Frame Control field shall be set to one.

The Source PAN Identifier subfield shall contain the value of macPANId, and the Source Address subfield shall contain the value of macShortAddress.

The Destination PAN Identifier subfield shall contain the identifier of the PAN to which to reply the DSME information, and the Destination Address subfield shall contain the address of the Destination device to which request the DSME information.

7.3.12.7 Beacon allocation notification command

The beacon allocation notification command is used by a device that selects vacant Superframe Duration (SD) for using transmission of beacon frame.

The beacon allocation notification command shall be formatted as illustrated in Figure 65.bb.

This command is mandatory for DSME-devices.

octets: (see 7.2.2.4)	1	2
MHR fields	Command Frame Identifier (see Table 123)	Allocation Beacon SD Index

Figure 65.bb—Beacon allocation notification command format

The Destination Addressing Mode and Source Addressing Mode subfields of the Frame Control field shall both be set to two (e.g., 16-bit short addressing).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledgment Request subfield shall be set to zero.

The PAN ID Compression subfield of the Frame Control field shall be set to one. In accordance with this value of the PAN ID Compression subfield, the Destination PAN Identifier field shall contain the value of macPANId, while the Source PAN Identifier field shall be omitted. The Destination Address field shall be set to 0xffff. The Source Address field shall contain the value of macShortAddress.

The Allocation Beacon SD Index field is 2 octets in length and shall contain the allocating SD index number for beacon frame which is allocated to the Source device.

7.3.12.8 Beacon collision notification command

The beacon collision notification command is used by a device that detects the collision of beacon frame.

The beacon collision notification command shall be formatted as illustrated in Figure 65.cc.

This command is mandatory for DSME-devices.

octets: (see 7.2.2.4)	1	2
MHR fields	Command Frame Identifier (see	Collision SD Index

octets: (see 7.2.2.4)	1	2
	Table 123)	

Figure 65.cc—Beacon collision notification command format

The Destination Addressing Mode and Source Addressing Mode subfields of the Frame Control field shall both be set to two (e.g., 16-bit short addressing).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledgment Request subfield shall be set to one.

The PAN ID Compression subfield of the Frame Control field shall be set to one. In accordance with this value of the PAN ID Compression subfield, the Destination PAN Identifier field shall contain the value of macPANId, while the Source PAN Identifier field shall be omitted. The Destination Address field shall be set to 0xffff. The Source Address field shall contain the value of macShortAddress.

The Conflict SD Index field is 2 octets in length and shall contain the SD index number of collision beacon frame.

7.3.12.9 Link status report command

7.3.12.9.1 General

The link status report command allows a source device to report its link quality parameters to a destination device.

This command shall only be sent by an associated device that wishes to report the link quality. All devices shall be capable of transmitting this command, although an RFD is not required to be capable of receiving it.

The link status report command shall be formatted as illustrated in Figure 65.dd.

octets: (see 7.2.2.4)	1	variable
MHR fields	Command Frame Identifier (see Table 123)	Link Status Specification

Figure 65.dd—Link status report command format

The Link Status Specification fields shall be formatted as illustrated in Figure 65.ee.

octets: 1	variable
Link Status Descriptor Count	Link Status List

Figure 65.ee—Link status specification field format

7.3.12.9.2 MHR fields

The Source Addressing Mode subfield of the Frame Control field shall be set to two (16-bit extended addressing), and the Destination Addressing Mode subfield shall be set to the same mode as the destination device to which the status report command refers.

The Frame Pending subfield of the Frame Control field shall be set to one, and the Acknowledgment Request subfield shall be set to one.

The Destination PAN Identifier field shall contain the identifier of the PAN of the destination device to which to report the link status. The Destination Address field shall contain the address of the destination device to which the status report command is being sent.

The Source PAN Identifier field shall contain the value of macPANId, and the Source Address field shall contain the value of macShortAddress.

The Frame Type subfield in MHR shall be set to 100 and the Frame Version subfield should be set to 0x10.

7.3.12.9.3 Link Status Descriptor Count field

The Link Status Descriptor Count field is 1 octet in length and specifies the number of the Link Status Descriptors in the link status List field.

7.3.12.9.4 Link Status List fields

The size of the Link Status List fields is defined by the values specified in the Link Status Descriptor Count field and contain the list of Link Status Descriptors that represents the link status of each link. The Link Status Descriptor shall be formatted as illustrated in Figure 65.ff.

octets: 1	1	1	1
Channel	avgLQI	avgRSSI	Reserved

Figure 65.ff—Link status Descriptor format

The Channel subfield is 1 octet in length and specifies the channel index reported by the source device.

The avgLQI subfield is 1 octet in length and contains the average received LQI of the channel specified in Channel subfield within LinkStatusStatisticPeriod symbols.

The avgRSSI subfield is 1 octet in length and contains the average received signal power by ED measurement during a period of LinkStatusStatisticPeriod symbols. The avgRSSI measurement shall be performed for each received packet, and the use of the avgRSSI result by the next higher layer is not specified in this standard.

7.3.12.10 Multi-channel beacon request command

The multi-channel beacon request command is used by a device that is performing asymmetric multi-channel active scan.

The multi-channel beacon request command shall be formatted as illustrated in Figure 65.gg. This command is mandatory for DSME-devices.

octets: (see 7.2.2.4)	1	4
MHR fields	Command Frame Identifier (see Table 123)	Scan Channels

Figure 65.gg—Multi-channel beacon request command format

The Destination Addressing Mode subfield of the Frame Control field shall be set to two (e.g., 16-bit short addressing), and the Source Addressing Mode subfield shall be set to zero (e.g., source addressing information not present).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledgment Request subfield shall be set to zero.

The Destination PAN Identifier subfield shall contain the broadcast PAN identifier (i.e., 0xffff). The Destination Address subfield shall contain the broadcast short address (i.e., 0xffff).

The Scan Channels subfield is represented in 27-bit bitmaps. The 27 bits (b0, b1,... b26) indicate which channels are to be scanned (1 = scan, 0 = do not scan) for each of the 27 channels supported by the ChannelPage parameter.

7.3.12.11 Multi-channel hello command

7.3.12.11.1 General

The multi-channel hello command is used to inform neighboring devices of the device's designated channel.

The multi-channel beacon request command shall be formatted as illustrated in Figure 65.hh. This command is mandatory for DSME-devices.

octets: (see 7.2.2.4)	1	1
MHR fields	Command Frame Identifier (see Table 123)	Hello Specification

Figure 65.hh—Multi-channel hello command format

7.3.12.11.2 MHR fields

The Destination Addressing Mode subfield of the Frame Control field shall be set to two (e.g., 16-bit short addressing), and the Source Addressing Mode subfield shall be set to zero (e.g., source addressing information not present).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledgment Request subfield shall be set to zero.

The Destination PAN Identifier subfield shall contain the broadcast PAN identifier (i.e., 0xffff). The Destination Address subfield shall contain the broadcast short address (i.e., 0xffff).

7.3.12.11.3 Hello Specification field

The Hello Specification field shall be formatted as illustrated in Figure 65.ii.

bits: 5	1	2
Designated Channel Index	Hello Reply Request	Reserved

Figure 65.ii—Hello specification field format

The Designated Channel Index subfield is 5 bits in length and shall contain the designated logical channel index number of the device.

The Hello Reply Request subfield is 1 bit in length and shall indicate whether the multi-channel hello command needs a hello reply. If the Hello Reply Request bit is set to'1', the device shall transmit a hello reply upon receiving the hello command.

7.3.12.11.4 Multi-channel hello reply command

The multi-channel hello reply command is used to

TBD ...

The multi-channel beacon request command shall be formatted as illustrated in Figure 65Q.

TBD ...

Comment [youcy36]: The format and description of the hello reply command should be added

7.3.12.12 Channel probe command

7.3.12.12.1 General

The channel probe command is used to check the link quality of the specified channel.

The channel probe command shall be formatted as illustrated in Figure 65.jj. This command is mandatory for DSME-device.

octets: (see 7.2.2.4)	1	2
MHR fields	Command Frame Identifier (see Table 123)	Channel Probe Specification

Figure 65.jj—Channel probe command format

7.3.12.12.2 MHR fields

The Source Addressing Mode subfield of the Frame Control field shall be set to two (16-bit extended addressing), and the Destination Addressing Mode subfield shall be set to the same mode as the destination device to which the channel probe command refers.

The Frame Pending subfield of the Frame Control field shall be set to zero, and the Acknowledgment Request subfield shall be set to one.

The Destination PAN Identifier field shall contain the identifier of the PAN of the destination device to which to check the link quality. The Destination Address field shall contain the address of the destination device to which the channel probe command is being sent.

The Source PAN Identifier field shall contain the value of macPANId, and the Source Address field shall contain the value of macShortAddress.

7.3.12.12.3 Channel Probe Specification field

The Channel Probe Specification field shall be formatted as illustrated in Figure 65.kk.

bits: 2	5	5	4
Channel Probe Subtype	Designated Channel	Probe Channel	Reserved

Figure 65.kk—Channel Probe specification format

The Channel Probe Subtype subfield is 2 bits in length and shall be set to one of the non-reserved values listed in Table 84.d.

Table 84.d—Values of the Channel Probe Subtype subfield

Channel Probe subtype value b_1b_0	Description
00	Request
01	Reply
10	Probe
11	Reserved

The Designated Channel subfield is 5 bits in length and indicates the originator’s designated channel.

The Probe Channel subfield is 5 bits in length and indicates the channel that needs to be probed.

7.3.13 SUN-commands

7.3.13.1 SUN-Enhanced Beacon request command

7.3.13.1.1 General

Editorial note RS:

Incoming frame security processing requires the extended address of the device originating the frame.

The language on p. 93, ll. 2-8, seems to be conflicting here. This requires more explanation.

The enhanced beacon request command is intended to be used by a SUN-device to locate a subset of all coordinators within its POS during an active scan.

This command is optional for an FFD and an RFD.

The enhanced beacon request command shall be formatted as illustrated in Figure 65.ll.

Octets:7	1	1	0/1	0/1	Variable
MHR fields	**Command Frame Identifier (see Table 123)**	**Request Field**	**Link Quality**	**Percent filter**	**Extended Payload**

Figure 65.ll—SUN-Enhanced Beacon request command

The Destination Addressing Mode subfield of the Frame Control field shall be set to two (i.e., 16-bit short addressing), and the Source Addressing Mode subfield shall be set to zero (i.e., source addressing information not present).

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception.

The Acknowledgment Request subfield and Security Enabled subfield shall be set to zero. If the enhanced beacon request is being sent on a particular PAN Identifier that is not the broadcast PAN identifier the Security Enable subfield may be set to 1, otherwise it shall be set to 0.

The Destination PAN Identifier field shall contain the any appropriate PAN identifier. If the broadcast PAN identifier is used, any device may respond however if a specific PAN identifier is used only devices using that PAN identifier will respond to the enhanced beacon request.

The Destination Address field shall contain the broadcast short address (i.e., 0xffff).

7.3.13.1.2 Request Field

7.3.13.1.2.1 General

The request field is a 1 octet field indicating what optional request discriminators are included in the Enhanced Beacon Request Command Payload. The Request Field is as shown in Table 84.e.

Table 84.e—Request field coding

Bit	SubField
0	Permit Joining On
1	LinkQuality
2	Percent filter
3 - 6	Reserved
7	Extended Payload

For any bits set in the request field the following is done.

7.3.13.1.2.2 Permit Joining On

Only devices with permit joining on shall respond to the enhanced beacon request.

7.3.13.1.2.3 LinkQuality Level

Following the Request field the enhanced beacon request will include a field containing a value for LinkQuality. The device will respond to the enhanced beacon request if the MCPS-DATA.indication indicates a mpduLinkQuality equal or lower than this value (where lower values represent higher quality links).

7.3.13.1.2.4 Percent filter

Following the Request Field a byte will be included of a scaled value from 0x00 to 0x64 representing zero to 100 percent probability for a given device to respond to the enhanced beacon request. The device will then randomly determine if it is to respond to the enhanced beacon request based on meeting this probability. For example if the probability is set to 10% then 1 of 10 devices would randomly be expected to respond.

7.3.13.1.2.5 Extended Payload

If this bit is set the extended payload field shall be included in the enhanced beacon request. The extended payload should be provided as the MSDU in the MCPS-DATA.indication to the local SSCS entity in which case the SSCS entity would be responsible for handling beacon responses based on data in this extended payload.

7.3.14 LE-commands

7.3.14.1 RIT data request command

7.3.14.1.1 General

The RIT data request command allows a device to request data from its neighboring devices in the RIT mode. This command shall only be sent and received in the RIT mode (macRitPeriod : non zero value). This command is optional and applicable for FFD only. The RIT data request command shall be formatted as illustrated in Figure 65.mm.

Octets : (see 7.2.2.4)	1	0 or 4
MHR fields	Command Frame Identifier (see Table 82)	Optional command payload

Figure 65.mm— Format of RIT data request command

7.3.14.1.2 MHR fields

The Frame Pending subfield of the Frame Control field shall be set to zero and ignored upon reception, and the Acknowledgement Request subfield shall also be set to zero. All other subfields shall be set appropriately according to the intended use of the command frame.

7.3.14.2 Command Frame Identifier

See Table 82.

7.3.14.3 Optional Command Payload

The command payload can be either 0 or 4 octets. The 4-octet payload is defined in Figure 65.nn:

Octets : 1	1	2
Time to 1st Listen (T0)	Number of Repeat (N)	Repeat Listen Interval (T)

Figure 65.nn— Format of Optional Command Payload

Time to 1st Listen (T0) and Repeat Listen Interval (T) are in the same unit as macRitPeriod. Number of Repeat Listen (N) is constrained by T0 + N*T < macRitPeriod.

7.3.14.4 Wake-up frame

7.3.14.4.1 General

The wake-up frame shall be formatted as illustrated in Figure 65.oo

octets: 1	1	4	0/5/6/10/14	2	variable	0/2
sFCF	Sequence number	Addressing fields	Auxiliary Security Header	RZ time	Frame payload	FCS
MHR				MAC payload		MFR

Figure 65.oo—Format of Optional Command Payload

The order of the fields of the wake-up frame shall conform to the order of the general MAC frame as illustrated in Figure 43.

7.3.14.4.2 Wake-up frame MHR fields

The MHR for a wake-up frame shall contain the Short Frame Control Field, the Sequence Number Field, the Destination PAN Id, and the Destination Address field.

In the Short Frame Control Field, the Frame Type shall contain the value that indicates a wake-up frame, as shown in Table {xxx}. If protection is used for the wake-up frame, the Security Enabled subfield shall be set to one. The Frame Version subfield of the Short Frame Control Field shall be set to the value indicating an 802.15.4-2006, as shown in Table {xxx}.

Comment [W37]: According the agreement in LA at 2010-01-21 shall be this term revised. (It is also not a complete sentence.

Comment [W38]: TBD by René.

The Sequence Number field shall be set to the current value of *macDSN*.

The addressing fields shall comprise only the destination addressing fields. The Destination PAN Identifier and the Destination Address fields shall contain the PAN Identifier and the short address of the device receiving the wake-up frame.

7.3.14.4.3 Wake-up frame RZ time field

Editorial note RS:

Keep as is in 09/604r3.

Comment [W39]: TBD by René.

7.3.14.4.4 Wake-up frame payload field

The wake-up frame payload field is an optional field specified to be transmitted in the wake-up frame.

NOTE—Inclusion of this field is determined via inspection of the Frame Length subfield of the PHY header field of the PPDU and the other frame fields.

7.3.14.5 New optional payload field

Editorial note RS:

Change "optional MHR payload" towards "optional payload field", so as to make this consistent with treatment of payload fields with secured ACK, etc.

For frames where the payload field has a fixed length without the presence of the CSL-piggy back field, the CSL-sync bit can be derived from the length of the frame (as contained in the PHY header) and the length of the other subfields of this frame. Hence, in those cases the CSL-synch bit is not necessary. For other frames, inclusion of this optional payload field needs to be indicated via a CSL-sync bit contained in the MHR.

To be discussed with Wei Hong:

For data frames, command frames, and beacon frames, the payload field may be split into at least two distinct subfields, as indicated by the Split Payload Field of the full Frame Control Field contained in the MHR. If the Split Payload Field is set, the specific decomposition of the payload field is indicated in Frame Payload Header field.

For Coordinated Sampled Listening (CSL), the format of the frame payload field is illustrated in Fig. {xxx}.

octets:1	2	2	variable
"CSL value"	CSL Phase	CSL Period	Frame payload
MAC Payload			

For EGTS, the format of the frame payload field is illustrated in Fig. {xxx}:

variable	1	variable	variable
Beacon header	"EGTS value"	EGTS beacon header	Frame payload
MAC Payload			

Note RS: right now, Myung Lee may wish to indicate the EGTS value in the superframe specification (bit b13, which is currently reserved). If indicated as above, the EGTS frame cannot contain CSL synch info at the same time, unless one defines a separate Frame Payload Header identifier to indicate this combination.

7.3.14.6 Secure acknowledgement frame

TBD Editorial note RS:

This frame should be aligned with the secured acknowledgement frame (see §7.2.3).

7.3.15 RFID-commands

7.3.15.1 RFID Blink commands

7.3.15.1.1 General

The RFID Blink commands indicate to the PAN coordinator the identifier of an RFID which is the source.

This command shall only be sent or processed by a device with both macRFIDcapable and macRFIDenabled set to TRUE.

Only PAN coordinators are requested to be capable of recieving this command, devices are required to be capable of transmitting it.

The command payload of the RFID Blink frames shall be formatted as illustrated in Figure 65.pp.

Octets: 1	1	2	8
Command Frame Identifier (see Table 123)	Sequence Number	Destination PAN Identifier	Source MAC Address

Figure 65.pp—RFID Blink commands MAC payload

7.3.15.1.2 MHR fields

The RFID Blink commands can be sent using MAC command frames with shortened frame control (**Error! Reference source not found.**).

Comment [W40]: TBD by René

In the Shortened Frame Control field, the Frame Type subfield shall contain the value that indicates a MAC frame with a shortened frame control, as shown in Table 79, and the Sub Frame Type subfield shall contain the value that indicates a MAC command frame, as shown in **Error! Reference source not found.**

Comment [W41]: TBD by René

7.3.15.1.3 Command Frame Identifier field

The Command Frame Identifier field contains one of the values for the RFID Blink frames as defined in Table 55.

The different Command Frame Identifiers of the RFID Blink command frames indicate the existence or non-existence of the Destination PAN Identifier field and the Source MAC Address field in the RFID Blink frame as shown in Figure 65.qq.

	—	64 bit Source
—	0x40	0x43
Dest PAN ID	0x60	0x63

Figure 65.qq—Existence of addresses in different RFID Blink command frames

The visual representation of the different RFID Blink command frames is shown in Figure 65.rr.

Bits: 0	1	2 - 3	4	5	6 - 7
0	**1** *RFID Blink*	**00 / 10** *(no) Destination PAN Identifier*	**0**	**0**	**00 / 11** *(no) 64-bit Source MAC Address*

Figure 65.rr—Visual representation of RFID Blink command frames

Note, that bit 1 is only an indication that this command frame might be an RFID Blink frame, but it is not exclusively set for RFID Blink frames. There might be other values for command frame identifiers where b1 of the Command Frame Identifier is 1 and the command frame is not an RFID Blink frame.

7.3.15.1.4 Sequence Number field

The Sequence Number field is 1 octet in length and specifies the sequence identifier for the frame. The Sequence Number field shall specify a data sequence number that is used to match an acknowledgment frame to the RFID Blink frame.

7.3.15.1.5 Destination PAN Identifier field

The Destination PAN Identifier field is 2 octets in length and specifies the unique PAN identifier of the intended recipient of the frame. A value of 0xffff in this field shall represent the broadcast PAN identifier, which shall be accepted as a valid PAN identifier by all devices currently listening to the channel.

The Destination PAN Identifier field is only present in RFID Blink command frames where the bits 2 and 3 of the Command Frame Identifier correspond to “10”.

7.3.15.1.6 Source MAC Address field

The Source MAC Address field is 8 octets in length and specifies the 64-bit address of the RFID which is the source of the frame.

The Source MAC Address field is only present in RFID Blink command frames where the bits 6 and 7 of the Command Frame Identifier correspond to “11”.

7.4 MAC constants and PIB attributes

7.4.1 MAC constants

7.4.2 MAC PIB attributes

Editorial note RS:

Adapt the formula for the macACKWaitDuration parameter (Equation (13)), so as

• Same outcome with instantiation of unsecured 802.15.4-2006 style 5-octet ACK;

• Make formula independent of frame size variations of new ACK with payload (one way to realize this would be for the originating device to time the received frame by recognizing the incoming frame as acknowledgment frame, in anticipation of proper and successful remainder of incoming processing.

Again, details to be looked up in the minutes of that meeting.

Insert after the heading of 7.4.2 the following subclause header.

7.4.2.1 General

Insert before 7.5 the following subclauses.

7.4.2.2 General MAC PIB attributes for functional organization

Table 86.a provides the General MAC PIB attributes for functional organization

Table 86.a— General MAC PIB attributes for functional organization

Attribute	Identifier	Type	Range	Description	Default
macPAcapable	0x99	Boolean	TRUE or FALSE	The device is capable of functionality specific to process automation	
macLLcapable	0x9a	Boolean	TRUE or FALSE	The device is capable of functionality specific to Low Latency Networks	
macCMcapable	0x9b	Boolean	TRUE or FALSE	The device is capable of functionality specific to CM	
macLEcapable	0x9c	Boolean	TRUE or FALSE	The device is capable of functionality specific to Low Energy	
macSUNcapable	0x9d	Boolean	TRUE or FALSE	The device is capable of functionality specific to SUN	
macRFIDcapable	0x9e	Boolean	TRUE or FALSE	The device is capable of functionality specific to RFID	
macPAenabled	0x9f	Boolean	TRUE or FALSE	The device is using functionality specific to process automation	
macLLenabled	0xa0	Boolean	TRUE or FALSE	The device is using functionality specific to Low Latency Networks	
macCMenabled	0xa1	Boolean	TRUE or	The device is using	

Attribute	Identifier	Type	Range	Description	Default
			FALSE	functionality specific to CM	
macLEenabled	0xa2	Boolean	TRUE or FALSE	The device is using functionality specific to Low Energy	
macSUNenabled	0xa3	Boolean	TRUE or FALSE	The device is using functionality specific to SUN	
macRFIDenabled	0xa4	Boolean	TRUE or FALSE	The device is using functionality specific to RFID	

7.4.2.3 TSCH-specific MAC PIB attributes

7.4.2.3.1 General

Subclause 7.4.2.1 applies except that the attributes macMinBE and macMaxBE in Table 86 shall be according to Table 86.b and an additional attribute macDisconnectTime is required, see Table 86.b.

Table 86.b—TSCH-specific MAC PIB attributes

Attribute	Identifier	Type	Range	Description	Default
macMinBE	0x4f	Integer	0–macMaxBE	The minimum value of the backoff exponent (BE) in the CSMA-CA algorithm or the TSCH-CA algorithm. See 7.5.1.4 for a detailed explanation of the backoff exponent. See 7.5.4.2 for use of the backoff exponent in TSCH-mode.	3/1
macMaxBE	0x57	Integer	3–8	The maximum value of the backoff exponent (BE) in the CSMA-CA algorithm or the TSCH- CA algorithm. See 7.5.1.4 for a detailed explanation of the backoff exponent. See 7.5.4.2 for use of the backoff exponent in TSCH-mode.	5/7
macDisconnectTime	0x8f	Integer	0x00-0xFFFF	Time to send out Disassociate frames before disconnecting.	

7.4.2.3.2 TSCH-MAC PIB attributes for macSlotframeTable

The attributes contained in the MAC PIB for macSlotframeTable are presented in Table 86.c.

Table 86.c—TSCH-MAC PIB attributes for macSlotframeTable

Attribute	Identifier	Type	Range	Description	Default
slotframeId	0x64	Integer	0x00-0xFF	Identifier of the slotframe	
slotframeSize	0x65	Integer	0x0000-0xFFFF	Number of timeslots in the slotframe	
activeFlag	0x66	Boolean	0x0-0x1	Flag indicating if the slotframe is currently activated	
channelPage	0x67	Integer	0x00-0x1F	Channel Page of channels used in this slotframe	
channelMap	0x68	Bitmap		Bitmap of active channels.	

7.4.2.3.3 TSCH-MAC PIB attributes for macLinkTable

The attributes contained in the MAC PIB for macLinkTable are presented in Table 86.d.

Table 86.d— TSCH-MAC PIB attributes for macLinkTable

Attribute	Identifier	Type	Range	Description	Default
linkId	0x69	Integer	0x00-0xFF	Identifier of Link	
linkOption	0x6a	Bitmap	0x00-0x7	Flags indicating whether the link is used for transmit, receive, or shared transmissions:	
linkType	0x6b	Integer	0x00-0x2	Enumeration indicating the type of link: Normal or Advertising	
slotframeId	0x6c	Integer	0x00-0xFF	Identifier of Slotframe to which this link belongs	
nodeAddress	0x6d	IEEE address	16 bit address	Address of the node connected to this link	
timeslot	0x6e	Integer	0x0000-0xFFFF	Timeslot for this link	
channelOffset	0x6f		0x00-0xFF	Channel offset for this link	

7.4.2.3.4 TSCH-MAC PIB attributes for macTimeslotTemplate

The attributes contained in the MAC PIB for macTimeslotTemplate are presented in Table 86.e.

Table 86.e—TSCH-MAC PIB attributes for macTimeslotTemplate

Attribute	Identifier	Type	Range	Description	Default
Timeslot Template Id	0x70	Integer	0x0-0xF	Identifier of Timeslot Template	
TsCCAOffset	0x71	Integer	0x0000-0xFFFF	The time between the beginning of timeslot and start of CCA operation	
TsCCA	0x72	Integer	0x0000-0xFFFF	Duration of CCA	
TsTxOffset	0x73	Integer	0x0000-0xFFFF	The time between the beginning of the timeslot and the start of packet transmission	
TsRxOffset	0x74	Integer	0x0000-0xFFFF	Beginning of the timeslot to when the receiver must be listening	
TsRxAckDelay	0x75	Integer	0x0000-0xFFFF	End of packet to when the transmitter must listen for Acknowledgment	
TsTxAckDelay	0x76	Integer	0x0000-0xFFFF	End of packet to start of Acknowledgment	
TsRxWait	0x77	Integer	0x0000-0xFFFF	The time to wait for start of packet	
TsAckWait	0x78	Integer	0x0000-0xFFFF	The minimum time to wait for start of an Acknowledgment	
TsRxTx	0x79	Integer	0x0000-0xFFFF	Transmit to Receive turnaround (12 symbols)	
TsMaxAck	0x7a	Integer	0x0000-0xFFFF	Transmission time to send Acknowledgment	
TsMaxTx	0x7b	Integer	0x0000-0xFFFF	Transmission time to send the maximum length packet (133 bytes)	

7.4.2.3.5 TSCH-MAC PIB attributes for macHoppingSequence

To be jointly defined with Channel Hopping/Channel Diversity subgroup.

The attributes contained in the MAC PIB for macHoppingSequence are presented in Table 86.d.

Table 86.f— TSCH-MAC PIB attributes for macHoppingSequence

Attribute	Identifier	Type	Range	Description	Default

7.4.2.4 LL-specific MAC PIB attributes

Subclause 7.4.2.1 applies and additional attributes are required, see Table 86.b.

Table 86.g—LL-specific MAC PIB attributes

Attribute	Identifier	Type	Range	Description	Default
macFAlowLatencyPAN	0x7c	Boolean	TRUE or FALSE	Indicates that the PAN is using the mechanisms as described in 5.3.3, 5.5.1.2, and related clauses.	Set by configuration
macFAnumTimeSlots	0x7d	Integer	0 … 254	Number of time slots within superframe excluding time slot for beacon frame	20
macFAnumSensorTS	0x7e	Integer	0 … macFAnum-TimeSlots	Number of sensor time slots within superframe for unidirectional communication (uplink)	20
macFAnumRetransmitTS	0x7f	Integer	0 … macFAnum-SensorTS/2	Number of sensor time slots reserved for retransmission (see 5.5.1.2 and 7.5.1.6.1)	0
macFAnumActuatorTS	0x80	Integer	0 … macFAnum-TimeSlots	Number of actuator time slots within superframe for bidirectional communication	0
macFAmgmtTS	0x81	Boolean	TRUE or FALSE	Indicates existence of management time slots in Online Mode	FALSE
macFAlowLatencyNWid	0x82	Integer	0x00–0xff	The 8-bit identifier of the LLNW on which the device is operating. If this value is 0xff, the device is not associated.	0xff

7.4.2.5 DSME-specific MAC PIB attributes

Subclause 7.4.2.1 applies and additional attributes are required, see Table 86.h.

Table 86.h—DSME-specific MAC PIB attributes

Attribute	Identifier	Type	Range	Description	Default
Channel Index	0x83	Integer	0-31	Specifies the Channel index of the channel‘s link status reported by the source device.	
avgLQI	0x84	Integer	0x00-0xff	A characterization of the link quality between a source device and a destination device on the channel defined by Channel Index, the measurement shall be performed for each received packet during a period of *LinkStatusStatisticPeriod*.	
avgRSSI	0x85	Integer	0-255	Average RSSI.	
LinkStatisticPeriod	0x86	Interger	0x0000-0xffff	The time interval between two times of link status statistics	16
macLowEnergySuperframe Supported	0x87	Boolean	TRUE or FALSE	Indication of whether the low energy superframe is operational or not. If this attribute is TRUE, the coordinator shall not transmit beacon frames regardless of BO value. This attribute shall be set to FALSE if the device is aware of the existence of allocated GTS or DSME in its two-hop neighborhood.	Implementation Specific
macFAuseGACKmechanis m	0x88	Integer	0x00–0x01	This flag indicates if the coordinator is currently using the group acknowledge mechaisn for GTS frame receiptions.	0x00

7.4.2.6 LE-specific MAC PIB attributes

Subclause 7.4.2.1 applies and additional attributes are required, see Table 86.i.

Table 86.i—LE-specific MAC PIB attributes

Attribute	Identifier	Type	Range	Description	Default
macCSLPeriod	0x88	Integer	0 … 65535	CSL sampled listening period in unit of 10 symbols. 0 means always listening, i.e., CSL off.	0
macCSLMaxPeriod	0x89	Integer	0 … 65535	Maximum CSL sampled listening period in unit of 10 symbols in the entire PAN. This determines the length of the wakeup sequence when communicating to a device whose CSL listen period is unknown. NHL may set this attribute to 0 to stop sending wakeup sequences with proper coordination with neighboring devices.	macCSLPeriod
macCSLChannelMask	0x8a	Integer		32-bit bitmap relative to phyCurrentPage of channels. It represents the list of channels CSL operates on. 0 means CSL operates on phyCurrentChannel of phyCurrentPage.	0
macCSLFramePendingWaitT	0x8b	Integer		Number of symbols to keep the receiver on after receiving a payload frame with FCF frame pending bit set to 1.	
macSecAckWaitDuration	0x8c	Integer		The maximum number of symbols to wait for a secure acknowledgement frame to arrive following a transmitted data frame.	
macRitPeriod	0x8d	Integer	0x000000 - 0xffffff	The interval (in unit periods) for periodical transmission of RIT data request command in RIT mode. The unit period is aBaseSuperframeDuration. 0 means RIT is off	0
macRitDataWaitPeriod	0x8e	Integer	0x00 – 0xff	The maximum time (in unit period) to wait for Data frame after transmission of RIT data request command frame in RIT mode. The unit period is aBaseSuperframeDuration.	0
macRitTxWaitTime	0x8f	Integer	macRitPeriod - 0xffffff	The maximum time (in unit periods) that a transaction is stored by a device in RIT mode. The unit period is aBaseSurperframeDuration.	0

7.4.2.7 MAC Performance Metrics-specific MAC PIB attributes

Subclause 7.4.2.1 applies and additional attributes are required for Metrics to enable higher layers to assess network performance and behavior and aide in MSDU segmentation decisions, see Table 86.j.

MAC PIB attributes in Table 86.j are provided to enable the assessment of network performance and connection quality by higher layers.

MAC PIB attribute macCounterOctets defines the size of the counters providing attributes 0x91 through 0x98. The values of 1 though 4 correspond to counters of 8, 16, 24, or 32 bits. Attribute macCounterOctets is read-only and set by the implementer depending on the PHY characteristics and other considerations.

The counters implementing MAC PIB attributes 0x91 through 0x98 shall wrap to 0 when incremented beyond their maximum value (2^(8*macCounterOctets)-1). Attributes 0x91 through 0x98 are read/write and may be reset by higher layers by writing a 0 value.

The attributes macRetryCount, macMultipleRetryCount, macTXFailCount, macTXSuccessCount relate to data frame transmission. Each MSDU transferred into the MAC layer through the MCPS-DATA.request primitive shall increment exactly one of these four attribute counters depending on the final disposition of the frame as described in Table 86.

The attributes macFCSErrorCount, macSecurityFailure, macDuplicateFrameCount, macRXSuccessCount relate to data frame reception. Each MSDU transferred out of the MAC layer through the MCPS-DATA.indication primitive shall increment at least one of these four attribute counters based on the status of the frame as described in Table 86.

To create a list of PHY-related metrics, it is recommended that higher layers store relevant parameters of MCPS-DATA.indication such as mpduLinkQuality and DataRate for each unique Source Address (SrcAdr). The higher layers may also query PLME-ED.request to establish an idle channel noise measurement.

Table 86.j—Metrics-specific MAC PIB attributes

Attribute	Identifier	Type	Range	Description	Default
macCounterOctets[a]	0x90	Integer	1 – 4	Defines the counter size in octets for attributes 0x81 through 0x88.	—[b]
macRetryCount	0x91	Integer	0 – (2^(8*macCounterOctets)-1)	The number of transmitted frames that required exactly one retry before acknowledgement	0
macMultipleRetryCount	0x92	Integer	0 – (2^(8*macCounterOctets)-1)	The number of transmitted frames that required more than one retry before acknowledgement	0
macTXFailCount	0x93	Integer	0 – (2^(8*macCounterOctets)-1)	The number of transmitted frames that did not result in an acknowledgement after macMaxFrameRetries	0
macTXSuccessCount	0x94	Integer	0 – (2^(8*macCounterOctets)-1)	The number of transmitted frames that were acknowledged within macAckWaitDuration after the initial data frame transmission	0
macFCSErrorCount	0x95	Integer	0 – (2^(8*macCounterOctets)-1)	The number of received frames that discarded due to an incorrect FCS	0
macSecurityFailure	0x96	Integer	0 – (2^(8*macCounterOctets)-1)	The number of received data frames that were returned from the procedure described in 7.5.9.2.3 (Incoming frame security procedure) with any status other than "SUCCESS"	0
macDuplicateFrameCount	0x97	Integer	0 – (2^(8*macCounterOctets)-1)	The number of received data frames that contained the same sequence number as a frame previously received (accounting for wrap-around of macDSN).	0
macRXSuccessCount	0x98	Integer	0 – (2^(8*macCounterOctets)-1)	The number of received data frames that were received correctly	0

[a] Read-only;
[b] Implementation-dependant.

7.5 MAC functional description

7.5.1 Channel access

7.5.1.1 Superframe structure

Insert after the first paragraph the following text.

For LL-applications is required an additional superframe structure with beacons using a shortened frame control, see 7.5.1.6.

7.5.1.4 CSMA-CA algorithm

Insert after the heading of 7.5.1.4 the following subclause.

7.5.1.4.1 General

Insert before 7.5.2 the following subclauses.

7.5.1.4.2 TSCH-CCA Algorithm

When a device is operating in the TSCH-mode (see 7.1.18.3) the CCA is used to promote coexistence with other users of the radio channel. For other devices in the same network the start time of transmissions, TxTxOffset, is closely aligned making intra-network collision avoidance using CCA ineffective. The TSCH- devices also do channel hopping so there is no backoff period used when CCA prevents a transmission.

When a device has a packet to transmit. it waits for a link it can transmit it in. If CCA has been enabled, the MAC requests the PHY to perform a CCA at the designated time in the timeslot, TxCCAOffset, without any backoff delays. Figure 69.a extend Figure 107 for the TSCH-mode.

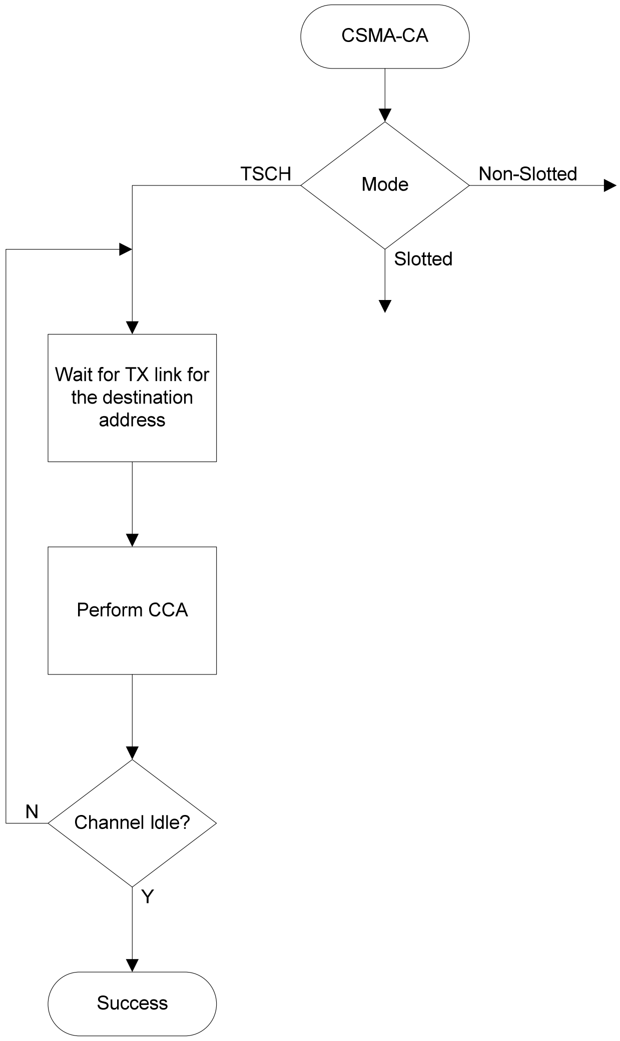


Figure 69.a—TSCH-CSMA-CA Algorithm

7.5.1.4.3 TSCH-CA Algorithm

Shared links (links with the linkOption shared bit set) are intentionally assigned to more than one device for transmission. This can lead to collisions and result in a transmission failure detected by not receiving an acknowledgement. To reduce the probability of repeated collisions when the packets are retransmitted a retransmission backoff algorithm shall be implemented for shared links.

When a packet is transmitted on a shared link for which an acknowledgement is expected and none is received, the transmitting device shall invoke the TSCH- CA retransmission algorithm. Subsequent retransmissions may be in either shared links or dedicated links. This backoff algorithm has the following properties:

- The retransmission backoff wait applies only to the transmission on shared links. There is no waiting for transmission on dedicated links.
- The retransmission backoff is calculated in the number of shared link transmission links.
- The backoff window increases for each consecutive failed transmission in a shared link.
- A successful transmission in a shared link resets the backoff window to the minimum value.
- The backoff window does not change when a transmission is a failure in a dedicated link.

— The backoff window does not change when a transmission is successful in a dedicated link and there transmission queue is still not empty afterwards.

— The backoff window is reset to the minimum value if the transmission in a dedicated link is successful and the transmit queue is then empty.

In TSCH-mode, backoff is calculated in shared links, so the CSMA-CA aUnitBackoffPeriod is not used.

macMaxBE and macMinBE have different default values when the device is in TSCH-mode (see table 86).

The device shall use an exponential backoff mechanism analogous to that described in 7.5.1.4.1. A device upon encountering a transmission failure in a shared link shall initialize the backoff exponent (BE) to macMinBE. The MAC sublayer shall delay for a random number in the range 0 to 2^{BE}–1 shared links (on any slotframe) before attempting a retransmission on a shared link. Retransmission on a dedicated link may occur at any time. For each successive failure on a shared link, the device should increase the backoff exponent until the backoff exponent = macMaxBE. Successful transmission on a shared link resets the backoff exponent to macMinBE.

If an acknowledgment is still not received after macMaxFrameRetries retransmissions, the MAC sublayer shall assume the transmission has failed and notify the next higher layer of the failure.

7.5.1.4.4 LL-Simplified CSMA-CA

This subclause defines a simplified CSMA-CA algorithm that is used during Management Time slots and Shared Group Timeslots in low latency networks.

The simplified CSMA-CA is a slotted CSMA-CA mechanism and follows the same algorithm as described in 7.5.1.4.1. However, some MAC PIB attributes have different default values as shown in Table 86.k.

Table 86.k—Default values for MAC PIB attributes for slotted CSMA-CA in LL-Networks

MAC PIB attribute	Default Value in Low Latency Networks
macMinBE	3
macMaxBE	3
macMaxCSMABackoffs	0

The backoff slots of *aUnitBackoffPeriod* symbols are aligned with the start of the beacon transmission in management time slots and with tSlotTxOwner in shared group time slots.

7.5.1.5 TSCH-Slotframe structure

7.5.1.5.1 General

A slotframe is a collection of timeslots repeating in time. The number of timeslots in a given slotframe (slotframe size) determines how often each timeslot repeats, thus setting a communication schedule for nodes that use the timeslots. When a slotframe is created, it is associated with a slotframe ID for identification. Every new slotframe instance in time is called a slotframe cycle. Figure 69.b shows how nodes may communicate in a sample three-timeslot slotframe. Nodes A and B communicate during timeslot 0, nodes B and C communicate during timeslot 1, and timeslot 2 is not being used. Every three timeslots, the schedule repeats. The total number of timeslots that has elapsed since the start of the network is called the Absolute Slot Number (ASN). The pair-wise assignment of a directed communication between devices in a given timeslot on a given channel offset is a link. Logical channel selection in a link is made by taking (Absolute Slot Number + channel offset) % Number of channels. A hopping sequence is a sorted list of physical channels that correspond to the logical channels selected. There must be a one-to-one mapping of logical channel to physical channel for two devices operating in a TSCH-network to agree on which channel is in use, and as such hopping sequence lengths are limited in a TSCH-network to the number of channels in use. Hopping sequence ID 0 is reserved to indicate that the logical channel and physical channel match. If other hopping sequences are to be used, a higher layer must define them and these must be configured prior to joining a TSCH-network.

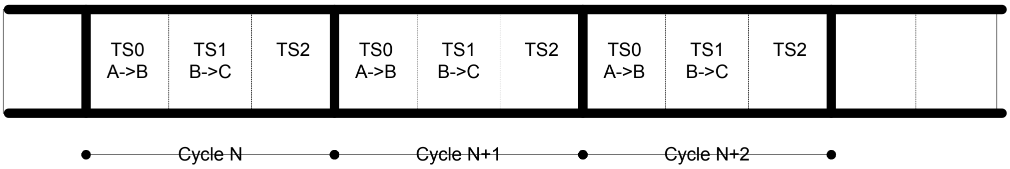


Figure 69.b—Example of a three-timeslot slotframe

Several performance parameters are determined by slotframe size and how timeslots are assigned within a slotframe for communication. In general, shorter slotframes result in lower latency and increased bandwidth, but at the expense of increased power consumption. Long slotframes generally result in higher latency and lower bandwidth, but power consumption is reduced and the number of communication resources (links) is increased. This affects the scale of the network.

7.5.1.5.2 Multiple slotframes

A given network using timeslot-based access may contain several concurrent slotframes of different sizes. Slotframe size defines the bandwidth of a timeslot. A timeslot within a slotframe of a particular size repeats twice as fast as a timeslot within a slotframe that is twice as long, thus allowing for double throughput on any given link. Multiple slotframes may be used to define a different communication schedule for various groups of nodes or to run the entire network at different duty cycles.

A network device may participate in one or more slotframes simultaneously, and not all devices need to participate in all slotframes. By configuring a network device to participate in multiple overlapping slotframes of different sizes, it is possible to establish different communication schedules and connectivity matrices that all work at the same time.

Slotframes can be added, removed, and modified while the network is running. Even though this is the case, all slotframes logically start in the same place in time. Cycle 0, timeslot 0 of every slotframe occurs at the beginning of epoch, which is determined by the network device that starts the network. Because of this, timeslots in different slotframes are always aligned, even though beginnings and ends of slotframes may not be (see Figure 69.c). Because all slotframes begin at the same time, it is always possible to identify time of a given slotframe cycle and timeslot, and ASN is the same across slotframes.

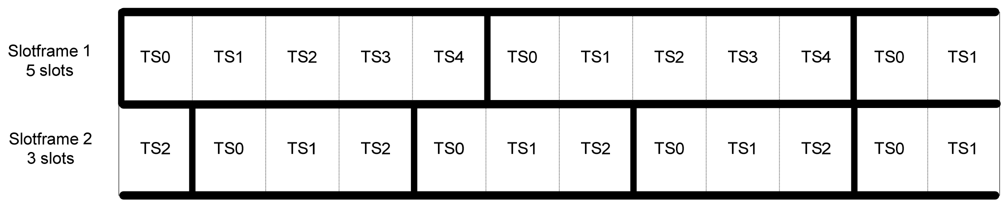


Figure 69.c—Multiple slotframes in the network

7.5.1.6 LL-Superframe structure

7.5.1.6.1 General Structure of Superframe

The superframe is divided into a beacon slot and *macFAnumTimeSlots* base time slots of equal length, see Figure 69.d.

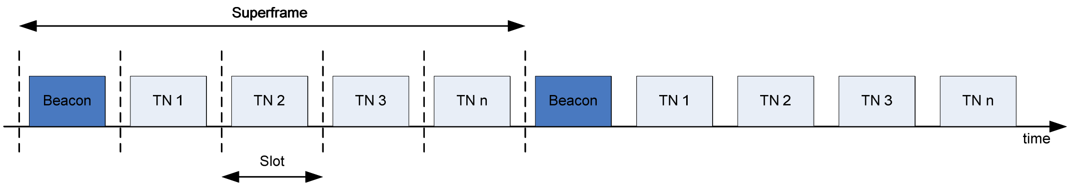


Figure 69.d—Superframe with dedicated time slots

The first time slot of each superframe contains a beacon frame. The beacon frame is used for synchronization with the superframe structure. It is also used for re-synchronization of devices that went into power save or sleep mode.

The remaining time slots are assigned to specific devices of the network. Each time slot may have assigned a so-called slot owner. The slot owner has access privileges in the time slot (dedicated time slot). There is no explicit addressing necessary inside the frames if the slot owner transmits in its time slot. The determination of the sender is achieved through the number of the time slot. More than one device can be assigned to a time slot (shared group time slot). The devices use a contention-based access method (modified CSMA/CA as described in 7.5.1.4.4) and a simple addressing scheme with 8-bit addresses in shared group time slots.

Multiple adjacent base time slots can be concatenated to a single, larger time slot.

As shown in Figure 69.e, there is a specific order in the meaning or usage of the time slots.

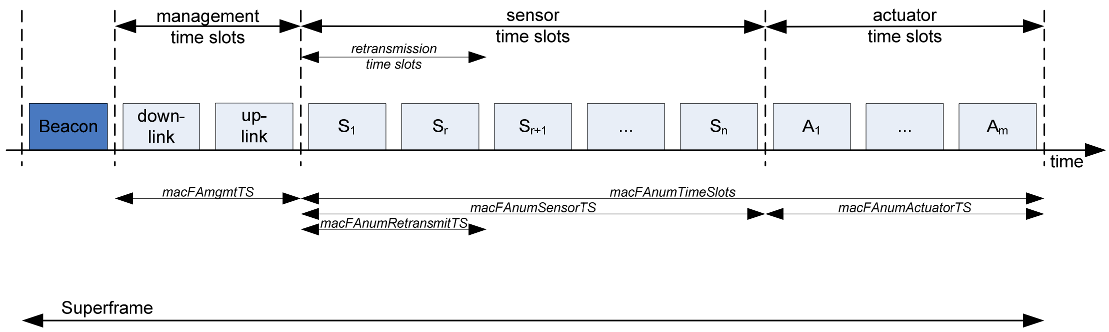


Figure 69.e—Usage and order of slots in a superframe

- Beacon Time Slot: always there (see 5.2)
- Management Time Slots: one time slot downlink, one time slot uplink, existence is configurable in *macFAmgmtTS* during setup (see 5.3)
- Time slots for sensors: *macFAnumSensorTS* time slots uplink (uni-directional communication), *macFAnumRetransmitTS* time slots at the beginning can be reserved for retransmissions (see 5.4)
- Time slots for actuators: *macFAnumActuatorTS* time slots uplink / downlink (bi-directional communication) (see 5.5)

7.5.1.6.2 Beacon Time Slot

The beacon time slot is reserved for the LL_NW PAN coordinator to indicate the start of a superframe with the transmission of a beacon. The beacon is used to synchronize the devices and to indicate the current transmission mode. The beacon contains also acknowledgements for the data transmitted in the last superframe.

The beacon time slot is available in every superframe.

7.5.1.6.3 Management Time Slots

The first portion of a superframe after the beacon time slot is formed by the management time slots, i.e. the downlink/uplink management time slots.

The downlink direction is defined as sending data *to* the device (sensor, actuator). The uplink direction is defined as sending data *from* the device (sensor, actuator).

Management time slots provide a mechanism for bidirectional transmission of management data in downlink and uplink direction. Downlink and uplink time slots are provided in equal number in a superframe. There are two management time slots per superframe at maximum. Management down-/uplink time slots are implemented as shared group access time slots.

Management down-/uplink time slots are used in discovery and configuration mode and are optional in the online mode.

7.5.1.6.4 Sensor Time Slots

After the management time slots, time slots for the transmission of sensor data are contained in a superframe. Sensor time slots allow for unidirectional communication (uplink) only.

The first *macFAnumRetransmitTS* of the *macFAnumSensorTS* sensor time slots are dedicated time slots for retransmissions of failed uplink transmission attempts in dedicated time slots of the previous superframe. The dynamic assignment of nodes to retransmission time slots is described in 7.5.9.4.

7.5.1.6.5 Actuator Time Slots

Actuator time slots allow for bidirectional communication between the LL_NW PAN coordinator and the device (actuator). The direction of the communication is signalled in the beacon as described in **Error! Reference source not found.**. Actuator time slots are used for the transmission of device data to the LL_NW PAN coordinator (uplink) as well as of actuator information from the LL_NW PAN coordinator to the device (downlink).

Comment [W42]: TBD by René

7.5.1.6.6 Channel access within time slots

Each time slot is described by four time attributes as illustrated in Figure 69.f and described in Table 86.l.

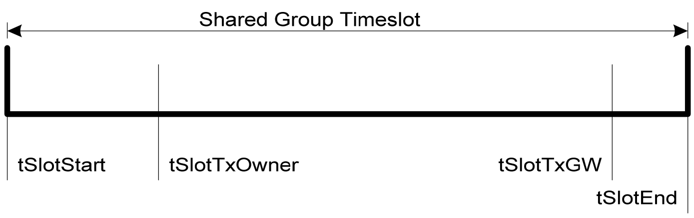


Figure 69.f—Time attributes of time slots

Table 86.l—Time attributes of time slots

Attribute	Description
tSlotStart	starting time of time slot
tSlotTxOwner	end time of privileged access by device that owns the time slot
tSlotTxGW	if time slot is unused, LL_NW PAN coordinator can use the time slot
tSlotEnd	end time of time slot

From tSlotStart till tSlotTxOwner, the device that owns the slot, the slot owner, has exclusive access to the time slot.

From tSlotTxOwner till tSlotTxGW, any device may use the time slot with a modified CSMA/CA access scheme as described in 7.5.1.5, if the time slot is not used by the slot owner.

From tSlotTxGW till tSlotEnd, the LL_NW PAN coordinator may use the time slot, if the time slot is still unused.

Dedicated time slots are reserved for a single device (slot owner). This is achieved by setting tSlotTxOwner and tSlotTxGW to tSlotEnd. A dedicated time slot allows the transmission of exactly one packet. Dedicated time slots are only used during online mode (see 7.5.9.4).

Shared group time slots with contention-based access for every allowed device can be achieved by setting tSlotTxOwner to tSlotStart.

7.5.1.7 LE-Functional description

7.5.1.7.1 LE-Contention access period (CAP)

When macCSLPeriod is set to non-zero, CSL is deployed in CAP.

macRitPeriod shall not be set to non-zero in a beacon-enabled PAN.

7.5.1.7.2 LE-Scanning through channels

When macCSLPeriod is set to non-zero, CSL is deployed in channel scans. When macCSLMaxPeriod is set to non-zero, each coordinator broadcasts beacon frames with wakeup sequence. This allows devices to perform channel scans with low duty cycles.

7.5.2 Starting and maintaining PANs

7.5.2.1 Scanning through channels

7.5.2.1.1 ED channel scan

7.5.2.1.2 Active channel scan

7.5.2.1.3 Passive channel scan

7.5.2.1.4 Orphan channel scan

Insert before 7.5.2.2 the following subclause.

7.5.2.1.5 LE-Scan

When macCSLPeriod is set to non-zero, CSL is deployed in channel scans. When macCSLMaxPeriod is set to non-zero, each coordinator broadcasts beacon frames with wakeup sequence. This allows devices to perform channel scans with low duty cycles.

7.5.2.2 PAN identifier conflict resolution

7.5.2.5 Device discovery

Insert before 7.5.3 the following subclause.

7.5.2.6 TSCH-network formation

7.5.2.6.1 Overview

There are two components of network formation in the TSCH-network:

— advertising and

— joining.

As a part of advertising, network devices that are already part of the network may send command frames announcing the presence of the network. Advertisement command frames include time synchronization information and a unique PAN ID. A new device trying to join listens for the Advertisement command frames. If the device is pre-provisioned with a PAN ID, then it matches the advertised PAN ID with the provisioned one at the higher layer. If there is no provisioned PAN ID, the device does not look for a match. When at least one acceptable Advertisement command frame is received, the new device can attempt to join the network. A new device joins the network by sending a Join request command frame to an advertising node. In a centralized management system this join command is routed to the PAN coordinator. In a distributed management system it can be processed locally. When the device is accepted into the network, the advertiser activates the device by setting up slotframes and links between the new device and other existing devices. These slotframes and links can also be deleted and modified and new slotframes and links added any time after a device has joined the network. The sequence of messages exchanged to synchronize a device to the networks is shown in Figure 69.g. The join sequence is shown in Figure 69.h.

A new network starts when the PAN coordinator starts to advertise (typically at the request of Network Manager residing in the PAN coordinator). Being the first node in the network, the PAN coordinator starts at least one slotframe, to which other network devices may later synchronize.

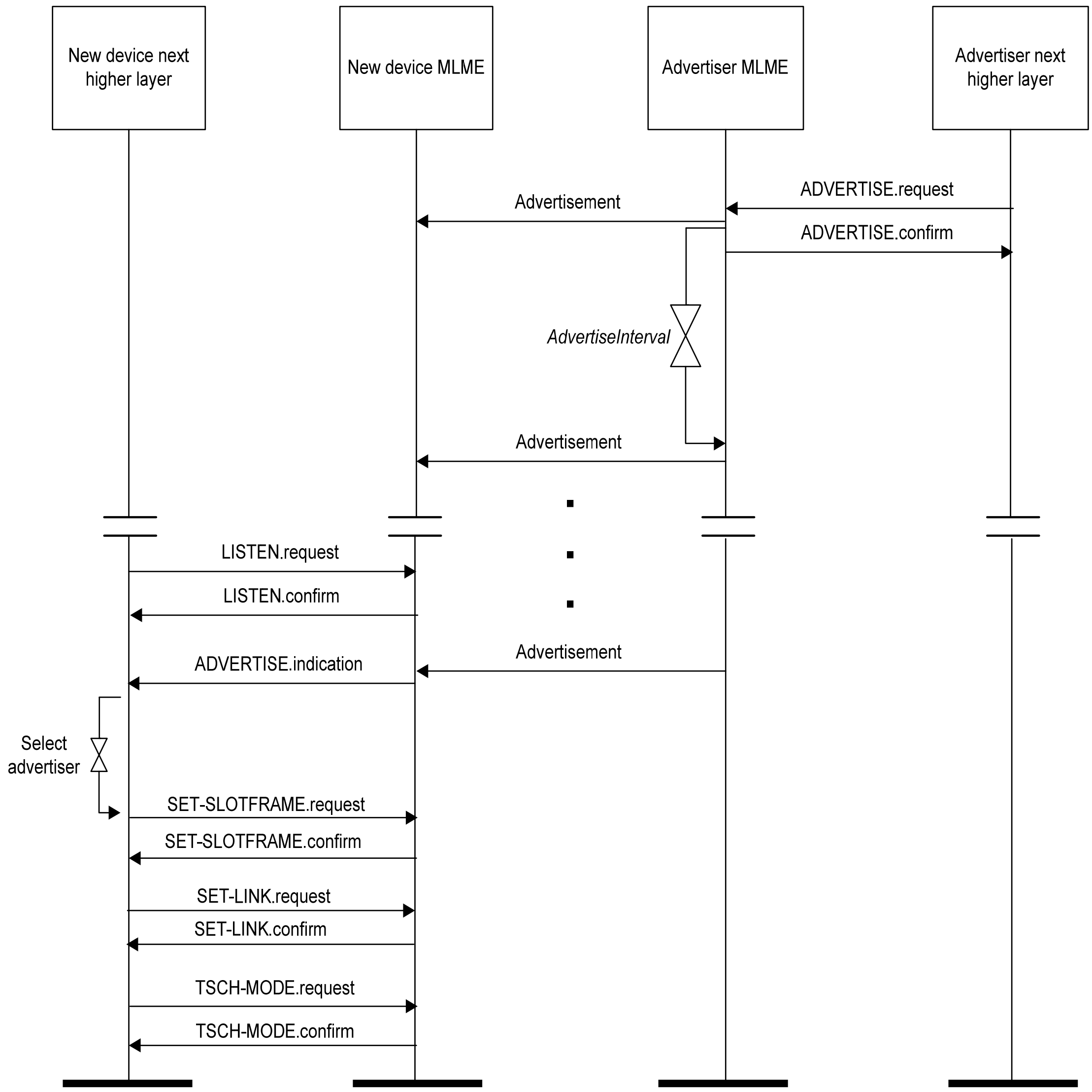


Figure 69.g—Message sequence chart for TSCH- procedure to find an advertising device

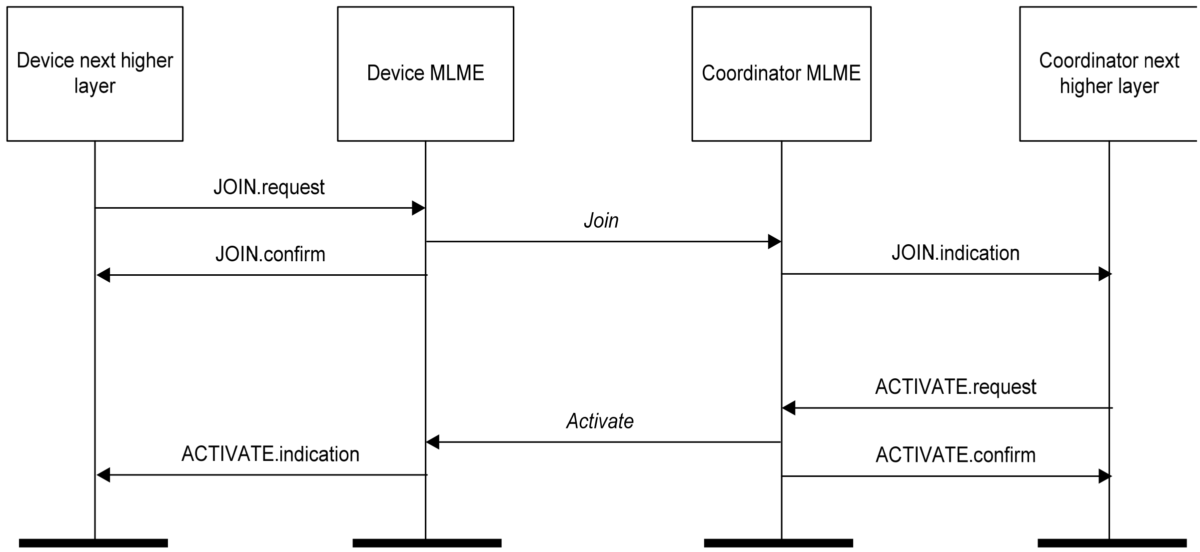


Figure 69.h—Message sequence chart for join and activate procedures

7.5.2.6.2 Advertising

In order for new devices to join a network they must first learn network information from some devices that are already part of the network. This is done through advertising. Network devices may send Advertisement command frames to invite new devices into the network. This is shown in Figure 69.g. The advertising device begins advertising on receipt of a ADVERTISE.request command from its NHL (next higher layer). At some time the device wishing to join the network begins listening (as result of receiving a LISTEN.request from its NHL). Once the listening device has heard an advertisement, it will generate an ADVERTISE.indication to a higher layer. The higher layer may initialize the slotframe and links contained in the advertisement and switch the device into TSCH-mode with a TSCH-MODE.request or wait for additional ADVERTISE.indications before doing so. At this point the device is synchronized to the network and may send in a Join request.

Advertisement command frames contain the following information:

- PAN ID.
- Time information so new devices can synchronize to the network.
- Channel page and a list of RF channels in that channel page being used.
- Link and slotframe information so new devices know when they can transmit to the advertising device.
- Link and slotframe information so new devices know when to listen for transmits from the advertising network device.

7.5.2.6.3 Joining

After a new device hears at least one valid Advertisement command frame, it may synchronize to the network and start joining. Advertisement command frames contain information about the links through which the new device may communicate with the advertising neighbor, and through it forward frames to the Network Manager. The joining procedure may include a security handshake to mutually authenticate the joining device and the Network Manager and establish the secure session between the new device and the Network Manager in addition to allocating the communication resource to the joining device. The content of authentication messages is beyond the scope of this document.

The joining process is shown in Figure 69.h. The joining device sends in a join message which contains its identity, capability and security information, and a list of potential neighbors heard during listening. The advertising device that receives this join request may process it locally or send it to a Network manager. If the device is to be allowed into the network, then an activate command is sent containing some slotframes and links that the device may use to communicate to its neighbors, which may or may not be the neighbor to whom the join request was sent. After receiving the activate command, the device may be instructed to remove slotframes and links obtained from advertisements. The device may receive additional slotframes and links from a Network Manager or peer as required by the application.

7.5.3 Association and disassociation

7.5.4 Synchronization

Insert before 7.5.4.1 the following paragraph.

For TSCH, Subclause 7.5.4 specifies in addition the procedures for coordinators to generate beacon frames for devices to synchronize to the TSCH-network. For PANs not supporting beacons, synchronization is performed by time synchronized communication within a timeslot of the slotframe.

7.5.4.1 Synchronization with beacons

Insert before 7.5.5 the following subclauses.

7.5.4.4 Synchronization in TSCH-network

7.5.4.4.1 Timeslot communication

During a timeslot in a slotframe, one node typically sends a frame, and another sends back an acknowledgement if it successfully receives that frame. An acknowledgement can be positive (ACK) or negative (NACK). A positive acknowledge indicates that the receiver has successfully received the frame and has taken ownership of it for further routing. A negative acknowledgement indicates that the receiver cannot accept the frame at this time, but has heard it with no errors. Both ACKs and NACKs carry timing information used by nodes to maintain network synchronization. Frames sent to a unicast node address require that a link-layer acknowledgement be sent in response during the same timeslot as shown in Figure 69.i. If an acknowledgement is requested and not received within the timeout period, retransmission of the frame waits until the next assigned transmit timeslot (in any active slotframe) to that address occurs.

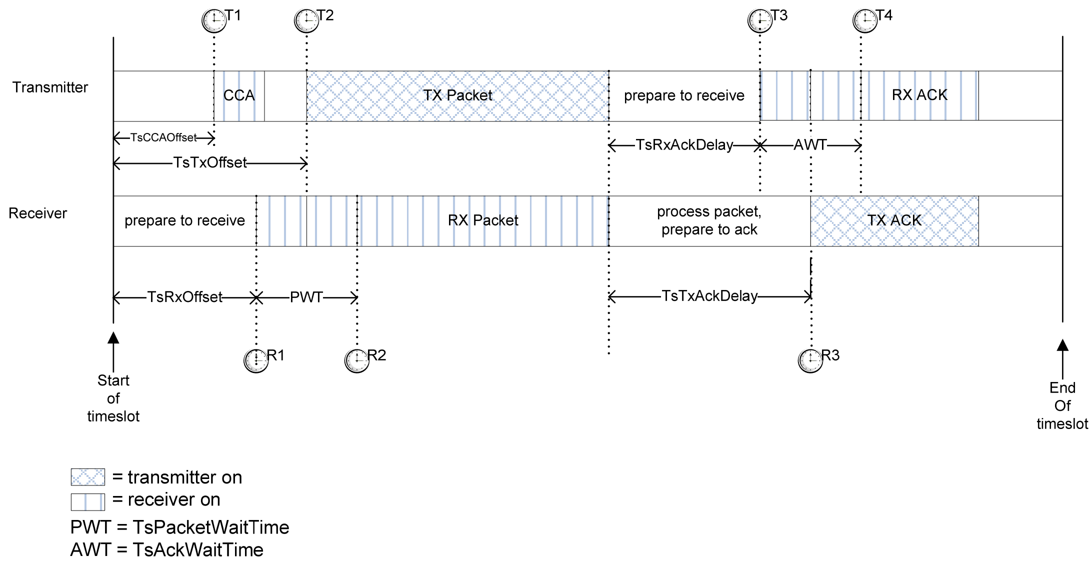


Figure 69.i—Timeslot diagram of acknowledged transmission

As shown in Figure 69.i, the timeslot starts at time T=0 from the transmitting device's perspective. The transmitter waits TsCCAOffset µs, and then performs CCA (if active). At TsTxOffset µs, the device begins transmitting the packet. The transmitter then waits TsRxAckDelay µs, then goes into receive mode to await the acknowledgement. If the acknowledgement doesn't arrive within TsAckWait (AWT) µs the device may idle the radio and that no acknowledgement will arrive.

On the receiver's side, at it's estimate of T=0 it waits TsRxOffset µs and then goes into receive for TsRxWait (PWT) µs. If the frame has not started by that time, it may idle the receiver. Otherwise, once the frame has been received, the receiver waits TsTxAckDelay µs and then sends an acknowledgement.

The transmitter or receiver may resynchronize clocks as described in 7.5.4.4.2.

EXAMPLE:

Below is the calculation of a 10 ms length timeslot template (from the transmitter's perspective):

TsTxOffset	2 120 µs
TsMaxPacket	4 256 µs
TsRxAckDelay	800 µs
TsAckWait	400 µs
TsMaxAck	2 400 µs
Total	9 976 µs

This allows for a maximum 133 octet frame (total including all SHR, PHR, MHR, etc.) to be sent, and an acknowledgement of up to 75 octets to be returned within 10 ms.

7.5.4.4.2 Node synchronization

7.5.4.4.2.1 General

Device-to-device synchronization is necessary to maintain connection with neighbors in a slotframe-based network. There are two methods for a device to synchronize to the network.

7.5.4.4.2.2 Acknowledgement-based synchronization

Unicast communication provides a basic method of time synchronization through the exchange of data and acknowledgement frames. The algorithm involves the receiver calculating the delta between the expected time of frame arrival and its actual arrival, and providing that information to the sender node.

The algorithm can be described as follows:

- Transmitter node sends a frame, timing the start symbol to be sent at TsTxOffset.
- Receiver records the timestamp TsRxActual of receiving the start symbol of the packet.
- Receiver calculates TimeAdj = TsTxOffset – TsRxActual.
- Receiver send back TimeAdj as part of acknowledgement packet.
- Transmitter receives the acknowledgement. If the receiver node is a clock source node, the transmitter adjusts its network clock by TimeAdj.

7.5.4.4.2.3 Frame-based synchronization

A node may synchronize its own network clock if it receives a frame from a clock source neighbor. The mechanism is similar to that of ACK-based synchronization. The receiver calculates the delta between expected time of frame arrival and its actual arrival time, and adjusts its own clock by the difference.

The algorithm can be described as follows:

- Receiver records the timestamp *TsRxActual* of receiving the start symbol of the packet.
- Receiver calculates *TimeAdj = TsTxOffset – TsRxActual.*
- Receiver adjusts its own network time by *–TimeAdj.*

Note that this procedure should only be executed if the node from which the frame is received is a clock source for the receiver.

Figure 69.j illustrates both time synchronization mechanisms. In both cases, the receiver calculates TimeAdj to either send back to the transmitter or to use locally.

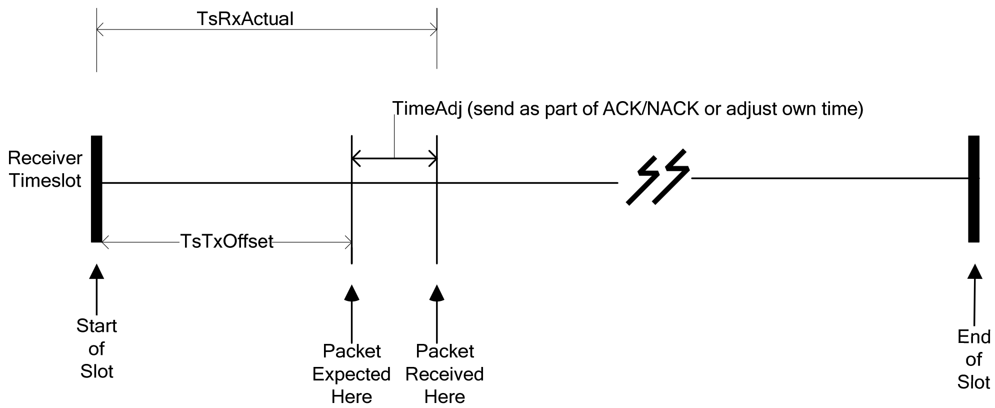


Figure 69.j—Time synchronization

7.5.4.4.2.4 Network time synchronization

Precise time synchronization is critical to the operation of networks based on time division multiplexing. Since all communication happens in timeslots, the network devices must have the same notion of when each timeslot begins and ends, with minimal variation. The acknowledgement and frame-based synchronization are used for pair-wise synchronization, as outlined below. In a typical TSCH-network, time propagates outwards from the PAN coordinator. It is very important to maintain unidirectional time propagation and avoid timing loops. A network device must periodically synchronize its network clock to at least one other network device. It may also provide its network time to one or more network devices. A network device determines whether to follow a neighbor's clock based on the presence of a ClockSource flag in the corresponding neighbor's record (configured by the Network Manager). The direction of time propagation is independent of data flow in the network. Neighbors included in a device's activate packet are marked as clock sources. A higher layer may add or change clock source neighbors later.

A network device may have more than one neighbor as its clock source. In such cases, the device may synchronize its clock to any of the neighbors that are acting as its clock source.

Figure 69.k shows typical time propagation in TSCH-network. The arrows indicate the direction of clock distribution. In this example, the PAN coordinator acts as the clock source for the entire network. Network Device (ND) 20 synchronizes to the PAN coordinator only, while ND 22 synchronizes its clock to both ND 20 and ND 21. If ND 20 and ND 23 were to be connected, ND 20 must provide time to ND 23. Setting it up otherwise would create a timing loop.

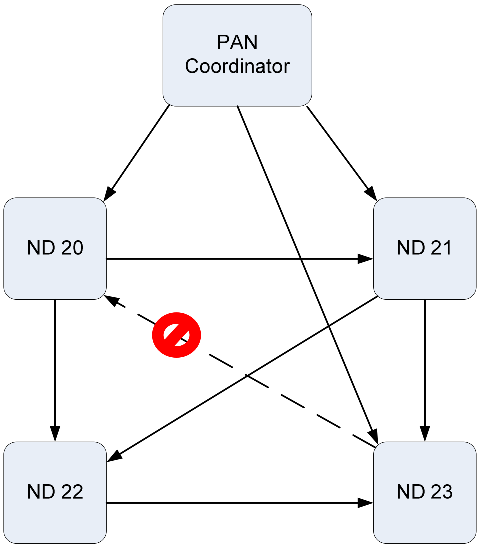


Figure 69.k—Time propagation in TSCH-network

7.5.4.4.2.5 Keep-Alive mechanism

In order to ensure that it remains synchronized with the TSCH-network (and to detect when paths may be down) a network device shall ensure that it communicates with each of its clock sources at least once per Keep Alive period.

If a network device has not sent a packet to its clock parent within this interval, it shall send a Keep-Alive command frame and use the ACK to perform ACK-based synchronization as usual.

7.5.5 Transaction handling

7.5.6.2 Reception and rejection

Change text in 7.5.6.2.

For valid frames that are not broadcast, if the Frame Type subfield indicates a data or MAC command frame and the Acknowledgment Request subfield of the Frame Control field is set to one, the MAC sublayer shall send an acknowledgment frame. Prior to the transmission of the acknowledgment frame, the sequence number included in the received data or MAC command frame shall be copied into the Sequence Number field of the acknowledgment frame. This step will allow the transaction originator to know that it has received the appropriate acknowledgment frame. If the PAN ID Compression subfield of the Frame Control field is set to one and both destination and source addressing information is included in the frame, the MAC sublayer shall assume that the omitted Source PAN Identifier field is identical to the Destination PAN Identifier field.

The device shall process the frame using the incoming frame security procedure described in 7.5.8.2.3. If the status from the incoming frame security procedure is not SUCCESS, the MLME shall issue the corresponding confirm or MLME-COMM-STATUS.indication primitive with the status parameter set to the status from the incoming frame security procedure, indicating the error, and with the security-related parameters set to the corresponding parameters returned by the unsecuring process.

For the first level of filtering, the MAC sublayer shall discard all received frames that do not contain a correct value in their FCS field in the MFR (see 7.2.1.9 and **Error! Reference source not found.**). The FCS field shall be verified on reception by recalculating the purported FCS over the MHR and MAC payload of the received frame and by subsequently comparing this value with the received FCS field. The FCS field of the received frame shall be considered to be correct if these values are the same and incorrect otherwise.

Comment [W43]: TBD by René

The second level of filtering shall be dependent on whether the MAC sublayer is currently operating in promiscuous mode. In promiscuous mode, the MAC sublayer shall pass all frames received after the first filter directly to the upper layers without applying any more filtering or processing. The MAC sublayer shall be in promiscuous mode if *macPromiscuousMode* is set to TRUE.

If the MAC sublayer is not in promiscuous mode (i.e., *macPromiscuousMode* is set to FALSE), it shall accept only frames that satisfy all of the following third-level filtering requirements:

— The Frame Type subfield shall not contain a reserved frame type.

— The Frame Version subfield shall not contain a reserved value.

— If a destination PAN identifier is included in the frame, it shall match *macPANId* or shall be the broadcast PAN identifier (0xffff).

— If a short destination address is included in the frame, it shall match either *macShortAddress*, macVeryShortAddress, or the broadcast address (0xffff). Otherwise, if an extended destination address is included in the frame, it shall match *aExtendedAddress*.

— If the frame type indicates that the frame is a beacon frame (frame type b000), the source PAN identifier shall match *macPANId* unless *macPANId* is equal to 0xffff, in which case the beacon frame shall be accepted regardless of the source PAN identifier. If the frame type indicates that the frame is a beacon frame of an LLNW (frame type b100, subframe type b00) and indicates online mode, the Gateway ID field shall match macFAlowLatencyNWid.

— If only source addressing fields are included in a data or MAC command frame, the frame shall be accepted only if the device is the PAN coordinator and the source PAN identifier matches macPANId.

If any of the third-level filtering requirements are not satisfied, the MAC sublayer shall discard the incoming frame without processing it further. If all of the third-level filtering requirements are satisfied, the frame shall be considered valid and processed further. For valid frames that are not broadcast, if the Frame Type subfield indicates a data or MAC command frame and the Acknowledgment Request subfield of the Frame Control field is set to one, the MAC sublayer shall send an acknowledgment frame. Prior to the transmission of the acknowledgment frame, the sequence number included in the received data or MAC command frame shall be copied into the Sequence Number field of the acknowledgment frame. This step will allow the transaction originator to know that it has received the appropriate acknowledgment frame.

If the PAN ID Compression subfield of the Frame Control field is set to one and both destination and source addressing information is included in the frame, the MAC sublayer shall assume that the omitted Source PAN Identifier field is identical to the Destination PAN Identifier field.

The device shall process the frame using the incoming frame security procedure described in 7.5.9.2.3.

If the status from the incoming frame security procedure is not SUCCESS, the MLME shall issue the corresponding confirm or MLME-COMM-STATUS.indication primitive with the status parameter set to the status from the incoming frame security procedure, indicating the error, and with the security-related parameters set to the corresponding parameters returned by the unsecuring process.

If the valid frame is a data frame, the MAC sublayer shall pass the frame to the next higher layer. This is achieved by issuing the MCPS-DATA.indication primitive containing the frame information. The security related parameters of the MCPS-DATA.indication primitive shall be set to the corresponding parameters returned by the unsecuring process.

If the valid frame is a MAC command or beacon frame, it shall be processed by the MAC sublayer accordingly, and a corresponding confirm or indication primitive may be sent to the next higher layer. The security-related parameters of the corresponding confirm or indication primitive shall be set to the corresponding parameters returned by the unsecuring process.

7.5.6.4.2 Acknowledgment

Editorial note RS:

The so-called ACK delay was discussed during the IEEE 802 meeting, Atlanta, Georgia, November 10-15, 2009 (cf., e.g., 09/782r1). During that discussion, it was suggested to essentially keep the turn-around time the same (for 2.4 GHz PHY). Details to be looked up in the minutes of that meeting.

Insert before 7.5.6.4.3 the following paragraph.

When in TSCH mode, incoming frames are acknowledged using the secure acknowledge frame as described in 7.2.5.2.4. Security of the acknowledge should match that of the incoming frame.

Comment [W44]: Ref shall be set to a valid subclause.

When operating in TSCH-mode (see 7.1.18.3), the acknowledgement frame is sent at the time specified by the macTimeslotTemplate being used (see 7.4.2 and 7.5.4.4.1); i.e. macSecAckWaitDuration should be set to a value corresponding to TsAckWait.

7.5.6.4.3 Retransmissions

Insert after the heading of 7.5.6.4.3 the following subclause.

7.5.6.4.3.1 General

Insert before 7.5.6.5 the following subclause.

7.5.6.4.3.2 TSCH-Retransmissions

A device that sends a data or MAC command frame with its Acknowledgment Request subfield set to one shall wait for TsRxAckDelay µs. If an acknowledgment frame is received within macAckWaitDuration symbols and contains the same DSN as the original transmission, the transmission is considered successful, and no further action regarding retransmission shall be taken by the device. If an acknowledgment is not received within the appropriate timeout or an acknowledgment is received containing a DSN that was not the same as the original transmission, the device shall conclude that the single transmission attempt has failed.

If a single transmission attempt has failed and the transmission was indirect, the coordinator shall not retransmit the data or MAC command frame. Instead, the frame shall remain in the transaction queue of the coordinator and can only be extracted following the reception of a new data request command. If a new data request command is received, the originating device shall transmit the frame using the same DSN as was used in the original transmission.

If a single transmission attempt has failed and the transmission was direct, the device shall repeat the process of transmitting the data or MAC command frame and waiting for the acknowledgment, up to a maximum of macMaxFrameRetries times. The retransmitted frame shall contain the same DSN as was used in the original transmission. Each retransmission shall only be attempted if it can be completed within the same portion of the superframe, i.e., the CAP or a GTS in which the original transmission was attempted. If this timing is not possible, the retransmission shall be deferred until the same portion in the next superframe. In TSCH-mode (see 7.1.18.3), retransmissions only occur on subsequent transmit links to the same recipient on any active slotframe. If an acknowledgment is still not received after macMaxFrameRetries retransmissions, the MAC sublayer shall assume the transmission has failed and notify the next higher layer of the failure.

7.5.6.5 Promiscuous mode

7.5.7 GTS allocation and management

7.5.7.6 GTS expiration

7.5.8 Frame security

Editorial note RS:

Replace this clause entirely by the corresponding clause of the draft text submitted to the Editing Team by August 15, 2009. This takes into account certain errors that are to be tackled with the Corrigendum to 802.15.4-2006, as also discussed during the IEEE 802 meeting, Atlanta, Georgia, November 10-15, 2009 (cf., e.g., 09/782r1). During that discussion, it was suggested that for TG4e editing purposes, one could anticipate the Corrigendum to include these updates. Details to be looked up in the minutes of that meeting.

Insert before 7.5.7a the following subclauses.

In a consolidated/integrated new edition the new subclauses should be moved before 7.5.7a Ranging. The numbering scheme don't allow to have a number between 7.5.7 and 7.5.7a, so that it is assigned here to 7.5.9 to 7.5.11.

7.5.9 LL-Transmission Modes in star networks using short MAC headers

7.5.9.1 General

The transitions between the different transmission modes are illustrated in Figure 73.a.

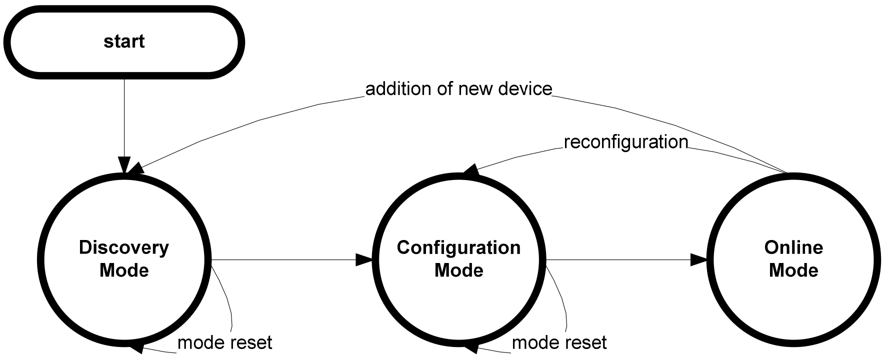


Figure 73.a—Transitions between transmission modes

The discovery mode is the first step during network setup: the new devices are discovered and configured in the second step, the configuration mode. After the successful completion of the configuration mode, the network can go into online mode. Productivity data, that is, data and readings from the devices such as sensors and actuators, can only be transmitted during online modus. In order to reconfigure a network, the configuration mode can be started again.

7.5.9.2 Discovery Mode

The Discovery Mode is the first step during network setup or for the addition of new devices to an existing network.

In discovery mode, the superframe contains only the time slot for the beacon (see 7.5.1.6.2) and two management time slots, one downlink and one uplink (7.5.1.6.3).

A new device scans the different channels until it detects a LL_NW PAN coordinator sending beacons that indicate discovery mode.

If a new device received a beacon indicating discovery mode, it tries to get access to the transmission medium in the uplink management time slot in order to send a Discover Response frame to the LL_NW PAN coordinator. The Discover Response frame is described in 7.3.11.1. The Discover Response frame contains the current configuration of the device. The new device shall repeat sending the Discover Response frame until it receives an Acknowledgement frame for it or the Discovery Mode is stopped by the LL_NW PAN coordinator. The Acknowledgement frame is described in **Error! Reference source not found.**.

Comment [W45]: TBD by René

Figure 73.b illustrates the discovery mode.

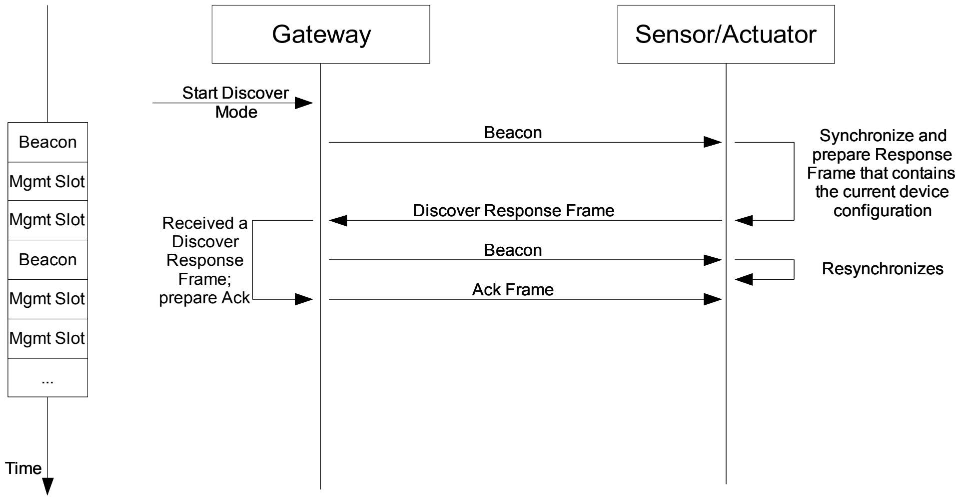


Figure 73.b—Flow diagram of Discovery Mode

7.5.9.3 Configuration Mode

The Configuration Mode is the second step during network setup. It is also used for network reconfiguration.

In configuration mode, the superframe contains only the time slot for the beacon (see 7.5.1.6.2) and two management time slots, one downlink and one uplink (see 7.5.1.6.3).

If a device received a beacon indicating configuration mode, it tries to get access to the transmission medium in the uplink management time slot in order to send a Configuration Response frame to the LL_NW PAN coordinator. The Configuration Response frame is described in 7.3.11.2. The Configuration Response frame contains the current configuration of the device. The new device shall repeat sending the Configuration Response frame until it receives a Configuration Request frame for it or the Configuration Mode is stopped by the LL_NW PAN coordinator. The Configuration Request frame is described in 7.3.11.3. The Configuration Request frame contains the new configuration for the receiving device. After successfully receiving the Configuration Request frame, the device sends an Acknowledgement frame to the LL_NW PAN coordinator. The Acknowledgement frame is described in **Error! Reference source not found.**.

Comment [W46]: TBD by René

Figure 73.c illustrates the configuration mode.

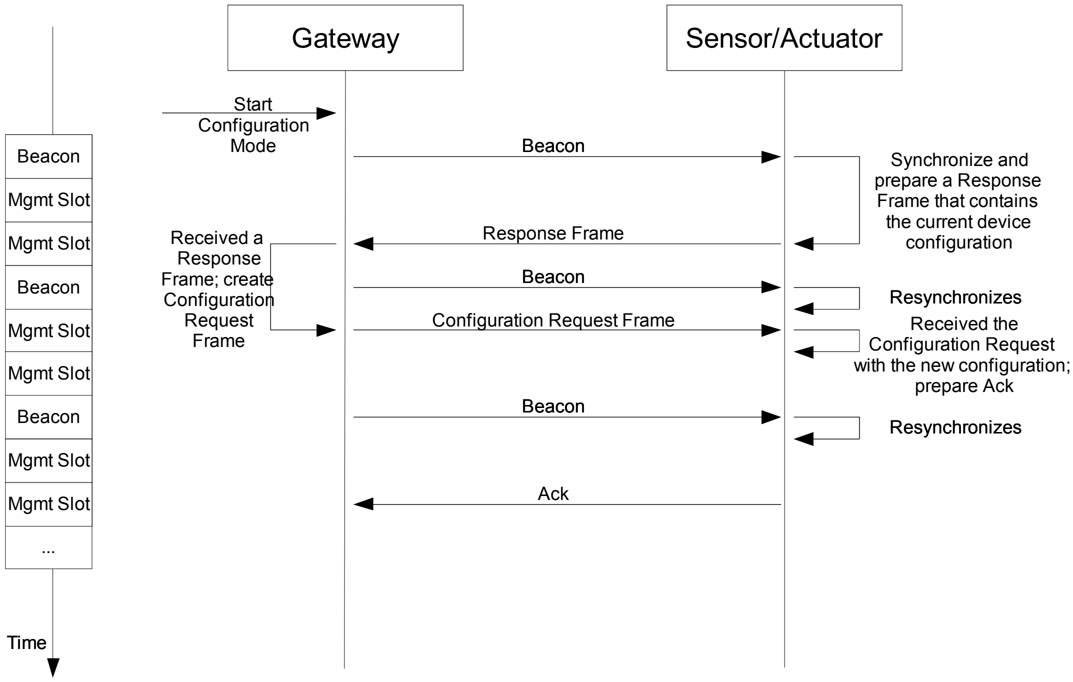


Figure 73.c—Flow diagram of configuration mode

7.5.9.4 Online Mode

User data is only sent during Online mode. The superframe starts with a beacon and is followed by several time slots. The devices can sent their data during the time slots assigned to them during configuration mode. The different types of time slots are described in clause 5.

The existence and length of management time slots in online mode is signalled in the configuration request frame.

The successful reception of data frames by the LL_NW PAN coordinator is acknowledged in the Group Acknowledgement bitmap of the beacon frame of the next superframe (see **Error! Reference source not found.**) or in a separate Data Group Acknowledgement frame (see **Error! Reference source not found.**) if so configured. This is the case for both sensor time slots and actuator time slots if the actuator direction is uplink. Figure 73.d illustrates an example of the online mode for uplink transmissions. The network has 3 dedicated time slots, and sensor 2 is assigned to time slot 2.

Comment [W47]: TBD by René

Comment [W48]: TBD by René

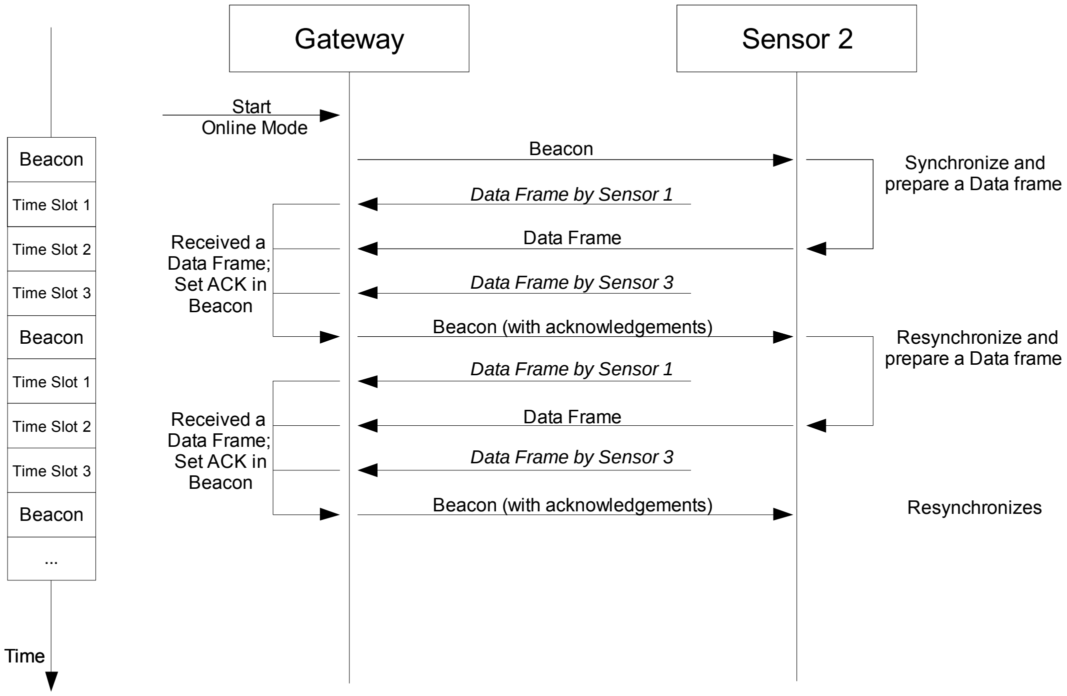


Figure 73.d—Flow diagram of online mode for sensor devices

If retransmission time slots are configured (*macFAnumRetransmitTS* > 0), the retransmission slots are assigned to the owners of the first *macFAnumRetransmitTS* with the corresponding bit in the group acknowledgement bitmap set to 0. Each sensor node has to execute the following algorithm in order to determine its retransmission time slot *r*. The LL_NW PAN coordinator has to execute a similar algorithm in order to determine the senders of the frames in the retransmission slots.

Assume that the sensor node has been assigned to sensor time slot *s*. ack[i] means the bit b_{i-1} in the group acknowledgment bitmap according to **Error! Reference source not found.** in **Error! Reference source not found.**.

Comment [W50]: TBD by René

Comment [W49]: TBD by René

```
if (ack[s] == false) {
            num_failed    :=    number    of    (ack[i]    ==    0    with
(macFAnumRetransmitTS+1) ≤ i ≤ (s-1))
            if (num_failed < macFAnumRetransmitTS) {
                        retransmission_possible = true
                        r = num_failed + 1
            }
            else {
                        retransmission_possible = false
            }
}
```

The successful reception of data frames by actuator devices (actuator direction is downlink) is acknowledged by an explicit acknowledgement frame by the corresponding actuator devices in the following superframe. This means that after setting the actuator direction bit in the beacon (see **Error! Reference source not found.**) to downlink and sending a data frame to one or more actuator devices, the LL_NW PAN coordinator shall set the actuator direction bit to uplink in the directly following superframe. Actuator devices having successfully received a data frame from the LL_NW PAN coordinator during the previous superframe shall sent an acknowledgement frame to the LL_NW PAN coordinator. Actuator devices that did not receive a data frame from the LL_NW PAN coordinator, may send data frames to the LL_NW PAN coordinator during this superframe with actuator direction bit set to uplink. Figure 73.e illustrates the online mode with actuator devices. The network has 3 dedicated actuator time slots, and actuator 2 is assigned to time slot 2.

Comment [W51]: TBD by René

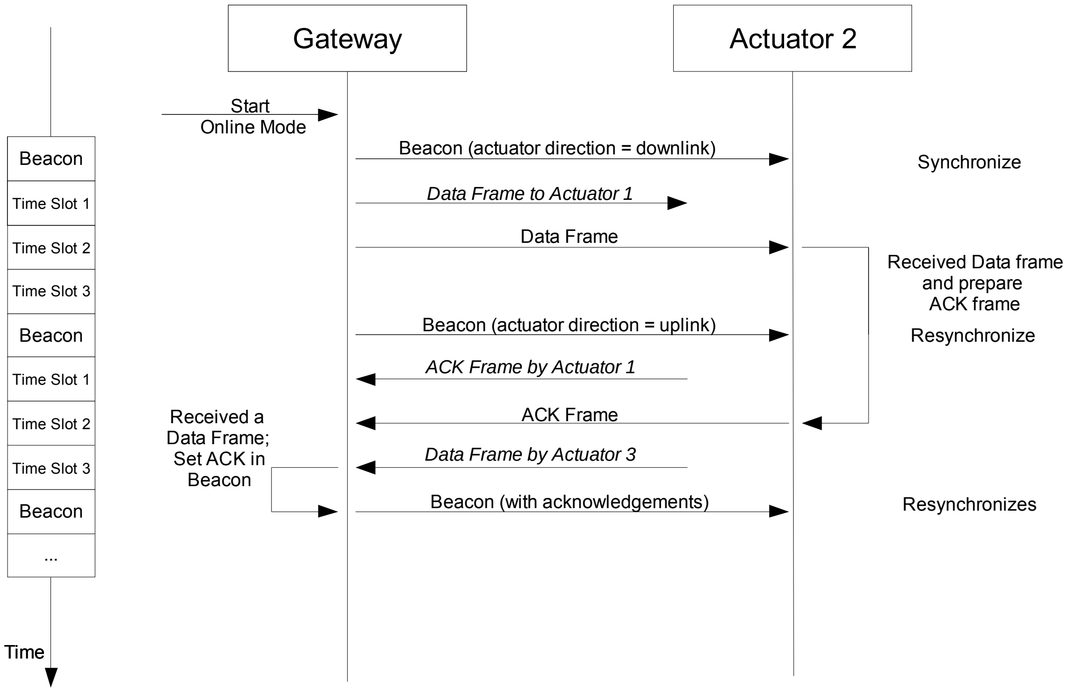


Figure 73.e—Flow diagram of online mode for actuator devices

7.5.10 DSME-DSME-based Multi-superframe Structure

7.5.10.1 DSME-based Multi-superframe Structure Definition

A coordinator on an DSME-based PAN can optionally bound its channel time using a multi-superframe structure. A multi-superframe is a cycle of repeated superframes, each of which consists of a beacon frame, a CAP and a CFP.

The structure of this multi-superframe is described by the values of *macBeaconOrder, macSuperframeOrder, and macMulti-superframeOrder.*

The MAC PIB attribute *macBeaconOrder* describes the interval at which the coordinator shall transmit its beacon frames. The value of *macBeaconOrder*, *BO*, and the beacon interval, *BI*, are related as follows: for $0 \leq BO \leq 14$, $BI = aBaseSuperframeDuration * 2^{BO}$ symbols. If $BO = 15$, the coordinator shall not transmit beacon frames except when requested to do so, such as on receipt of a beacon request command. The value of *macSuperframeOrder* and *macMulti-superframeOrder* shall be ignored if $BO = 15$.

The MAC PIB attribute *macSuperframeOrder* describes the length of a superframe. The value of *macSuperframeOrder*, *SO*, and the superframe duration, *SD*, are related as follows: for $0 \leq SO \leq BO \leq 14$, $SD = aBaseSuperframeDuration * 2^{SO}$ symbols.

The MAC PIB attribute *macMulti-superframeOrder* describes the length of a multi-superframe, which is a cycle of repeated superframes. The value of *macMulti-superframeOrder*, *MO*, and the multi-superframe duration, *MD*, are related as follows: for $0 \leq SO \leq MO \leq BO \leq 14$, $MD = aBaseSuperframeDuration * 2^{MO}$ symbols.

In case, both active period and inactive period in the beacon interval are filled with cyclic multi-superframes.

Each superframe shall be divided into *aNumSuperframeSlots* equally spaced slots of duration 2^{SO} * *aBaseSlotDuration* and is composed of three parts: a beacon, a CAP and a CFP. The beacon shall be transmitted, without the use of CSMA, at the start of slot 0, and the CAP shall commence immediately following the beacon. The start of slot 0 is defined as the point at which the first symbol of the beacon PPDU is transmitted. The CFP follows immediately after the CAP and extends to the end of the superframe. Any allocated DSMEs shall be located within the CFP.

The DSME-based PANs shall use the multi-superframe structure, and set *macBeaconOrder* to a value between 0 and 14, both inclusive, and *macSuperframeOrder* to a value between 0 and the value of *macBeaconOrder*, *macMulti-superframeOrder* to a value between the value of *macSuperframeOrder* and the value of *macBeaconOrder*, both inclusive.

An example of a multi-superframe structure is shown in Figure 73.f. In this case, the beacon interval, *BI*, is eight times as long as the superframe duration, *SD*, and twice as long as the multi-superframe duration, MD.

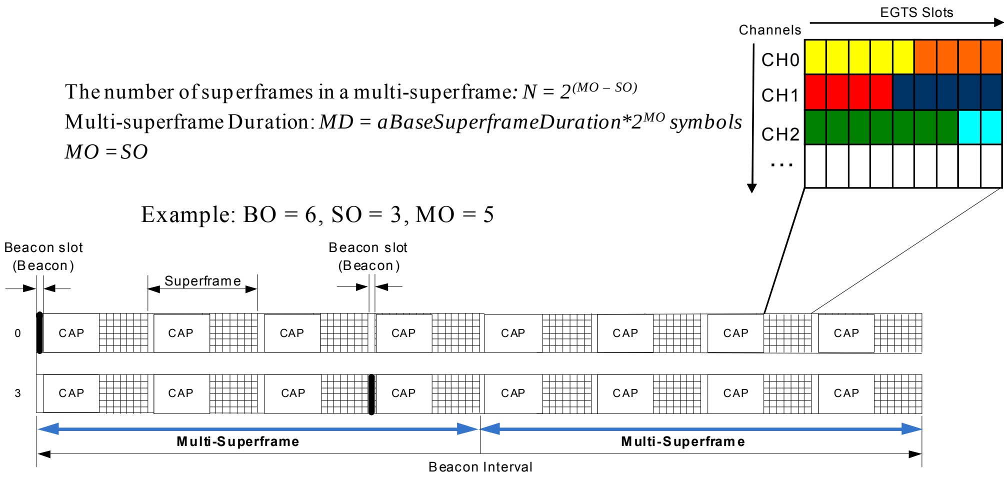


Figure 73.f—DSME-based multi-superframe structure

7.5.10.2 Channel Hopping Mode

In channel hopping mode (i.e., ChannelDiversityMode is set to '1'), each DSME slot shall use different channel to receive. Series of channels used at each DSME slots is called channel hopping sequence. Same channel hopping sequence shall be repeated over whole DSME slots in a multi-superframe. Device may select channel offset value to prevent same channel is used among devices within interfering range so as to minimize adverse interfering signals. Thus, devices in the PAN with single channel hopping sequence can access different channels at the given DSME slot if they have different channel hopping offset values, due to orthogonality in time and frequency.

An Example of the schedule of channels and DSMEs in Channel Hopping mode is illustrated in Figure 73.g.

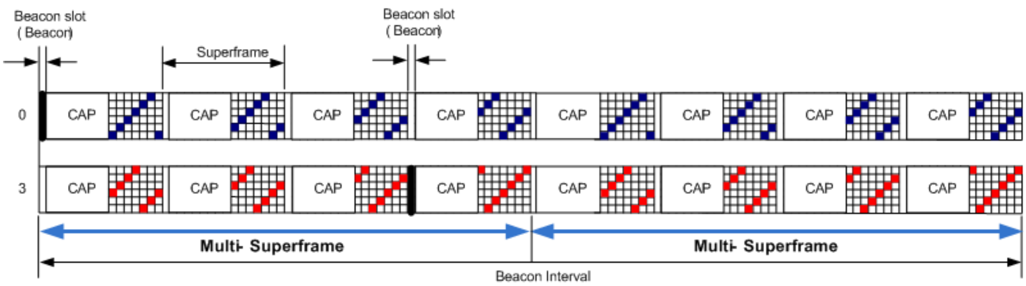


Figure 73.g—Channel usage of DSME slots in DSME-based multi-superframe structure

In this example, channel hopping sequence is {1, 2, 3, 4, 5, 6} and the channel hopping offset values of two devices are 0 and 2 respectively. For the device with channel hopping offset value of 0, DSME slots (timeslot, channel) for this device are (1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (7, 1), (8, 2), (9, 3), and so on. Similarly, for the device with channel hopping offset value of 2, DSME slots are given as (1, 3), (2, 4), (3, 5), (4, 6), (5, 1), (6, 2), and so on.

Thus, channel number *C* at the given DSME slots index *i* shall be determined as:

$$C(i) = CHSeq[\ (i + CHOffset)\ \%\ CHSeqLength\ ],$$

where *CHSeq*[*j*] represents the (*j*)th channel number in channel hopping sequence in use, *CHOffset* is the channel offset value and *CHSeqLength* is the length of channel hopping sequence.

Meanwhile, total number of DSME slots *NoSlot* in a multi-superframe is given by:

$$NoSlot = (7*2\text{^}(MO\text{-}SO))\ slots$$

7.5.10.3 Group Ack

7.5.10.3.1 General

In many application systems, it may be imperative to provide the sensor nodes with a retransmission opportunity, within the same superframe, for a data frame that failed in its GTS transmission. To satisfy that crucial requirement, the MAC sub-layer shall provide an optional feature of group acknowledgement whereby: (a) multiple GTS frames received by the coordinator in the CFP shall be acknowledged by a single transmission of GACK frame by the receiver (i.e. the coordinator); and (2) new GTS timeslots shall be allowed to be allocated to some of the transmitting sensor nodes for retransmission of their failed GTS transmission or for transmitting additional data frames. If the group ACK mechanism is activated by a coordinator, the structure of the MAC superframe shall be modified as described in the following sections.

7.5.10.3.2 Modified MAC Superframe Structure

The CFP of the MAC superframe shall be organized into several sub-periods as shown in **Error! Reference source not found.**. Specifically, each of these sub-periods shall consist of a set of standard CFP (or SCFP) slots, a group acknowledgement (GACK 1) slot, extended CFP (ECFP) slots, and another group acknowledgement (GACK 2) slot. Beacon frame, transmitted by the coordinator at the beginning of the slot 0, shall specify the information about the CFP. That information shall be used to determine the start of the SCFP and its duration. For the failed GTS transmissions in SCFP, the coordinator shall dynamically allocate new time slots, in the ECFP, and inform the sensor nodes of these new allocations by transmitting the GACK 1 frame at the end of the SCFP. In addition to transmitting the information about dynamic resource assignment (i.e. allocation of time slots for use in ECFP), the GACK 1 frame shall also include a group acknowledgement (in the form of a bitmap) for the GTS frame successfully received by the coordinator during the SCFP. The use of group acknowledgement bitmap eliminates the need for indivisual acknowledgements for the received frames and allows the coordinator to avoid from switching over from Rx mode to Tx mode and then back to Rx mode while acknowledgeing individual GTS transmissions. Based on the resource allocation, as specified in the GACK 1 frame, the sending nodes shall retransmit their data frames in the allocated time slots in the ECFP. The coordinator shall transmit the GACK 2 frame after the completion of the ECFP. The GACK 2 frame shall contain the bitmap only indicating successful and failed reception of GTS frames during the ECFP. If the GACK 1 frame contains no resource assignments, the ECFP and GACK 2 shall be non-existent in that superframe.

Comment [W52]: TBD by René

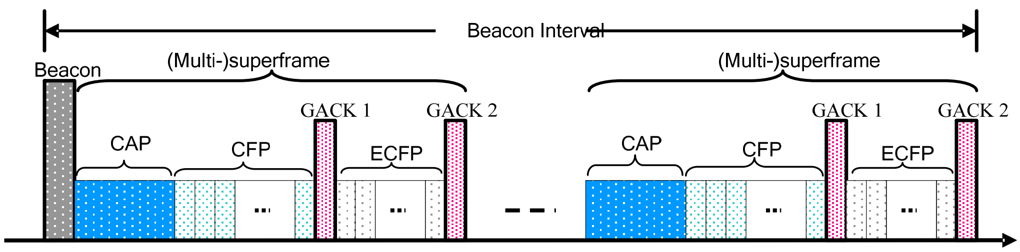


Figure 73.h—Details of DSME Superframe with ECFP and GACK

In addition to allow a retransmission of failed GTS transmissions in the SCFP, the ECFP shall also be used for allocating additional time slots on-demand to requesting nodes. A node shall request an additional GTS in the ECFP by setting a flag in the primitive while forwarding its data frame in the SCFP. The coordinator shall decide, based on the availability of time slots and/or priority mechanism, if to allocate the requested additional slots. The requesting node shall find the result of its request by checking the resource allocation in the following GACK 1 frame.

Comment [LL53]:

7.5.10.3.3 Using Group Ack Mechanism

Only a LL_NW coordinator device shall control the group acknowledgement (GACK) mechanism. Sensor nodes shall not initiate the request to the coordinator for the use of GACK mechanism. In a network that allows the use of group ACK mechanism, nodes shall use appropriate flags and fields in the beacon frame, data frame header, and GACK frame in order to facilitate the use of the GACK mechanism. The GACK Flag (i.e. b32) in the beacon frame, as described in Section 7.2.6.2.2.10, transmitted by the coordinator shall indicate if the GACK mechanism is activative and being used by the coordinator. GACK Flag set to '0' shall indicate that the coordinator is not using the mechanism. In such a case, all data transmissions by the sensor nodes to the coordinator shall be acknowledged individually as is the case of the normal operational mode of the original IEEE 802.15.4 specification. If the GACK Flag is set to '1' in the beacon frame, the transmitting coordinator shall use the GACK mechanism for its communication with sensor nodes. In such a case, the coordinator shall not transmit individual ACK frames for the data frames it received in GTS slots. Rather, all GTS transmissions received by the coordinator shall be acknowledged together in a GACK frame transmitted by the coordinator after the last GTS time slot in the CFP.

While the group ACK mechanism is active, all those sensor nodes that have transmitted their data farme in a GTS shall wait, after completing their GTS transmission, for the GACK frame transmitted by the receiving coordinator. The bitmap in the Group Ack Flags field of the GACK frame shall be used to determine if the GTS transmission by a sensor node was successful. A sensor node shall use Channel Index, DSME Device List and DSME Index to determine when to start its next GTS transmission in DSME portion of the superframe. That is the case when either the previous GTS transmission by the sensor node failed or it requested for allocation of an extra GTS for transmitting its additional data frames.

If a sensor node has additional data frames to be transmitted during the current superframe period, it shall indicate its desire for additional GTS allocation by in the header of its data frame transmitted in its allocated GTS in the CFP. Specifically, the node shall use '101' value for Frame Type subfield (i.e. b2b1b0) in the Frame Control field of the MAC frame header as specified in Table 120. The requesting sensor node shall check Channel Index, DSME Device List and DSME Index to determine if its request for additional GTS was granted by the coordinator. If granted, the node shall transmit its additional data frame in the allocated GTS in ECFP portion of the MAC superframe.

The application running on an LL_NW coordinator shall enable or disable the GACK mechanism by using MLME_SET.request primitive in order to set the PIB attribute macFAuseGACKmechanism, as listed in Table 127.f, to appropriate value. The application shall be able to setermine if the GACK mechanism is currently being used by the MAC by finding the current setting for macFAuseGACKmechanism PIB attribute. The application shall use MLME_GET.request primitive for finding the current setting for macFAuseGACKmechanism attribute.

7.5.10.4 CAP Reduction

If *macCAPReductionFlag* or the CAP Reduction Flag subfield in the DSME Superframe Specification field of a beacon frame is set to TRUE, the CAP reduction shall be enabled, except the first superframe in the multi-superframe, other superframes do not have the CAP. Figure 73.i shows an example of the multi-superframe structure when CAP reduction is enabled.

Number of available EGTS slots in a multi-superframe

- Without CAP reduction: $S_1 = 16\ channels * (7 * 2^{(MO-SO)})\ time\ slots$
- With CAP reduction: $S_2 = 16\ channels * (7 + (2^{(MO-SO)} - 1) * 15)\ time\ slots$

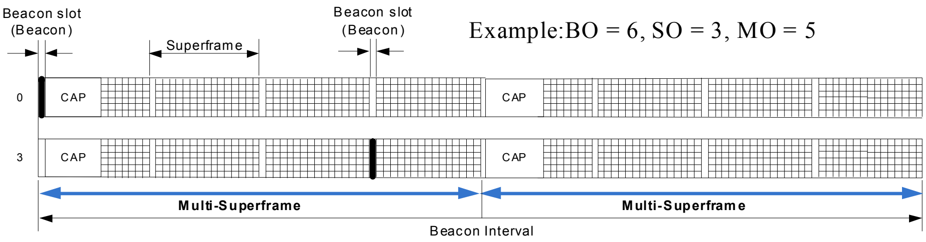


Figure 73.i— CAP Reduction in DSME-based Multi-superframe Structure

7.5.10.5 DSME allocation and management

An Enhanced Guaranteed Time Slot (DSME) functionality allows a DSME-device to operate on the channel within a portion of the superframe that is dedicated (on the PAN) exclusively to that device. An DSME shall be allocated by the destination device, and it shall be used only for communications between the source device and the destination device. A single DSME may extend over one or more superframe slots. The destination device may allocate up to seven DSMEs at the same time, provided there is sufficient capacity in the superframe.

An DSME shall be allocated before use, with the destination device deciding whether to allocate an DSME based on the requirements of the DSME request and the current available capacity in the superframe. DSMEs shall be allocated on a first-come-first-served basis, and all DSMEs shall be placed contiguously at the end of the superframe and after the CAP (or after the beacon slot if CAP reduction is enabled). Each DSME shall be deallocated when the DSME is no longer required, and an DSME can be deallocated at any time by the destination device or the source device that originally requested the DSME. A device that has been allocated an DSME may also operate in the CAP.

A data frame transmitted in an allocated DSME shall use only short addressing.

The management of DSMEs shall be undertaken by both of the destination device and the source device. To facilitate DSME management, the destination device and the source device shall be able to store all the information necessary to manage DSMEs. For each DSME, the destination device and the source device shall be able to store its starting slot, length, and associated device address.

Each DSME requested by the source device must be the transmit DSME for the source device, and the receive DSME for the destination device, so for each allocated DSME, there is no need for a device to store the direction. If a destination device has been allocated an DSME, it shall enable its receiver for the entirety of the DSME. If a data frame is received during an DSME and an acknowledgment is requested, the destination device shall transmit the acknowledgment frame as usual. Similarly, the source device shall be able to receive an acknowledgment frame during the DSME it requested.

A source device shall attempt to request a new DSME only if it is synchronizing with the destination device. The MLME of the source device is instructed to get the timestamp and the parameters of its DSMEs from the destination device by issuing the MLME-DSMEinfo.request primitive, and then the source device will send the DSME information request command frame.

If a source device loses synchronization with the destination device, all its DSMEs allocations shall be lost.

The use of DSMEs is optional.

7.5.10.6 DSME allocation

A DSME-device is instructed to request the allocation of a new DSME through the MLME-GTS.request primitive, with DSME characteristics set according to the requirements of the intended application and DSMEFlag set to TRUE.

To request the allocation of a new DSME, the MLME of the Source device shall send an DSME handshake command (see 7.3.10) to the Destination device. The Characteristics Type subfield of the DSME Characteristics field shall be set to one (DSME allocation) and the Handshake Type subfield shall be set to zero (DSME request). The DSME Length subfield of the DSMEDescriptor field shall be set according to the desired characteristics of the required DSME. The DSME ABT Specification subfield shall be set according to the current allocation status of all one-hop neighborhoods of the Source device.

After sending the DSME handshake request command frame, the source device shall wait for at most *anDSMERequestWaitingTime* symbols, if no DSME handshake reply command frame appears within this time, the MLME of the source device shall notify the next higher layer of the failure. This notification is achieved when the MLME issues the MLME-GTS.confirm primitive (see 7.1.7.2) with a status of NO_DATA.

On receipt of an DSME handshake command frame indicating an DSME allocation request, the Destination device shall first check if there is available capacity in the current multi-superframe, based on the ABT sub-block maintained by the Destination device, the desired length of the request DSME and the ABT sub-block subfield in the DSME handshake request command frame from the Source device. The Multi-superframe shall have available capacity if enough vacant slots exist in both ABT sub-block subfields of the Destination device and the Source device to satisfy the requested length. DSMEs shall be allocated on a first-come-first-served basis by the Destination device provided there is sufficient bandwidth available.

When the Destination device determines whether capacity is available for the requested DSME, it shall generate an DSME descriptor (see 7.3.10.2) with the requested specifications and the 16-bit short address of the requesting source device. If the DSME was allocated successfully, the destination device shall set the DSME Slot Identifier subfield in the DSME descriptor to the multi-superframe slot at which the allocated DSME begins from, the DSME Length subfield in the DSME descriptor to the length of the DSME and the Device short address to the address of the source device. In addition, the destination device shall notify the next higher layer of the newly allocated DSME. This notification is achieved when the MLME of the destination device issues the MLME-GTS.indication primitive (7.1.7.3) with the characteristics of the allocated DSME and the DSMEFlag set to TRUE. If there was not sufficient capacity to allocate the requested DSME, the DSME Slot Identifier shall be set to zero and the length set to the largest DSME length that can currently be supported.

The Destination device shall then include the DSME descriptor in its DSME handshake command frame and broadcast it to its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field shall be set to one (DSME allocation) and the Handshake Type subfield shall be set to one (DSME reply).The DSME ABT Specification subfield shall be set to represent the newly allocated slots.

On receipt of an DSME handshake command frame indicating an DSME allocation reply, the device shall process the DSME descriptor.

If the address in the Device Short Address subfield of the DSME descriptor does not correspond to *macShortAddress* of the device, the device updates its ABT to reflect the neighbor's newly allocated DSME.

If the newly allocated DSME is conflicting with the device's known DSME, the device shall send an DSME handshake command frame to the origin device of the DSME handshake reply command frame. The Characteristics Type subfield of the DSME Characteristics field set to three (DSME duplicate allocation notification) and the Handshake Type subfield set to two (DSME notify), with the DSME Slot Identifier subfield in the DSME descriptor set to the multi-superframe slot at which the DSME duplicate allocated, the DSME Length subfield in the DSME descriptor to the length of the duplicate allocated DSME and the Device short address to the address of the device for which the DSME allocation replied.

If the address in the Device Short Address subfield of the DSME descriptor corresponds to *macShortAddress* of the device, the MLME of the device shall then notify the next higher layer of whether the DSME allocation request was successful. This notification is achieved when the MLME issues the MLME-GTS.confirm primitive with a status of SUCCESS (if the DSME Slot Identifier in the DSME descriptor was greater than zero) or DENIED (if the DSME Slot Identifier in the DSME descriptor was equal to zero or if the length did not match the requested length). After that, the Source device shall broadcast an DSME handshake command frame to all its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field shall be set to one (DSME allocation) and the Handshake Type subfield shall be set to two (DSME notify), with the DSME Slot Identifier subfield in the DSME descriptor set to the value of the multi-superframe slot at which the new allocated DSME begins, the DSME Length subfield in the DSME descriptor to the length of the allocated DSME and the Device short address to the address of the destination device.

On receipt of an DSME handshake command frame indicating an DSME allocation notify, the device shall process the DSME descriptor. The device updates its ABT to reflect the neighbor's newly allocated DSME. If the newly allocated DSME conflicts with the device's known DSME, the device shall send an DSME handshake command frame to the origin device of the DSME handshake notify command frame. The Characteristics Type subfield of the DSME Characteristics field shall be set to three (DSME duplicate allocation notification) and the Handshake Type subfield shall be set to two (DSME notify), with the DSME Slot Identifier subfield in the DSME descriptor set to the multi-superframe slot at which the DSME duplicate allocated, the DSME Length subfield in the DSME descriptor to the length of the duplicate allocated DSME and the Device short address to the address of the device which sent the DSME allocation notify.

On receipt of an DSME handshake command frame indicating an DSME duplicate allocation notification, the device shall reallocate the DSME (see 7.5.10.3).

An example of DSME allocation is shown in Figure 73.j.

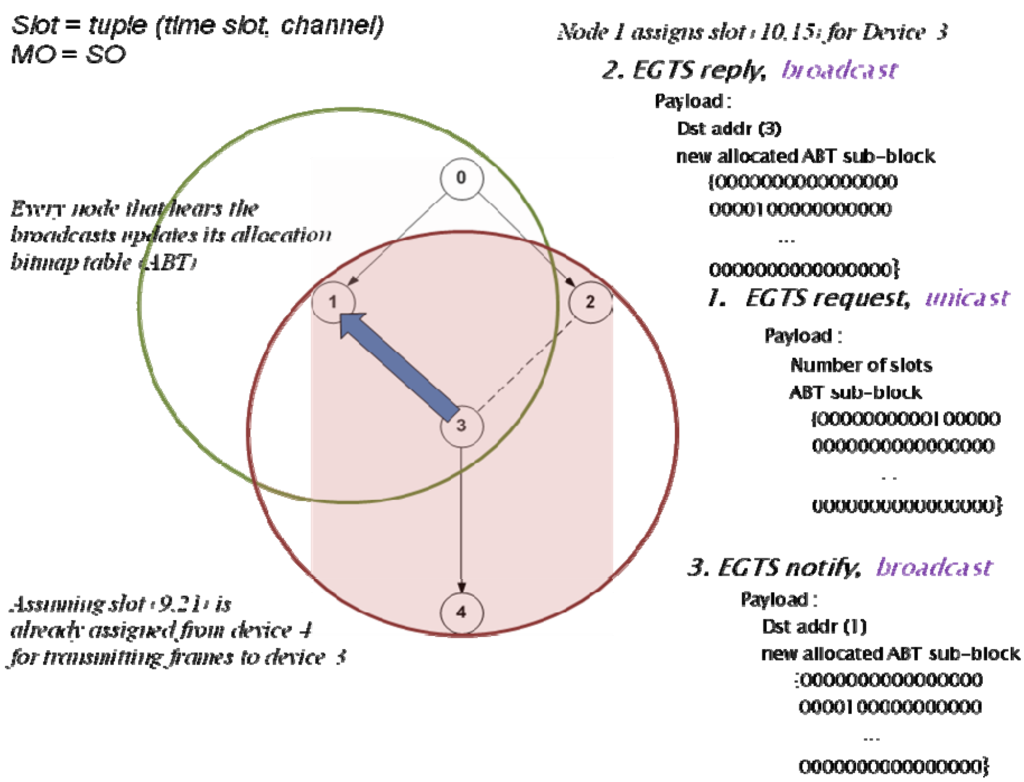


Figure 73.j— Three-way Handshake for DSME Allocation

7.5.10.7 DSME deallocation

The DSME-Source device is instructed to request the deallocation of an existing DSME through the MLME-GTS.request primitive (see 7.1.7.1) using the characteristics of the DSME it wishes to deallocate. The Destination device can request the deallocation of an existing DSME if a deallocation request from the next higher layer, or the expiration of the DSME. From this point onward, the DSME to be deallocated shall not be used by the device, and its stored characteristics shall be reset.

When an DSME deallocation is initiated by the next higher layer of the device, the MLME shall receive the MLME-GTS.request primitive with the DSMEFlag set to TRUE, the Characteristics Type subfield of the DSMECharacteristics parameter set to zero (DSME deallocation) and the DSME Length subfield set according to the characteristics of the DSME to deallocate.

When an DSME deallocation is due to the DSME expiring, the MLME shall notify the next higher layer of the change. This notification is achieved when the MLME issues the MLME-GTS.indication primitive with the DSMEFlag set to TRUE, the DSMECharacteristics to the characteristics of the deallocated DSME and the Characteristics Type subfield set to one.

In the case of any the deallocation of an existing DSME, the MLME shall send the DSME handshake command (see 7.3.10) to the corresponding device (the Source or Destination of which the DSME to be deallocated). The Characteristics Type subfield of the DSME Characteristics field shall be set to zero (DSME deallocation), and the Handshake Type subfield shall be set to zero (DSME request). The DSME Length subfield of the DSMEDescriptor shall be set according to the characteristics of the DSME to deallocate. The DSME ABT Specification subfield shall be set according to the current allocation status of all one-hop neighborhoods of the device request to deallocate the DSME.

After sending the DSME handshake request command frame, the device shall wait for at most *anDSMERequestWaitingTime* symbols, if no DSME handshake reply command frame appears within this time, the MLME of the device shall notify the next higher layer of the failure. This notification is achieved when the MLME issues the MLME-GTS.confirm primitive (see 7.1.7.2) with a status of NO_DATA. Then the device shall determine whether stop using its DSME by the procedure described in 7.5.10.4.

On receipt of an DSME handshake command frame indicating an DSME deallocation request, the device shall attempt to deallocate the DSME.

If the DSME characteristics contained in the command do not match the characteristics of a known DSME, the device shall ignore the request.

If the DSME characteristics contained in the DSME request command match the characteristics of a known DSME, the MLME of the device shall deallocate the specified DSME, update its ABT and notify the next higher layer of the change. This notification is achieved when the MLME issues the MLME-GTS.indication primitive (see 7.1.7.3) with the DSMEFlag set to TRUE, the DSME Characteristics parameter containing the characteristics of the deallocated DSME and the Characteristics Type subfield set to one. Then, the device shall broadcast an DSME handshake command to its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field of the DSME handshake command shall be set to zero (DSME deallocation), and the Handshake Type subfield shall be set to one (DSME reply). The DSME Length subfield in the DSME descriptor to the length of the successfully deallocated DSME and the Device Short Address to the address of the device request deallocate DSME. The DSME ABT Specification subfield shall be set to represent the slots status after successful deallocation.

On receipt of an DSME handshake command indicating an DSME deallocation reply, the device shall process the DSME descriptor.

If the address in the Device Short Address subfield of the DSME descriptor does not correspond to *macShortAddress* of the device, the device updates its ABT to reflect all the neighbor's deallocated DSME.

If the address in the Device Short Address subfield of the DSME descriptor corresponds to *macShortAddress* of the device, the MLME of the device shall then notify the next higher layer of whether the DSME deallocation request was successful. This notification is achieved when the MLME issues the MLME-GTS.confirm primitive with a status of SUCCESS (if the length in the DSME descriptor matched the requested deallocation length) or DENIED (if the length in the DSME descriptor did not match the requested deallocation length). Then, the device shall broadcast an DSME handshake command to all its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field shall be set to zero (DSME deallocation) and the Handshake Type subfield shall be set to two (Notify), with the DSME Slot Identifier subfield and the DSME Length subfield in the DSME descriptor set to the identifier and the length of the DSME deallocated respectively.

On receipt of an DSME handshake command indicating an DSME deallocation notify, the device shall process the DSME descriptor. The device updates its ABT to reflect the neighbor's deallocated DSME.

7.5.10.8 DSME reallocation

A DSME-device shall reallocate the DSMEs to fill the gap result from the deallocation of an DSME or regulate the DSME allocation when the duplicate allocation occurs.

If the DSME reallocation is initiated by the next higher layer of the device, the MLME shall receive the MLME-GTS.request primitive with the DSMEFlag set to TRUE, the Characteristics subfield of the DSMECharacteristics parameter set to two (DSME Reallocation) and the DSME Length subfield set according to the characteristics of the DSME to reallocate.

If the DSME reallocation is due to the receipt of the DSME handshake duplicate allocation notification, the MLME shall notify the next higher layer of the conflicts. This notification is achieved when the MLME issues the MLME-GTS.indication primitive with the DSMEFlag set to TRUE, the DSMECharacteristics set to the characteristics of the duplicate allocation DSME and the Characteristics Type subfield set to three (DSME Duplicate allocation).

If the device instructed to request reallocate DSME is the source device which has requested the allocation of DSME, the MLME shall generate an DSME handshake command (see 7.3.10) to the destination device which has allocated the DSME, with the Characteristics Type subfield of the DSME Characteristics field shall be set to two (DSME reallocation) and the Handshake Type subfield shall be set to zero (DSME request). The DSME Length and the DSME Slot Identifier subfields of the DSMEDescriptor field shall be set according to the desired characteristics of the reallocation DSME. The DSME ABT (Allocation Bitmap Table) Specification subfield shall be set according to the current allocation status of all the one-hop neighborhoods of the device.

On receipt of an DSME handshake command frame indicating an DSME reallocation request, the destination device shall attempt to reallocate the DSME.

If the DSME characteristics contained in the command do not match the characteristics of a known DSME, the destination device shall ignore the request.

If the DSME characteristics contained in the DSME request command match the characteristics of a known DSME, the destination device shall first check if there is available capacity in the current Multi-superframe, based on the ABT sub-block maintained by the destination device, the length of the request reallocation DSME and the ABT sub-block subfield in the DSME handshake request command frame from the Source device. The Multi-superframe shall have available capacity if enough vacant slots exist in both ABT sub-block subfields of the Destination device and the Source device to satisfy the requested reallocation length. DSMEs shall be reallocated on a first-come-first-served basis by the Destination device provided there is sufficient bandwidth available.

If the DSME is successfully reallocated, the destination device shall updates its ABT, and notify the next higher layer of the reallocated DSME by primitive MLME-GTS.indication with the DSMEFlag set to TRUE.

Then the destination device shall broadcast an DSME handshake command to all its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field of the DSME handshake command shall be set to two (DSME reallocation) and the Handshake Type subfield shall be set to one (DSME reply). If the DSME was reallocated successfully, the DSME Slot Identifier subfield in the DSME descriptor shall be set to the multi-superframe slot at which the reallocated DSME begins from, the DSME Length subfield in the DSME descriptor to the length of the reallocated DSME and the DSME ABT Specification subfield shall be set to represent the slots status after reallocation. If there was not sufficient capacity to reallocate the requested DSME, the DSME Slot Identifier shall be set to zero and the length set to the largest DSME length that can currently be supported.

On receipt of an DSME handshake command indicating an DSME reallocation reply, the device shall process the DSME descriptor.

If the address in the Device Short Address subfield of the DSME descriptor does not correspond to *macShortAddress* of the device, the device updates its ABT to reflect the neighbor's reallocated DSME. If the newly reallocated DSME is conflicting with the device's known DSME, the device shall send an DSME handshake command to the origin device of the DSME handshake reallocation reply command frame. The Characteristics Type subfield of the DSME Characteristics field shall be set to three (Duplicate Allocation Notification) and the Handshake Type subfield shall be set to zero (request).

If the address in the Device Short Address subfield of the DSME descriptor corresponds to *macShortAddress* of the device, the MLME of the device shall then notify the next higher layer of whether the DSME reallocation request was successful. This notification is achieved when the MLME issues the MLME-GTS.confirm primitive with a status of SUCCESS (if the DSME Slot Identifier in the DSME descriptor was greater than zero) or DENIED (if the DSME Slot Identifier in the DSME descriptor was equal to zero or if the length did not match the requested length). Then, the Source device shall broadcast an DSME handshake command to all its one-hop neighbors. The Characteristics Type subfield of the DSME Characteristics field shall be set to two (DSME reallocation) and the Handshake Type subfield shall be set to two (DSME notify).

On receipt of an DSME handshake command indicating an DSME reallocation notify, the device shall process the DSME descriptor. The device updates its ABT to reflect the neighbor's reallocated DSME. If the newly reallocated DSME conflicts with the device's known DSME, the device shall send an DSME handshake command to the origin device of the DSME handshake notify command frame. The Characteristics Type subfield of the DSME Characteristics field shall be set to three (DSME Duplicate Allocation Notification) and the Handshake Type subfield shall be set to zero (request).

7.5.10.9 DSME expiration

The MLME of the device shall attempt to detect when a device has stopped using an DSME using the following rules:

— The MLME of the Destination device of DSME shall assume that the source device is no longer using its DSME if a data frame is not received from the source device in the DSME at least every 2**n* multi-superframes, where *n* is defined below.

— The MLME of the Source device of DSME shall assume that the destination device is no longer using its DSME if an acknowledgement frame is not received from the destination device at least every 2**n* multi-superframes, where *n* is defined below. If the data frames sent in the DSME do not require acknowledgment frames, the MLME of the source device will not be able to detect whether the destination device is using the corresponding DSME.

The value of *n* is defined as follows:

$$n = 2^{(8\text{-}macBeaconOrder)} \qquad 0 \le macBeaconOrder \le 8$$

$$n = 1 \qquad 9 \le macBeaconOrder \le 14$$

7.5.10.10 DSME retrieve

If a loss of synchronization occurs before its allocated DSMEs of current superframe starting, the Source device shall be instructed to request the timestamp and the DSME information through the MLME-DSMEinfo.request primitive (see 7.1.18.1).

To request the timestamp and the DSME information, the MLME of the Source device shall send an DSME information request command frame to the Destination device.

On receipt of an DSME information request command, the Destination device shall determine whether it has allocated DSME slots to the requesting device. If so, the MLME of the Destination device shall send an DSME information reply command frame before the end of the Source device's DSME slot of current superframe excepting the beacon slot, including the timestamp and the DSME parameters information to the Source device. Otherwise, the MLME of the Destination device shall send an DSME information reply command frame indicating the failure of the Source device's DSME request at current superframe.

After sending the DSME information request command frame, the Source device shall wait for macDSMEInfoWaitTime symbols, if the DSME information reply command frame indicating a failure or no DSME information reply command frame is received, the MLME of the source device shall notify the next higher layer of the failure by the MLME-DSMEinfo.confirm primitive with a status of NO_DATA.

On receipt of an DSME information reply command frame containing the timestamp and the DSME information, the MLME of the Source device shall notify the next higher layer of the success. This notification is achieved when the MLME issues the MLME-DSMEinfo.confirm primitive with a status of SUCCESS. Then the Source device shall synchronize to the Destination device by using the received timestamp and continue to use its allocated DSMEs during current superframe.

7.5.10.11 DSME change

The Destination device allocates the DSME slots to the Source device according to the first-come-first-served basis. If the Destination device receives an DSME handshake allocation request command from a source device with a higher priority of data transmission when there is no available DSME slots, the Destination device shall reduce part or all of the DSME slots which are being used for the lower priority data transmission and allocate the reduced DSME slots for the higher priority data transmission. If the Destination device receives more than one DSME handshake allocation request command with the same priority of data transmission, the Destination device shall allocate the DSME slots according to the first-come-first-served basis.

After the higher priority data transmission in the DSME slots is finished, if there are no more DSME handshake allocation request commands with higher priority of data transmission are received, the Destination device shall restart the DSME slots for the lower priority data transmission which were reduced previously. Otherwise, the higher priority data transmission will use the DSME slots first. If the lower priority data transmission has been suspended for a certain time, the Destination device shall allocate the next available DSME slots to the corresponding Source device (see Table 86.h in 7.4.2.5).

The procedure of DSME change shall be initiated when a Destination device wants to reduce or restart the allocated DSMEs through the MLME-GTS.request primitive (see 7.1.7.1).

When an DSME change is initiated by the next higher layer of the Destination device, the MLME shall receive the MLME-GTS.request primitive with the DSMEFlag set to TRUE, the DSME Characteristics Type subfield of the DSME Characteristics parameter set accordingly (i.e., 101 for DSME Reduce or 110 for DSME Restart)

To request the change of an existing DSME, the MLME of the Destination device shall send the DSME handshake request command frame (see 7.3.10) to the Source device. The DSME Characteristics Type subfield of the DSME Characteristics field shall be set accordingly (i.e., 101 for DSME Reduce or 110 for DSME Restart), and other subfields set according to the characteristics of the DSME which the Destination device requests the Source device to change its original DSME to.

The DSME handshake request command frame for DSME change contains an acknowledgment request (see 7.3.12), and the Source device shall confirm its receipt of DSME handshake change request command frame by sending an acknowledgment frame to the destination device.

On receipt of the acknowledgment from the source device, the MLME of the Destination device shall notify the next higher layer of the DSME change. This notification is achieved when the MLME issues the MLME-DSME.confirm primitive (see 7.1.20.1.3) with a status of SUCCESS, the DSMEFlag set to TRUE, the DSME Characteristics Type subfield of the DSMECharacteristics parameter set to 101 for DSME Reduce or 110 for DSME Restart accordingly, and other subfields set according to the characteristics of the DSME which the Destination device requests the Source device to change its original DSME to.

On receipt of an DSME handshake request command frame for DSME change from the destination device, the Source device shall immediately change its DSME according to the DSME Characteristics field in the DSME handshake change request command frame. Then the MLME of the Source device shall notify the next higher layer of the change. This notification is achieved when the MLME issues the MLME-GTS.indication primitive (see 7.1.20.1.4) with an DSMECharacteristics parameter set according to the characteristics of the DSME which the Destination device requests the Source device to change its original DSME to.

7.5.10.12 Robust DSME allocation

If the data transmitted in the DSME requires higher transmission reliability, the device is instructed to request the allocation of a new Channel Hopping DSME through the MLME-GTS.request primitive with the DSMEFlag set to TRUE and the DSME Characteristics type subfield set to 100 (Robust DSME Allocation).

To request the allocation of a new Robust DSME, the MLME shall send an DSME handshake command to the Destination device with the DSME Characteristics type subfield of the DSME characteristics parameter set to 100 (Robust DSME Allocation) and the Handshake Type subfield shall be set to zero (DSME request). The DSME Length subfield of the DSMEDescriptor shall be set according to the desired characteristics of the required DSME. The DSME ABT Specification subfield shall be set according to the current allocation status of all one-hop neighborhoods of the Source device.

After sending the DSME handshake request command frame, the source device shall wait for at most *anDSMERequestWaitingTime* symbols, if no DSME handshake reply command frame appears within this time, the MLME of the source device shall notify the next higher layer of the failure. This notification is achieved when the MLME issues the MLME-GTS.confirm primitive (see 7.1.7.2) with a status of NO_DATA.

On receipt of an DSME handshake command frame indicating a Robust DSME Allocation request, the Destination device shall first decide whether allocating the Robust DSME or the regular DSME based on its own availability.

If the Destination device decides to allocate the Robust DSME, it shall behave the same as allocating the regular DSME except allocating DSME ABT in a Channel Hopping mode, i.e. adjacent slots will be allocated different channels, and the channel selection depends on the Destination device's knowledge of current channel condition. The remaining parts of the Robust DSME allocation are same with the regular DSME allocation. Within the Robust DSME, the device switches to the next channel every slot according to the channel sequence in the ABT.

7.5.10.13 Beacon Scheduling

When a new node wants to join a network, first it scans the channel. The new device uses the MLME-SCAN.request primitive in order to initiate a channel scan over a given list of channels. It searches for all coordinators transmitting beacon frames within the maximum BI period. Then these neighboring nodes would share their information of beacon bitmap with the new node. The beacon bitmap indicates the beacon frame allocation information for neighboring nodes. This field is expressed by bitmap method which orderly represents the schedule of beacons. Corresponding bit shall be set to 1 if a beacon is allocated in that SD. The new node will search the SD which is vacant (not set to 1) in all of the received beacon bitmap of beacon frame. Once new node finds vacant SD, It uses it as its own SD.

There can be beacon slot collision when two or more nodes are trying to compete for same SD slot number. As shown in Figure 73.k, node D and node E are new nodes that join the network. These new nodes will receive the beacon bitmap from their neighboring nodes. As it can be seen that node A is a common neighboring node. Thus it can be the case when both new nodes E and D request same vacant SD number within same CAP. This happens due to hidden node problem, because node E and D are hidden to each other, and cannot listen to each others transmission. When node A receives the SD number 2 request, within same CAP, by nodes E and D then it would determine which node has requested first. Node A will reply the beacon collision notification to the node which has requested later.

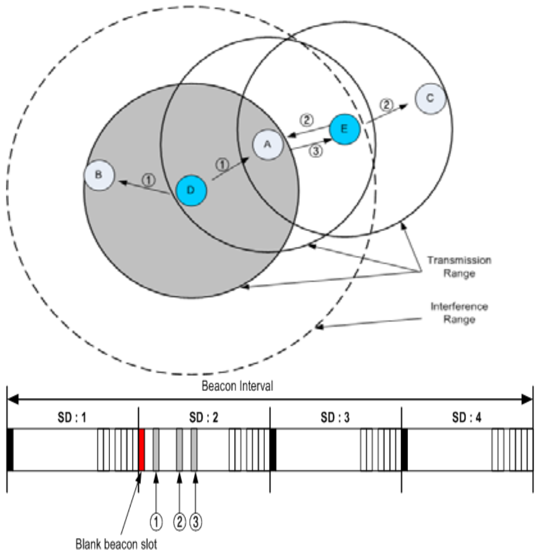


Figure 73.k— New node joining the network

7.5.10.14 DSME Synchronization

The new node discovers its neighboring node through the scanning process. Then it associate with one of the neighboring node in order to be part of the network. The node which the new node associates with is called its parent node.

Now the coordinator knows that the device would track its beacon. Thus the coordinator will determine the transmission time of its beacon. This beacon timestamp value is set just before transmitting the beacon frame. When the device gets this beacon timestamp value then it can synchronize with its parent coordinator.

The effect of collision is inevitable in multiple nodes scenario, where more than one node try to use the same channel at the same time. In the case of collision, the node wait for some backoff duration and then it tries to re-send. Thus in case of collision, beacon transmission timestamp does not become valid. In order to avoid this problem, the coordinator sets *macDefferedBeaconUsed* value to be TRUE. When the device notice *macDefferedBeaconUsed* value to be TRUE then it knows that coordinator uses CCA for transmitting its beacon. Also the coordinator would add the number of tries it made for successful transmission. If the coordinator would be able to send its beacon after 3^{rd} try then it would set *DefferedBeaconFlag* value to be 3. On receipt of the beacon, the device would exactly knows when the beacon is sent (by adding beacon timestamp with (*DefferedBeaconFlag* value * 20 symbols)). Thus perfect time synchronization becomes possible.

7.5.10.15 Passive channel scan

Channel Diversity Specification in the received beacon frame shall update the value of ChannelDiversitySpecification in PANDescriptor. This value is sent to the next higher layer via the MLME-SCAN.confirm primitive. The value of Channel Offset subfield in the received beacon shall update the value of *macChannelOffsetBitmap* in MAC PIB attributes. For instance, if ChannelOffset is set to 0x01, the value of *macChannelOffsetBitmap* corresponding channel shall set to ‘1’. Thus, the value of *macChannelOffsetBitmap* shall represent if the channel offset value is used among one hop neighbor devices.

7.5.10.16 Updating superframe configuration and channel PIB attributes

Subclause 7.5.2.3.4 applies. For DSME-devices the following is additionally required.

If a PAN uses both of the DSME and Channel Hopping mode (i.e., DSMEFlag is TRUE and ChannelDiversityMode is ‘1’), the MAC sublayer shall update the values of *DCHDescriptor* with the values of the DCHDescriptor parameter.

7.5.10.17 Beacon generation

Subclause 7.5.2.4 applies. For DSME-devices the following is additionally required.

If DSME and Channel Hopping mode (i.e., DSMEFlag is TRUE and ChannelDiversityMode is ‘1’) are used in the PAN, the MAC sublayer shall set the Channel Diversity Specification field of the beacon frame. The value of ChannelOffsetBitmap field, representing channel offset used among one hop neighbor devices, shall be set to the value of *macChannelOffsetBitmap* in MAC PIB attributes.

7.5.10.18 Coexistence of beacon-enabled and non-beacon-enabled mode

PANs that contain both the devices of beacon-enabled mode and the devices of non-beacon-enabled mode shall include the Connection Devices.

The device of beacon-enabled mode shall either transmit periodic beacon or track the beacon for communication in the PAN. The device of non-beacon-enabled mode shall neither transmit periodic beacon nor track the beacon for communication in the PAN, and it shall either request the data from other devices of non-beacon-enabled mode or transmit the data upon receipt of the data request commands from the devices of non-beacon-enabled mode in the PAN. Both modes shall be operated in the Connection Devices at the same time, which means the Connection Device can transmit or receive frame in either beacon-enabled mode or non-beacon-enabled mode.

In order to maintain the beacon order consistent, the Connection Device shall store the superframe structure parameters in the beacon it tracks and use those parameters in its own beacon to transmit. Moreover, the Connection Device shall send the stored parameters to the neighbors actively or upon receipt of a request from other devices. In order to avoid frame conflict, the Connection Device shall not communicate with the device of non-beacon-enabled mode when it is tracking or transmitting beacon or communicating in beacon-enabled mode.

7.5.10.19 DSME-Superframe structure

Subclause 7.5.1.1 applies and for DSME-devices with Low Energy Superframe Support shall apply the following in addition.

If BO = 15 and macLowEnergySuperframeSupported is FALSE, the coordinator shall not transmit beacon frames except when requested to do so, such as on receipt of a beacon request command. The value of macSuperframeOrder shall be ignored if BO = 15. Moreover, if macLowEnergySuperframeSupported is TRUE the coordinator shall not transmit beacon frames except when requested to do so, regardless of BO value.

If BO = 15 and macLowEnergySuperframeSupported is FALSE, the superframe shall not exist (the value of macSuperframeOrder shall be ignored). This MAC PIB attribute enables basic low energy performance that is defined by the superframe structure in the beacon-enabled PAN, not driven by LE-MAC operations (see 7.5.11).

7.5.10.20 DSME-Contention access period (CAP)

Subclause 7.5.1.1.1 applies and for DSME-devices with Low Energy Superframe Support shall apply the following in addition.

All frames, except acknowledgment and data frames that quickly follows the acknowledgment of a data request command (see 7.5.6.3), transmitted in the CAP shall use a slotted CSMA-CA mechanism to access the channel. A device transmitting within the CAP shall ensure that its transaction is complete (i.e., including the reception of any acknowledgment) one IFS period (see 7.5.1.3) before the end of the CAP when *macLowEnergySuperframeSupported* is FALSE. If this is not possible, the device shall defer its transmission until the CAP of the following superframe. When *macLowEnergySuperframeSupported* is TRUE, on the other hand, transaction shall be ensured to be completed one IFS period before the end of the inactive period. Finally, if a device senses frame in CAP that does not end within CAP when *macLowEnergySuperframeSupported* is set to TRUE, the device may continue receiving the frame until it ends before the end of the inactive period. When *macLowEnergySuperframeSupported* is TRUE, the coordinator shall not locate DSMEs in order to avoid the interference from the frames in CAP. When *macLowEnergySuperframeSupported* is TRUE, the coordinator shall notify the devices that already associated or intend to associate the condition of *macLowEnergySuperframeSupported* in the beacon frames.

7.5.10.21 DSME-Incoming and outgoing superframe timing

Subclause 7.5.1.2 applies and for DSME-devices shall apply the following in addition.

The beacon order and superframe order may be equal for all superframes on a PAN. All devices may interact with the PAN only during the active portion of a superframe.

7.5.10.22 Multi-Channel adaptation

7.5.10.22.1 General

Single common channel approach may not be able to connect all devices in the PAN. The variance of channel condition can be large and channel asymmetry between two neighboring device can happen. Multi-channel adaptation is a solution to handle such case.

Two types of multi-channel adaptation is specified, which are synchronous multi-channel adaptation and asynchronous multi-channel adaptation. The synchronous multi-channel adaptation is performed in beacon-enabled mode, and is handled by DSME as described in 7.5.4.4. The asynchronous multi-channel adaptation is performed in non-beacon mode, and is described in this subclause.

7.5.10.22.2 Receiver-based communication

It is possible that there exists no common channel that two devices can communicate in DSME mode as there are many available channels. In that case, each device selects its designated channel based on its local link quality, and keep listening to its designated channel. When another device wants to communicate with it, the sender device shall switch to the designated channel of the receiver device and transmit a DATA frame. Then the sender device shall switch back to its own designated channel and keep listening. On receipt of the data frame from the sender device, the receiver device shall switch to the designated channel of the sender device and transmit an ACK frame (if requested). After sending the acknowledge frame, the receiver device shall switch back to its own designated channel and keep listening at last.

Figure 73.l illustrated the receiver-based communications.

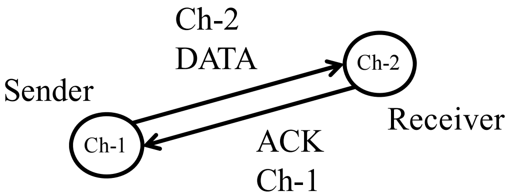


Figure 73.l— Receiver-based communication

7.5.10.23 Asymmetric multi-channel active scan

An asymmetric multi-channel active scan allows device to detect the designated channel of each coordinator or detect the best channel for the device.

The asymmetric multi-channel active scan over a specified set of logical channels is requested using the MLME-SCAN.request primitive with the ScanType parameter set to 0x04.

For each logical channel, the device shall first switch to the channel, by setting *phyCurrentChannel* and *phyCurrentPage* accordingly, and send a multi-channel beacon request command (see 7.3.11). Upon successful transmission of the multi-channel beacon request command, the device shall enable its receiver for [*aBaseSuperframeDuration* * (2^n + 1)] symbols, where *n* is the value of the *ScanDuration* parameter. During this time, the device shall reject all non-beacon frames and record the information contained in all unique beacons in a PAN descriptor structure (see Table 55 in 7.1.5.1.1). After this time, the device shall switch to the next channel and repeat the same procedure. The device shall stop repeating this procedure after visiting every channel twice.

If *linkqualityscan* flag is FALSE, the device may stop after it receives a beacon and decide the current channel as its designated channel. If *linkqualityscan* flag is TRUE, the device make decision on its designated channel comparing LQI or RSSI of the received beacons.

On receipt of the multi-channel beacon request command, the coordinator shall transmit a beacon (see 7.2.2.1) over a set of logical channels specified in the asymmetric multi-channel beacon request command. Upon successful transmission of the beacon, the coordinator shall switch to the next channel after [*aBaseSuperframeDuration* * (2^n + 1)] symbols, where *n* is the value of the *ScanDuration* parameter, and send another beacon. The coordinator shall repeat the same procedure over all the logical channels specified in the asymmetric multi-channel beacon request command.

7.5.10.24 Multi-Channel Hello

Multi-channel hello mechanism allows a device to announce its designated channel to its one-hop neighbor devices.

After successfully performing the asymmetric active scan and the association, the device shall transmit the same multi-channel hello command on each channel sequentially starting from its designated channel. The device can request multi-channel hello reply by setting the Hello Reply Request of the multi-channel hello command to ‘1’. When its neighbors receives the multi-channel hello command with Hello Reply Request set to'1', each neighbor shall transmit a multi-channel hello reply command on designated channel of the requesting device.

7.5.10.25 Three-way Handshake Channel Probe

If the channel condition is bad, the device can probe other channels and switch to a better channel. After switching to the new channel, the device shall broadcast a multi-channel hello command to its one-hop neighbors to notify the new channel.

The channel probe over a specified logical channel is requested using the MLME-SCAN.request primitive with the ScanType parameter set to 0x05.

The device will check the condition of its designated channel by using the three-way handshake mechanism. The procedure of the three-way handshake channel probing is described as follows.

The request device sends a channel probe request command frame to one of its neighbors on the designated channel of the neighbor. On receipt of the channel probe request command, the neighbor sends a channel probe reply frame back to the request device on the originator's channel indicating in the channel probe request command. The request device shall check the LQI or RSSI of the channel probe reply frame upon receiving it. The request device determines that the link quality of the channel is bad if the device have not received the channel probe reply frame after [*aBaseSuperframeDuration* * (2^n + 1)] symbols from the reception of probe reply, where *n* is the value of the *ScanDuration* parameter.

7.5.11 LE-Transmission, reception and acknowledgement

7.5.11.1.1 Coordinated Sampled Listening (CSL)

7.5.11.1.1.1 General

The coordinated sampled listening (CSL) mode is turned on when the PIB attribute macCSLPeriod is set to non-zero and turned off when macCSLPeriod is set to zero. In CSL mode, transmission, reception and acknowledgement work as follows. Figure 73.m illustrates the basic CSL operations.

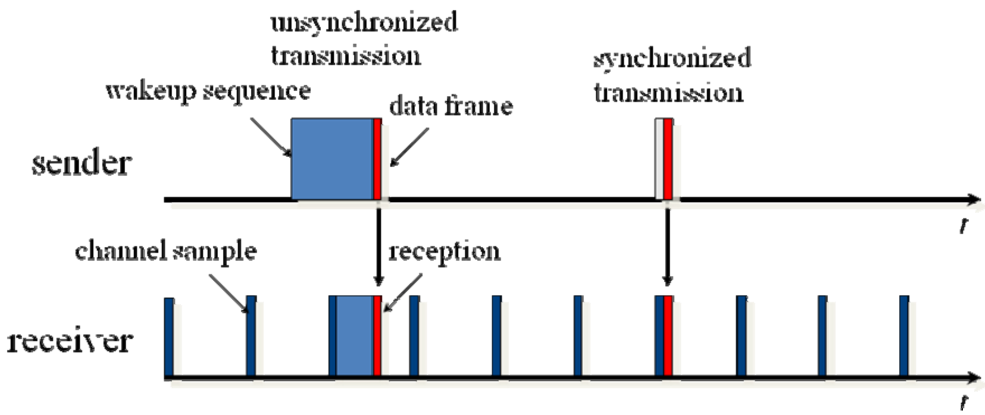


Figure 73.m— Basic CSL operations

7.5.11.1.1.2 CSL idle listening

During idle listening, CSL performs a channel sample every macCSLPeriod milliseconds. If the channel sample does not detect energy on the channel, CSL disables receiver for macCSLPeriod milliseconds and then perform the next channel sample. If the channel sample receives a wakeup frame, CSL checks the destination address in the wakeup frame. If it matches macShortAddress, CSL disables receiver until the

Rendezvous Time (RZTime) in the wakeup frame from now and then enables receiver to receive the payload frame. Otherwise, CSL disables receiver until RZTime from now plus the transmission time of the payload frame and the secure acknowledgment frame and then resume channel sampling.

7.5.11.1.1.3 CSL transmission

Each CSL transmission of a payload frame is preceded with a sequence of back-to-back wakeup frames (wakeup sequence).

7.5.11.1.1.4 Unicast transmission

In unicast transmissions, the wakeup sequence length can be long or short based on the following two cases:

- Unsynchronized transmission: This is the case when the MAC layer does not know the CSL phase and period of the destination device. In this case, the wakeup sequence length is macCSLMaxPeriod.
- Synchronized transmission: This is the case when the MAC layer knows the CSL phase and period of the destination device. In this case, the wakeup sequence length is only the guard time against clock drift based on the last time when CSL phase and period updated about the destination device.

If the next higher layer has multiple frames to transmit to the same destination, it can set the FCF frame pending bit to 1 in all but the last frame to maximize the throughput.

CSL unicast transmission is performed in the following steps by the MAC layer:

a) Perform CSMA-CA to acquire the channel
b) If the previous acknowledged payload frame to the destination has the frame pending bit set and is within macCSLFramePendingWaitT, go to step 5.
c) If it is a synchronized transmission, wait until the destination device's next channel sample.
d) For the duration of wakeup sequence length (short or long)
    1) Construct wakeup frame with the destination short address and remaining time to payload frame transmission (at the end of wakeup sequence)
    2) Transmit wakeup frame
e) Transmit payload frame
f) Wait for up to macSecAckWaitDuration symbol time for the secure acknowledgement frame if the ack request subfield in the payload frame is set to 1.
g) If the secure acknowledgment frame is received, update CSL phase and period information about the destination device from the acknowledgment CSL sync field.
h) If the secure acknowledgement frame is not received, start retransmission process.

7.5.11.1.1.5 Multicast transmission

Multicast transmission is the same as unicast transmission except the following:

- It is always unsynchronized transmission.
- The destination address in wakeup frames is set to 0xffff.

7.5.11.1.1.6 Utilizing the optional CSL sync field

Selectively the next higher layer may set the CSL sync bit in FCF in a frame to propagate CSL phase and period information among the neighboring devices. When the bit is set, the MAC layer automatically appends the CSL sync fields to the end of MHR.

7.5.11.1.1.7 CSL reception

When a payload frame is received, the MAC layer performs the following steps:

a) Immediately send back a secure acknowledgment frame with the destination address set as the transmitting device and its own CSL phase and period filled in the CSL sync field. The acknowledgment frame can be optionally authenticated and/or encrypted depending on the current security mode.

b) If CSL sync bit in the received payload frame is set to 1, the CSL phase and period information about the transmitting device is updated with the information in the CSL sync field.

c) If FCF frame pending bit in the received payload frame is set to 1, keep receiver on for macCSLFramePendingWaitT milliseconds before going back to CSL idle listening. Otherwise, start CSL idle listening.

7.5.11.1.1.8 CSL over multiple channels

When macCSLChannelMask is set to non-zero, the CSL operations are extended to all the channels selected in the bitmap. CSL idle listening performs channel sample on each channel from the lowest number to the highest in a round-robin fashion. In the unsynchronized case, CSL transmission transmits a wakeup sequence of the length *number_of_channels*macCSLMaxPeriod* before each payload frame. In the synchronized case, CSL transmission calculates the next channel sample time and channel number and transmits at the next channel sample time on the right channel with a short wakeup sequence. In this case, CSL phase is the duration from now to the next channel sample on the first channel selected in macCSLChannelMask.

7.5.11.1.1.9 Turning off CSL mode to reduce latency

The next higher layer has the option to turn off sampled listening and stop sending wakeup sequences to reduce latency for urgent messages. This assumes that the higher layer manages the coordination between the sender and receiver in turning on and off sampled listening. To turn off sampled listening, the higher layer simply sets macCSLPeriod to zero. To turn on sampled listening, the high layer restores macCSLPeriod to their previous non-zero values. Similarly, to stop sending wakeup sequences, the higher layer sets macCSLMaxPeriod to zero and restores it to its previous value to return to normal CSL mode. To request a neighboring device to turn off sampled listening, the higher layer must send a frame to the device with frame pending bit set to 1. This prevents CSL from turning off the radio before the request is processed.

7.5.11.2 Receiver Initiated Transmission (RIT)

7.5.11.2.1 General

The Receiver Initiated Transmission (RIT) is an alternative low energy MAC for non beacon-enabled PAN (BO=15). RIT mode is turned on when PIB attribute macRitPeriod is set to non-zero value and is turned off when macRitPeriod is set to zero. In RIT mode, transmission, reception and acknowledgement work as follows.

RIT mode is applicable to low duty cycle, low traffic load type of applications and especially suitable in the case that consecutive radio emission time is limited by regional or national regulation (e.g., 950MHz band in Japan). macCSLPeriod (in coordinated sample listening) and macRitPeriod cannot be set to non-zero value at the same time. Figure 73.n illustrates the basic RIT operations. Figure 73.o illustrates the RIT operations when RIT data request command payload carries schedule information (see 7.3.14.3).

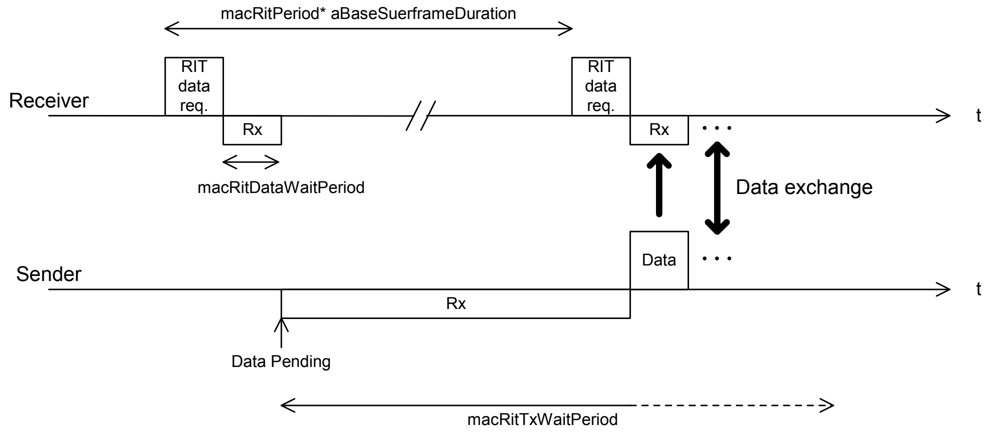


Figure 73.n— Basic RIT operations

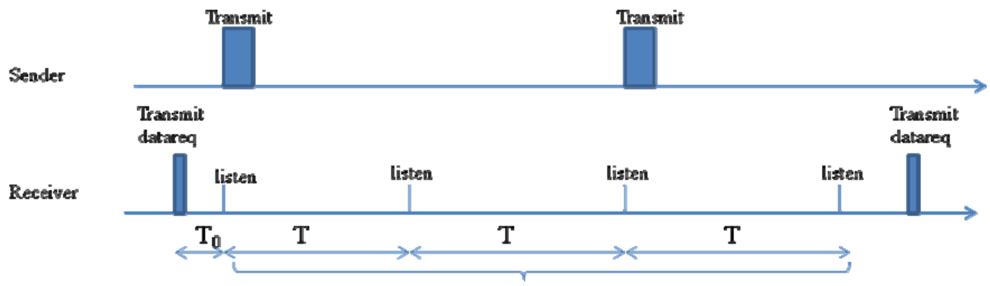


Figure 73.o— RIT operations when datareq carries schedule information

7.5.11.2.2 Periodical RIT data request transmission and reception

In RIT mode, a device transmits RIT data request command every macRitPeriod using unslotted CSMA-CA. The destination address of the command may be broadcast address (0xffff) or the address of intended transmitter of data (associated coordinator). The command may optional contain a 4-octet payload defined in 7.3.14.4.3. When the command carries no payload, after the transmission of RIT data request command frame, the device listens for macRitDataWaitPeriod for incoming frame (except RIT data request frame) and goes back to idle state till the next periodical transmission of RIT data request command. When a device is in the receiving state after transmission of RIT data request command, RIT data request command frame from another device shall be discarded. When data request command carries a 4-octet payload (time to 1st listen T0, number of repeat N, repeat listen interval T), the device goes back to sleep for T0 period of time then listen for macRitDataWaitPeriod before going back to sleep. The first listen on, it repeats a listen interval of macRitDataWaitPeriod every T period of time for N times. The device shall start listening slightly before each scheduled listen time based on a guard time computed from possible clock skew since the last data request command transmission.

Upon reception of a data frame after the transmission of RIT data request command, it notifies its arrival to the next higher layer by instigating MCPS-DATA.indication. Upon reception of a data frame with error (FCS or security), it notifies its erroneous reception to the next higher layer by instigating MLME-FRAME-ERROR.indication.

At this point (instigation of MCPS-DATA.indication or MLME-FRAME-ERROR.indication), the device may set *macRitPeriod* primitive to zero (RIT off) at the discretion of the next higher layer. If this is the case, it will stop periodical transmission of RIT data request command and become always active until *macRitPeriod* primitive is set to non zero value by the next higher layer again. During this period (*macRitPeriod* equals to zero), all transactions will be handled as those of normal non beacon-enabled PAN (RxOnWhenIdle: False).

Figure 73.p shows the Message sequence chart for starting RIT mode.

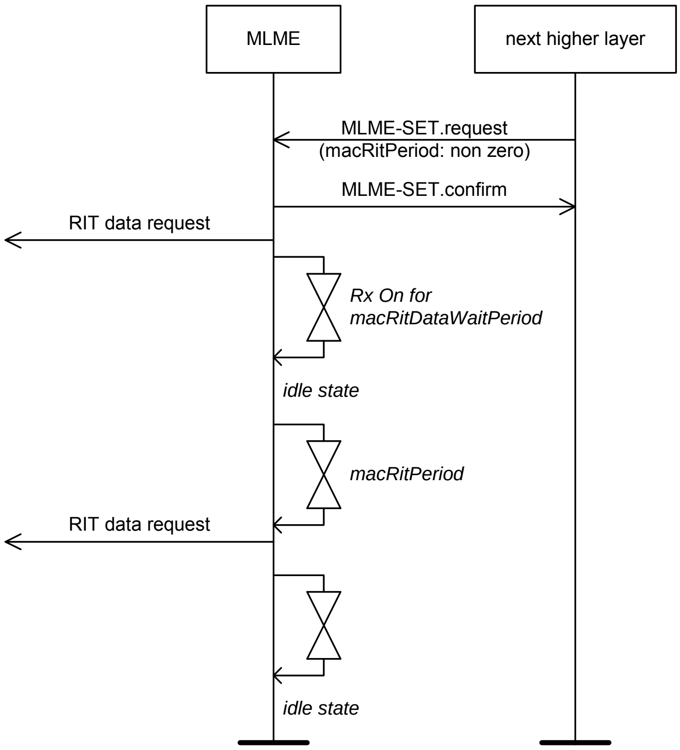


Figure 73.p— Message sequence chart for starting RIT mode

7.5.11.2.3 Data transmission in RIT mode

In order to transmit data frame in RIT mode, MCPS-DATA.request primitive (with TxOption indirect) shall be instigated by the next higher layer at first. When the primitive is instigated, the device shall stop its periodical transmission of RIT data request, enable its receiver and wait reception of RIT data request command frame from neighboring devices for at most macRitTxWaitTime. During period, all other frames except RIT data request command shall be discarded.

Upon reception of RIT data request command frame, the MAC sublayer sends the pending data with a use of unslotted CSMA-CA. In case that the Destination PAN identifier field and the Destination address field of the received RIT data request command are broadcast (0xffff) and the DstPANId and DstAddr parameters of instigated MCPS-DATA.request are also broadcast, the Destination PAN identifier field and the Destination Address field of the outgoing data frame shall be set as the Source PAN identifier field and the Source Address field of the received RIT data request command, respectively.

At the completion of data transmission, MCPS-DATA.confirm shall be instigated by the MAC sublayer to the next higher layer. At this point, the device shall restart its transmission of periodical RIT data request transmission. Also at this point, the device may set *macRitPeriod* primitive to zero (RIT off) at the discretion of the next higher layer. If this is the case, it will continue to stop periodical transmission of RIT data request command and become always active until *macRitPeriod* primitive is set to non zero value by the next higher layer again. During this period (*macRitPeriod* equals to zero), all transactions will be handled as those of normal non beacon-enabled PAN (RxOnWhenIdle: False).

When the data request commands carry the listen schedule payload, the device can either wait to receive a data request frame from the receiving device as described above, or sleep until the next scheduled listen time by the receiving device then wakeup to transmit the intended frame.

7.5.11.2.4 Multicast transmission

Multicast transmission shall not be supported in RIT mode.

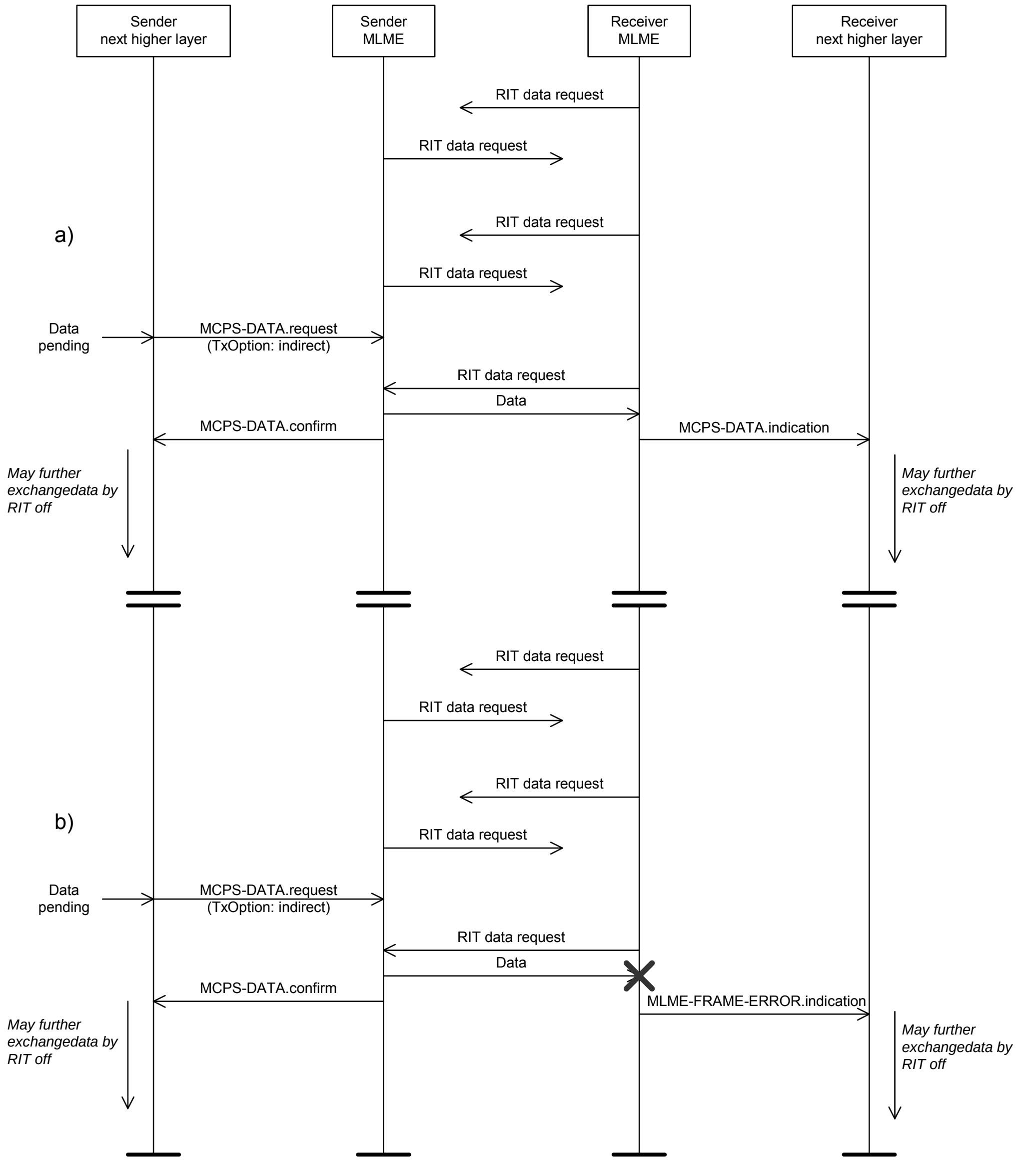


Figure 73.q— Message sequence chart for data transmission in RIT mode

7.6 Security suite specifications

Editorial note RS:

Replace this clause entirely by the corresponding clause of the draft text submitted to the Editing Team by August 15, 2009. This takes into account certain errors that are to be tackled with the Corrigendum to 802.15.4-2006, as also discussed during the IEEE 802 meeting, Atlanta, Georgia, November 10-15, 2009 (cf., e.g., 09/782r1). During that discussion, it was suggested that for TG4e editing purposes, one could anticipate the Corrigendum to include these updates. Details to be looked up in the minutes of that meeting.

7.6.1 PIB security material

7.6.2 Auxiliary security header

7.6.3 Security operations

7.6.3.1 Integer and octet representation

7.6.3.2 CCM* Nonce

Insert at the end of 7.6.3.2 the following text.

When operating in TSCH mode (including when in Listen waiting to join a TSCH network), the nonce shall be formatted as shown in Figure 77.a.

Octets: 8	**5**
Source address	ASN

Figure 77.a—CCM* Nonce in TSCH mode

The source address shall be set to the extended address aExtendedAddress of the device originating the frame, and ASN shall be set to the 5 LSBs of the absolute slot number, i.e. the number of timeslots elapsed in the network (see 7.5.1.5.1).

7.7 Message sequence charts illustrating MAC-PHY interaction

Annex L
(informative)

Bibliography

Annex L.1 will disappear in the final version. The references are given for the reviewer to get more background information for the new technologies in 802.15.4e.

L.1 Documents for MAC enhancements in support of LL-applications

— 15-09/0254r0 Proposal for Factory Automation presentation of proposal for factory automation at March 09 IEEE 802.15.4e meeting

— 15-08/0827r0 Shared Group Timeslots presentation with further details on Shared Group Timeslots

— 15-09/0228r0 Proposal for Factory Automation text of proposal for factory automation at March 09 IEEE 802.15.4e meeting

— 15-08/0420r2 Extending the MAC Superframe of 802.15.4 Spec presentation with separate GACK mechanism

— 15-08/0503r0 Preliminary Proposal for Factory Automation presentation of preliminary proposal for factory automation at July 08 IEEE 802.15.4e meeting

— 15-08/0571r1 Proposal for Factory Automation presentation of proposal for factory automation at September/November 08 IEEE 802.15.4e meetings

— 15-08/0572r0 Proposal for Factory Automation text of proposal for factory automation at September 08 IEEE 802.15.4e meeting

Insert the following Annex M before Bibliography (Annexes H, I, J, K, and L are used in existing Amendments).

Annex M
(informative)

Requirements of industrial and other application domains

M.1 General

The intentions of these add-ons are to enhance and add functionality to the 802.15.4-2009 MAC to

a) better support the industrial markets and

b) permit compatibility with modifications being proposed within the Chinese WPAN.

This functionality will facilitate industrial applications (such as addressed by IEC 62591 and the ISA100.11a), and those enhancements defined by the Chinese WPAN standard that aren't included in the Amendment of TG4c.

Industrial applications have requirements that are not addressed by the edition 2009 such as low latency, robustness in the harsh industrial RF environment, and determinism.

The Chinese Wireless Personal Area Network (CWPAN) standard has identified enhancements to improve network reliability and increase network throughput to support higher duty-cycle data communication applications.

This amendment addresses coexistence with wireless protocols such as 802.11, 802.15.1, 802.15.3, and 802.15.4.

Specifically, the MAC enhancements are grouped into two categories:

a) Industrial and other application domains such as Process automation, Factory automation and

b) Additional functional improvements such as Low energy.

To identify easier the specific amendments to which category these apply in the normative clauses, the specific subclauses are named with the following acronyms in the order as they appear here.

a) Time Slotted Channel Hopping, e.g. for Process automation, (TSCH).

b) Low latency networks, e.g. for Factory automation, (LL).

c) Distributed Synchronous Multi-Channel Extension, e.g. for Process automation and Commercial applications, (DSME).

d) Blink frame, e.g. ???

Comment [W54]: TBD by TG4f

e) (RFID) reserved for future requests.

Additional functional improvements are:

a) Low Energy (LE)

b) Enhanced Beacon request

c) Overhaed reduction and enhanced Security

d) MAC Performance Metrics

The convention as used in Clause 7 is that same headings needed for different solutions based on different requirements have a prefix as the given acronyms above to differentiate the subclauses.

EXAMPLES;

— TSCH-Heading

— LL-Heading

— DSME-Heading

M.2 Time Slotted Channel Hopping (TSCH)

Typical parts of the application domain of process automation are facilities for

- — Oil & gas industry,
- — Food & beverage products,
- — Chemical products
- — Pharmaceutical products
- — Water/waist water treatments
- — etc.

For this application domain exists the following major requirements:

- — IEEE 802.15.4 header extensions for mesh support
    - Additional addresses (source, destination)
    - Sequence number
    - TTL („transmissions to live“)
- — Framework for choosing path selection mechanisms
- Path selection protocol
- Link metrics

M.3 Low latency networks (LL)

M.3.1 Typical application domains for LL-networks

Typical parts of the application domain of low latency networks are facilities for

- — Factory automation as for automotive manufacturing
- — Robots
- — Suspension tracks
- — Portable machine tools
- — Milling, turning
- — Robot revolver
- — Filling
- — Cargo
- — Airport logistics
- — Post
- — Packaging industry
- — Special engineering
- — Conveyor technique
- — etc.

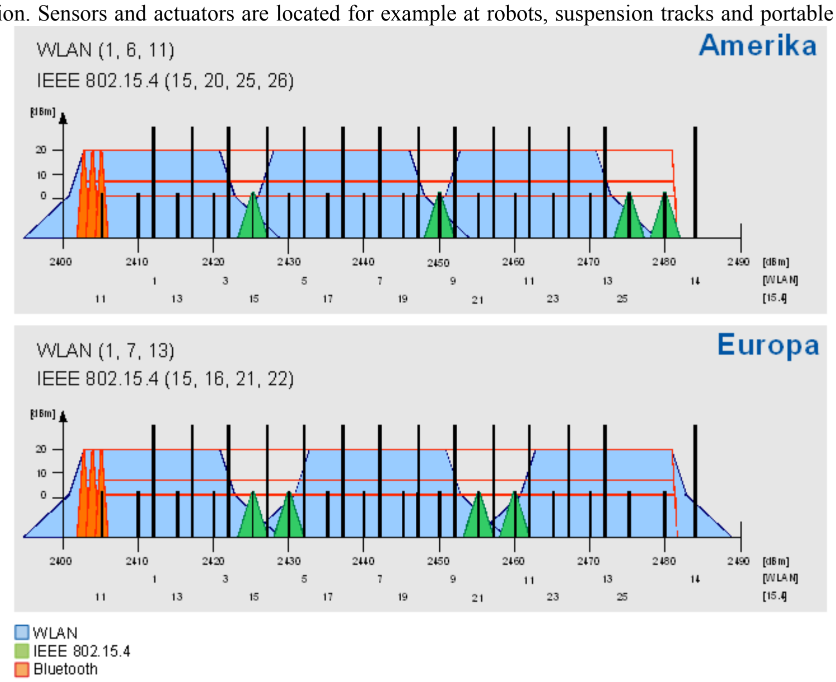
For this application domain exists the following major requirements:

- — High determinism

— High reliability
— Low latency:
  - transmission of sensor data in □ 10 ms
  - low round-trip time
— Many sensors per LL_NW PAN coordinator
  - might be more than 100 sensors per LL_NW PAN coordinator
— Assume controlled environment (factory floor)
— Configuration for optimal performance
— Network management and frequency planning for avoidance of co-existence issues.
— Roaming capability (no channel hopping)

M.3.2 Application overview

Factory automation comprises today a large number of sensors and actuators observing and controlling the production. Sensors and actuators are located for example at robots, suspension tracks and portable tools in



the automotive industry, collect data on machine tools, such as milling or turning machines and control revolving robots. Further application areas are control of conveyor belts in cargo and logistics scenarios or special engineering machines. Depending on the specific needs of different factory automation branches many more examples could be named.

Common to these sensor applications in factory automation context is the requirement of low latency and high cyclic determinism. The performance should allow for reading sensor data from 20 sensors within 10ms.

Cabling these sensors is very time consuming and expensive. Furthermore, cables are a frequent source for failures due to the harsh environment in a factory and may cause additional costs by production outage.

Wireless access to sensors and actuators solves the cabling issue and provides also advantages in case of mobility and retrofit situations.

Wireless technologies that could be applied for the factory automation scenario include 802.11 (WLAN), 802.15.1 (Bluetooth) and 802.15.4. 804.15.4 is designed for sensor applications and offers the lowest energy consumption as well as the required communication range and capacity. Moreover, four 802.15.4 channels can be utilized in good coexistence with three non-overlapping WLAN channels (see Figure M.1). Bluetooth offers good real-time capabilities, but interferes inevitably with any existing WLAN installations.

802.15.4 is a worldwide and successfully applied standard for wireless and low power transmission of sensor data. Different protocols on top of 802.15.4 (WirelessHART™ according to IEC 62591, ISA100.11a or ZigBee) in the context of process automation are already in the process of standardization. Those protocols aim at different requirements, but employ the same physical layer hardware as the proposed solution for factory automation, which indicates potential hardware synergies and cost savings. Thus, a solution for factory automation based on 802.15.4 would be beneficial.

802.15.4 operates usually in Carrier Sense Multiple Access (CSMA) mode which gives no guarantees for media access. Optionally, 802.15.4 specifies the beacon-enabled mode which defines a TDMA like superframe structure with Guaranteed Time Slots (GTS) for deterministic access. The performance of 7 GTS in an interval of 15ms does not fulfill the factory automation requirements and makes not full use of the available capacity. Therefore a modification of the 802.15.4 MAC for application in industrial factory automation, i.e. defining a fine granular deterministic TDMA access, is envisaged.

Figure M.1—RF technology coexistence in the 2,4GHz ISM band

M.3.3 Requirements and Assumptions

The above mentioned factory automation applications impose the following requirements to a wireless system:

- high determinism,
- high reliability,
- low latency, i.e. transmission of sensor date in ≤ 10ms,
- low round trip time,
- support for many sensors per LL_NW PAN coordinator.

The proposed TDMA scheme, as described in the remainder of this document, supports these requirements. Allocating a dedicated time slot for each sensor provides a deterministic system. The 802.15.4 DSSS coding together with the exclusive channel access for each sensor ensures high reliability of the system. Small time slots and short packets lead to superframes as small as 10ms, which provides a latency of less than 10ms and a low round trip time. The number of slots in a superframe determines the number of sensors that can access each channel. By operating the LL_NW PAN coordinator with multiple transceivers on different channels, a high number of sensors is supported.

The proposed system needs to be operated in a controlled configuration to achieve the required performance. Thus, it is assumed that the system is operated in a controlled environment with frequency planning. The TDMA channels are allocated in a way that.

M.4 Enhanced GTS (DSME)

Typical parts of the application domain of commercial networks are facilities for

- etc.

For this application domain exists the following major requirements:

- ...

TBD by DSME technology provider.

M.5 Low energy (LE)

The Low Energy (LE) mechanisms are suitable for applications that are willing to trade low latency for low energy consumption. They allow radios to operate down to a fraction of 1% duty cycles while presenting an always-on illusion. Devices can always talk to each other without any pre-arranged synchronization schedules. The Low Energy mechanisms are applicable in the non-beacon mode as well as in the CAP periods of the beacon mode. It is also possible for the upper layer to temporarily turn off the Low Energy mechanisms by operating the radio at 100% duty cycle for emergency messages.

There are two Low Energy mechanisms CSL and RIT. CSL is suitable for applications with relatively low latency requirements, e.g., < 1 second. RIT is suitable for applications with a high latency tolerance, e.g., tens of seconds. RIT is also required in cases where the local regulation limits the duration of continuous transmissions to too small a period for CSL to be effective.

M.6 Channel Diversity

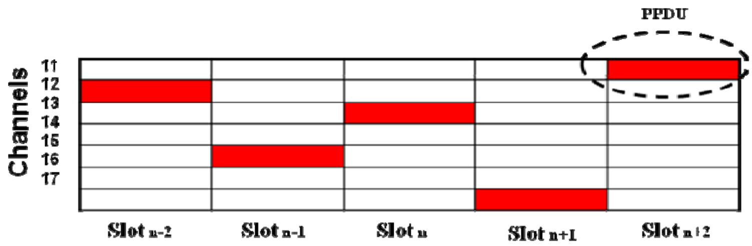
Annex M.6 provides tutorial material for a better understanding for the different solutions specified in Clauses 7.

Wireless PAN suffers severe receiver channel variation which results in poor signal reception quality. The main cause of physical impairments is called multi-path fading, and mutual RF interferences.

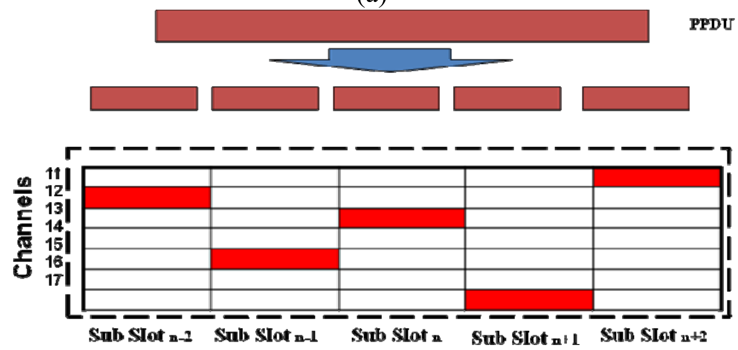
IEEE802.15.4e MAC provides two types of channel diversity methods to overcome these impairments: channel adaption and channel hopping. Channel adaptation does not change a channel in use until the received signal quality drops down lower than a threshold value. When channel quality is poor, it switches the channel to another one which is expected to show statistically different reception quality. On the other hand, channel hopping enforce the channel to switch at each time slots at most according to predefined channel hopping pattern. Channel hopping pattern, called channel hopping sequence, is set by NHL. Basic idea behind these channel diversity methods is to exploit the nature of receiver channel quality varying over whole available RF channel spectrum. A chance for a channel suffering channel impairments is statistically much lower than another one suffering deep fading located far apart. Thus, the reception signal quality is expected to be improved significantly by switching a channel with poor quality to other one located far apart.

IEEE 802.15.4 provides two types of PAN operation modes: beacon enabled mode and non beacon enabled mode. In IEEE802.15.4e MAC, channel adaptation is implemented over DSME structure in beacon enabled PAN, while channel hopping can be implemented in either of PAN operation modes. See 7.1.21 and 7.1.23 for more detail.

Channel diversity methods herein can coexist with channel hopping method performed in PHY such as physical layer frequency hopping (PHY-FH) in SUN. The fundamental difference of channel hopping method as in MAC and PHY is whether channel switching occurs during the transmission of a PPDU. Figure M.2 illustrates the hopping methods in two layers. In MAC channel hopping (MAC-CH) scheme, each PPDU is transmitted in different frequency channel (Figure M.2 (a)), while a PPDU is fragmented into segments and each segment is transmitted in different sub time slots with different frequency channel in PHY-FH (Figure M.2 (b)).



(a)



(b)

Figure M.2—Illustration of channel hopping in (a) MAC (b) PHY Layer

However, the notion of channel number is not clear when this cooperative channel diversity scheme is applied. To understand this, let us consider channel hopping sequence of {1,2,3,4} for MAC-CH. Before the transmission of the first PPDU, PHY would set physical channel information obtained from channel hopping sequence for MAC-CH and from that for PHY-FH. Now, PHY cannot determine using which channel a frame is transmitted. In order to resolve this, we introduce a notion of logical channel number. Table M.1 shows how a logical channel number maps into a channel hopping sequence used for PHY-FH. For instance, if PHY-FH employs channel hopping sequences, {1,3,5,7}, {2,4,6,8}, {9,11,13,15} and {10,12,14,16}, each sequence is numbered as logical channel numbers 1 through 4. Thus, when MAC sets logical channel number of 1, PHY uses channel hopping sequence {1,3,5,7} for the transmission of a PPDU, and so forth. An example of PHY channel hopping sequence for the given sequence of logical channel numbers are illustrated in Table M.2.

Table M.1—Logical channel numbering

PHY Hopping Sequence	Logical Channel Number
{1,3,5,7}	1
{2,4,6,8}	2
{9,11,13,15}	3
{10,12,14,16}	4

Table M.2—PHY channel hopping sequences using the notion of logical channel

MAC Hopping Sequence	PHY Channel Hopping Sequences
{1,2,3,4}	{{1,3,5,7},{2,4,6,8},{9,11,13,15}, {10,12,14,16}}
{2,3,4,1}	{{2,4,6,8},{9,11,13,15}, {10,12,14,16},{1,3,5,7}}
{3,4,1,2},	{{9,11,13,15},{10,12,14,16},{1,3,5,7}, {2,4,6,8}}
{4,1,2,3},	{{10,12,14,16},{1,3,5,7}, {2,4,6,8},{9,11,13,15}}

M.7 Blink frame

The Blink Frame provides a mechanism for a device to communicate its ID (i.e. the EUI-64 Source Address) and/or an alternate ID (in payload), and optionally additional payload data to other devices without prior association and without an acknowledgement. The frame can be used by “transmit only” devices to co-exist within a network, utilizing Aloha protocol. Any devices that are not interested in this Blink Frame have an opportunity to reject the frame at early stage during frame processing and not burden the MAC or higher communication layers with this, potentially high volume, data traffic.