IEEE P802.15

Wireless Personal Area Networks

Project	IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)	
Title	MAC Enhancements for Time Slotted Channel Hopping	
Date Submitted	[15 August 2009]	
Source	[Chol Su Kang] [Kuor Hsin Chang] [Rick Enns] [Kris Pister] [Ludwig Winkel] [Jonathan Simon] [Clint Powell] [Jose Gutierrez]	Voice: [ ] Fax: [ ] E-mail: [cholsukang@gmail.com] [kuor-hsin.chang@us.elster.com] [enns@stanfordalumni.org] [pister@eecs.berkeley.edu] [ludwig.winkel@siemens.com] [jsimon@dustnetworks.com] [cpowell@ieee.org] [Jose.Gutierrez@emerson.com]
Re:	[Task Group 15.4e: Enhancements to the 802.15.4 MAC]	
Abstract	This document describes the Time Slotted Channel Hopping extension to the current IEEE802.15.4-2006 MAC.	
Purpose	The purpose of this document is to specify the Time Slotted Channel Hopping extension to enhance the current MAC in IEEE802.15.4-2006 standard.	
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.	

New MAC service primitives

To be appended to **Table 46**

Name	Request	Indication	Response	Confirm
MLME-SET-SLOTFRAME	**7.1.17-A.1**			**7.1.17-A.2**
MLME-SET-LINK	**7.1.17-B.1**			**7.1.17-B.3**
MLME-TSCH-MODE	**7.1.17-C.1**			**7.1.17-C.2**
MLME-LISTEN	**7.1.17-D.1**			7.1.17-D.2
MLME-ADVERTISE	**7.1.17-E.1**	**7.1.17-E.2**		**7.1.17-E.3**
MLME-KEEP-ALIVE	**7.1.17-F.1**			**7.1.17-F.2**
MLME-JOIN	**7.1.17-G.1**	**7.1.17-G.2**		**7.1.17-G.3**
MLME-ACTIVATE	**7.1.17-H.1**	**7.1.17-H.2**		**7.1.17-H.3**
MLME-DISCONNECT	**7.1.17-I.1**	**7.1.17-I.2**		**7.1.17-I.3**

7.1.17-A MLME-SET-SLOTFRAME

7.1.17-A.1 MLME-SET-SLOTFRAME.request

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

7.1.17-A.1.1 Semantics

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

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

The following table specifies parameters for the MLME-SET-SLOTFRAME.request primitive.

Table 77-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 slots in the new frame.
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.

7.1.17-A.1.2 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.

7.1.17-A.1.3 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.

7.1.17-A.2 MLME-SET-SLOTFRAME.confirm

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

7.1.17-A.2.1 Semantics

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

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

The following table specifies parameters for the MLME-SET-SLOTFRAME.confirm primitive.

Table 77-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_EXCEEDED	Results of the MLME-SET-SLOTFRAME.request command.

7.1.17-A.2.2 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.

7.1.17-A.2.3 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.

7.1.17-B MLME-SET-LINK

7.1.17-B.1 MLME-SET-LINK.request

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.

7.1.17-B.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
)

Or

MLME-SET-LINK.request to delete a link (
operationType (DELETE_LINK),
linkHandle
)

The following table specifies parameters for the MLME-SET-LINK.request primitive with the ADD_LINK or MODIFY_LINK operationType.

Table 77-C MLME-SET-LINK.request parameters

Name	Type	Valid Range	Description
operationType	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	0x01-0xFF	Channel offset of the link.
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-

Name	Type	Valid Range	Description
			ADVERTISE.request.
nodeAddr	Integer	0x0000-0xffff	Address list of neighbor devices connected to the link. 0xffff means the broadcasting to every node.

7.1.17-B.2.1 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.

7.1.17-B.2.2 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 section 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.

7.1.17-B.3 MLME-SET-LINK.confirm

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

7.1.17-B.3.1 Semantics

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

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

The following table specifies parameters for the MLME-SET-LINK.confirm primitive.

Table 77-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.

Name	Type	Valid Range	Description
		SUCCESS LINK_NOT_FOUND	Result of the delete link operation.
linkHandle	Integer	0x00 – 0xFF	Unique (local to specified slotframe) identifier for the link.

7.1.17-B.3.2 When generated

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

7.1.17-B.3.3 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.

7.1.17-C MLME-TSCH-MODE

7.1.17-C.1 MLME-TSCH-MODE.request

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

7.1.17-C.1.1 Semantics

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

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

The following table specifies parameters for the MLME-TSCH-MODE.request primitive.

Table 77-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.

7.1.17-C.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).

7.1.17-C.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.

7.1.17-C.2 MLME-TSCH-MODE.confirm

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

7.1.17-C.2.1 Semantics

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

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

The following table specifies parameters for the MLME-TSCH-MODE.confirm primitive.

Table 77-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

Name	Type	Valid Range	Description
			mode ON request or OFF request.
status	Enumeration	SUCCESS NO_SYNC	

7.1.17-C.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.

7.1.17-C.2.3 Effect on receipt

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

7.1.17-D MLME-LISTEN

7.1.17-D.1 MLME-LISTEN.request

7.1.17-D.1.1 Semantics

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

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

The following table specifies parameters for the MLME-LISTEN.request primitive.

Table 77-G MLME-LISTEN.request parameters

Name	Type	Valid Range	Description
onTime	Integer	0x0000 – 0xFFFF	The amount of time (10–millisecond units) to stay on each channel. 0x0000 indicates that the MAC stops listening
offTime	Integer	0x0000 – 0xFFFF	The amount of time (10–millisecond units) to wait between channel changes.
numPageChannel	Integer	0x01-0xFF	The number of page channel descriptors in the page channels array.
pageChannelsDes[]	Table 77-H	Table 77-H	Array of page channel descriptor. See Table 77-H for the format of page channel descriptor.

Table 77-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 2.	The array of channels on which to listen. See Table 2 for the valid range of channels in each channel page.

7.1.17-D.1.2 When generated

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

7.1.17-D.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.

7.1.17-D.2 MLME-LISTEN.confirm

7.1.17-D.2.1 Semantics

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

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

The following table specifies parameters for the MLME-LISTEN.confirm primitive.

Table 77-I MLME-LISTEN.confirm parameters

Name	Type	Valid Range	Description
status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.17-D.2.2 When generated

The MAC layer shall generate MLME-LISTEN.confirm when it completes the listen operation started by MLME-LISTEN.request.

7.1.17-D.2.3 Effect on receipt

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

7.1.17-E MLME-ADVERTISE

7.1.17-E.1 MLME-ADVERTISE.request

7.1.17-E.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[]
                       )
```

The following table specifies parameters for the MLME-ADVERTISE.request primitive.

Table 77-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 95	Security level in the Advertisement command. See Table 95 in IEEE802.15.4-2006.
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[]	Table 77-K	Table 77-K	See Table 77-K.

Table 77-K MLME-ADVERTISE.request Slotframe parameters (per slotframe)

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

7.1.17-E.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.

7.1.17-E.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.

7.1.17-E.2 ADVERTISE.indication

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

7.1.17-E.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[]
                          )
```

The following table specifies parameters for the MLME-ADVERTISE.indication primitive.

Table 77-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	Channel page.

Name	Type	Valid Range	Description
		pages supported by the PHY (see 6.1.2)	
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 95	Security level in advertisement packet See Table 95 in IEEE802.15.4-2006.
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[]	Table 77-K	Table 77-K	See Table 77-K.

7.1.17-E.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.

7.1.17-E.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.

7.1.17-E.3 MLME-ADVERTISE.confirm

7.1.17-E.3.1 Semantics

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

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

The following table specifies parameters for the MLME-ADVERTISE.confirm primitive.

Table 77-M MLME-ADVERTISE.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.17-E.3.2 When generated

The MAC layer shall generate MLME-ADVERTISE.confirm when it starts sending the Advertisement command.

7.1.17-E.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.

7.1.17-F MLME-KEEP-ALIVE

7.1.17-F.1 MLME-KEEP-ALIVE.request

7.1.17-F.1.1 Semantics

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

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

The following table specifies parameters for the MLME-KEEP-ALIVE.request primitive.

Table 77-N 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 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.

7.1.17-F.1.2 When generated

7.1.17-F.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).

7.1.17-F.2 MLME-KEEP-ALIVE.confirm

7.1.17-F.2.1 Semantics

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

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

The following table specifies parameters for the MLME-KEEP-ALIVE.confirm primitive.

Table 77-O MLME-KEEP-ALIVE.confirm parameters

Name	Type	Valid Range	Description

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.17-F.2.2 When generated

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

7.1.17-F.2.3 Effect on receipt

7.1.17-G MLME-JOIN

7.1.17-G.1 MLME-JOIN.request

7.1.17-G.1.1 Semantics

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

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

The following table specifies parameters for the MLME-JOIN.request primitive.

Table 77-P MLME-JOIN.request parameters

Name	Type	Valid Range	Description
dstAddr	Integer	0x0000 - 0xFFFF	Address of neighbor device to send Join command
securityInformation	Table 77-Q	Table 77-Q	See Table 77-Q for the detail.
numNeighbors	Integer	0x0 – 0xF	Number of neighbors found by the joining device.
neighbors	Table 77-R	Table 77-R	Neighbor information for the number of neighbors specified in numNeighbors. See Table 77-R for the definition of a neighbor.

Table 77-Q MLME-JOIN.request securityInformation parameters

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

Table 77-R 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.

7.1.17-G.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.

7.1.17-G.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 Section 7.3.12. 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).

7.1.17-G.2 MLME-JOIN.indication

7.1.17-G.2.1 Semantics

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

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

The following table specifies parameters for the MLME-JOIN.indication primitive.

Table 77-S MLME-JOIN.indication parameters

Name	Type	Valid Range	Description
linkHandle	Integer	0x00 – 0xFF	Unique identifier for the link that Join command frame is received on.
newNodeAddr	array of octets	64-bit binary string	64-bit long address of new device sending the Join command.
securityInformation	Table 77-Q	Table 77-Q	See Table 77-Q.
numNeighbors	Integer	0x0 – 0xF	Number of neighbors reported by the joining device.
Neighbors	Table 77-R	Table 77-R	Neighbor information for the number of neighbors specified in numNeighbors. See Table 77-R for the definition of a neighbor in neighbors.

7.1.17-G.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.

7.1.17-G.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.

7.1.17-G.3 MLME-JOIN.confirm

7.1.17-G.3.1 Semantics

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

MLME-JOIN.confirm (
status
)

The following table specifies parameters for the MLME-JOIN.confirm primitive.

Table 77-T MLME-JOIN.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.17-G.3.2 When generated

The MAC layer shall generate MLME-JOIN.confirm to acknowledge that it received the MLME-JOIN.request primitive.

7.1.17-G.3.3 Effect on receipt

7.1.17-H MLME-ACTIVATE

7.1.17-H.1 MLME-ACTIVATE.request

7.1.17-H.1.1 Semantics

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

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

The following table specifies parameters for the MLME-ACTIVATE.request primitive.

Table 77-U MLME-ACTIVATE.request parameters

Name	Type	Valid Range	Description
dstAddr	Integer	0x0000 - 0xFFFF	Address of neighbor device to send Activate command.
securityInformation	Table 77-V	Table 77-V	See Table 77-V for details.
slotframes[]	Table 77-K	Table 77-K	See Table 77-K.

Table 77-V 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 77-W 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 77-x	Table 77-x	Table 77-x for parameters (per link)

Table 77-X MLME-ACTIVATE.request 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.

7.1.17-H.1.2 When generated

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

7.1.17-H.1.3 Effect on receipt

Upon receipt of the request, the MAC layer shall send either the Activate command frame to activate the new joining device, 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).

7.1.17-H.2 MLME-ACTIVATE.indication

7.1.17-H.2.1 Semantics

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

```
MLME-ACTIVATE.indication        (
                                srcAddr,
                                securityInformation,
                                        )
```

The following table specifies parameters for the MLME-ACTIVATE.indication primitive.

Table 77-Y 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 77-V	Table 77-V	See Table 77-V.

7.1.17-H.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.

7.1.17-H.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. 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).

7.1.17-H.3 MLME-ACTIVATE.confirm

7.1.17-H.3.1 Semantics

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

MLME-ACTIVATE.confirm (
status
)

The following table specifies parameters for the MLME-ACTIVATE.confirm primitive.

Table 77-Z MLME-ACTIVATE.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS INVALID_PARAMETER	

7.1.17-H.3.2 When generated

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

7.1.17-H.3.3 Effect on receipt

7.1.17-I MLME-DISCONNECT

7.1.17-I.1 MLME-DISCONNECT.request

7.1.17-I.1.1 Semantics

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

MLME-DISCONNECT.request (

)

7.1.17-I.1.2 When generated

MLME-DISCONNECT.request primitive is used to initiate the graceful disconnection from TSCH network.

7.1.17-I.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.

7.1.17-I.2 MLME-DISCONNECT.indication

7.1.17-I.2.1 Semantics

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

MLME-DISCONNECT.indication (

srcAddress,

)

The following table specifies parameters for the MLME-DISCONNECT.indication primitive.

Table 77-AA MLME-DISCONNECT.indication parameters

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

7.1.17-I.2.2 When generated

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

7.1.17-I.2.3 Effect on receipt

Upon receipt of the MLME-DISCONNECT.indication, the device management layer shall process the disconnection of the neighbor from which the Disconnect 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).

7.1.17-I.3 MLME-DISCONNECT.confirm

7.1.17-I.3.1 Semantics

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

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

The following table specifies parameters for the MLME-DISCONNECT.confirm primitive.

Table 77-BB MLME-DISCONNECT.confirm parameters

Name	Type	Valid Range	Description
Status	Enumeration	SUCCESS	

7.1.17-I.3.2 When generated

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

7.1.17-I.3.3 Effect on receipt

Modified MAC Service Primitives

7.1.1.1 MCPS-DATA.request

The MCPS-DATA.request primitive requests the transfer of a data SPDU (i.e., MSDU) from a local SSCS entity to a single peer SSCS entity specified in any of the destination addresses in DstAddr and additionalDstAddr.

7.1.1.1.1 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 specifies parameters for the MCPS-DATA.request primitive.

Table 41-MCPS-DATA.request parameters

Name	Type	Valid Range	Description
SrcAddrMode	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
DstAddrMode	Table 41	Table 41	See Table 41 in IEEE802.15.4-2006.
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.

Name	Type	Valid Range	Description
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

The MCPS-DATA.request primitive is generated by a local SSCS entity when a data SPDU (i.e., MSDU) is to be transferred to a peer SSCS entity specified in any of destination addresses in DstAddr and additionalDstAddr. When sending to alternate destinations, all correspondent devices must be operating within the same security context or they will not possess appropriate information to authenticate or decrypt the frame.

7.1.1.1.3 Effect on receipt

On receipt of the MCPS-DATA.request primitive…the MAC sublayer will issue the MCPS-DATA.confirm primitive with a status of INVALID_PARAMETER.

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 using the linkHandle provided in linkHandleList. MAC should transfer the requested data SPDU on one of these links. MAC will select a link to transmit (or retransmit if ACK is not received) in earliest possible opportunity.

7.1.1.2 MCPS-DATA.confirm

The MCPS-DATA.confirm primitive reports the results of a request to transfer a data SPDU (MSDU) from a local SSCS entity to a single peer SSCS entity.

7.1.1.2.1 Semantics of the service primitive

The semantics of the MCPS-DATA.confirm primitive are as follows:

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

Table 42 specifies the parameters for the MCPS-DATA.confirm primitive.

7.1.1.2.2 When generated

The MCPS-DATA.confirm primitive is generated by the MAC sublayer entity in response to an MCPS-DATA.request primitive. The MCPS-DATA.confirm primitive returns a status of either SUCCESS, indicating that the request to transmit was successful, or the appropriate error code. The status values are fully described in 7.1.1.1.3 and subclauses referenced by 7.1.1.1.3.

Table 42-MCPS-DATA.confirm parameters

Name	Type	Valid Range	Description
moduleHandle	Table 42	Table 42	See Table 42 in IEEE802.15.4-2006.
status	Table 42	Table 42	See Table 42 in IEEE802.15.4-2006.
Timestamp	Table 42	Table 42	See Table 42 in IEEE802.15.4-2006.
DstAddr	Device address	As specified by the DstAddrMode parameter of MCPS-DATA.request	Destination address to which the data SPDU was transferred.

7.1.1.2.3 Appropriate usage

On receipt of the MCPS-DATA.confirm primitive, the SSCS of the initiating device is notified of the result of its request to transmit. If the transmission attempt was successful, the status parameter will be set to SUCCESS. Otherwise, the status parameter will indicate the error.

If the transmission attempt was successful, DstAddr is set to the address of the destination to which the data SPDU was transferred

7.2 New Frame Formats

Secure Extended ACKs with payload is to be jointly defined with Security subgroup. ACK/NACK frame below is authenticate only/no payload and should be a subset of the Extended ACK definition.

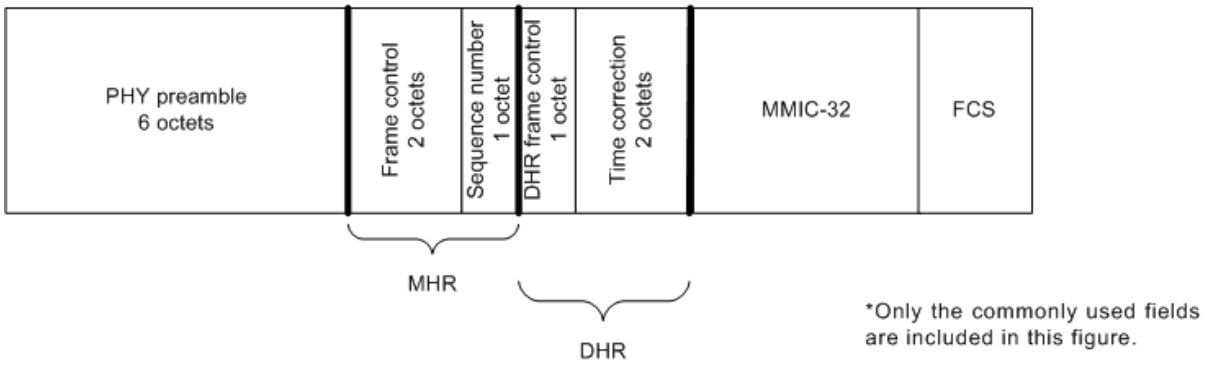

Figure 85 – Typical acknowledgement frame layout

The source address of an ACK/NACK is the address of the device that transmits the ACK/NACK. The destination address is the address of the intended recipient of the ACK/NACK.

Every ACK/NACK frame shall be authenticated with a MMIC, but not encrypted. Some fields are virtual, used in creating the MMIC but not actually transmitted.

The format of an IEEE Std 802.15.4:2006 MHR is summarized in table 109

Table 109 – Acknowledgement frame MHR

octets	bits							
	7	6	5	6	3	2	1	0
1	Frame control (octet 0)							
2	Frame control (octet 1)							
3	Sequence number							

As shown in table 109, attributes include:

Frame control attributes for ACK/NACKs, as follows:

- Frame type shall be data.
- Security shall be disabled, as it is handled in the DHR.
- Frame pending shall be false.
- Ack.Request shall be false (IEEE Std 802.15.4 does not recognize this as an ACK).
- Source addressing mode shall be 0x00 (i.e., implicit), except for cases described below where the source address and PAN ID are included in the MHR.
- Destination addressing mode shall be 0x00 (i.e., implicit).
- Frame version shall be 0x~~01~~ 0x10.

Sequence number shall be incremented after each use as described in 7.3.

The acknowledgers EUI-64 shall be included in the source address field of the ACK/NACKs MHR if so requested in the received DPDU's DHDR. Normally, the 16-bit DL source address of the ACK/NACK is not transmitted, because it matches the destination address of the received DPDU. However, in duocast or n-cast acknowledgements, one or more acknowledgements may be sent by different devices. Therefore, in cases where the acknowledger's address is different from the destination address of the received DPDU, the acknowledgement shall also include the acknowledger's 16-bit DL source address in the MHR, with the assumption that the acknowledger's EUI-64 is already known by the recipient of the acknowledgement. All correspondent devices must be operating within the same security context or they will not possess appropriate information to authenticate the frame.

A prototype DHR following a MHR is summarized in table 110.

Table 110 – Acknowledgement frame DHR

octets	bits							
	7	6	5	4	3	2	1	0
1 octet	DHR ACK/NACK Frame control							
4 octets (virtual)	Echoed MMIC of received DPDU							
0-2 octets	Time correction (LSB) When requested							
0-1 octets	Timeslot offset When needed							
0-4 octets	DAUX sub-header Usually absent							

As shown in table 110, attributes include:

The DHR ACK/NACK frame control octet is described in table 111

Echoed MMIC of received DPDU. For a discussion of handling of this virtual field, see 7.3.4. To unambiguously connect the ACK/NACK with the DPDU, the MMIC of the DPDU is included in the ACK/NACK's DHR as a virtual field, with octet ordering matching the DPDUs MMIC. This virtual field is used to calculate the ACK/NACK's MMIC, but not transmitted. If the received MMIC is longer than 4 octets, only the final 4 octets of the MMIC are echoed as a virtual field.

Time correction (LSB). Used by DL clock sources to correct the time of the DL clock recipient, if it is requested in the received DPDU's DHDR. This 2-octet value, when included in the ACK/NACK, echoes the time that the DPDU was received. The value, in 2-20 s (approximately 0.954 μs), reports an offset from the scheduled start time of the current timeslot in the acknowledger's time base. The reported value is based on DPDU's start time. See.9.1.9.3.2

Acknowledger's timeslot offset is provided, when needed, within a slow hopping period. This one-octet value, when included in the NACK/ACK, indicates the current timeslot in the acknowledger's time base. It shall be included only when the received DPDU is received in a different slow hopping timeslot than is used for the acknowledgement. The first timeslot in a slow hopping period has an offset of zero. When the corrected timeslot offset is nonzero, the time correction (previous field), when included, shall be an offset of the corrected scheduled timeslot time, Security requires that a device's time increases from timeslot to timeslot. Therefore, if the timeslot is corrected to an earlier timeslot by a clock recipient, there shall be an interruption in service, equal to the magnitude of the timeslot correction plus at least one timeslot. See 9.1.9.4.9

Auxiliary sub-header (DAUX). DAUX may be included in an ACK/NACK, for the limited purpose of echoing received signal quality (see).9.3.5.5

In an ACK/NACK DPDU, the DHR frame control octet communicates the ACK/NACK selections, as shown in table 111

Table 111– DHR ACK/NACK frame control

octets	bits							
	7	6	5	4	3	2	1	0
1	Include clock correction 0= no 1= yes	Include slow-hopping timeslot offset 0= no 1= yes	ACK/NACK type 0= ACK 1=ACK/ECN 2= NACK0 3=NACK1		Auxiliary sub-header 0= no DAUX 1= DAUX included	MMIC alternative (LSB) 0= reserved 1= MMIC-32 2= MMIC-64 3= reserved		Reserved= 0

The DL protocol version number and MAC security key always match the received DPDU, and therefore are not indicated in the ACK/NACK.

Bit content is as follows:

Bit 7 shall indicate whether the ACK/NACK includes clock correction information.

Bit 6 shall indicate whether the ACK/NACK includes a slow hopping offset.

Bits 5 and 4 shall indicate whether the PDU is an ACK or a NACK, as follows:

00 is an acknowledgement.

01 is an acknowledgement with an explicit congestion notification (ECN). See. A router that is signaling ECN bits in the forward direction should also signal the ECN through DL acknowledgements, if the priority of the DPDU is 7 or less. A device receiving an ECN through a DL acknowledgement may treat this signal as early notification that it is likely to receive an ECN at upper layers.

02 is a NACK0, signaling that the DPDU was received but could not be acknowledged due to message queue congestion. See.9.1.9.4.4

03 is a NACK1, signaling that the DPDU was received but was not accepted due to recent history of forwarding problems along the route. See .9.1.9.4.4

Bit 3 shall indicate whether the ACK/NACK includes a DAUX sub-header. DAUX may be included in an ACK/NACK for the limited purpose of reporting received signal quality.

Bits 2 and 1 shall indicate the MMIC alternative.

Bit 0 is reserved and shall be set to 0.

7.3 New MAC command frame

This table entry should be appended to Table 82.

Command frame identifier	Command name	RFD		Subclause
		TX	RX	
0x0a	Advertisement		X	7.3.10
0x0b	Join	X		7.3.12
0x0c	Activate		X	7.3.13

7.3.10 Advertisement command

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.

Octets: Variable (see 7.2.2.4)	1	6	1	1	1	1	1	variable	1	Variable	0/4/8/16	2
MHR	Command Frame Identifier (see table 82)	Timing Informa-tion	Security Control Field	Join Contr ol	Timeslot Template and Hopping Sequence ID	Chan nel Page/ Map Lengt h	Chan nel Page	Channel Map	Number of Slot-frames	Slotframe Info. and Links (for Each Slotframe)	MIC	MFR

Figure 64-A Advertisement Command Format

7.3.10.1 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.

7.3.10.2 Command Frame Identifier field

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

7.3.10.3 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.

7.3.10.4 Security Control field

Bit 4–7	Bit 0–3
Reserved	Security Level

Figure 64-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 in Section 7.6.2.2.1.

7.3.10.5 Join Control field

Bit 4–7	0–3
Reserved	Join Priority

Figure 64-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.

7.3.10.6 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.

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

Figure 64-C Timeslot Template and Hopping Sequence ID field

7.3.10.7 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.

7.3.10.8 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.

7.3.10.9 Channel Map field

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

7.3.10.10 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.

7.3.10.11 Slotframe Information and Links (for each slotframe) field

Slotframe Information and Links field is included for each slotframe. The format of Slotframe Information and Links field is depicted as follow:

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

Figure 64-D Slotframe and Links field

Slotframe ID subfield

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

Slotframe Size subfield

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

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.

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 follow:

Octets: 2	1	1
Timeslot	Channel Offset	Link Option

Figure 64-E Link Information field

Timeslot subfield

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

Channel Offset Information subfield

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

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.

7.3.10.12 MIC

The message integrity check of the Advertisement command frame.

7.3.12 Join command

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 64.G.

Octets: (see 7.2.2.4)	1	1	1	1	0/3	var	0/3	TBD	0/4/8/16
MHR fields	Command frame Identifier (see Table 82)	Capability Information (see Figure 56)	Clock Accuracy Capability	Number of Neighbors	Neighbor 1	…	Neighbor n	Join Security Information (TBD by Security sub-group)	MIC

Figure 64.G-Join command format

7.3.12.1 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 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.

7.3.12.2 Command Frame Identifier field

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

7.3.12.3 Capability Information field

The Capability Information field shall be formatted as illustrated in Figure 56.

7.3.12.4 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

7.3.12.5 Join Security Information field

The definition of Join Security Information field shall contain the security information that the new device to mutually authenticate the new joining device and Security Manager. This will be defined by Security sub-group.

7.3.12.6 Number of Neighbor field

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

7.3.12.7 Neighbor field

The Neighbor field shall contain the information about the neighbors of the new device.

Octets: 2	1
16 bit address of the neighbor	RSSI

Figure 65.I-Neighbor

7.3.12.8 MIC

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

7.3.13 Activate command

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.
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.

Octets:	1	2	1	Variab	TBD	0/4/8/16

The command formatted illustrated 65-J.

(see 7.2.2.4)				le		
MHR fields	Command frame Identifier (see Table 82)	Short Address	Number of Slot-frames	Slotframe Info. and Links (for Each Slotframe)	Activate Security Information (TBD by Security sub-group)	MIC

Activate shall be as in Figure

Figure 65-J- Activate command format

7.3.13.1 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 *aCoordShortAddress*.
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.

7.3.13.2 Command Frame Identifier field

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

7.3.13.3 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.

7.3.13.4 Number of Links field

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

7.3.13.5 Link field

Link field shall have the description of link allocated to new device being activated. The format of Link field can be found in Section 7.3.10.10.

7.3.13.6 Activate Security Information field

The definition of Activate Security Information field shall contain the security information that the new device should use to securely communicate to the TSCH network. It may include keys for data link and session layers.

7.3.13.7 MIC

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

7.4

7.4.2 MAC PIB attributes

macDisconnectTime entry needs to be appended in Table 86 as follows. macMinBE and macMaxBE need to be modified as follows.

attribute	Identifier	Type	Range	Description	Default
macDisconnectTime		Integer	0x00-0xFFFF	Time to send out Disconnect frames before disconnecting	
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	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

7.4.3 MAC PIB Tables

macSlotframeTable

macSlotframeTable attribute	Identifier	Type	Range	Description	Default
slotframeId		Integer	0x00-0xFF	Identifier of the slotframe	
slotframeSize		Integer	0x0000-0xFFFF	Number of timeslots in the slotframe	
activeFlag		Boolean	0x0-0x1	Flag indicating if the slotframe is currently activated	
channelPage		Integer	0x00-0x1F	Channel Page of channels used in this slotframe	
channelMap		Bitmap		Bitmap of active channels.	

macLinkTable

macLinkTable attribute	Identifier	Type	Range	Description	Default
linkId		Integer	0x00-0xFF	Identifier of Link	
linkOption		Bitmap	0x00-0x7	Flags indicating whether the link is used for transmit, receive, or shared transmissions:	
linkType		Integer	0x00-0x2	Enumeration indicating the type of link: Normal, Join, or Advertising	
slotframeId		Integer	0x00-0xFF	Identifier of Slotframe to which this link belongs	
nodeAddress		IEEE address	16 bit address	Address of the node connected to this link	
timeslot		Integer	0x0000-0xFFFF	Timeslot for this link	
channelOffset			0x00-0xFF	Channel offset for this link	

macTimeslotTemplate

Entry Name	Identifier	Type	Range	Description	Default
Timeslot Template Id		Integer	0x0-0xF	Identifier of Timeslot Template	
TsCCAOffset		Integer	0x0000-0xFFFF	The time between the beginning of timeslot and start of CCA operation	
TsCCA		Integer	0x0000-0xFFFF	Duration of CCA	
TsTxOffset		Integer	0x0000-0xFFFF	The time between the beginning of the timeslot and the start of packet transmission	
TsRxOffset		Integer	0x0000-0xFFFF	Beginning of the timeslot to when the receiver must be listening	
TsRxAckDelay		Integer	0x0000-0xFFFF	End of packet to when the transmitter must listen for Acknowledgment	
TsTxAckDelay		Integer	0x0000-0xFFFF	End of packet to start of Acknowledgment	
TsRxWait		Integer	0x0000-0xFFFF	The time to wait for start of packet	
TsAckWait		Integer	0x0000-0xFFFF	The minimum time to wait for start of an Acknowledgment	
TsRxTx		Integer	0x0000-0xFFFF	Transmit to Receive turnaround (12 symbols)	
TsMaxAck		Integer	0x0000-0xFFFF	Transmission time to send Acknowledgment	
TsMaxTx		Integer	0x0000-0xFFFF	Transmission time to send the maximum length packet (133 bytes)	

macHoppingSequence

To be jointly defined with Channel Hopping/Channel Diversity subgroup.

7.5 Functional description

7.5.1.5 Slotframe structure

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.A 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 pairwise 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. Mapping of logical channel to physical channel is to be jointly defined with Channel Hopping/Channel Diversity subgroup

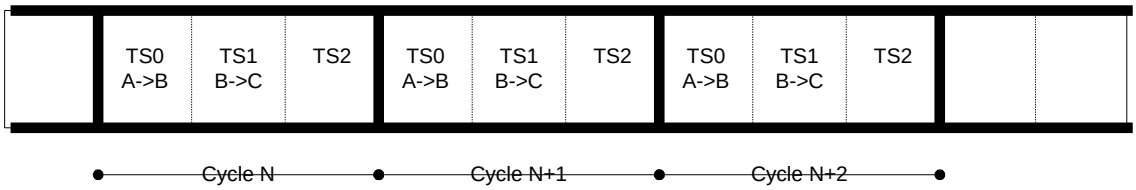

Figure 69.A -Example of a three-timeslot slotframe

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.

7.5.1.5.1 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.B). 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

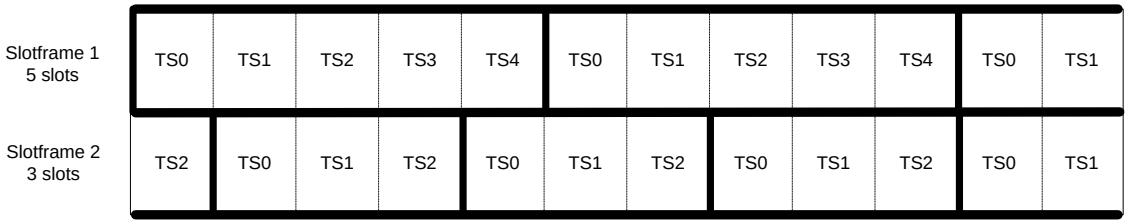

Figure 69.B - Multiple slotframes in the network

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 40.A. The join sequence is shown in figure 40.B.

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.

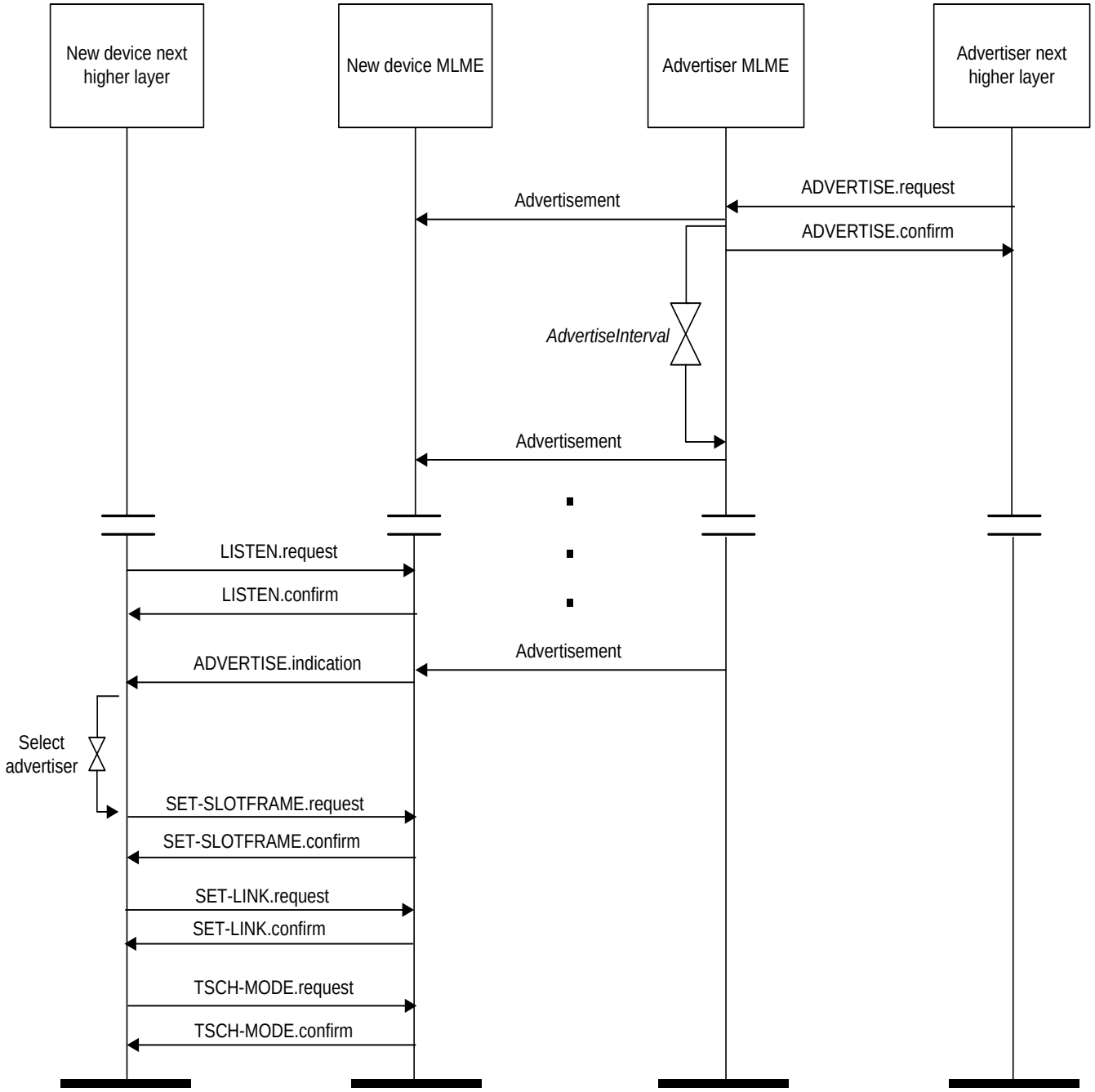

Figure 40.A-Message sequence chart for TSCH procedure to find an advertising device

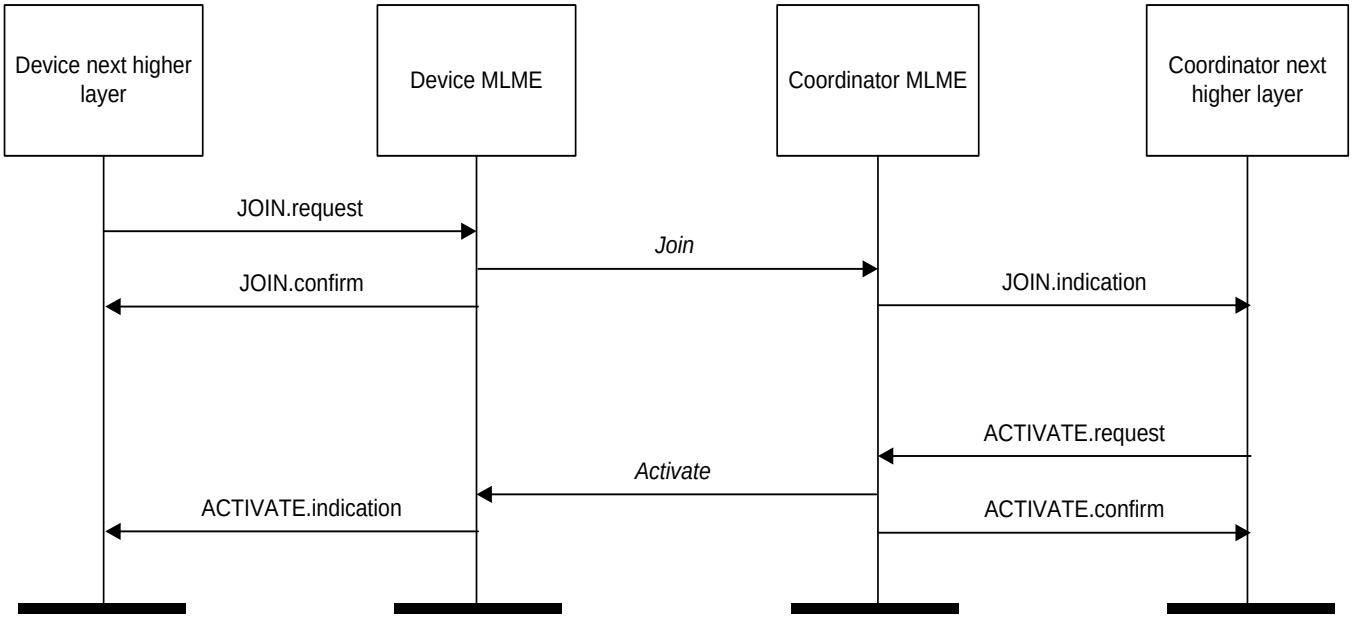

Figure 40.B-Message sequence chart for join and activate procedures

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 40-A. 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 40-B. 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.

7.5.4 Synchronization

This subclause specifies the procedures for coordinators to generate beacon frames, ~~and~~ for devices to synchronize with a coordinator, and for devices to synchronize to the TSCH network. For PANs supporting beacons, synchronization is performed by receiving and decoding the beacon frames. For PANs not supporting beacons, synchronization is performed by polling the coordinator for data or by time synchronized communication within a timeslot of the slotframe.

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.C. 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.

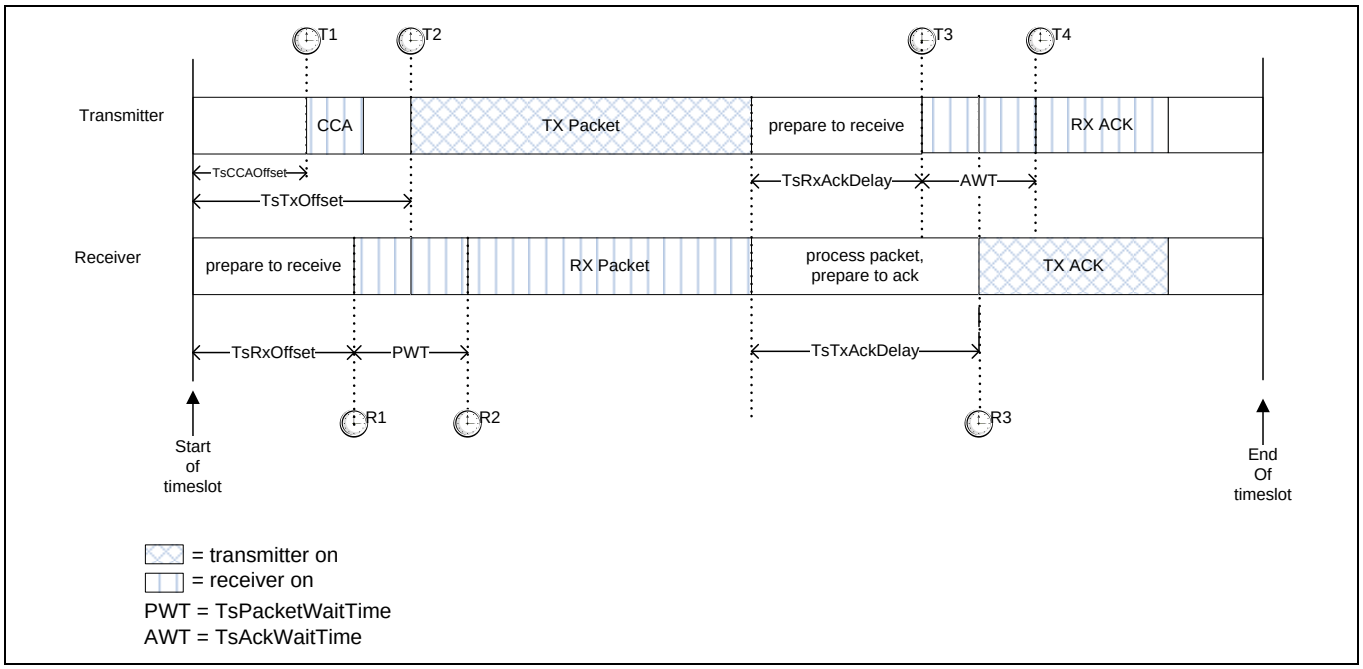

Figure 69.C -Timeslot diagram of acknowledged transmission

As shown in Figure 96C, 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

An example of a 10 ms length timeslot template (from the transmitter's perspective):

TsTxOffset	2120 µs
TsMaxPacket	4256 µs
TsRxAckDelay	800 µs
TsAckWait	400 µs
TsMaxAck	2400 µs
Total	9976 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

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.

7.5.4.4.2.1 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:

1. Transmitter node sends a frame, timing the start symbol to be sent at *TsTxOffset.*
2. Receiver records the timestamp *TsRxActual* of receiving the start symbol of the packet.
3. Receiver calculates TimeAdj = TsTxOffset – TsRxActual.
4. Receiver send back *TimeAdj* as part of acknowledgement packet.
5. Transmitter receives the acknowledgement. If the receiver node is a clock source node, the transmitter adjusts its network clock by *TimeAdj.*

7.5.4.4.2.2 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:

1. Receiver records the timestamp *TsRxActual* of receiving the start symbol of the packet.
2. Receiver calculates *TimeAdj = TsTxOffset – TsRxActual.*
3. 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.D illustrates both time synchronization mechanisms. In both cases, the receiver calculates TimeAdj to either send back to the transmitter or to use locally.

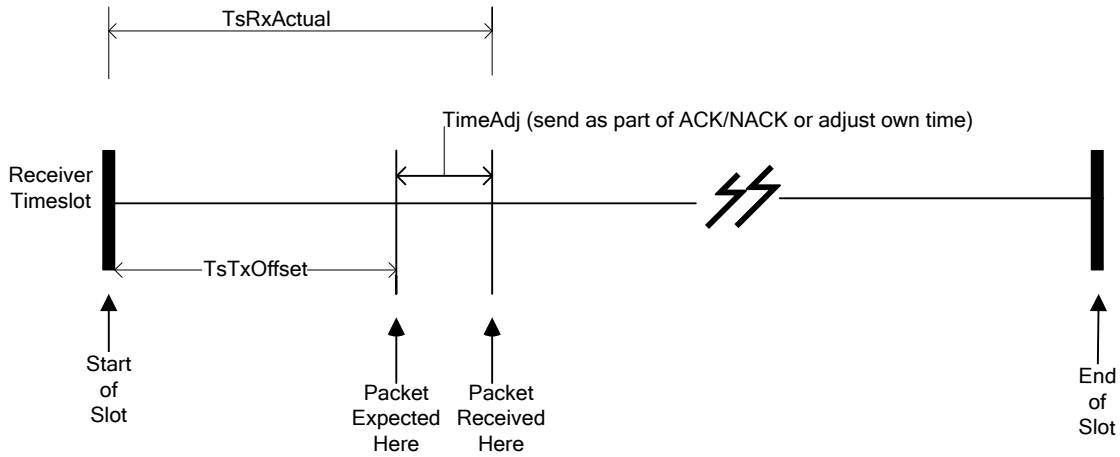

Figure 69.D -Time synchronization

7.5.4.4.3 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.

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.E 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.

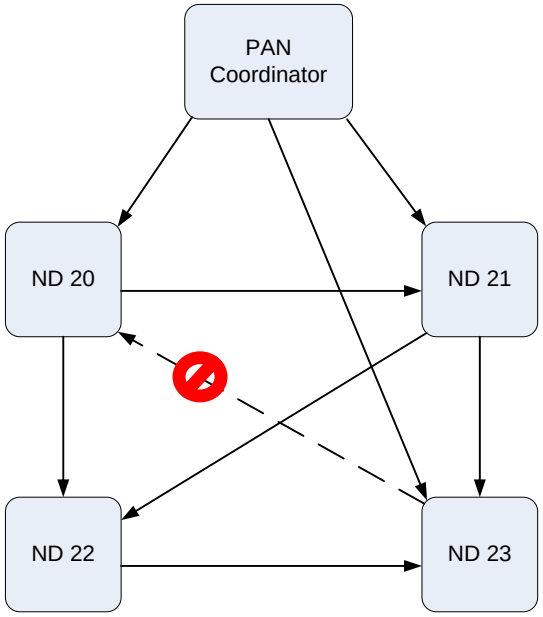

Figure 69.E -Time propagation in TSCH network

7.5.4.4.4 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.

7.5.1.4.1 TSCH CCA Algorithm

When a device is operating in the TSCH mode (7.1.17-C) 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. (ADD TSCH path to Figure 69-A).

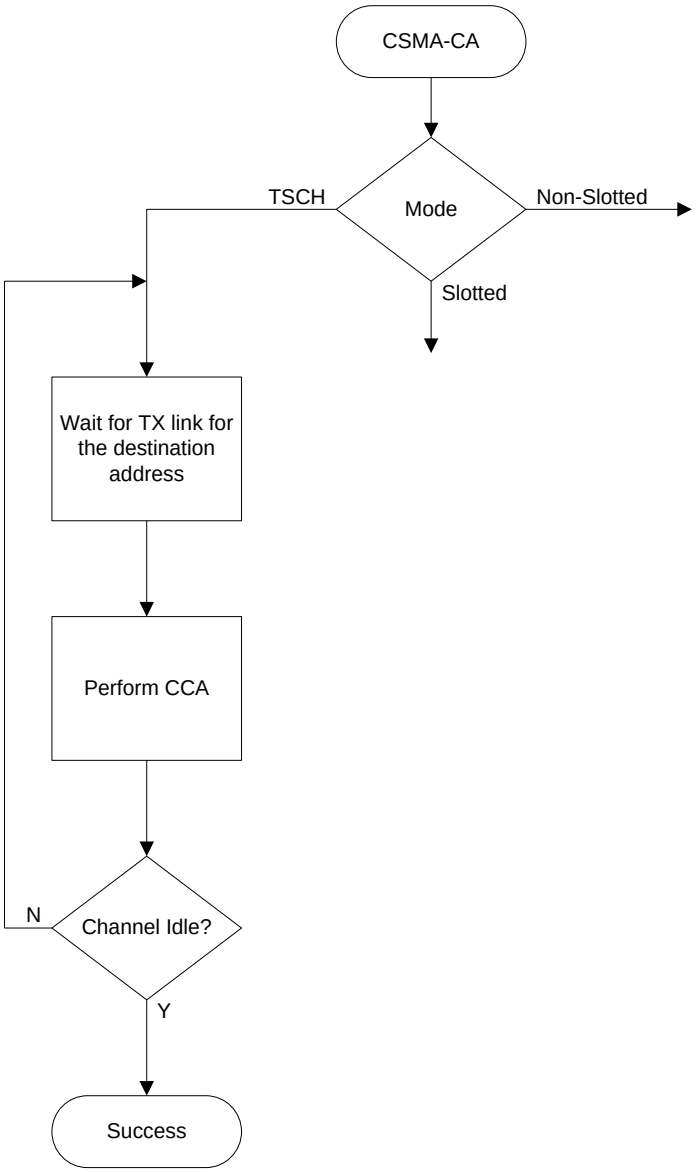

Figure 69 CSMA-CA Algorithm

(Added TSCH path)

7.5.4.2 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. 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.

7.5.6.4.2 Acknowledgment

When operating in TSCH mode, the acknowledgement frame is sent at the time specified by the macTimeslotTemplate being used (see 7.4.3 and 7.5.4.4.1).

7.5.6.4.3 Retransmissions

A device that sends a data or MAC command frame with its Acknowledgment Request subfield set to one shall wait for at most macAckWaitDuration symbols for the corresponding acknowledgment frame to be received, or in the case of operation in TSCH mode, 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 ~~macAckWaitDuration symbols~~ 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, 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.

7.6 Security

To be jointly defined with Security subgroup