IEEE P802.15
Wireless Personal Area Networks

Project	IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)	
Title	**TG3 Draft D08 to D09 changes**	
Date Submitted	[7 December, 2001]	
Source	[James P. K. Gilb] [Mobilian] [12707 High Bluff Dr., Suite 335, San Diego, CA 92130]	Voice: [858-436-2201] Fax: [858-436-2301] E-mail: [gilb@ieee.org]
Re:	[]	
Abstract	[This document contains the addtions made by the TG3 technical editor to the draft standard D08 to produce D09.]	
Purpose	[To provide a record of the changes to the TG3 draft standard D08 to make D09.]	
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.	

1. Changes from "Austin Draft Ammendements," document 01/496r2

The list of changes to be made are:

1) Make changes identified in document 01/485r4
2) Make changes identified in document 01/488r1
3) Make changes identified in document 01/476r3
4) Make changes identified in document 01/503r0
5) Make changes identified in document 01/328r4 with caveat to add CTRB parameter of desired maximum GTS.
6) Make changes identified in document 01/517r2
7) Make changes identified in document 01/502r1
8) Make changes identified in document 01/410r0
9) Make changes identified in document 01/469r3
10) Make changes identified in document 01/530r2
11) Items in 01/374r12, entered into D09 and noted in the minutes.
12) Change the MLME commands to reflect the frame formats and information described in clause 7 and 8.
13) Change backoff algorithm to use PHY dependent parameters rather than numeric times. Use 802.11 as a model to write this.
14) Update neighbor piconet information to reflect changes in 01/481r4
15) Add mapping of supported data rates from 5 bits to 8 bits by adding 3 binary 0's as the MSB.
16) Move supported data rates field in figure 19, 7.4.3, to be bits b0-b4. Add bit b10 to be neighbor piconet bit. Text for neighbor piconet bit is: "The neighbor piconet bit shall be set to 1 if the DEV is intending to be a neighbor PNC in the current piconet and shall be set to 0 otherwise." Change "is set to" to be "shall set to" in 7.4.3. Change 01/481r4 to use neighbor piconet bit instead of 0 capability field to identify and neighbor association request.
17) Change "is shown in figure xx" to be "shall be formatted as illustrated in figure xx" for frame formats, information elements, command types and field format figures in clause 7.

1.1 Document 01/485r4

7.5.11 EPS Configuration Request

MkS Editor note: Eliminate this command from D08

7.5.12 EPS Configuration Response

MkS Editor note: Eliminate this command from D08

7.5.13 EPS PS Configuration command

MkS Editor note: Eliminate this command and replace the Action Commands shown below

Ed. action: The above three commands have been deleted from the Frame Formats clause. The EPS PS configuration command was called the PS PNC configuration command in D08.

7.5.13.1 EPS Action command

This command is found in Figure 1 along with the interpretation of the fields as shown in Figure 2.

7.5.13.2 EPS Action response

This command is found in Figure 1 along with the interpretation of the field as shown in Figure 3.

This is used to create and maintain EPS Sets as well as EPS Set membership. When an EPS Set is confirmed as created, the PNC shall begin keeping the time base specified for that EPS Set.

Octets:2	**2**	**1**	**1**	**2**	**4**
Command type	Length (=2 or 8 based on request type)	Request/Response type	EPSSet value	EPSTime	EPSNext

Figure 1 PS PNC configuration command/response

Request type	***EPS set value***	***EPSTime & EPSNext***
Release request (0 value)	Value Required	Not present
New request (1 value)	Set to 0	Required
Place me in set request (2 value)	Value Required	Not present
Give me information on the EPS set request (3 value)	Value Required	Not present

Figure 2 Request entries

Reply type	***EPS set value***	***EPSTime & EPSNext***
Release confirm (0 value)	Value released	Not present
New confirm (1 value)	New value provided	Not present
Place me in set confirm (2 value)	Value of set place into	Not present
Give me information on the EPS set response (3 value)	Value Required	Part of reply
Failure code Incorrect command length	Set to 0	Not present
Failure code – Already a member	Set to 0	Not present
Failure code – EPS set does not exist	Set to 0	Not present
Failure code – Cannot create new set	Set to 0	Not present
Failure code – Illegal command	Set to 0	Not present

Figure 3 Response entries

PS editor note: Security methods may impact which devices are permitted to use this command. TBD

Ed. action: new commands are listed below:

7.5.5.1 EPS action request command

The EPS action request command shall be formatted as illustrated in Figure 1. This command is used to create and maintain EPS sets as well as EPSSet membership. When an EPS set is confirmed as created, the PNC shall begin keeping the time base specified for that EPS set.

octets: 2	**2**	**1**	**1**	**2**	**4**
Command type	Length (=2 to 8)	Action type	EPSSet value	EPSTime	EPSNext

Figure 1—EPS action request/response command format

The value of the action type determines the length of the command since the EPSTime and EPSNext fields may be left out for certain action types. The valid request types and the corresponding values for EPS set, EPSTime and EPSNext are given in Table 1.

The EPS set value is a octet that that is assigned by the PNC to a group of DEVs that share the same EPSTime and EPSNext.

Table 1—EPS action request command entries

Action type	Action type value	EPSSet value	EPSTime and EPSNext
Release request	0	Value required	Not present
New request	1	Set to 0	Required
Place in set	2	Value required	Not present
Information request	3	Value required	Not present
Reserved	4-255		

The EPSTime has a range of 0 to 65,535 ms. A value of zero indicates that the DEV is waking for each superframe. Depending on the value of superframe duration parameter, values of EPSTime that are less than the current value of superframe duration indicate that the DEV wakes for each superframe. Since the wake time is bounded by superframe beacon location, the beacon start point immediately preceding the completion of EPSTime shall be the wake point.

EPSNext is a beacon number as defined in piconet synchronization parameters element, 7.4.2. EPSNext informs the PNC or DEVs when the next EPSTime will occur. For this command, the value of EPSNext is taken from the EPSSync parameter in the MLME-POWERMGT.request primitive. The current beacon number when that primitive is received by the SME is used to calculate the beacon number for the next EPSTime event and inserts that beacon number as EPSNext when building the EPS configuration request command.

7.5.5.2 EPS action response command

The EPS action request command shall be formatted as illustrated in Figure 1. This command is used to create and maintain EPS Sets as well as EPS Set membership. When an EPSSet is confirmed as created, the PNC shall begin keeping the time base specified for that EPSSet.

The defintions of the EPSSet value, EPSTime and EPSNext fields in the command are the same as for the EPS action request, .

The value of the action type determines the length of the command since the EPSTime and EPSNext fields may be left out for certain action types. The valid action types for an EPS action response and the corresponding values for EPSSet, EPSTime and EPSNext are given in Table 2.

7.5.14 DEV to PNC PS information

MkS Editors Note: Keep this command in D07. It is needed to give the PNC information about PS capabilities to the PNC.

Table 2—EPS action response command entries

Action type	Action type value	EPSSet value	EPSTime and EPSNext
Release confirm	0	Value released	Not present
New confirm	1	New value provided	Not present
Place in set confirm	2	Value of set place	Not present
Return information on the EPS set response	3	Value required	Part of reply
Incorrect command length	4	Set to 0	Not present
Already a member	5	Set to 0	Not present
EPS set does not exist	6	Set to 0	Not present
Cannot create new set	7	Set to 0	Not present
Illegal command	8	Set to 0	Not present
Reserved	9-255		

MkS Editors Note: D07, Table 63, printed page 80, line 18-19. Fix momentary command entry.

Command type **Hex value**	**Command name**
0x0002	Channel time request (*enhancement*)
0x0003	Channel time grant (*enhancement*)
0x0015	EPS action request
0x0016	EPS action response
0x0017	DEV to PNC PS information
0x0018	Switch to ACTIVE mode
0x0019	Switch to EPS mode
0x001a	Momentary EPS CTA slot

MkS Editors note: D08 The channel time request block with will have to be modified to add a 1 octet field for "EPS Set". This modification will not be shown in this document.

Ed. action: The command DEV to PNC PS information is retained. The command summary table has been updated and is now cross-linked to the section headings so the words and locations are always current. The CTRB has a new EPS set element. The new figure and item description is shown below:

octets: 1	**1**	**1**	**1**	**2**	**2**	**2**	**2**
Target AD-AD	CTRB type	EPS set	Stream index	Allocation period	Minimum GTS time	Desired GTS time	Maximum allocation delay

Figure 2—Channel time request block for a particular stream

The EPS set is the one for which this channel time is requested if the CTRB type indicates that this is for an EPS mode CTA. The DEV that sends this command shall be a member of EPS set before it sends this request for an EPS mode CTA. The use of this field for EPS CTRs is described in 8.13.3.4

7.5.21.1 Channel time request

MkS Editors note: D08 printed page 101, line 45-48.

Replace paragraph with text below.

EPS status indicates whether the CTA requested is for ACTIVE mode channel time or EPS mode channel time. The PNC shall maintain separate ACTIVE mode type CTAs and EPS mode type CTAs. Values of 0, or 1 are used for making ACTIVE mode channel time requests, and the value of 2 is used for making EPS mode channel time requests.

For a device without EPS capability: A value of 0 shall be used for an ACTIVE mode channel time request. For an EPS capable device, a value of 0 or 1, shall be used for an ACTIVE mode channel time request and a value of 2 shall be used for an EPS mode channel time request.

The difference between using a 0 and a 1 EPS status is the persistence of the CTR. The values 0 and 1 are used to tell the PNC whether it should delete this CTR when the "Switch to EPS mode" command is received by the PNC. A 0 indicates that the PNC shall delete this CTR and de-allocate the associated channel time, and a 1 indicates that the PNC shall retain the CTR, and if possible, return the channel time to the device when the PNC it the device returns to ACTIVE mode from EPS mode.

The EPS status value of 2 is used to create an EPS channel time request. The PNC shall create and retain this EPS CTR based on this request. If possible, the PNC shall provide the requested channel time when the PNC switches the device back to EPS mode from ACTIVE mode.

See section 8..c.m for details about CTA management.

All other values of EPS status are reserved.

MkS Editors note: I have deleted the use of the value 3 in the EPS status field (broadcast/multicast) until a specific reason for this feature is indicated to the MAC group. No one seems to know whyit was put in.

Ed. action: Small re-arrangement of the text, actual text follows:

CTRB type indicates whether the CTA requested is for ACTIVE mode channel time or EPS mode channel time. The PNC shall maintain separate ACTIVE mode type CTAs and EPS mode type CTAs. Values of 0, or 1 are used for making ACTIVE mode channel time requests, and the value of 2 is used for making EPS mode channel time requests. All other values of CTRB type are reserved. CTA management is discussed in 8.13.3.5.

For a device without EPS capability a value of 0 shall be used for an ACTIVE mode channel time request. For an EPS capable device, a value of 0 or 1, shall be used for an ACTIVE mode channel time request and a value of 2 shall be used for an EPS mode channel time request.

The difference between using a 0 and a 1 CTRB type is the persistence of the CTR. The values 0 and 1 are used to tell the PNC whether it should delete this CTR when the "Switch to EPS mode" command is received by the PNC. A 0 indicates that the PNC shall delete this CTR and de-allocate the associated channel time, and a 1 indicates that the PNC shall retain the CTR, and if possible, return the channel time to the DEV when the PNC it the device returns to ACTIVE mode from EPS mode.

The CTRB type value of 2 is used to create an EPS channel time request. The PNC shall create and retain this EPS CTR based on this request. If possible, the PNC shall provide the requested channel time when the PNC switches the device back to EPS mode from ACTIVE mode.

7.5.17 Momentary EPS CTA slot command.

MkS Editor's Note: D08 Page 96, line 42-53. Replace txt with the following.

The structure of the command is indicated in Figure 45 This command instructs the PNC to use the EPS CTR slot size in the EPS CTA of the next WAKE beacon. This substitution is only in effect for one EPS superframe. If the WAKE beacon already has an EPS slot, there is no change to the CTA, andif the EPS CTA scheduled was a null CTA, then the null CTA shall be replaced with a non-zero CTA, the length specified by the EPS CTR for that DEV.

Ed. action: text added as indicated.

7.4.10 Channel time allocation (CTA) element

MkS Editors note: The text does not change until D08 printed page 84, line 31.

Figure 26 shows the structure of the CTA element. This shall be used by the PNC to describe the location of a dynamic or pseudo static time slot for the specified Source AD-AD, Destination AD-AD, and Stream Index field values. The use of the next field, Slot Location, is summarized in Table xxcc. Slot Location is slot start time if there is a time slot in the corresponding superframe, and as the next slot's beacon number (SFN-ext) if the next time slot is in a future superframe. The Beacon-Time bit of the CTA Control field shall be set by the PNC to indicate how the two bytes of the Slot Location field are interpreted. A Beacon-Time of 0 shall cause the Slot Location field to be interpreted as the Slot Start Time.

The slot stop time is Slot Start time of the next GTS slot minus the aSlotGuardTime. The slot stop time is also measured relative to the start of the beacon frame and in the same units. In EPS mode only, the values of Slot Start time and slot stop time may both be identical to indicate a zero length time slot.

A value of 1 shall cause Slot Location to be interpreted as the least significant two bytes of a beacon count corresponding to the superframe in which the next slot will be allocated. Figure 26 shows this Slot Location field as SFNext in Figure 26. The interpretation of the Slot Location field is the same for devices in ACTIVE mode as it is for devices in EPS mode. For a device is in EPS mode, the SFNext contains the two least significant bytes of EPSNext.

Figure 26 also shows the CTA Type bit contained in the CTA control field. This bit specifies whether the Source DEV and Destination DEV are in ACTIVE mode, 0, or EPS mode, 1.

The Slot Start time field indicates the start time or end time of the time slot. The value of this field is always measured relative to the start of transmission of the beacon frame sent by the PNC. The resolution of this field is 8 µs and so the range is [0-524280] µs.

The third bit in the CTA Control block is Key Change. If this bit is set to 1, this flag indicates that the DEV must update its symmetric keys before continuing peer to peer communication. The security section (x.?.?.?) explains this process in detail.

The remaining 5 bits in the CTA block are reserved.

Figure 25-Channel time allocation element

MkS Editors Note: no change to figure 25 and to the associated text.

Ed. action: Lots of changes to format this like the rest of the Frame Formats clause. Actual text for the CTRB is below:

Each CTA element consists of multiple, 6-octet wide CTA blocks, which shall be formatted as illustrated in Figure 3.

octets: 1	**1**	**1**	**1**	**2**
Source DEV address	Destination DEV address	Stream index	CTA control	Slot Start time or SFNext

Figure 3—Channel time allocation block

The source DEV address indicates the DEV to whom the channel time is being allocated.

The destination DEV address indicates the DEV to whom the source DEV may send the frames. If this is a broadcast address, then the source DEV shall send broadcast frames only during that time slot.

For a child PNC, the source DEV and destination DEV addresses shall both be the AD-AD of the DEV that is the child piconet's PNC.

For a neighbor piconet, the source and destination addresses shall both be the AD-AD assigned by the parent PNC for the neighbor piconet and shall be one of the reserved neighbor piconet addresses, 7.2.3.

The stream index is the number assigned by the PNC that indicates the stream associated with the channel time.

The CTA control field shall be formatted as illustrated in Figure 4..

bits: b0	**b1**	**b2-b7**
Time-beacon	CTA type	Reserved

Figure 4—CTA control block

The time-beacon bit of the CTA Control field shall be set by the PNC to indicate how the two bytes of the slot location field are interpreted. The bit shall be set to 0 if the slot location field to be interpreted as the slot start time. It shall be set to 1 if the slot location field is to be interpreted as the SFNext.

The CTA type specifies whether the source DEV and destination DEV are in ACTIVE mode or EPS mode. The bit shall be set to 0 if they are in ACTIVE mode and shall be set to 0 if they are in EPS mode.

~~The key change bit is reserved for possible security implementation with TBD meaning.~~

(This has been moved to the piconet synchronization parameters element. See below)

The slot location field is interpreted as either the slot start time field or the SFNext field, depending on the value of the time-beacon bit. The use of this field is summarized in Table 3.

Table 3—Summary of slot location field usage

Type of activity for destination DEV	Slot location octets	Time-beacon bit value	CTA type bit value
ACTIVE CTA, GTS present in this super-frame	Slot start time	0	0
ACTIVE CTA, no GTS in this superframe	Next GTS slot start time	1	0
EPS CTA, AWAKE superframe	Slot start time	0	1
EPS CTA, no GTS, just WAKE	Slot start time = 0	0	1
EPS CTA, Momentary EPS CTA GTS	Slot start time	0	1
EPS CTA, not a WAKE superframe	SFNext	1	1

If the slot location field is to be interpreted as the slot start time, then the field contians the start time of the allocated slot. The value of this field is always an offset from the start of superframe and hence the start of transmission of beacon frame from the PNC. The resolution of this field is 8 μs and so the range is [0-524280] μs.

The end of each GTS slot is the start time of the next GTS slot minus the guard time indicated in the beacon.

If the slot location field is to be interpreted as the SFNext field, then the field contains the least significant two bytes of a beacon count corresponding to the superframe in which the next AWAKE slot will be allocated.

Ed. action: New element in the piconet synchronization parameters called key number.New figure and associated text is shown below:

octets: 1	**1**	**4**	**2**	**2**	**2**	**1**	**1**
Element ID	Length (=12)	Beacon number	Superframe duration	CFP duration	CAP MaxBurstDuration	Key number	CAP mode

Figure 5—Piconet synchronization parameters element.

The key number is used to identify the current data encyrption key for the piconet. The PNC increments this number as a rollover counter every time it changes the DEK. If the piconet does not use data encryption, this field shall be set zero.

Clause 8 additions ...

Ed action: Text added by Jay Bain. JPKG moved the definitions to the Definitions clause. New definitions are below:

1.2 active mode: This mode is the default mode after a device joins the piconet. A device in active mode listens to every beacon.

1.3 active channel time allocation: A type of channel time allocation where the device will be using the allocated time slot in the current superframe.

1.4 awake mode: In this mode, a device that is using enhanced power save mode begins using the previously allocated time slots while still maintaining its enhances power save mode status.

1.5 enhanced power save: A the power management technique used to reduce the amount of power used by devices in the piconet.

1.6 enhanced power save time: The fundamental operating parameter for enhanced power save modemode. It is the nominal time value for the inter-wake periods for enhanced power save devices. The actual wake beacon is the beacon for the superframe when the nominal time is reached.

1.7 enhanced power save next: The beacon number value for the very next wake beacon for an enhanced power save set.

1.8 enhanced power save set: A grouping of devices where at least one member of the group will use enhanced power save mode. A single enhanced power save mode set shares a common timing information with regards to the enhanced power save time and enhanced power save next.

1.9 enhanced power save channel time allocation: A type of channel time allocation element generated to enable a power saving mode by the device.

1.10 reduced power save: The mode of a device that reduces its power level for part and only part ofa superframe, excluding the beacon.

1.11 wake beacon: The beacon to which the enhanced power save devicewill listen. For other beacons, the enhanced power save mode device can be presumed to be unavailable for communications.

1.12 wake superframe: A superframe when the enhanced power save device will listen to the beacon and also be available for sending or receiving operations.

Ed. action: New acronyms were added as well, as shown below:

ACTIVE	active mode
ACTIVE CTA	active mode channel time allocation
AWAKE	awake mode
CTR	channel time request
EPSTime	enhanced power save time
EPSNext	enhanced power save next
EPS set	enhanced power save set
EPS CTA	enhanced power save channel time allocation
WAKE	wake mode

Ed. action: Some changes to “Overview of CTA managment.” New text is given below

8.13.3.5 Overview of CTA Management

The PNC shall create CTA elements in every beacon, after the PNC has allocated their slot time. An EPS CTA is the CTA of a DEV in EPS mode. An ACTIVE CTA is the CTA of a DEV in ACTIVE mode. If a DEV does not have an ACTIVE or EPS slot in a particular superframe, the PNC shall include a CTA with the beacon number of the next ACTIVE or EPS time slot according to the CTA block definitions of 7.4.11.

The constant presence of CTA element allows any member of the network that has missed hearing some number of beacons, to re-synchronize with other members of the piconet after hearing just one beacon. Any DEV may read the CTA mode of a device as well as its slot start time or beacon number value.

In order for a device to determine the duration of the time slot identified in a particular CTA, that same device shall use the Slot Location field of the next contiguous CTA whose Time-Beacon bit is set to 0. This slot location field shall be

interpreted as "slot stop time + aSlotGuardTime". The device shall ignore all intermediate CTA blocks that have time-beacon bit set to 1.

When a DEV is in EPS mode the EPS set and the EPS channel time request are used together by the PNC to create the EPS CTAs, and time slots with the correct characteristics and at the correct times.

Figure 84 describes the three sequences of switching CTAs for an EPS DEV that is depicted as DEV B. DEV A may be operating as EPS or ACTIVE for this description. Without loss of generality, the direction of information flow is shown here from DEV A to DEV B.

The first sequence describes the transition of DEV B from ACTIVE mode to EPS mode and the operation of the DEV when it is in EPS mode. DEV A first sends a switch to EPS CTA mode command, 7.5.5.5, to the PNC either in the CAP or an MTS. The PNC then switches both the mode of DEV B and the specification it uses to create CTAs. The CTA specification switches from the DEV B's ACTIVE CTRB to the combination of the DEV B's EPS CTRB and the EPS Set specified therein. The first sequence shows that DEV B previously specified the EPS CTRB using an allocation period field set to 2. The resulting CTAs have a time slot allocated in every other superframe, with a null CTA allocated in the superframes in-between.

The second sequence set describes the operation for the momentary EPS CTA, 7.5.5.6. This command forces an EPS CTA into the next WAKE beacon. It may be issued by a source DEV by sending it in superframe prior to the desired WAKE superframe. The PNC shall create an EPS CTA in the next WAKE superframe with a slot size specified by the original EPS CTR block. It is possible for an EPS DEV to specify that all WAKE times have null slots except when the momentary EPS CTA command is issued. Doing so gives simple direct control over the creation of EPS slots that are still synchronized to the EPS set time base.

The third sequence describes the transition of DEV B from EPS mode back to ACTIVE mode. The switch to ACTIVE mode command, 7.5.5.4, is send by DEV A in the CAP or MTS slot allocated in the superframe before the WAKE superframe. Since this command is sent prior to the WAKE superframe, the PNC is able to switch DEV B to ACTIVE mode and begin using the corresponding CTA elements starting with the next (WAKE) superframe. Since this is the normal WAKE superframe, all devices hearing the beacon will see from the mode bit of the CTA element for DEV B that it is now in ACTIVE mode. The transition is both synchronized and seamless.

PS Editors note: make sure that Mark's Allocation Interval and not the older Duration Between Slots is correct in clause 7. Also pickup the value of Allocation Interval and place above. Make sure that "EPS Set" added as a new field to the Channel Time Request

Ed. action: The allocation interval in clause 8 was renamed to be the allocation period to match the clause 7 usage. EPS set is a new field in the channel time requrest (but not the stream management command).

Ed. action: Add CTR acronym (noted above), Also added a new EPS set element to the channel time request block, the table and definition follow:

octets: 1	**1**	**1**	**1**	**2**	**2**	**2**	**2**
Target AD-AD	CTRB type	EPS set	Stream index	Allocation period	Minimum GTS time	Desired GTS time	Maximum allocation delay

Figure 6—Channel time request block for a particular stream

The EPS set is the one for which this channel time is requested if the CTRB type indicates that this is for an EPS mode CTA. The DEV that sends this command shall be a member of EPS set before it sends this request for an EPS mode CTA. The field shall be set to zero for an ACTIVE mode channel time request.

Ed. action: Edited text for "ACTIVE and EPS channel time requests and allocations", actual text follows:

8.13.3.4 ACTIVE and EPS channel time requests and allocations

The operation of channel time requests for EPS mode is shown in Figure 82. Channel time requests (CTRs), 7.5.8.1 and 7.5.8.3, that correspond to the operation of ACTIVE CTA and EPS CTA are required for DEVs operating in EPS mode.

The ACTIVE CTA in a beacon by the PNC is the result of an ACTIVE CTR. The creation of an EPS CTA in a beacon by the PNC is the result of an EPS CTR. However, there is one additional requirement placed on a DEV for the creation of the EPS CTA. The DEV shall also be a member of at least one EPS set. The DEV shall first become a member of an EPS set, and then may issue an EPS CTR using the number of an EPS set to which it belongs as a parameter.

The role of the EPS CTR, as defined in 7.5.8.1, is to specify two parameters:

1) The size of the EPS time slots used for data
2) The value of N such that a proportion 1:N of EPS time slots that will be of that length.

For an EPS CTR, the allocation period field, 7.5.8.1, is used to specify the value of N. If the value of N is 1, then every EPS CTA has the slot duration specified by the CTR block. If instead, the value of N is 4, then 1 EPS CTA out of each 4 occurrences of the CTA has this specified size. In either case, each data slot is followed by N-1 null EPS slots (i.e. zero time duration slots).

For EPS channel time requests, N = 0 is a special case in which the PNC shall create all EPS CTA slots with zero length. The zero length, or null CTA identifies that the EPS DEV shall listen to this beacon, and that the EPS DEV does not have GTS time allocated for data transmission.

After 2 or more EPS mode CTA devices have submitted EPS type CTR blocks with the same EPS set value, the PNC follows the rules defined in 8.13.1.

The PNC shall decline the EPS channel time request if the DEV is not a member of the EPS set specified, which could be due to the condition that the EPS set does not exist. This rejection shall be sent with a channel time grant command containing the reject code, not a member of requested EPS set, 7.5.8.3.

1.2 Document 01/488r1

Ed action: The two MSCs from 01/488r1 were added to D09 by Jay Bain.

1.3 Document 01/476r3

Ed. note: Draft D08 contains 01/476r0. Text that is new from 01/476r0 is listed below:

1.3.1 Management time slots

New Clause 8.3.4 MTS slot Access

Management Time Slots (MTSs) are identical to GTSs except that the PNC address (zero) is the source or the destination address in the CTA. A PNC can choose to use MTSs instead of the CAP for command frames. When MTSs are used, the PNC shall ensure that sufficient MTSs are allocated to allow for transmission of commands to and from the PNC. There can be as little as a single MTS in a superframe whose ownership changes from superframe to superframe. At the other extreme, there can be one or more uplink and downlink MTS per associated DEV per superframe plus MTSs for associatnion. It is up to the PNC to determine the appropriate number of MTS in a superframe the same way the PNC is responsible for choosing the CAP size if a CAP is used. The PNC determines which DEVs to allocate MTSs to and how often. The PNC determines which DEVs to allocate MTSs to and how often. The PNC shall allocate at least one association MTS every aMTSAssocPeriod.

Ed. Note: add aMTSAssocPeriod with a value of 0.6s.

An open MTS is an MTS where the source address in the CTA for the MTS is the broadcast address. Any DEV associated to the piconet can attempt to send a command frame to the PNC in an open MTS. An MTS

with the association address as the SA in the CTA for the MTS is called an association MTS. Any station not currently associated to the piconet can attempt to send an association command to the PNC in an association MTS. Association commands are not permitted in open MTSs. Likewise, only association commands are allowed in association MTSs.

Open MTSs enable the PNC to service a large number of DEVs with low MTS requirements using a minmum number of MTSs. When there are few DEVs in a piconet it will be more efficient to use assigned an MTS to a DEV instead of using open MTS. It is the PNC's responsibility to determine how many and what type of MTSs to use.

The PNC shall assign an uplink MTS within aMTSAssocPeriods of a successful association command in order to support the 1 second connection target.

Ed. action: The above text was merged by Jay Bain with the existing text. The parameter aMTSAssocPeriod was added to the table at the end of clause 8 with a value of 0.6 s. JPKG performed minor changes to formalize the language, actual text is below:

8.4.3.4 Management Time Slots

Management Time Slots (MTSs) are identical to GTSs except that the PNC address is the source or the destination address in the CTA. A PNC may choose to use MTSs instead of the CAP for command frames. When MTSs are used, the PNC shall ensure that sufficient MTSs are allocated to allow for the transmission of commands to and from the PNC. There may be as few as a single MTS in a superframe where the ownership of the MTS changes from superframe to superframe. At the other extreme, there may be one or more uplink and downlink MTSs per associated DEV per superframe plus MTSs for association. The PNC is responsible for determining the appropriate number of MTSs in a superframe in the same way that the PNC is responsible for choosing the CAP size if a CAP is used. The PNC determines which DEVs will be allocated MTSs and how often. The PNC shall allocate at least one association MTS every aMTSAssocPeriod.

An open MTS is one where the source address in the CTA for the MTS is the broadcast address. Any DEV that is associated to the piconet may attempt to send a command frame to the PNC in an open MTS. An MTS with the association address as the SA in the CTA for the MTS is called an association MTS. Any station not currently associated to the piconet may attempt to send an association command to the PNC in an association MTS. Association commands shall not be sent in open MTSs. Likewise, only association commands shall be sent in association MTSs. Open MTSs enable the PNC to service a large number of DEVs with low MTS requirements by using a minmum number of MTSs. When there are few DEVs in a piconet it would be more efficient to use MTSs assigned to a DEV instead of using an open MTS. It is the PNC's responsibility to determine how many and what type of MTSs to use for each superframe.

The PNC shall assign an uplink MTS within aMTSAssocPeriod of a successful association command in order to support a 1 second connection target.

Clause 8.12.3.5 Additional Traffic to EPS DEVs

Add the following paragraph:

If management time slots are used, the PNC shall assign management time slots for a device in EPS mode during the superframes when the EPS device is scheduled to be awake and the superframe before that.

Ed. action: Text added by Jay Bain, minor edits by JPKG, actual text is below:

If management time slots are used, the PNC shall only assign management time slots for a DEV in EPS mode during superframes when the EPS device is scheduled to be listening to the beacon. The preceding superframe shall have an MTS allocated with the EPS DEV as the source to allow for a Switch to AWAKE CTA mode command, 7.5.5.5, or a Momentary EPS CTA command, 7.5.5.7, to be sent to the PNC.

1.3.2 Static GTS

Clause 7.5.21 Figure 49

Change Reason code to 4 bits. Use 1 bits for GTS type. 3 bits reserved.

Note: Modified stream management command appears below.

Ed. action: JPKG changed as indicated.

Clause 8.3.3.1

Add the following sentence:

There are two types of GTS: dynamic GTS, and pseudo-static GTS. The type of GTS slots are indicated in the stream management command as specified in 7.5.21.

New Clause 8.3.3..1.1 Dynamic Guaranteed Time Slots

The PNC is free to move dynamic GTSs within the superframe on a superframe by superframe basis. This allows the PNC the flexibility to rearrange GTS assignments to optimize the utilization of the slot assignments. The PNC can move a dynamic GTS by simply changing the CTA parameters in the Beacon.

New Clause 8.3.3.1.2 Pseudo-Static Guaranteed Time Slots

Pseudo static GTSs require a stream connection - non-stream GTSs cannot be pseudo-static. Pseudo static GTSs can be moved within the CFP by the PNC, but the PNC shall notify the affected DEVs by sending them acknowledged Channel Time Grant frames with the new CTA. As with dynamic GTSs, the PNC can rearrange pseudo static GTSs so that the GTS assignments can be optimized, but it must use the Channel Time Grant and coordinate the channel time grants with the CTAs in the beacon.

Before a pseudo static GTS is moved, the PNC shall ensure that the new position is unoccupied by another GTS. Then, the PNC sends a directed channel time grant to the receiving DEV so that the receiving DEV is listening to both the old GTS position and the new position. This channel time grant contains both the old and the new CTA. If the old and the new position overlap, the CTA can be one larger CTA.

Next the PNC sends a channel time grant to the transmitting DEV that contains only the new CTA. By moving the receiver first, the PNC ensures that no frames are lost if Channel time requests are corrupted.

Lastly, the PNC sends a channel time grant to the receiving DEV which only contains the new CTA.

Ed. action: Most of the text was already in D08, small changes to D09 made by Jay Bain and corrections to the language by JPKG (an evil “must” was purged).

Modified Clause 8.3.3.2 paragraph 1

(Ed. note: the changes are in the following sentences)

The slot assignments for dynamic GTS may change from superframe to superframe as required by the PNC. Slot assignments for pseudo-static GTS slots require directed channel time grant commands.

Ed. action: Changes made by Jay Bain, small change to the wording by JPKG, actual text below:

The slot assignments for dynamic GTSs may change from superframe to superframe as required by the PNC. Changing the slot assignments for pseudo-static GTSs requires directed channel time grant commands, as described in 8.4.3.1.

Modified Clause 8.3.3.2 paragraph 4

In any superframe there may be one or more DEVs in the piconet that receives the Beacon in error. This may not happen to the same DEV all the time but may happen to different DEVs at different times depending upon their location and type of interference they are subjected to. If a DEV did not receive the ~~CTA information~~ beacon correctly, it shall not access the ~~channel~~ dynamic GTS during CFP. Stations with pseudo-static GTS(s) are allowed to transmit during these GTS(s) as long as the number of consecutive lost beacons is less than or equal to MaxLostBeacons. A DEV shall stop transmitting in its pseudo-static GTS when the number of consecutive lost beacons exceeds MaxLostBeacons.

Ed. action: Corrections made by Jay Bain, editorial changes by JPKG, actual text is below:

If a DEV did not receive the beacon, it shall not access any dynamic GTSs during the CFP. Stations with pseudo-static GTSs are allowed to transmit during these GTSs as long as the number of consecutive lost beacons is less than or equal to aMaxLostBeacons. A DEV shall stop transmitting in its pseudo-static GTS when the number of consecutive lost beacons exceeds aMaxLostBeacons.

1.3.3 Private GTS (was formerly known as Unassigned GTS)

New Clause 3.3.3.3 Private GTS

A private GTS is a GTS where the same DEV is the source and destination. A private GTS is not used for communication in the piconet. Rather, it is used to reserve channel time for some other use. The other use may be for another 802.15.3 piconet, or a different type of network sharing the same channel.

Private GTS slots will usually be pseudo-static GTS slots, so that the slot is periodic for the other use. A DEV requests a private GTS slot by inserting it's own AD-AD in the source and destination address for the stream and channel time request.

Ed. action: Corrections made by Jay Bain, editorial changes by JPKG, actual text is below:

A private GTS is a GTS where the same DEV is both the source and the destination. A private GTS is not used for communication in the piconet. Instead, it is used to reserve channel time for some other use. The other use may be for another 802.15.3 piconet, or a different type of network sharing the same channel.

Private GTSs will usually be pseudo-static GTSs, so that the slot is periodic for the other use. A DEV requests a private GTS by using it’s own AD-AD as the originator and target address for the stream management command, 7.5.8.3, or channel time request command, 7.5.8.1.

1.3.4 Modified Stream Management Command

Reserved Byte for max Delayed ACK frames in stream Mgt Command (Issue 310) and source/target address field in stream management command (Issue 400). Modified Action Type and Reason Code address Issues

Figure 49 Modified Stream Management Command

Octets: 2	2	2	1	1	1	1	1	1	20
Command type	Length (=30)	Stream request identifier	Originator AD-AD	Target AD-AD	DSAA	Max Frames (delayed ACK)	Reason code/GTS type	reserved	Stream QoS parameters

Note: Reserved field will not be needed if the stream QoS parameters become odd after pending changes.

New Figure Reason code / GTS Type field (after figure 50)

4 bits	1 bits	3 bits
Reason Code	GTS Type	reserved

Replace line 50 and 51 of page 95 with the following:

The Originator AD-AD is the 8 bit address of the originator of the stream management command. The Target AD-AD is the 8 bit address of the target of the stream management command.

Add the following text after line 29, page 96:

Max Frames specifies the maximum number of frames that can be outstanding when the ACK policy for the stream is Delayed ACK.

Ed action: Changed command illustration, new figures and text shown below:

octets: 2	**2**	**1**	**1**	**1**	**1**	**2**	**20**
Command type	Length (=26)	Stream request indentifier	Originator AD-AD	Target AD-AD	Max frames (Del-ACK	Control Information	Stream QoS parameters

Figure 7—Stream management command format

The originator AD-AD is the allocated address of the DEV that is the originator of the stream manament command.

The target AD-AD is the allocated address of the DEV that is the target of the stream managment command.

The max frames field specifies the maximum number of frames that can be outstanding when the ACK policy for the stream is Del-ACK.

The control information field is illustrated in Figure 8.

bits b0:b1	**b2:b3**	**b4:b6**	**b7**	**b8:b11**	**b12**	**b13-b15**
Action Type	ACK Policy	Security	Direction	Reason code	GTS type	Reserved

Figure 8—Control information field in the stream management command

Modify the Action Type description as follows:

— Value of '0' means that this is a request for stream connection. This request is sent from the DEV that originates the stream management request to the PNC.
— Value of '1' means that this is the indication forwarded request frame sent from the PNC to the target of the stream management frame. This command contains the QoS-parameters, except the retransmission window, set by the PNC.
— Value of '2' means that this is a response to the stream connection sent from the target DEV to the PNC. The target DEV can lower the QoS-paremeters in the response.

- Value of '3' means that this is a Stream Management confirm forwarded response command for the stream connection. This request is sent by the PNC to the originator DEV to complete the stream connection. It is also sent from the PNC to the originator and target DEV to reject or disconnect a stream.
- Value of "4" means that the frame is sent by one of the DEVs to the PNC to reject or disconnect the stream
- Value of "5" means that the frame is sent by the PNC to one of the DEVs to reject or disconnect the stream

Ed. action: Text updated with editorial changes (JPKG), actual text follows:

- A value of '0' indicates that this is a request for stream connection. This request is sent from the DEV that originates the stream management request to the PNC.
- A value of '1' indicates that this is a forwarded request frame sent from the PNC to the target of the stream. This command contains the QoS-parameters set by the PNC, except for the retransmission window.
- A value of '2' indicates that this is a response to the stream connection. This is sent from the target DEV to the PNC. The target DEV can modify the QoS-parameters to smaller values in the response.
- A value of '3' indicates that this is a confirmation of the acceptance of the stream connection. This request is sent by the PNC to the originator DEV to complete the stream connection.
- A value of "4" indicates that the frame is sent by one of the DEVs to the PNC to reject or disconnect the stream
- A value of "5" indicates that the frame is sent by the PNC to one of the DEVs to reject or disconnect the stream

Modify the reason code as follows:

The reason code is a 4 bit field that is valid in the Stream Management Confirm when a stream connection is being completed, rejected or disconnected. It is also valid in the Stream Management response from the target to the PNC. Otherwise this field is ignored. Valid reason codes are:

- 0 ->Success
- 1 ->invalid stream parameters
- 2 -> non-negotiable stream parameters
- 3 -> system resources unavailable
- 4 -> bandwidth allocation failure
- 5 -> currently disassociating from the piconet
- 6 -> too many streams
- 7 -> lack of required security
- 8 -> unauthorized stream
- 9-> Stream rejected or disconnected by other DEV
- 10-16-> reserved

Add the following text:

GTS type zero signifies that the stream shall use dynamic GTS. GTS type one indicates that the stream shall use pseudo-static GTS.

Ed action: Reason codes and GTS type modified, actual text is below:

The reason code is a 4 bit field that is valid when a stream connection is being completed, rejected or disconnected. It is also valid in the response from the target to the PNC. Otherwise this field shall be ignored. Valid reason codes are:

- 0 -> success

— 1 -> invalid stream parameters
— 2 -> non-negotiable stream parameters
— 3 -> system resources unavailable
— 4 -> insufficient channel time available
— 5 -> currently disassociating from the piconet
— 6 -> too many streams
— 7 -> lack of required security
— 8 -> unauthorized stream
— 9-> Stream rejected or disconnected by other DEV
— 10 -> target unreachable
— 11-16-> reserved

The GTS type bit is set by the requesting DEV and shall be set to 0 for dynamic GTSs and shall be set to 1 for pseudo-static GTSs.

1.3.5 DEV GTS Status Information Element

Add new element GTS status information element to the beacon frame with type in every frame.

New Clause 7.4.x DEV GTS Status Information Element

This element is a 256 bit bitmap where each bit corresponds to an AD-AD. The PNC sends the DEV GTS Status element in the Beacon. The purpose of this element is to enable a DEV to know if any GTSs where it is a SA or DA have change since the last Beacon. If any of the GTSs for a given DEV has changed since the last Beacon, the bit corresponding to the AD-AD for that DEV shall be set to a one. If none of the GTSs for that DEV have changed since the last beacon, the bit corresponding to the AD-AD for that DEV shall be set to zero. If a broadcast GTS has changed since the last beacon, the broadcast GTS bit will be set, and not all bits in the bitmap.

Octets:1	1	8	
ElementID	Length=8	lsb	DEV GTS Status bitmap msb

Figure 2 DEV GTS Status Information Element

If a DEV correctly received beacon n, it does not need to process the CTAs in beacon n+1 if its DEV GTS Status bit and the broadcast GTS status bit in beacon n+1 is set to zero. If its DEV GTS Status bit is set to one or the broadcast GTS status bit, the DEV shall process the CTAs in that beacon.

Ed. action, add element and update information element table. New section for the information element is below:

7.4.13 DEV GTS status

The DEV GTS status element, illustrated in Figure 29, is a 256 bit bitmap where each bit corresponds to an AD-AD. The DEV GTS status element may only be sent by the PNC in the beacon. The purpose of this information element is to enable a DEV to know if any GTSs where it is either the SA or DA have changed since the last beacon. If any of the GTSs for a given DEV has changed since the last beacon, the bit corresponding to the AD-AD for that DEV shall be set to a one. If none of the GTSs for that DEV have changed since the last beacon, the bit corresponding to the AD-AD for

that DEV shall be set to zero. If a broadcast GTS has changed since the last beacon, only the broadcast GTS bit shall be set, and not all bits in the bitmap

octets: 1	1	8
Element ID	Length (=8)	(lsb) DEV GTS status bitmap (msb)

Figure 9—DEV GTS status information element

If a DEV correctly received beacon n, it does not need to process the CTAs in beacon n+1 if it's DEV GTS status bit and the broadcast GTS status bit in beacon n+1 are set to zero. If either it's DEV GTS status bit or the broadcast GTS status bit is set to one, then the DEV needs to process the CTAs in that beacon.

1.4 Document 01/503r0

1.4.1 Clause 8.8 Probe

- Fixed timeout not acceptable for EPS DEVs
- Require sending DEV to use EPS information, EPSTime & EPSNext to determine allowable time to resend probe
- Allow sender to switch EPS DEV to ACTIVE to speed up the exchange.
- Allow sender an aggressive resend

Ed. action: Text added to D09 by Jay Bain, new text in 8.9 follows:

EPS DEVs present a special case for use of the probe request command. It is the responsibility of the DEV sending a probe request command to understand when the EPS DEV will be able to receive it. Use of aProbeResponseDelay is not appropriate as a timeout. It is acceptable for the sending DEV to switch an EPS DEV into ACTIVE mode to improve the responsiveness. The sending DEV shall return the receiving DEV to EPS mode after completion of probing.

1.4.2 Clause 7.5.19 Device information

- This serves to support discovery
- Add fields to provide
  1) Status information
  2) A single EPS set

Ed. action: Changed the terminology to from "requested channel time" to be CTRB, updated the table and the text. The CTRB has both the CTRB type and the EPS set for the CTA. New text for the device information response command is given below:.

octets: 1	1	6	2	2	12	12	…	12
AD-AD	EPS info	Device ID	Capability field	Number of TX slots (= n)	CTRB for stream-1	CTRB for stream-2	…	CTRB for stream-n

Figure 10—Format of a record in device information response command

...

The EPS info field shall be formatted as illustrated in Figure 35 and is defined in 7.4.13.

...

The CTRB is the channel time request block for a given stream which shall be formatted as illustrated in Figure 13.

1.4.3 Clause 7.4.x Power management parameters element

- Add this element to the text
- Provides EPS sets a DEV is a member of
- Provides status information
  1) Power management mode
  2) Current state (ACTIVE/EPS)

Ed. action: Add new information element and update summary table, text given below:

7.4.14 Power management parameters

The power managment parameters element shall be formatted as illustrated in Figure 11. The purpose of this information element is to communicate the EPS information to the requesting DEV.

octets: 1	**1**	**1**	**1**	**1**	**…**	**1**
Element ID	Length (=2+n)	EPS info	EPS set 1	EPS set 2	…	EPS set n

Figure 11—Power management parameters element

The EPS set(s) are a listing of the numbers assigned by the PNC to all of the EPS set(s) to which the DEV belongs.

The EPS info field shall be formatted as illustrated in Figure 12

bits: b0-b1	**b2**	**b3-b7**
PowerManagementMode	EPS status	Reserved

Figure 12—EPS info field

PowerManagementMode indicates the power management mode of the DEV. If the device uses EPS power management at any time during a session, this field is set to 2. If the DEV will only use RPS power management, this field is set to 1. If the DEV will not use any power management, the field will be set to 0. This field is the result of information provided to the DEV and PNC by the DEV-host for use in this session.

EPS status indicates the current operation (EPS or ACTIVE) for an EPS DEV and has meaning for a DEV with Power-ManagementMode set = 2. A value of 1 is set if the DEV is currently in EPS mode. A value of 0 is set if the DEV is currently in ACTIVE mode.

1.4.4 Clause 6.3.1 changes

Parameter name	Request	Response	Indicate
Request type (xref 7.5.13)	Add	Add	--------
EPS set (xref 7.5.13)	Add	Add	--------
EPS status	Add	Add	
EPSTime (xref 7.5.13)	Retain	add	-------
EPSSync (develops EPSNext, xref 7.5.13)	Retain	------	--------
PeerPowerManagementMode	OK (note)	--------	delete
PeerPowerManagementRole	delete	--------	delete
PowerManagement Priority (xref 7.5.14)	OK	--------	--------

PeerWakeup	———	———	OK
DeviceID	———	———	delete
PowerManagement RecoveryMode	OK	———	———
Wakeup (xref 7.5.15, .16, .17)	OK (note)	———	———
Resultcode (match xref 7.5.13)	———	Retain and add values	———
ActualEPSTime	———	delete	———
PeerEPSTime	——	———	delete

Ed. action: This results in a re-write of 6.3.1, the results of which are given below:

6.3.1 Power management

This mechanism supports the process of establishment and maintenance of the power management mode of a DEV. The parameters used for these commands are defined in Table 4

6.3.1.1 MLME-POWERMGT.request

This primitive requests a change in the power management operation. The modes and operational considerations for sending and receiving DEVs in a piconet that want to use EPS operation in a variety of configurations. It is available to the DEV prior to association and may be used at additional times during operation to change configurations. This primi-

Table 4—MLME-POWERMGT.xxx primitive parameters

Name	Type	Valid Range	Description
RequestType	Integer	As defined in 7.5.7.1	Determines the type of power management request that is being made.
EPSSet	Octet	0-255	The EPS set for which the command applies, as defined in 7.5.7.1
EPSStatus	Enumeration	ACTIVE, EPS	The status of the DEV. The EPS states are defined in 8.13
EPSTime	Integer	0-65,535 ms	Time interval for an EPS device to be in reduced power state and unavailable for reception of packets. The operating super-frame length adjusts this value. A value of zero is to wake for each beacon. This element has no meaning if the EPS DEV is not in power management mode.Defined in 7.5.7.1
EPSSync	Boolean	True, False	When true, the MAC will force the phase of EPSTime to zero across the EPS set and determines the value of EPS next. False has no effect. Defined in 7.5.7.1
PowerManagementMode	Enumeration	PM_OFF, RPS, EPS	Describes the desired power management mode of the DEV.
PowerManagementRecoveryMode	Boolean	True, False	When true, the DEVs will perform recovery from errors rather than waiting for next EPSTime.
PowerManagementPriority	Enumeration	LOW, MEDIUM, HIGH	An indication of battery sensitivity. It is used by the PNC to allocate CTA locations. High indicates a very battery sensitive device requiring optimal CTA locations. Used in RPS and EPS modes.
Wakeup	Boolean	True, false	When true, the MAC is forced immediately into the ACTIVE state. This parameter has no effect if the current power management mode is ACTIVE.
PeerWakeup	Enumeration	EPS,ACTIVE, SHORTTERM	An indication that a mode change has occured based on network information as described in clause 8.13.3.7
ResultCode	Octet	0-255	The result of a power management command. The codes are the action type returned by an EPS action response command as defined in 7.5.7.2.

tive works in conjunction with the MLME-STREAM-CONNECT.xxx primitives, 6.3.13, to set the EPS modes. The semantics of the primitive are as follows:

```
MLME-POWERMGT.request (
                      RequestType
                      PowerManagementMode,
                      PowerManagementRecoveryMode,
                      PowerManagementPriority,
                      WakeUp,
                      EPSSet
                      EPSStatus
                      EPSTime,
                      EPSSync
                      )
```

The parameters for this command are defined in Table 4

6.3.1.1.1 When generated

This primitive is generated by the DME to implement the power-saving strategy of an implementation.

6.3.1.1.2 Effect of receipt

This request sets the DEVs power management parameters. The MLME subsequently issues a MLME-POWER-MGT.confirm that reflects the results of the power management change request.

6.3.1.2 MLME-POWERMGT.confirm

This primitive confirms the change in power management mode. The semantics of the primitive are as follows:

```
MLME-POWERMGT.confirm (
                      RequestType,
                      EPSSet,
                      EPSStatus,
                      EPSTime,
                      ResultCode
                      )
```

The parameters for this command are defined in Table 4

6.3.1.2.1When generated

This primitive is generated by the MLME as a result of an MLME-POWERMGT.request. It is not generated until the change has completed.

6.3.1.2.2 Effect of receipt

The DME is notified of the change of power management mode.

6.3.1.3 MLME-POWERMGT.indication

This primitive reports power management changes from a specific peer MAC entity. The semantics of the primitive are as follows

MLME-POWERMGT.indication (PeerWakeup)

The parameters for this command are defined in Table 4

6.3.1.3.1When generated

This primitive is generated by the DEV as a result of a command or activity by another DEV in the piconet.

6.3.1.3.2 Effect of receipt

The DME is notified of changes in power management configuration or to wake up for information reception.

1.4.5 Clause 7.5.15, 16 ,17 changes

- Switch to – commands call for informing DEV as well as PNC.
- 485r4 provides alternate mechanism to provide this the destination DEV

Ed. action: No specific suggestions here, apparently taken care of by changes in 01/485r4, which was added as a part of the ammendments. No action taken from the above information.

1.5 Document 01/328r4

With caveat to add CTRB parameter of desired maximum GTS.

Definitions;

- f_D = bits per second of delivered data.
- N_B = bits of source buffer available to store the data to be communicated.
- N_{MPDU} = bits of the data portion of one packet of data.
- N_{OH} = Equivalent bits of overhead of one packet of data including: actual MAC header bits, slot guard times, PHY or PLCP overhead, etc., everything-but-data, etc. It simplifies the explanation to express this as an equivalent number of bits.
- N_E = Channel (PHY) encoding, bits per symbol
- f_S = Channel symbol rate, symbols per second.
- T_{BCN} = Beacon Period

Octets: 1	1	1	1	2	2	2	2
Target Address	EPS status	EPS Set	Stream index	Allocation Period	Minimum GTS Time	Desired GTS Time	Maximum Allocation Delay

CTRB field that exists in draft D0.8

CTRB field proposed in this proposal

CTRB field proposed in document 01/485

Allocation period

a) $T_{AP} = N_B/f_D$
b) This is how often the N_B buffer must be sent to get the desired delivered data rate f_D.
c) This is the size of the source buffer divided by the desired data rate.
d) We need to send N_P packets over the network in order to transmit one buffer:
e) $N_P = N_B/N_{MPDU}$ Assume an integer for simplicity.
f) $N_T = N_{MPDU}+N_{OH}$, the total number of bits that would have to be sent over the network to cover both the data and the overhead.

Minimum GTS time

a) T_{GTS} is the total time requested for GTS allocated in order to send one buffer of data.
b) $T_{GTS} = (N_P N_T / N_E f_S)$.
c) For the sending N_P packets of equivalent size, $N_T = (N_{MPDU} + N_{OH})$ with a PHY encoding of N_E bits per symbol and a PHY symbol rate of f_S.\
d) $1/(N_E f_S)$ is a constant as long as the symbol rate and encoding method is unchanged.
e) From before: $N_P = N_B/N_{MPDU}$
f) Stating the obvious: N_T and N_P will also be constants if the transmit packet size and the transmit buffer size both remain constant.

Desired GTS time, i.e. GTS time per allocation period

a) T_{DMG} defines the maximum amount of GTS time per allocation period that a DEV is capable of using. $T_{DMG} > T_{GTS}$
b) If there is unused bandwidth, the extra channel bandwidth can be divided up among DEVs based on what they can actually use.
c) Allows channel utilization to be maximized.

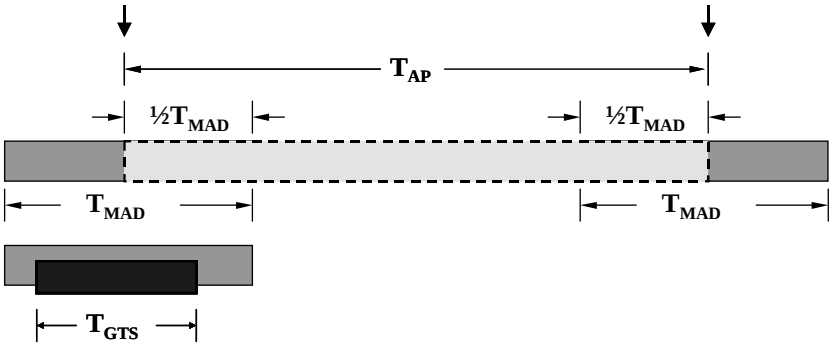
Maximum allocation delay

a) T_{MAD} defines an allowable time jitter to the allocation of time slots by the piconet coordinator, PNC.
b) The maximum allocation delay starts before the end of the allocation period, but does not affect the PNC's reference timing of the T_{AP} period. It allows some variability in position of the GTS slot(s).

Requirements for Data Rate Only QoS

a) The transmit buffer is large enough to accept GTS slots anywhere in the superframe.
b) The amount of bandwidth only depends on the data rate required fd, the data rate available, and the amount of overhead.
c) $T_{AP} / TGTS = (f_D/f_S) / (N_T / N_E N_{MPDU})$
d) Define T_{MAD} = 0xFFFF, which will be a special value indicating "anywhere in CFP".

The T_{GTS} (blue) time is the total amount of slot time that must be allocated by the PNC within the T_{MAD} (green) time centered around the start of the T_{AP} interval (vertical arrows).



Ed. action: Change the CTRB to match the figure. The new figure and new text are below:

The allocation period has different meanings depending on the value of the CTRB type field.

If the CTRB type field indicates that this request is for an EPS mode channel time request, then the allocation period is an integer, N, ranging from 1-65535 that indicates what fraction, 1/N, of the EPS slots require this allocation. A zero value is not allowed for this field and shall cause the command to be ignored by the recipient.

Otherwise, the allocation period is for an ACTIVE mode CTA and defined as the block of time that the DEV is using to calculate the other parameters in this block. The resolution of this field is 1 ms and so the range of this field is [0-65535] ms.

The minimum GTS time is the minimum duration of the time that is acceptable at the requesting DEV in any time slot. The resolution of this field is 8 ms and so the range of this field is [0-524280] μs.

The desired GTS time is the amount of time that the DEV would prefer to have allocated. The resolution of this field is 8 ms and so the range of this field is [0-524280] μs.

The maximum allocation delay defines the allowable time jitter in the allocation of the GTSs with respect to the allocation period. The value 0xFFFF indicates that the DEV has no jitter requirements for the GTS. The resolution of the channel time field is 8 ms and so the range of requested time is [0-524272] μs.

The relationship of the allocation period, minimum GTS time, desired GTS time and maximum allocation delay is discussed in 8.4.3.3.

New section added to clause 8, is 8.4.3.3, text follows;

8.4.3.3 QoS considerations for channel time allocation

The DEV needs to map its QoS requirements into the parameters of the channel time request block, Figure 73. This subclause describes one way to map the DEVs requirements into the CTRB parameters. A compliant DEV may use another method to determine what values of the CTRB are required to fulfill its throughput and QoS requirements. In this subclause, the following terms are used:

- f_D = Bits/s of delivered data.
- N_B = Number of bits of source buffer available to store the data to be communicated.
- N_{MPDU} = Number of bits in the data portion of one packet of data.
- N_{OH} = Equivalent bits of overhead of one packet of data including: actual MAC header bits, slot guard times, PHY overhead, etc. (i.e. everything but the data). It simplifies the explanation to express this as an equivalent number of bits.
- N_E = Channel (PHY) encoding in bits/symbol
- f_S = Channel symbol rate in symbols/s.
- T_{BCN} = Beacon period in seconds.
- T_{AP} = The allocation period, in seconds.
- T_{GTS} is the total time, in seconds, requested for GTS allocated in order to send one buffer of data.
- T_{MAD}, defines an allowable time jitter to the allocation of time slots by the piconet coordinator, PNC.

The allocation period, is a time reference that the DEV uses in calculating the other parameters. In general, the allocation period is how often the N_B buffer must be sent to get the desired delivered data rate f_D, i.e . $T_{AP} = N_B/f_D$. In order to transmit the buffer, the DEV needs to send N_P packets over the network, where each packet is $N_T = N_{MPDU}+N_{OH}$ bits in length.

The minimum GTS time is then be calculated using $T_{GTS} = (N_P N_T / N_E f_S)$. The desired GTS time should not be more than the maximum amount of GTS time per allocation period that a DEV is capable of using. If there is unused bandwidth in the piconet, the PNC could choose to allocate the extra channel bandwidth among the DEVs based on what they can actually use. This allows channel utilization to be maximized.

The maximum allocation delay starts before the end of the allocation period, but does not affect the PNC's reference timing of the T_{AP} period. This allows some variability in position of the GTS(s). The DEV determines this time based on the jitter requirements of the data stream that it needs to send. Figure ?? illustrates the position of T_{MAD} within T_{AP}. The

T_{GTS} time is the total amount of slot time that needs to be allocated with in the T_{MAD} time cntered around the start of the T_{AP} interval.

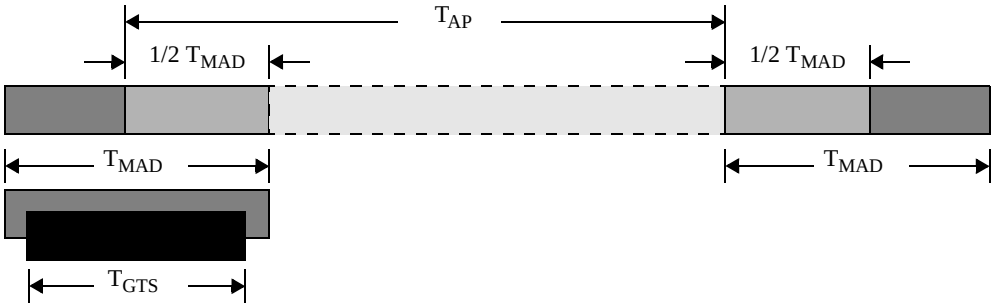


Figure 14—Illustration of the position of T_{MAD} within T_{AP}

Based on the requirements for T_{MAD} within T_{AP}, the PNC has many possible ways to allocate the GTSs required by the DEV, as illustrated in .

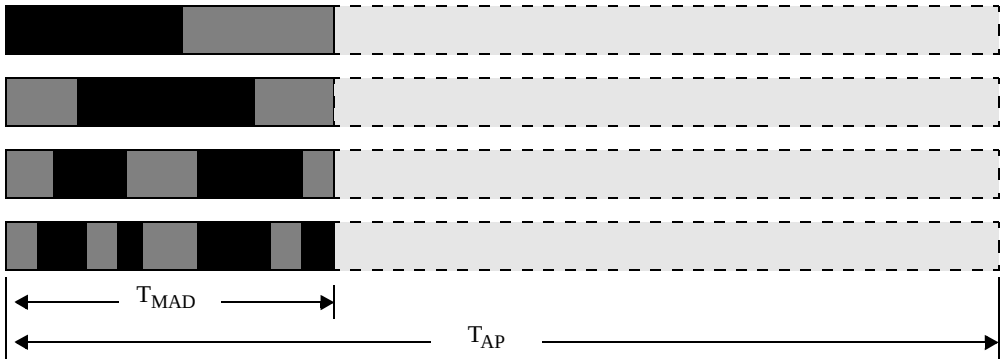


Figure 15—Possible GTS allocations that meet T_{MAD} requirements

For streams with only data rate requirements for QoS, the transmit buffer needs to be large enough to accept GTS slots anywhere in the superframe. The amount of bandwidth only depends on the data rate required, f_D, the data rate available, and the amount of overhead. In this case, $T_{AP}/ T_{GTS} = (f_D/f_S) / (N_T / N_E N_{MPDU})$ and the value of T_{MAD} is set to 0xFFFF, which is the special value indicating “anywhere in CFP”, 7.5.10.1.

1.6 Document 01/517r2

Octets: 1	1	1	N
Element ID	Element Length	AD-AD	Application Specific Data

Figure ??--Application Specific Information Element (ASIE)

The Element ID field shall have the value (the next available reserved ID).

The Element Length shall be K+1 where K is the number of octets in the Application Specific Data field.

The AD-AD is set by the PNC as the index of the application specific capable device that will make use of this information element.

The Application Specific Data is specified by the PNC. Its use by the application specific capable device is beyond the scope of this standard.

More than one ASIE may be placed in any beacon by the PNC. All ASIEs shall be the last information elements in the beacon.

The ASIE shall be the last information element in the beacon.

The ASIE shall only be used by the PNC after negotiating the application specific capability with a DEV using a standard a GTS or CFP message exchange.

The negotiation of the application specific capability between the DEV and the PNC is beyond the scope of this specification.

The use of the DATA field of the information element is beyond the scope of this specification.

Ed. action: Added new element and updated the summary table. Added following acronym definition:

ASIE application specific information element

New text and figure follows:

7.4.15 Application specific information

The application specific information element shall be formatted as illustrated in Figure 16. The purpose of this information element is to allow custom information for enhanced operation that is outside of the scope of this standard .

octets: 1	**1**	**1**	**variable**
Element ID	Length	AD-AD	Application specific data

Figure 16—Application specific information element

The AD-AD is set by the PNC as the index of the application specific capable device that will make use of this information element.

The application specific data is specified by the PNC. Its use by the application specific capable device is beyond the scope of this standard.

More than one application specific information element (ASIE) may be placed in any beacon by the PNC. All ASIEs shall be the last information elements in the beacon. The ASIE shall only be used by the PNC after negotiating the application specific capability with a DEV using a standard a GTS or CFP message exchange. The negotiation of the application specific capability between the DEV and the PNC is beyond the scope of this specification. The use of the application specific data field of the information element is beyond the scope of this specification.

1.7 Document 01/502r1

1.7.1 Remove Delayed ACK Expedite (Issue 248)

Issue: This is redundant with 7.2.1.8 Del-ACK Request in the frame control field. Section and figure numbers are from D07.

Clause 7.2.4 Stream ID

Remove pg 75 line 1 and 2:

"The Del-ACK expedite field is used to request that the accumulated Del-ACK's be sent as soon as possible. This is described in more detail in 8.7.3."

Modified Figure 5 Stream ID Field

its: 0	**1:3**	**4:11**	**12:15**
tream type	**Priority**	**Stream index**	**Reserved**

Ed. action: Deleted the Del-ACK, re-arranged the bits so that stream index was on a byte boundary and fiddled some of the words. Complete stream control section is below:

7.2.4 Stream control

The stream control field is 16 bits in length and is used to uniquely identify a data stream. This field is valid only for data frames. This field is is set to zero, and ignored upon reception, in all other frame types.

bits 0:7	**8**	**9:11**	**12:15**
Stream index	Stream type	Priority	Reserved

Figure 17—Stream control field

This field contains three sub-fields, stream index, stream type and priority.

The stream index field is an 8-bit field with the value of zero reserved for non-stream data. The DEVs use the rest of the values of the stream index as dynamically assigned by the PNC during the setup of the data stream. The PNC allocates a unique value of stream index for each stream in the piconet.

The stream type shall be set to ‘1’ for streams requiring isochronous services and shall be set to ‘0’ otherwise.

The priority field indicates the priority of the stream and is defined in A.3.

Any frame that does not belong to an established stream and does not need a stream connection is a non-stream data frame. Any non-stream data frame is transmitted with the stream index value of zero. The use of a stream connection for asynchronous or isochronous data is up to the DEV.

1.7.2 Multicast Stream Establishment (Issue 339)

Issue: There is currently no mechanism specified to set up a multicast stream

Clause 7.5.21 Figure 49

Modify

For a broadcast stream, the involvement of intended receiver is precluded.

To

For a broadcast or multicast stream, the involvement of intended receiver is precluded.

Ed. action: This sentence occurs in 8.6.1, not in 7.5.21 Figure 49. The sentence was changed as indicated in D09. Excerpt from D09 is below

8.6.1 Stream connection

A stream shall be connected only after tripartite communication/negotiation among the DEV that is originating the stream, the DEV that is the intended receiver of the stream and the PNC. For a broadcast or multicast stream, the involvement of intended receiver is precluded. Once connected, the stream is sent in a peer-to-peer style.

1.7.3 Slotted ALOHA Reference for the Bibliography (Requested by Jim Allen)

Stallings, Data and Computer Communications, Second Edition, Macmillan, 1988, pp300-302

Ed. action: reference added to Bibliography clause, text is below:

[B1] Stallings, *Data and Computer Communications*, *Second Edition*, Macmillan, 1988, pp. 300-302

~~1.7.4 Channel Time Requests - only from stream source (Issue 412)~~

~~Issue: Since the originator of a stream request can be the source or the target of the stream, it is not clear which DEV can make Channel time requests: the originator of the request, the source of the stream, either if they are not the same?~~

~~8.3.3.2 Channel Time Allocation (CTA) and channel time usage~~

~~Add the following sentence the first paragraph:~~

~~Only the DEV that is the source of a stream can send Channel Time Request commands for that stream.~~
Tabled 11/13/01

Ed. action: No action taken.

1.7.5 Association Response Success Reason Code (Issue 410)

Issue: DEV should not have to look at AD-AD field to determine success or failure of the association. That should be in the reason code.

Association Response Command Format 7.5.3

~~Remove the following sentence from the sixth paragraph:~~

~~If this field contains the association address (0xFE), the DEV is not allowed to associate for the reason mentioned in the reason code.~~ Keep this 11/13/01, James will modify as appropriate.

Change the reason codes as follows:

The valid reason codes are:

- 0 -> Success
- 1->Already serving maximum number of DEVs
- 2 -> Lack of available bandwidth to serve the DEV
- 3 -> Channel is severe to serve the DEV
- 4 -> PNC is turning off with no AC in the piconet
- 5 -> DEV wishes to disassociate
- 6 -> Channel change is in progress
- 7 -> PNC hand over is in progress
- 8 -> DEV authentication failed
- 9-255 -> reserved

Ed. action: Reason codes modified as indicated above. I changed code 3 to read “Channel is too severe to serve the DEV”

1.7.6 MAC Frame Formats (Issue 393)

Issue: Since there are now only 4 frame types, all DEVs shall be able to process all frames, except only PNC capable devices must be capable of creating beacon frames.

Clause 7

Replace "In addition" through the end of the paragraph with:

In addition, every DEV shall be able to construct these frame formats for transmission, and to decode frame formats upon validation following reception. The only exception is that a DEV that is not PNC capable need not be able to construct beacon frames.

Ed. action: Changed as indicated in D09.

1.7.7 Beacon Information Elements (Issue 398)

Issue: D07 does not explicitly specify which information elements are optional or mandatory in the Beacon.

Table 62

Information Element	Note	Present in beacon
Device ID	…	In Every Beacon
Piconet Synchronization Parameters	…	In Every Beacon
TPC element	…	As Needed
Channel change	…	As Needed
Channel time allocation (CTA)	…	As Needed
Parent Device ID (if child or neighbor piconet)	…	As Needed

Ed. action: Added new column to the beacon table, new table is below:

Table 5—Beacon frame body

Information element	Sub-clause	Note	Present in beacon
Device identifier	7.4.1	IEEE 802 address of the PNC	In every beacon
Piconet synchronization parameters	7.4.2	Beacon number and other time duration elements	In every beacon
Piconet maximum transmit power	7.4.10	Sets the max TX power level in the piconet	As needed
Channel change	7.4.5	During change to new channel	As needed
Channel time allocation	7.4.11	All the channel time allocation in the current superframe	As needed
Device identifier of parent PNC (if child or neighbor piconet)	7.4.1	IEEE 802 address of the parent PNC	As needed

1.7.8 Open Scan (Issue 317)

Issue: Need to define passive scanning for any PNID. 8.2 currently only addresses searching for a specific PNID.

Clause 8.2.1 Scanning through Channels

Replace the following sentence:

While scanning, the DEV shall ignore all the received frames with a different PNID than the one for which the DEV is searching.

With:

If open scan is specified in the MLME-SCAN.request, the DEV shall perform open scan. In open scan, the DEV searches for any PNID. If open scan is not specified, the DEV shall ignore all the received frames with a different PNID than the one for which the DEV is searching.

Ed. action: Changed as indicated by Jay Bain.

Clause 6.3.2.1 MLME-SCAN.request

Modify the primitive as follows:

```
MLME-SCAN.request (     OpenScan,
                        PNID,
                        ChannelList,
                        ChannelScanDuration)
```

Add the following Entry to Table 5

Open Scan	Boolean	TRUE/FALSE	Identifies if scan is Open Scan or not

Ed. action: OpenScan added as a parameter and entry added to the tabel for MLME-SCAN.request as indicated above. Added “Open scan is defined in 8.2.1” to the description element.

1.7.9 Peer Discovery (Issue 365)

Issue: … any DEV may send any directed command frame to any other DEV in the piconet to determine if the destination DEV is still present in the piconet." Comment: This sentence implies that the DEV can send a Retransmission req, Channel Status request, Association req, Disassociation req, and Sleep state req as "pings" to the destination device. Doesn't this get confusing if all the transmitting DEV wants to do is determine whether the destination DEV is present?

Clause 8.8, paragraph 3

Modify the sentence as follows:

In addition to the above, any DEV may send probe request command with the information request field set to zero and ACK set to immediate ACK ~~any appropriate directed command frame~~ to any other DEV in the piconet to determine if the destination DEV is still present in the piconet.

James will add additional clarification to the probe request text to specify that if the request field is null, the DEV shall respond with immediate ACK only.

Ed action: Sentence modified as indicated above in D09. The modified probe request command is shown below, text that was added is underlined.

The probe request command is used either to request information about a DEV or to see if a DEV is still present in the piconet. This command may be exchanged between any two DEVs in the piconet. The individual information elements used in this frame are described in 7.4. The stream control field in the probe request frame header shall be set to 0x00 and shall be ignored upon reception. The probe request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	2	2	Variable
Command type	Length	Information request	Information elements

Figure 18—Probe request command format

The least significant 15 bits of the information request field is a bitmap to indicate the information requested of the destination of DEV. The sender sets a value of '1' in a bit to request the information element that corresponds to the bit position. Otherwise the sender sets the bit to '0'. The bit position for an information element is same as the value of the element-ID for that information element. That is, the bit position of 'n' in information request field corresponds the information element whose element ID, Table 74, is 'n'. An all-zero value in this field means that the source DEV is not expecting any probe information from the destination DEV, but is providing the information about itself to the destination DEV in the elements following this field. In this case, the destination DEV only ACKs the frame if it is received correctly and does not respond with a probe response command.

1.8 Document 01/410r0

Ed. action: Changes made by Allen Heberling, in D09. New MLMEs include: MLME-CHANNEL-STATUS, MLME-CHANGE-CHANNEL, MLME-PROBE-PNC, MLME-DEV-INFO and MLME-STREAM-CTA.

1.9 Document 01/469r3

Ed. action: All changes from r2 are in D08, new information in r3 is editorial, i.e. it is in MLMEs and MSCs that reflect what is in the normative text but do not provide new technical information. All edits and updates performed by Allen Heberling.

1.10 Document 01/530r2

1.10.1 Security commands

This set of commands is used to establish security and privacy functions between DEVs and a PNC in the piconet. In all cases involving the establishment and maintainence of security and privacy in the piconet, the PNC is defined as hosting the Piconet Security Manager (PSM) function in the piconet. Identically, the PNC, which is defined as hosting the PSM, shall always operate as the active security manager in a piconet.

Ed. action: Added to D09 as "Authentication and security commands."

Ed. action: All of the new commands were added to the command summary table.

The commands below required the following changes:

3) Retyr, Frag-start and frag-end shall not be set to zero, otherwise you cannot fragment the command.

4) There is a lot of clause 10 stuff in here (i.e. it is a functional description rather than just a description of the frame format.

5) Frame position no longer exists, thanks for finding it.

6) "shall always" is redundant, shall means always.

1.10.1.1 Authentication request command

The authentication request command is used to request authentication of the requesting DEV. This command may be exchanged between a DEV and a PNC operating as the active security manager in the piconet.

An associated DEV uses this command to request authentication within a piconet from the security manager. The DEVAuthenticationPublicKey may be simply a public key, a public-key certificate or some other construct that communicates public-key related information.

Only a DEV currently associated with a piconet with non-zero Authentication Mode shall send this command to the PNC. This command initiates authentication of the DEV with the PNC.

The ACK policy shall always be set to request immediate acknowledgement.

The frame position, frag-start, frag-end, retry, Del-ACK request, SEC and Repeater sub-fields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The DA shall always be set to all-zero address, meant to indicate the PNC address.

The individual information elements used in this frame are described in 7.4. The authentication request command frame structure shall be formatted as illustrated in Figure 19

octets: 2	**2**	**2**	**Variable**
Command type	Length	Public Key Length	DEV AuthenticationPublicKey

Figure 19—Authentication request command format

PublicKeyLength and AuthoritySignatureLength are set to the lengths of the variable length DEVAuthenticationPublicKey and AuthoritySignature fields.

DEVAuthenticationPublicKey is the public key provided to the DEV by a key management authority.

Ed. action: Inserted with some changes.

7.5.3.1 Authentication request command

An associated DEV uses this command to request authentication within a piconet from the PNC.

The ACK policy shall be set to request immediate acknowledgement. The Del-ACK request, SEC and Repeater subfields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The authentication request command frame structure shall be formatted as illustrated in Figure 19

octets: 2	2	2	Variable
Command type	Length	PublicKeyObjectLength	DEVPublicKeyObject

Figure 20—Authentication request command format

The DEVPublicKeyObject may be simply a public key, a public-key certificate or some other construct that communicates public-key related information.

The PublicKeyObjectLength is the length in octets of the variable length DEVPublicKeyObject.

1.10.1.2 Authentication response command

The authenitcation response command is used to respond to a challenge confirm command from the PNC operating as the active security manager, to an associated DEV. This command may be exchanged between the active PNC and the authenticating DEV in the piconet.

The AuthenticationInfoLength is a 2 octet integer that specifies the length in bytes of the AuthenticationInfo parameter.

The AuthenticationInfo parameter is variable in length, as specified by the active cipher suite. The format of the information contained in the AuthenticationInfo parameter is also defined by the active cipher suite.

The result code is TRUE if the PSM accepts the DEV for authentication, and FALSE if it does not.

Only a DEV currently associated with a piconet with non-zero Authentication Mode shall send this command to the PNC.

The ACK policy shall always be set to request immediate acknowledgement.

The frame position, frag-start, frag-end, retry, Del-ACK request, SEC and Repeater sub-fields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The DA shall always be set to all-zero address, meant to indicate the PNC address.

The individual information elements used in this frame are described in 7.4. The authentication response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	2	2	Variable	1
Command type	Length	AuthenticationInfoLength	AuthenticationInfo	ResultCode

Figure 21—Authentication response command format

Upon receiving an challenge confirm command from an associated DEV, the PNC acting as the PSM generates an authentication response command that is sent to the authenticating DEV indicating whether or not the authentication has been approved by the PNC operating as the active security manager, and including additional authentication information for authentication, if needed.

Ed. action: New command below:

7.5.3.2 Authentication response command

The authentication response command is used by the PNC to respond to an authentication request command from the DEV.

The ACK policy shall be set to request immediate acknowledgement. The Del-ACK request, SEC and repeater sub-fields in frame control field of the MAC header in this command shall be set to 0 and shall be ignored upon reception.

The authentication response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**2**	**Variable**	**1**
Command type	Length	AuthenticationInfoLength	AuthenticationInfo	ResultCode

Figure 22—Authentication response command format

The AuthenticationInfoLength is a 2 octet integer that specifies the length in bytes of the AuthenticationInfo parameter.

The AuthenticationInfo parameter is variable in length, as specified by the active cipher suite. The format of the information contained in the AuthenticationInfo parameter is also defined by the active cipher suite.

The ResultCode shall be set to 1 if the PNC accepts the DEV for authentication, and shall be set to 0 otherwise.

1.10.1.3 Challenge request command

The challenge request command is used to initiate a public key challenge from the PNC operating as the active security manager, to an associated DEV. This command may be exchanged between the active PNC and the authenticating DEV in the piconet. It includes a public key challenge that is dependent on the cipher suite that is being used and the public key of the PNC operating as the active security manager.

The PublicKeyChallengeLength is a 2 octet integer that specifies the length in bytes of the PublicKeyChallenge parameter.

The PublicKeyChallenge parameter is variable in length, as specified by the active cipher suite. The format of the information contained in the PublicKeyChallenge parameter is also defined by the active cipher suite

The AuthenticationFailureTimeout parameter is a 2 octet value that determines the maximum wait time in mS until the authentication request command is no longer valid as issued by the DEV to the PNC.

The ACK policy shall always be set to request immediate acknowledgement.

The frame position, frag-start, frag-end, retry, Del-ACK request, SEC and Repeater sub-fields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The DA shall always be set to all-zero address, meant to indicate the PNC address.

The individual information elements used in this frame are described in 7.4. The challenge request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	2	2	Variable	2
Commandtype	Length	PublicKeyChallengeLength	PublicKeyChallenge	AuthenticateFailureTimeout

Figure 23—Challenge request command format

Upon receiving an challenge confirm command from an associated DEV, the PNC acting as the PSM generates a challenge response command (authentication challenge) that is sent to the authenticating DEV.

Ed. action: New text below:

7.5.3.3 Challenge request command

The challenge request command is used to initiate a public key challenge from the PNC to an associated DEV. It includes a public key challenge that is dependent on the cipher suite that is being used and the public key of the PNC.

The ACK policy shall always be set to request immediate acknowledgement. The Del-ACK request, SEC and repeater sub-fields in frame control field of the MAC header in this command shall be set to 0 and shall be ignored upon reception.

The challenge request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	2	2	Variable	2
Commandtype	Length	PublicKeyChallengeLength	PublicKeyChallenge	AuthenticateFailureTimeout

Figure 24—Challenge request command format

The PublicKeyChallengeLength is the length in octets of the PublicKeyChallenge parameter.

The PublicKeyChallenge parameter is variable in length, as specified by the active cipher suite. The format of the information contained in the PublicKeyChallenge parameter is also defined by the active cipher suite

The AuthenticationFailureTimeout parameteris the maximum wait time in ms until the challenge request command issued by the DEV to the PNC is no longer valid.

1.10.1.4 Challenge response command

The challenge response command is used by an associated DEV to respond to a public key challenge from the PNC operating as the active security manager. This command may be exchanged between an authenticating DEV and the active PNC in the piconet. It includes a public key proof that is dependent on the cipher suite that is being used.

The PublicKeyProofLength is a 2 octet integer that specifies the length in bytes of the PublicKeyProof parameter.

The PublicKeyProof parameter is variable in length, as specified by the active cipher suite. The format of the information contained in the PublicKeyProof parameter is also defined by the active cipher suite.

Only a DEV currently associated with a piconet with non-zero Authentication Mode shall send this command to the PNC.

The ACK policy shall always be set to request immediate acknowledgement.

The frame position, frag-start, frag-end, retry, Del-ACK request, SEC and Repeater sub-fields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The DA shall always be set to all-zero address, meant to indicate the PNC address.

The individual information elements used in this frame are described in 7.4. The challenge response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**2**	**Variable**
Command type	Length	PublicKeyProofLength	PublicKeyProof

Figure 25—Challenge response command format

Upon receiving a public key challenge from the PNC operating as the active security manager, an associated DEV generates a challenge response command that is sent to the PNC.

Ed. action: New text below:

7.5.3.4 Challenge response command

The challenge response command is used by an associated DEV to respond to a public key challenge from the PNC. It includes a public key proof that is dependent on the cipher suite that is being used.

The ACK policy shall always be set to request immediate acknowledgement. The Del-ACK request, SEC and repeater sub-fields in frame control field of the MAC header in this command shall be set to 0 and shall be ignored upon reception.

The challenge response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**2**	**Variable**
Command type	Length	PublicKeyProofLength	PublicKeyProof

Figure 26—Challenge response command format

The PublicKeyProofLength specifies the length in octets of the PublicKeyProof parameter.

The PublicKeyProof parameter is variable in length, as specified by the active cipher suite. The format of the information contained in the PublicKeyProof parameter is also defined by the active cipher suite.

1.10.1.5 Request key request command

The request key request command is used by an associated DEV to to request the transmission of a key from the PNC operating as the active security manager. This command may be exchanged between an authenticated DEV and the active PNC in the piconet.

The KeyPurpose is a 2 octet integer that specifies the purpose for which the key is intended to be used, as enumerated in the cipher suite list.

The KeyRequestTimeout parameter is a 2 octet value that determines the maximum wait time in mS until the request key request command is no longer valid as issued by the DEV to the PNC.

Only a DEV currently associated and authenticated with a piconet with non-zero Authentication Mode shall send this command to the PNC.

The ACK policy shall always be set to request immediate acknowledgement.

The frame position, frag-start, frag-end, retry, Del-ACK request, SEC and Repeater sub-fields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The DA shall always be set to all-zero address, meant to indicate the PNC address.

The individual information elements used in this frame are described in 7.4. The request key request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**1**	**1**
Command type	Length	KeyPurpose	KeyRequestTimeout

Figure 27—Request key request command format

When an associated and authenticated DEV needs a key from the PNC operating as the active security manager, the DEV generates the request key request command that is sent to the PNC.

Ed. note: Text for new command is below:

7.5.3.5 Request key request command

The request key request command is used by an associated DEV to to request the transmission of a key from the PNC.

The ACK policy shall always be set to request immediate acknowledgement. The Del-ACK request, SEC and sepeater sub-fields in frame control field of the MAC header in this command shall be set to 0 and shall be ignored upon reception.

The request key request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**1**	**1**
Command type	Length (=2)	KeyPurpose	KeyRequestTimeout

Figure 28—Request key request command format

The KeyPurpose specifies the purpose for which the key is intended to be used, as defined in the cipher suite list.

The KeyRequestTimeout parameter is the maximum wait time in ms until the request key request command issued by the DEV to the PNC is no longer valid.

1.10.1.6 Request key response command

The request key response command is used by the PNC operating as the active security manager to respond to an associated and authenticated DEV with either an encrypted version of the requested key or an indication that the key request was denied. This command may be exchanged between the active PNC and an an associated and authenticated DEV in the piconet.

The KeyPurpose is a 2 octet integer that specifies the purpose for which the key is intended to be used, as enumerated in the cipher suite list.

The EncryptedKeyObjectLength is a 2 octet integer that specifies the length in bytes of the EncryptedKeyObject.

The EncryptedKeyObject is a variable length object specified by the active cipher suite. The EncryptedKeyObject contains a payload data key that may be encrypted using a public key or a symmetric private key, in a format defined by the active cipher suite.

The ReasonCode is TRUE if the PSM accepts the request key request command from the requesting DEV, and FALSE if it does not.

The individual information elements used in this frame are described in 7.4. The request key response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**1**	**2**	**Variable**	**1**
Command type	Length	KeyPurpose	EncryptedKeyObjectLength	EncryptedKeyObject	ReasonCode

Figure 29—Request key response command format

Upon receiving a key request from an associated DEV, the PNC as security manager to responds with either an encrypted version of the requested key or an indication that the key request was denied.

Ed. note: Text for new command is below:

7.5.3.6 Request key response command

The request key response command is used by the PNC to respond to an associated and authenticated DEV with either an encrypted version of the requested key or an indication that the key request was denied.

The request key response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**1**	**1**	**2**	**Variable**
Command type	Length	KeyPurpose	ReasonCode	EncryptedKeyObjectLength	EncryptedKeyObject

Figure 30—Request key response command format

The KeyPurpose specifies the purpose for which the key is intended to be used, as defined in the cipher suite list.

The EncryptedKeyObjectLength is the length in octets of the EncryptedKeyObject.

The EncryptedKeyObject is a variable length object specified by the active cipher suite. The EncryptedKeyObject contains payload data key that may be encrypted using a public key or a symmetric private key, in a format defined by the active cipher suite.

The ReasonCode shall be set to 1 if the PNC accepts the request key request command from the DEV and shall be set to 0 otherwise.

1.10.1.7 Distribute key request command

The distribute key request command is used by the active security manager to send a key to a specific DEV. This command may be exchanged between the active PNC and an associated and authenticated DEV in the piconet. It includes a EncryptedKeyObject that is dependent on the cipher suite that is being used.

The KeyPurpose is a 2 octet integer that specifies the purpose for which the key is intended to be used, as enumerated in the cipher suite list.

The EncryptedKeyObjectLength is a 2 octet integer that specifies the length in bytes of the EncryptedKeyObject.

The EncryptedKeyObject is a variable length object specified by the active cipher suite. The EncryptedKeyObject contains a payload data key that may be encrypted using a public key or a symmetric private key, in a format defined by the active cipher suite.

The DistributeKeyFailureTimeout parameter is a 2 octet value that determines the maximum wait time in mS until the distribute key request command is no longer valid as issued by the PNC to the DEV.

The individual information elements used in this frame are described in 7.4. The distribute key request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**1**	**2**	**Variable**	**2**
Command type	Length	KeyPurpose	EncryptedKey ObjectLength	EncryptedKeyObject	DistributeKeFailure Timeout

Figure 31—Distribute key request command format

Upon receiving a key request from an associated DEV, the PNC as security manager to responds with either an encrypted version of the requested key or an indication that the key request was denied.

Ed. note: Text for new command is below:

7.5.3.7 Distribute key request command

The distribute key request command is used by the PNC to send a key to a specific DEV.

The distribute key request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	2	1	2	Variable	2
Command type	Length	KeyPurpose	EncryptedKey ObjectLength	EncryptedKeyObject	DistributeKey FailureTimeout

Figure 32—Distribute key request command format

The KeyPurpose specifies the purpose for which the key is intended to be used, as defined in the cipher suite list.

The EncryptedKeyObjectLength is the length in octets of the EncryptedKeyObject.

The EncryptedKeyObject is a variable length object specified by the active cipher suite. The EncryptedKeyObject contains payload data key that may be encrypted using a public key or a symmetric private key, in a format defined by the active cipher suite.

The DistributeKeyFailureTimeout parameter is the maximum wait time in ms until the distribute key request command issued by the PNC to the DEV is no longer valid.

1.10.1.8 Distribute key response command

The distribute key response command is used by an associated and authenticated DEV to respond to the distribute key request command. A a result code indicating the success or failure of the distribute key response command is sent to the PNC operating as the active security manager. This command may be exchanged between an authenticated and associated DEV and the active PNC in the piconet.

Only a DEV currently associated and authenticated with a piconet with non-zero Authentication Mode shall send this command to the PNC.

The ACK policy shall always be set to request immediate acknowledgement.

The frame position, frag-start, frag-end, retry, Del-ACK request, SEC and Repeater sub-fields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The DA shall always be set to all-zero address, meant to indicate the PNC address.

The ResultCode is TRUE if the DEV correctly received the distribute key request command, and FALSE if it did not.

The individual information elements used in this frame are described in 7.4. The distribute key response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	2	1
Command type	Length	ResultCode

Figure 33—Distribute key response command format

Upon the associated and authenticated DEV receiving a distribute key request from the PNC as security manager, the DEV responds with the distribute key response frame or indicating whether the distribute key request command ws successful.

Ed. note: Text for new command is below:

7.5.3.8 Distribute key response command

The distribute key response command is used by an associated and authenticated DEV to respond to the distribute key request command.

The ACK policy shall be set to request immediate acknowledgement. The Del-ACK request, SEC and repeater sub-fields in frame control field of the MAC header in this command shall be set to 0 and shall be ignored upon reception.

The distribute key response command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**1**
Command type	Length	ResultCode

Figure 34—Distribute key response command format

The ResultCode shall be set to 1 if the DEV correctly received the distribute key request command and shall be set to zero otherwise.

1.10.1.9 Deauthenticate request command

The deauthenticate request command is used by the PNC operating as the active security manager to revoke authentication of an authenticated and associated DEV. This command may be exchanged between the active PNC and an authenticated and associated DEV in the piconet.

The individual information elements used in this frame are described in 7.4. The deauthenticate request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	**2**	**1**
Command type	Length (=1)	ResultCode

Figure 35—Deauthenticate request command format

Upon the PNC as security manager sending the deauthenticate request command to an associated and authenticated DEV, the DEV shall disassociate from the piconet, which inherently deauthenticates the DEV. Further, the PNC as security manager shall required that the deauthenticated DEV re-associate and re-authenticate if the deauthenticated DEV is to rejoin the piconet.

Ed. note: Text for new command is below:

7.5.3.9 Deauthenticate request command

The deauthenticate request command is used by the PNC to revoke the authentication of an authenticated and associated DEV.

The deauthenticate request command frame structure shall be formatted as illustrated in Figure 18

octets: 2	2
Command type	Length (=1)

Figure 36—Deauthenticate request command format

Ed. note: Futher changes based on email exchange:

1) More the sentence "If there is no response from the PNC within AuthenticateFailureTimeout, the Reason-Code shall be set to TBD." to the MLME-AUTHENTICATE.confirm primitive "When generated" and replace TBD with TIMEOUT. Add another sentence that says "Otherwise, the ReasonCode is the value that was returned in the authentication response command, 7.5.3.2"

Ed. action: New text follows:

6.3.6.4.1When generated

When an MLME receives an authentication confirmation message from the PNC. If there is no response from the PNC within AuthenticateFailureTimeout, the ReasonCode shall be set to TIMEOUT. Otherwise, the ReasonCode is the value that was returned in the authentication response command, 7.5.3.2.

2) Update the ReasonCode definition in Table 14 to be:.

Table 6—

Name	Type	Valid Range	Description
ReasonCode	Enumeration	SUCCESS, FAILURE, TIMEOUT	The result of the authentication command

Ed. action: Changed as indicated.

3) Delete the command "Distribute key response command," 7.5.3.8.

Ed action: Command deleted and removed from cross reference table.

4) Delete MLME-DISTRIBUTE-KEY.response primitive

Ed. action: Command deleted and removed from cross reference table. Also edited the “When generated” for MLME-DISTRIBUTE-KEY.confirm primitive, text is below:

6.3.9.3.1 When generated

When the MAC receives an ACK for a directed distribute key request command or when a broadcast distributed key request command is sent, it generates this message and sends it to the DME.

1.11 Items in 01/374r12, entered into D09 and noted in the minutes.

Ed. action: Changes already made at Austin meeting, notes on changes are included in 01/374r12.

1.12 Change the MLME commands to reflect the frame formats and information described in clause 7 and 8.

Ed. action: Changes made by Allen Heberling.

1.13 Change backoff algorithm to use PHY dependent parameters rather than numeric times. Use 802.11 as a model to write this.

Ed. action: New text for 8.4.2:

Except when transmitting an Imm-ACK, the following backoff procedure is performed when sending frames during the CAP.

The backoff algorithm uses the following information:

- retry_count: An integer that takes on values in the range [0,3].
- backoff_window(retry_count): A table which has values [7, 15, 31, 63]
- aBackoffSlot: A PHY dependent parameter that is based on the amount of time it takes to sense the channel. For the 2.4 GHz PHY, this is defined in 11.2.6.1
- bw_random(retry_count): A random integer drawn from a uniform distribution over the interval [0,backoff_window(retry_count)]. The method for choosing the random integer should be unique for each DEV and use the random number generator resident on the DEV. If the DEV does not possess a random number source, the random integer should be generated using its unique 48-bit device ID (and any other information that the implementer wishes to use) and a pseudo-random number generator (PRNG) such as MGF1 as defined in IEEE Std 1363-2000. Note that the current state of the PRNG should be maintained and subsequent backoffs should use subsequent bits in the pseudo-random sequence.
- bw_random(retry_count): A pseudorandom integer drawn from a uniform distribution over the interval [0,backoff_window(retry_count)]. It is important that designers recognize the need for statistical independence among the random number streams among DEVs.

The backoff time in the CAP is measured at the air interface and indicates when a DEV may begin transmitting data. The DEV first waits aRIFSTime from when the medium is determined to be idle before beginning the backoff algorithm.

The DEV shall then choose backoff_count = bw_random(retry_count) and shall maintain a counter for backoff_count which is decremented only when the medium is idle for the entire duration of aBackoffSlot. The retry_count shall be set to 0 for the first transmission attempt of a frame. Whenever the channel is busy, the backoff counter is suspended. This avoids the problem of unfair channel access when a backoff counter of a DEV ending in the middle of a reception and hence resulting in larger backoff for that DEV while another DEV starting after the current reception choosing smaller value for backoff. When the backoff counter reaches zero, the DEV shall transmit its frame.

The backoff counter shall also be suspended outside of the CAP duration. Note that the backoff counter is maintained across superframes and is not reset with each beacon.

When a directed frame is transmitted and the expected ACK is not correctly received by the DEV, the retry_count shall be incremented but shall not be set to more than 3. The backoff_count shall then be set to bw_random(retry_count). If the maximum number of retries for that frame has not been exceeded, the backoff procedure is again resumed. The DEV shall attempt transmission of a frame for a maximum of aMaxRetransmissionLimit times before the failure in frame transmission is reported through MLME interface via the MA-UNITDATA-STATUS.indication primitive, A.2.3.

1.14 Update neighbor piconet information to reflect changes in 01/481r4

Ed. note: 01/481r3 was included in D08. The changes from r3 to r4 of the document were:

Rev 40

1) Removed the neighbor association request command (see rev 1) and added text to the (normal association request format to accommodate the neighbor association.
2) Removed the neighbor channel time request from figure def.
3) Modified table 63 to remove the neighbor association request and the neighbor channel time request.

Ed. action: Removed the extra commands as noted with new text for the association request. Note that the text is changed based on item 16 below. Made changes to 8.2.6 in the first paragraph to reflect the new usage of the association request command. Actual text is below:

8.2.6 Neighbor piconet

If after following the scan procedure in 8.2.1, no channels are available, then a neighbor alternate coordinator (i.e. an AC from a different system), may attempt to start a neighbor piconet within an existing piconet. To start a neighbor piconet, the neighbor AC shall send an association request, defined in 7.5.2.1, using the association address, 7.2.3, as the source address of the frame. The neighbor PNC bit in the capability field shall be set as indicated in 7.4.3 when the association request command is sent.

1.15 Add mapping of supported data rates from 5 bits to 8 bits by adding 3 binary 0's as the MSB.

Ed. action: Added the following text to the PHY clause, last paragraph of the clause:

The encoding of the supported PHY data rates into an octet number is accomplished by adding bits b5-b7, all set to zero, to the encoding given in Table 102. Bit b0 is the lsb while bit 7 is the msb. Thus a DEV that supports 11, 22 and 33 Mb/s would have a capability information element 01000 (lsb to msb) and an octet encoding of 0x2.

1.16 Move supported data rates field in figure 19, 7.4.3, to be bits b0-b4. Add bit b10 to be neighbor piconet bit. Text for neighbor piconet bit is: "The neighbor piconet bit shall be set to 1 if the DEV is intending to be a neighbor PNC in the current piconet and shall be set to 0 otherwise." Change "is set to" to be "shall set to" in 7.4.3. Change 01/481r4 to use neighbor piconet bit instead of 0 capability field to identify and neighbor association request.

Ed. action: Changed as indicated, figure and text follows:

bits: b0-b5	b6-b9	b10	b11	b12	b13	b14	b15
Supported data rates	Reserved	Neighbor PNC	PSAVE	PSRC	SEC	PNC-Des-Mode	AC

Figure 37—Capability field format

The supported data rates element is a PHY dependent mapping of the optional data rates to a 5 bit field that indicates which of the optional data rates are supported by a DEV. For the 2.4 GHz PHY, this mapping is defined in Table 94.

The neighbor piconet bit shall be set to 1 if the DEV is intends to be a neighbor PNC, 8.2.6, in the current piconet and shall be set to 0 otherwise.

The PSAVE bit shall be set to 1 if if the DEV is planning to use sleep state as a part of power management. Otherwise the PS bit shall be set to 0. The PNC shall always set this bit to 0 in its capability field.

The PSRC field shall be set to 1 if the DEV is receiving power from the AC (alternating current) mains and shall be set to 0 otherwise.

The SEC bit shall be set to 1 if the DEV is capable of supporting encryption for its data streams. Otherwise SEC bit shall be set to 0.

The PNC-Des-Mode is the designated mode of the DEV as currently set. This bit shall be set to 1 if the DEV is in the PNC mode. Otherwise this bit shall be set to 0.

The AC bit shall be set to 1 if the DEV is capable of being a PNC in the piconet. Otherwise AC bit shall be set to 0.

1.17 Change "is shown in figure xx" to be "shall be formatted as illustrated in figure xx" for frame formats, information elements, command types and field format figures in clause 7.

Ed. action: Changed as indicated in clause 7 for all frame format figures.

2. Editorial changes from 01/374r14

Selected editorial issues from 01/374r14

Issue 11, WMS: It would be nice to have a list of primatives and a short description

Suggestion: Please add the appropriate definitions.

Resolution: R. Roberts will add an MLME primitives table similar to the one for the PHY-SAP. Also add MAC-SAP and PLME-SAP tables

Ed. action: The table for the MLME primitives is up to date. Added summary table for PLME primitives. Did not add table for MAC CPS SAP since there are only two commands. Updated the PHY SAP table so it is current and added cross references to the sub-clause in which they are defined.

Issue 334, ADH: Figure 49 Superframe structure does not include guardtime

Suggestion: Please add Guardtimes between GTSs in diagram.

Resolution: Update figure.

Ed. action: New figure added, to D09, shown below:

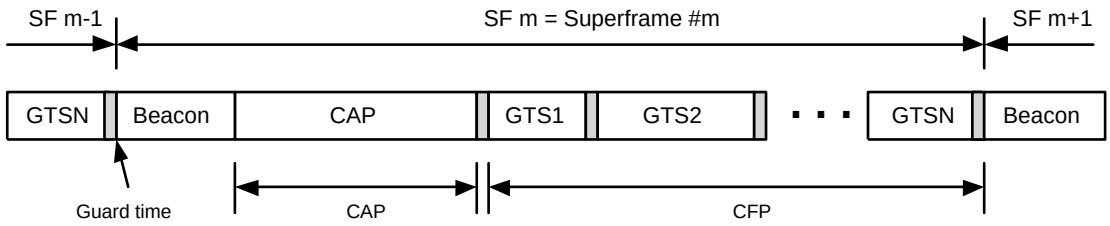


Figure 38—Superframe structure

Issue: 335, ADH: There is need, in this clause, for an IFS timing relationship diagram with parameters. This diagram shall be similar to the one illustrated on page 85 of ISO/IEC 8802-11:1999(E)

Suggestion: Please include the requested diagram.

Resolution: Allen will drow up in Visio

Ed. action: New figure added t

Issue 377, ADH: These parameters are missing from the MLME-ASSOCIATE.confirm primitive: Device-ID, AllocatedDeviceAddress, ATP. Also change ResultCode to ReasonCode

Suggestion: Please add the new parameters and change ResultCode to ReasonCode. The order of these parameters shall be Device-ID, ReasonCode, AD-AD, ATP.

Resolution:

Ed action: Items added, except for Device-ID, since the device knows its on ID. New primitive description follows:

```
MLME-ASSOCIATE.confirm      (
                            AssocDEVAddress,
                            AssociationTimeOutPeriod,
                            ReasonCode
                            )
```

Issue 378, ADH: Table 16 is missing these parameters: Device-ID, AllocatedDeviceAddress, ATP. Also change ResultCode to ReasonCode

Suggestion: Add the missing parameters to the table and these definitions for the various fields in the table: ReasonCode; AllocatedDeviceAddress Type: 1 Octet, Valid Range: 0x00-0xFF, Description: The address allocated to the requesting device by the PNC; Association Timeout Period(ATP) Type: 2 Octets, Description: ATP that the PNC supports.

Resolution:

Ed action: The items have been added and result code changed to reason code. New table appears as Table 8 in this document.

Issue 379, ADH: Clause 6.3.6.2 describes the MLME primitive that is generated when an assoc.rsp message is received by a DEV whose Device-ID is specified in the parm field of assoc.rsp. While, Clause 8.2.5, p98, line16-17, describes how assoc.rsp is a broadcast frametype which DEVs in the piconet may receive and use the info contained there in to add to their association table a newly associated DEV's Device-ID/AD-AD. The issue is there currently is not defined an MLME primitive that will pass the appropriate information from the Assoc.rsp upto the SME. Consequently, there is need for an MLME-xxxx.indication containing these parameters: Device-ID, AD-AD

Suggestion: Please add MLME-xxxx.indication(Device-ID, AD-AD). Also add a parameter table for this primitive describing Device-ID and AD-AD)

Resolution: Add MLME-Assoc-MAP.indication with DevInfo Table as the entry.

Ed action: This is now the MLME-ASSOCIATION-RESPONSE.indication element in D09. Text follows:

6.3.4.5 MLME-ASSOCIATION-RESPONSE.indication

This primitive is used to indicate to an associated DEV the reception of a broadcast ASSOCIATION-RESPONSE command. The semantics of this primitive are as follows:

```
MLME-ASSOCIATION-RESPONSE.indication
                            (
                            ReasonCode,
                            DeviceID
                            AssocDEVAddress,
                            )
```

Issue 380, ADH: MLME-ASSOCIATE.indication is missing these parameters: Capability info element, AssociationTimeoutPeriod(ATP)

Suggestion: Please add the missing parameters.

Resolution: Add indicated parameters in D09

Table 7—MLME-ASSOCIATION-RESPONSE.indication primitive parameters

Name	Type	Valid Range	Description
ReasonCode	Integer	As defined in 7.5.2.2	Indicates the response of the PNC to MLME-ASSOCIATE.request
DeviceID	MAC address	Any valid MAC address	The device ID of the DEV that has been associated.
AssocDEVAddress	Octet	1-255	Either the allocated device address for successful association or the association address for unsuccessful association, 7.5.2.2

Ed action: Parameters added in D09. Changed aAssocRespConfirmTime to be AssociationTimeOutPeriod in the .request and .indication since that is what is passed in the frame formats. aAssocRespConfirmTime is a constant and so it does not need to be passed.

Issue 381, ADH: MLME-ASSOCIATE.indication table is missing these parameters: Capability info element, AssociationTimeoutPeriod(ATP)

Suggestion: Please add missing parameters to table and reference appropriate sections of document

Resolution: Add indicated parameters in D09

Ed action: Table has been updated, The table is as follows:.

Table 8—MLME-ASSOCIATE primitive parameters (partial)

Name	Type	Valid Range	Description
DeviceID	MACAddress	Any valid individual MAC address	Specifies the MAC address of the DEV that is requesting association with the PNC.
CapabilityInformation	As defined 7.4.3	As defined in 7.4.3	Specifies the operational capability definitions to be used by the MAC entity
AssociationTimeOutPeriod	Integer	≥ 1	As defined in 7.5.2.1
AssocDEVAddress	Octet	1-255	Either the allocated device address for successful association or the association address for unsuccessful association, 7.5.2.2
ReasonCode	Octet	As defined in 7.5.2.2	Indicates the result of the association request

Issue 387, ADH: MLME-CHANNEL-TIME.request/indication/response/confirm are not defined.

Suggestion: Please define the indicated MLME primitives.

Resolution:

Ed action: Primitives now defined in D09, too big to include here.

Issue 389, ADH :MLME-COORDINATOR-HANDOVER.request/indication/response/confirm are not defined

Suggestion: Please define the indicated MLME primitives.

Resolution:

Ed action: None yet, issue remains open

Issue 391, ADH: MLME-REPEATER.request/indication/response/confirm are not defined.

Suggestion: Please define the indicated MLME primitives.

Resolution:

Ed action: None yet, issue remains open

Issue 392, ADH: MLME-TxPOWER-CONTROL.request/indication/response/confirm are not defined.

Suggestion: Please define the indicated MLME primitives.

Resolution: Check with Rick and Raju.

Ed action: None yet, issue remains open

Issue: 395, WMS: We probably need an MLME request and MLME indication primitive for PNC handover.

Suggestion: doc 01410

Resolution:

Ed action: Duplicate of 389.

Issue 411, WMS: Information Element lsb/msb not clear

Suggestion: Add lsb and msb indicators to each field in each

Resolution: Heberling to Add generic figure to 7.1.1 for lsb/msb. James to add text and change all figures to be lsb on left (e.g.Table 61)

Ed action: Still TBD, will submit new figure in letter ballot.

Issue 428, RDR: frame position is referenced in text but is not shown if figure 9

Suggestion: remove from text or add to figure

Resolution

Ed action: Good catch, frame position no longer exists. I have purged all references to it in the draft.

Issue 431, RDR: in all the figures of section 7.4 the words "Element ID" is explicitly used.

Suggestion: Replace the element ID with the number value of the ID

Resolution:

Ed action: Since the sections and number have been a moving target, the element ID numbers are not in the individual descriptions, but rather in the summary table. This is the easiest place to define them and they can only be defined in one place. Thus I am rejecting this comment, but suggest that if RDR desires, he may resubmit this in letter ballot.

Issue: 432, RDR: The letters AC (indicating alternating current) clash with the use of AC in figure 19 (indicating alternate coordinator)

Suggestion: remove the AC in line 49 and just use the phrase alternating current

Resolution:

Ed action: Changed as indicated in D09.

Issue: 433, RDR: frame position is used but I can't find where it is define

Suggestion: define or eliminate the words "frame position"

Resolution:

Ed action: Excellent catch, frame position no longer exists. I have deleted all occurances of this from D09.

Issue 434, RDR: The paragraph starting at line 6 with "The TX power …" is specific for the 2.4 GHz PHY.

Suggestion: Rewrite the text so it is generic with reference to 2.4 GHz PHY as an example.

Resolution:

Ed action: This issue was discussed and no alternative definition was proposed. While it is possible to write it as “PHY dependent, for the 2.4 GHz PHY this is defined in x.x.x”, -127 dBm to 127 dBm should be sufficient for any system. Alternate PHYs should be able to map into this. Comment is rejected.

Issue 435, RDR: the word "required" too strong???

Suggestion: replace "required" with "desired"???

Resolution:

Ed action: The new CTRB has deleted this definition. None of the new definitions use the word required.

Issue 436, RDR: Word "require

Suggestion: replace it with "not allowed"

Resolution:

Ed action: Changed sentence to read: “Therefore, an ACK is not allowed for the association response command.”

Issue 429, RDR: Why doesn't the PNC immediately utilize this information.

Suggestion: Clarify.

Ed action: D09 power managment text was heavily re-written. The quoted sentence no longer appears in the draft.

Issue: 430, RDR: A new term "EPSPhase" is introduced with no explaination of the term.

Suggestion: Add text to clarify.

Resolution:

Ed action: EPSPhase has been replaced by EPSNext and EPSNext has been defined and described.

3. Editorial changes

— 7.4.12 MAX CTAs element, reformat to match the previous sub-clauses in 7.4 and move the functional description to clause 8.

Ed. action: Changed as indicated, new format follows:

The max CTAs element shall be formatted as illustrated in Figure 39.

octets: 1	1	2	2
Element ID	Length (=4)	MaxAssignedCTAs	MaxProcessedCTAs

Figure 39—Max CTAs information element

The MaxAssignedCTAs field describes the maximum number of GTS slots that may be assigned to a DEV when the DEV is either the source or destination. The destination address may include group or multicast destinations.

The MaxProcessedCTAs field specifies the maximum number of CTAs that the DEV is able to process.

— Check the PNC handover process to make sure that we state that the current PNC checks the DEV-info table to find the most qualified AC to become the new PNC.

Ed. action: Text now says:

When a station joins a piconet, the coordinator shall compare the capabilities field of the new station to its own.

— 7.3.1 Add xref to the channel change element and the CTA element in table 58 verify the use of the correct terminology. Also, change the Device ID entry to be device identifier element with description “IEEE 802 address of the PNC, xref 7.4.1” Change to piconet synchronization element and xref in notes.
— 7.3.1 Add text that says that the beacon elements can occur in any order and that DEVs shall ingnore information elements other than the ones defined.
— 7.4.6 Add pad byte for word alignment and text for how to handle the pad byte.
— 7.4.3 Fix the TBD in the PHY table mapping by adding the appropriate section to the PHY and the xref to it in 7.4.3.
— 7.4.2 Fix length parameter for piconet sychronization parameter.
— 7.3.3 and 7.3.4, define the SA and DA for the command and data frames.
— 7.5.1, 7.5.2, 7.5.3, Make notes that the PNC selection commands, and the association commands are never sent with other commands. Or perhaps better is to say that a command can only be piggy-backed with commands that are sent with the same SA and DA pair.
— 7.2.1.9 Note that the beacon shall not have the SEC bit set (perhaps this is true of the command frames?).
— 7.5.2 ATP definitions disagrees with 8.2.5 definition of ATP. Did we fix this since Schaumburg.

Ed. action: ATP is now consistent in the document, changes were made in Austin and all issues related to this were closed.

— 7.5.17 (D06) add xref to stream identifier element to the appropriate information element (stream ID)
— Page 104, change aa to a

Ed. action: Fixed in D09

- 7.5.21 Fix QoS paramters length in table 49, fix table sizing for table 51
- 7.5.21 Need to get a good definition for Max Burst Size
- Figure 27, change length of the individual elements to be 7 instead of 6.
- 8.8 The acronyms TPC and DCS are mentioned, but are not defined. It should be DFS rather than DCS. Perhaps delete this sentence.

Ed. action: TPC was added to the acronym list in D08, DCS was added in D09 as “dynamic channel selection.

- Change format of annex A CS-SAP parameters to match the format in the layer management clause.
- MA-UNITDATA.request supports 3 types of ServiceClass:
  for asynchronous: reorderable and strictly ordered
  for stream based: only reorderable multicast.
  However, the MA-UNITDATA.indication indicates only two possible classes:
  reorderable and strictly ordered.
  Shouldn't reorderable multicast be an option?
- What about adding "channel sucks right now" to the MA-UNITDATA-STATUS.indication as a potential error code? (Actually, we would add something like "undeliverable, possibly due to poor channel conditions")
- Change kilo micro seconds (Kμs) to ms throughout and delete the definition from the Definition clause.

Ed. action: changed as indicated.

4. TBD’s and editorial notes

4.1 Front matter and General Description

4.2 References

4.3 Definitions

4.4 Acronyms

4.5 Overview

4.6 Layer management

Email from Bill Shvodian

MAC PIB PNC group parameters:

MACPIBCFPDuration (not Period) , 2 octets, Duration of the CFP, dynamic
MACPIBCFPMaxDuration 2 octets, Duration of the CFP, dynamic
MACPIBBeaconPeriod, 2 octets, Period of the superframe, dynamic
MACPIBPNCCapable, 1 bit, =1 if the DEV has the capability to become PNC, static
MACPIBPNCDesMode, 1 bit, =1 if the DEV is designated by the upper layers to be the PNC, dynamic

MAC PIB characteristic group

MACPIBDeviceID, 6 , Unique IEEE (was "Any valid") MAC address of the DEV, Static
MACPIBPowerManagementMode (Jay and Mark)
MACPIBNumMaxStreams, 1, Maximum total number of streams that the DEV can handle, static

MAC PIB Authentication group parameters - see Gregg Rasor

Not sure why these two are PIBs. They are MLME-DISASSOCIATE.indication parameters

MACPIBDisassociateReason
MACPIBDisassociateDevice

Ed. action: tables updated, given below:..

Table 9—MAC PIB PNC group parameters

Managed Object	Number of octets	Definition	Type
MACPIBCFPDuration	2	Duration of the CFP	Dynamic
MACPIBCFPMaxDuration	2	Maximum duration of the CFP	Dynamic
MACPIBBeaconPeriod	2	Period of the superframe	Dynamic
MACPIBPNCCapable	1 bit	1 if the DEV has the capability to become the PNC, 0 otherwise	Static
MACPIBPNCDesMode	1 bit	1 if it is desired that the DEV be the PNC	Dynamic

Table 11—MAC PIB authentication group parameters

Managed Object	Number of octets	Octets Definition	Type
MACPIBAuthenticationResponseTimeOut	2	Time in ms for the authentication procedure to be completed.	Dynamic
MACPIBPrivacyOptionImplemented	1	0x01 = privacy implemented 0x00= privacy not implemented	Static
MACPIBAuthenticateReasonCode	1	0x00 = authenticated 0x01 = failed 0x02 = timed out	Dynamic
MACPIBAuthenticateFailDevice	6	MAC address of the last device to authenticate	Dynamic
MACPIBDeauthenticateDevice	6	MAC address of the last device to deauthenticate	Dynamic

Table 12—MAC PIB association group parameters

Managed Object	Number of octets	Definition	Type
MACPIBAssociationResponseTimeOut	2	The length of time in ms to wait for the association response command.	Static

Table 10—MAC PIB characteristic group parameters

Managed Object	Number of octets	Definition	Type
MACPIBDeviceID	6	Any valid MAC address	Static
MACPIBPowerManagementMode	1	The current power management mode of the DEV. 0x00 = PM_OFF 0x01 = RPS 0x02 = EPS	Dynamic
MACPIBNumMaxStreams	1	Maximum total number of streams that the DEV can handle	Static
MACPIBMaxAssignedCTAs	2	Maximum number of CTAs that the PNC is able to assign to the DEV.	Static
MACPIBMaxProcessedCTAs	2	Maximum number of CTAs that the DEV is able to process	Static
MACPIBMaxAllocatableGTS	1	Maximum number of time slots that the DEV is able to handle in one superframe.	Static
MACPIBPowerSource	1	0x00 for mains power 0x01 for battery power	Dynamic

6.7.3 Security services is TBD

Ed. action: Section deleted, services are provided by a higher layer.

4.7 Frame formats

4.8 MAC functional description

Ed. note: This section is based on an email from Ari Singer.

Below are all the TBDs that I found in section 8 and comments when appropriate. I will put my own comments in [] if there are issues that I think may be controversial. If these are all you found as well, I will forward this along to the main list.

1) Two TBDs at the end of section 8.2.4. Recommend replacing final 2 sentences with:

"If the PNC-Des-Mode bit is set in the new station and not in the current PNC, the old PNC shall perform PNC handover. If the new station is more qualified to be the coordinator, the coordinator may perform PNC handover. If security controlled by the old PNC is in use during PNC handover, the established security parameters are invalidated and all devices must be authenticated by the new PNC in order to re-establish security in the piconet. Therefore, if re-authentication is not desirable, a PNC running security in the piconet should not perform PNC handover unless it is leaving the piconet."

[I believe this matches the concensus of the security group and is the simplest solution. In the two-tier model, this is the only solution. In the three-tier model, most of the members felt that the changing of the piconet membership was sufficient reason to change keys. In any case, it seems unconventional to pass authentication relationships from one device to another. If the devices remain the same, I could imagine passing over KEKs or something, but I don't think that is a good idea.]

Ed. action: Add reference to the PNC selection table in the second sentence. Change words to be more formal. New text follows:

If the PNC-Des-Mode bit is set in the new station and not in the current PNC, the old PNC shall perform PNC handover. If the new station is more qualified to be the coordinator, based on the PNC selection criteria in Table 68, the coordinator may perform PNC handover. If security controlled by the old PNC is in use during PNC handover, the established security parameters shall be invalidated and all devices shall authenticate with the new PNC in order to re-establish security in the piconet. Therefore, if re-authentication is not desirable, a PNC running security in the piconet should not perform PNC handover unless it is leaving the piconet.

2) TBD in 8.14.2 "The other DEV shall increase or decrease its transmit power level as indicated in the TBD command if the power level setting . . . " Not my area, so I don't have any suggestions. Sorry . . .

Ed. action: Changed transmit power control from an information element to a command called transmit power change. Changed TBD to reference the transmit power change command.

3) TBD in 8.14.2 "2) DEV-1 sends a TBD command with a TPC element to the DEV-2 with the requested TX power level change." No helpful comment.

Ed. action: Changed TBD to reference the transmit power change command.

4) TBD in table in 8.15 "Authentication request <TBD>- ACK" Recommend removing this row. The authentication request occurs (I believe) in an allocated slot by an associated DEV, not in the CAP.

Ed. action: Row deleted in the table. It can be added back in if we get different information later.

5) Editor's note in 8.13.3.4 "PS Editors note: make sure that Mark's Allocation Interval and not the older Duration Between Slots is correct in clause 7. . . " No helpful comment.

Ed. action: I have verified that the allocation interval is correct in clause 7 and have removed this editorial comment.

Also in there but missed by Ari

8.6.1, Editor’s note:

note: the figure number on next line needs to be fixed. It is not showing the figure number and I was not successful in changing it.

Ed action: I fixed the crossreference and deleted the comment.

8.8.5, Editor’s note:

Ed note: The draft now has interframe packet timing scattered throughout this clause, possibly being defined more than once. All of the interframe timing requirements need to be reviewed so that each requirement is defined in only one place.

Ed. action: Delete note, review will happen in letter ballot.

8.12, generic note:

Fix this silly picture, make it an MSC

Ed. action: remove the note, add the MSC as a part of letter ballot phase.

4.9 Quality of Service

Ed. note: The following is based on comments from Mark Schrader on clauses 9-Annex D.

p 161, lines 5-6: The use of the word "association"in the context of QoS is confusing.

A term should be used that makes a it morre like an ordered pair.

Ed action: New text: A procedure during which pre-configured default service flows and their corresponding QoS parameters are applied.

page 161, lines 7: How can the PNC use "min channel time" and "burst size"?

The use of "max channel time" is well defined in the standard, but the others should be defined as available for use but not convered in the standard.

Ed. action: OK, but you need to propose a specific change. In any event, this is technical, so resubmit during letter ballot

page 161, line 29: SFID length is TBD

16 bits

Ed. action: Changed as indicated.

page 161, line 31: Stream Index length is TBD

16 bits

Ed. action: Changed as indicated.

4.10 Security

page 163 Missing Text, Insert Text

Ed. action: Waiting for security text.

4.11 2.4 GHz PHY

11.2.3, page 166, lines 20-39: How does one select what channelization to use? If by the PNC then we need to add mechanism. If not, we still need to add some comment about how they are supported in the PHY.

There should be a "may" or a "shall" for whether a device must support both channelizations and how both shall be supported if the device includes both channelizations. How do you select?

Ed action: The PNC chooses the channel based on decisions by the DME and possibly higher layers. (check out the scan and start commands) The PHY is required to support all 5 channels. No change to the text.

11.2.6.1, page 167, lines 2-4: The last column of table 62 should be lenghened- "Defintion" is broken

see previous column

Ed. action: Both tables (now 86 and 87) have been re-formatted.

11.2.8, page 167, line 44: "Note: More details and an example calculation will be added later."

Add from 802.11b-1999: page 19, Figure 130

Ed. action: Add the following sentence and entry to the bibliography:

This CRC is the same one used in IEEE Std 802.11b-1999. An example implementation and a calculation example are available in 18.2.3.6 of [B15].

[B2] IEEE Std 802.11b-1999, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band

11.4.1, page 179, line 2: Since SB appears in the figure, perhaps (SB) should be added to the text after "stuff bits". The definition of stuff bit could be made more explicit and less implied.

Elaborate slightly and add (SB)

Ed. action: Changed as indicated.

11.4.3, page 180, line 36: Figure 94 not correct

Figure 95

Ed. action: Changed cross reference.

11.4.4, page 181, line 11: Ed note: Add location where s_n is xor-ed with the x_n in the data stream.

Add according to IEEE 802.11b, page 21, Figure 131.

Ed action: Added the change to the figure, reviewed by Jeyhan Karaoguz, included below:

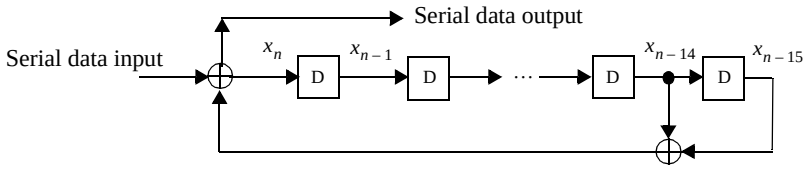


Figure 40—Realization of side-stream scrambler by linear feedback shift registers

11.4.4, page 181, line 11: Missing Descrambler

Add according to IEEE 802.11b, page 21, Figure 132.

Ed action: None, please submit comment as a part of letter ballot.

11.4.4, page 181, line 44: "burst" is unclear

use the definition of burst instead of the word burst

Ed. action: Change burst to PHY frame.

11.4.4, page 82, line 4: Resolve Ed note

Resolve Ed note

Ed. action: Wording has been verified, the editorial note has been removed.

11.4.6, page 182, line 54: change "stuff bits"

stuff bits (SB)

Ed. action: None, I don't want to use another acronym here. Stuff bits are only used in a couple of places, it is OK to just spell it out wherever it is used.

11.4.6, page 183, line 4: change "A compliant PHY shall less than …symbol."

to: A compliant PHY shall insert the number of stuff bits required to make the number of MPDU bits an integer number of symbols. The minimum MPDU size is equal to the size of one symbol.

Ed. action: Changed "A compliant PHY shall less than the" to "A compliant PHY shall add less than the". The smallest MPDU is actually 12 bytes long.

11.6.1, page 190, lines 28-29: PN23 sequence as defined in TBD

X**23 + X**5 + 1, prime Galois polynomial

Ed. action: changed to: "... a PN23 sequence as defined by $x_{n+1} = x_n^{23} + x_n^5 + 1$."

11.7.2.4, page 193, lines 1-17: Table 104: Expand first column of the table slightly to eliminate word wrap

Table 104: Expand first column of the table slightly to eliminate word wrap

Ed. action: changed as indicated.

4.12 Annex A

A.1, page 195, lines 17-46: The PCS, is not shown in Figure A.1 and it should be there consistent with the text lines 44-46.

Add PCS to figure A.1. Particularly when there is a CPS and a PCS, their relationship in the text and the diagram should be consistent and clear.

Ed. action: The figure is intended to be generic, the PCS is one of the many possible SSCS. No change, resubmit in letter ballot.

A.1, page 198, lines 42-44: Why is RoutingInformation specified at all if "For IEEE 802.15.3 this field shall be set to null"

Even for an annex, isn't specifying an always null parameter unnecessary.

Ed. action: This parameter is included for compatibility with the higher layers. No change made.

A.1, page 199, line 4: Doesn't the indication also support reorderable multicast as does the request?

Add reorderable multicast if appropriate.

Ed. action: Technical change, please submit in letter ballot.

A.1, page 200, lines 23-48: Best Effort does not apply to 802.15.3

We should change the description, so that it matches the expectations of the 802.15.3 access method. Or change Best Effort to Don't Care.

Ed. action: This parameter is included for compatibility with the higher layers. No change made.

A.2.1, page 197, line 41: Eliminate <TBD>

change to aMaxTransferUnitSize.

Ed. action: Changed as indicated.

4.13 Annex B

B.2, page 203, line 27: Remaining text TBD

Provide text.

Ed. action: Note removed. Additional text will be added if it is provided.

4.14 Annex C

C.0, page 205, lines 1-54: Missing Text

Provide text.

Ed. action: Will add text when it is available.

4.15 Annex D

D.0, 207, lines 1-54: Missing Text

Provide text.

Ed. action: Will add text when it is available.

5. Security changes

The following is based on an email from Gregg Rasor

7.2.1.9 - Replace second sentence with: "When the SEC bit is set, the frame body shall be encrypted using the currently assigned data encryption key for the piconet." (from 01/530r3)

Ed. action: New section below:

7.2.1.9 SEC field

The SEC field is one bit in length. When the SEC bit is set to 1, the frame body is encrypted using the currently assigned data encryption key for the piconet.

7.3.4 - Remove the second field from the figure - Encryption information. Remove first sentence following figure 13. Remove "-(length of encryption field)" from the next sentence.

Ed. note: Changed as indicated, new section and figure are as follows:

7.3.4 Data frame format

The frame format of data frame is as shown in Figure 41.

octets: 12	variable	0 or 4
MAC frame header	Data	FCS

Figure 41—Data frame format

The length of the data field is 0 to aMaxFrameSize-4, inclusive.

7.4.2 - Rename Reserved octet in Figure 16 to be "Authentication Mode" (Instead of Authentication Version which is in 01/530r3. I don't think 'Version' is the right term to use. 'Mode' reflects what these bits will determine.) After Figure 17 add the following text: "Authentication Mode determines what type of authentication policy is used for this piconet. The following table defines the possible modes: 0 - none required, 1 - authentication required, 2 - authentication and data encryption required."

Ed. action: The reserved octet became the key index, so instead I used 2 bits in the CAP mode octet and renamed it the piconet mode field. The additional text is below:

The piconet mode indicates 2 types of information: 1) what type of information is allowed to be sent in the CAP of the current superframe and 2) the current security requirements. The encoding of this octet shall be formatted as illustrated in Figure 42.

bits: b0	b1	b2	b3	b4-b5	b6-b7
Data	Commands (except association)	Association commands	EPS	SEC mode	Reserved

Figure 42—Piconet mode field

If a bit is set for data, commands, association or EPS, i.e. its value is 1, then that type of data or command is allowed to be sent in the CAP of the current superframe. Otherwise, that type of frame is not allowed to be sent in the CAP. The use of this command is described in 8.4.2.

The SEC mode indicates the current security settings in the piconet. The field is encoded as follows (b5, the msb, is listed first, b4, the lsb, last):

0b00: neither authentication nor data encryption are required.

0b01: authentication is required.

0b11: both authentication and data encryption are required

Note that 0b10, i.e. having the data encryption but not authentication, is a disallowed state.

7.4.7 - (Editor - Add an element ID for security parameters) Add the following text: "This element, illustrated in Figure XX, shall be used to communicate the security parameters used in a piconet. The OID field is defined by IEEE P1363: Standard Specifications for Public-Key Cryptography. Authentication and data encryption operations in this piconet shall use the cryptography algorithms defined by this OID specification." Add figure showing "Element ID" (1 octet), "OID length" (1 octet), "OID (cipher suite selector)" (Variable)

Ed. action: New information element added, actual text and figure are below:

7.4.7 Security parameters

The security parameters element is used to communicate the security parameters used in a piconet.The security parameters element shall be formatted as illustrated in Figure 26.

octets: 1	**1**	**2**	**variable**
Element ID	Length (=2+variable)	OID length	OID

Figure 43—Security parameterts element

The OID length is the length in octets of the OID field.

The OID field specifies a cipher suite according to IEEE P1363: Standard Specifications for Public-Key Cryptography.

7.5 - Add the following command types to Table 66:

- c1 - Authenticate Request
- c2 - Authenticate Response
- c3 - Authentication Challenge Request
- c4 - Authentication Challenge Response
- c5 - Distribute Key Encryption Key
- c6 - Distribute Data Encryption Key
- c7 - Data Encryption Key Request
- c8 - Data Encryption Key Response (This could be a directed Distribute Data Encryption Key command)

Ed. action: New commands added, including the deauthentication request command.

7.5.4 - Page 89, Line 46: Remove "Dev authentication failed" as a reason code. Associate will not require authentication. It is a separate action.

Ed. action: I moved this reason code to the disassociate command since a DEV shall be disassociated if it fails to authenticate.

Remove line 52.

Ed. action: The challenge response description and its position in the frame format have been deleted.

Add the following commands to section 7.5 (7.5.cN corresponds to command ID)

7.5.c1 Authentication Request

The structure of the command is indicated in Figure c1-nn.

Only a DEV currently associated with a piconet with non-zero Authentication Mode shall send this command to the PNC. This command initiates authentication of the DEV with the PNC.

The ACK policy shall always be set to request immediate acknowledgement.

The frame position, frag-start, frag-end, retry, Del-ACK request, SEC and Repeater sub-fields in frame control field of the MAC header in this command shall be set to zeros and shall be ignored upon reception.

The DA shall always be set to all-zero address, meant to indicate the PNC address.

Figure c1-nn: Command Type (2 octets), Length (2 octets), PublicKeyLength (2 octets), DEVAuthenticationPublicKey (PublicKeyLength octets - Variable), KeySigned (2 octets), AuthoritySignatureLength (2 octets), AuthoritySignature (Variable), AuthenticationFailureTimeout (2 octets).

PublicKeyLength and AuthoritySignatureLength are set to the lengths of the variable length DEVAuthenticationPublicKey and AuthoritySignature fields.

DEVAuthenticationPublicKey is the public key provided to the DEV by a key management authority.

KeySigned indicates whether the DEVAuthenticationPublicKey is signed by a key management authority.

Authority signature is the key of the key management authority that signed the DEVAuthenticationPublicKey.

Ed. action: None, authentication command added based on another document that had the rest of the commands. See Figure 19 in section 1.10.

6. Other editorial submissions

Via email from Robert Huang.

Review of Clause 6 in D09pre1: (footer page numbers used for reference)

Page - line Comment

32 10 Text reads "The following primitives describe how a DEV ~~becomes~~ disassociates with a PNC and how the PNC disassociates a DEV from the piconet."

Ed. action: Changed as indicated.

35 29 There is a TBD for AuthenticateFailureTimeout. Discussion: Two cases the PNC responds in time of fails to respond. There the result code is SUCCESSFUL or FAILURE.

Ed. action: Suggest moving the text to the .confirm commands and using SUCCESS, FAILURE and TIMEOUT as ReasonCodes.

56 16 change 'inthe' to 'in the' (add space)

Ed. action: Changed as indicated.

57 24 Table 26 is cited. I believe there is no need (table 26 is not relevant).

Ed. action: Changed as indicated.

77 22 TBD appears. This sentence was to be deleted (but still appears): The definition of these managed objects, attributes, actions, and notifications, as well as their structure, are presented in Annex TBD.

Ed. action: Sentence deleted.

Email from Bill Shvodian

3.16 MAC management protocol data unit should be MAC command protocol data unit

Ed. action: Above changes made as indicated. Also deleted the definition for abstract symbolic notation one, since we are deleting it from the acronyms anyway.

3.21 "frame" should be "packet"

Ed. action: Must of already changed it, frame does not appear in that clause (other than as superframe).

Further communication, “packet” should be “frame” and we need a defintion for frame. So, change:

6.1 packet: Format of aggregated bits that are transmitted together in time.

to be

6.2 frame: Format of aggregated bits that are transmitted together in time.

and place alphabetically.

4.
Delete ARQ - not used
Delete ARQN - not used
Delete ASN.1 - not used
ISM is Industrial Scientific and Medical (not Medicine)
delete LM - not used
LSB should be lsb (bit not BYTE)
MSB should be msb
SME should be DME (global change, probably already done...)

Ed. action: Above changes made as indicated.

The next one is apparently in clause 5

Figure 1 - replace Asynchronous and Isochronous with MTS and GTS

Ed. action: Changed to MTS and GTS and added a few more. Hope it looks OK

The next one is from the Layer Managment clause.

Table 4 - Do a global change from KuS to ms

Ed. action: Tried to change them, however the search is a little flaky, so some might have been missed.

7. modify as follows:

In addition, every DEV shall be able to construct a subset of the command frames for transmission, and to decode another (potentially different) subset of the command frames upon validation following reception. The particular subsets of these commands that a DEV shall construct and decode are determined by the functional capabilities supported by that particular DEV, as declared by them in the capability information specified in 7.4.3.

Ed. action: We had just changed it, but now the authentication, key exchange, neighbor piconet, etc. mean that there are a slew of frames and data types that not every DEV will be required to construct. Both changes were suggested by Bill Shvodian, so the final one will take precedence. The capabilities field does not explicitly specify which frame formats correspond to which capabilities, so I left this cross reference out. The new text follows:

In addition, every DEV shall be able to construct a subset of the command frames for transmission, and to decode another (potentially different) subset of the command frames upon validation following reception. The particular subsets of these commands that a DEV shall construct and decode are determined by the functional capabilities supported by that particular DEV.

7.2.5 Para3, change "frame types other than data" to "commands"

Ed. action: But that leaves out the other frame types, i.e. association and the beacon. While these two probably don't need sequence numbers, we would need to define what to set the field to. Accept change, but need to add text that describes what to set the sequence number to for the ACK and beacon frames. New text for the section follows:

The sequence numbers for all command frames shall be assigned from a single modulo-65536 counter.

The sequence number for Imm-ACK frames and the beacon frame shall be set to 0.

7.4.8 Para3 "devices has" should be "devices have"

Ed. action: Changed to the following text:

The TX step size is a one octet field that is the TX power level step size in 1 dB resolution, e.g. a number 4 in this field means that the DEV has nominaly 4 dB steps.

7.4.10 do global search on xref placeholders

Ed. action: Did search and replaced with appropriate cross-references.

Figure 30 - Length should say Length =10

Ed. action: This command was deleted in D09.

8.2.5 first para: add the following at the end: or MTSs.

Ed. action: New text follows:

Before a DEV has completed the association process, all frames between the PNC and the DEV shall be exchanged either in the CAP of the superframe or in an association MTS.

The following is based on email exchanges with Ari Singer regarding defintions for security

6.3 authentication: The process of assuring that an entity is authorized to perform certain operations.

6.4 certificate authority: an entity that provides assurance that a particular public key is associated with additional information through the creation of a public-key certificate

6.5 certificate revokation: the process of invalidating a public-key certificate.

6.6 confidentiality: assurance that communicated data remains private to the parties for whom the data is intended.

6.7 data integrity: assurance that the data has not been modified from its original form.

6.8 digital signature: a data string generated with an entity's private key that is typically appended to data in order to provide data integrity and source authentication

6.9 key establishment: a public-key process by which two entities securely establish a symmetric key that is known only by the participating entities

6.10 key transport: a process by which an entity sends a key to another entity

6.11 identification: The process of assuring that an entity is who they claim to be.

6.12 message authentication code: a data string generated using a symmetric key that is typically appended to data in order to provide data integrity and source authentication similar to a digital signature.

6.13 mutual entity authentication: a process by which two entities authenticate each other

6.14 private key: the secret portion of a public-key pair that may be used for digital signature creation, data decryption or key establishment procedures depending on the type of key pair

6.15 pseudo-random number generation: the process of generating a deterministic sequence of bits from a given seed that has the statistical properties of a random sequence of bits when the seed is not known

6.16 public key: the public portion of a public-key pair that may be used for signature verification, data encryption or key establishment procedures depending on the type of key pair

6.17 public-key pair: a related pair of data elements including a public key and a private key

6.18 public-key certificate: A data element usually created by a certificate authority that associates a particular public key with additional information and provides assurance as to the validity of the relationship and the integrity of the data being associated

6.19 random number generator: a device that provides a sequence of bits that is unpredictable

6.20 security manager: the entity that is responsible for the control of security relationships, authentication and key distribution

6.21 signature verification: a process by which a public key is used to verify that the signed data has not been modified and that the owner of the private key signed the data

6.22 signed data: data which has had a digital signature appended to it that assures the integrity of that data and the source of the signer

6.23 source authentication: authentication of the sender of the data

6.24 symmetric key: a secret key that is shared between two or more parties that may be used for encryption/decription and/or integrity protection/integrity verification depending on its intended use

6.25 trusted third party: an entity that may provide assurances to and is trusted by two or more parties who do not necessarily trust each other

New acronyms:

CA	certificate authority
KEK	key encryption key
PRNG	pseudo-random number generator
RNG	random number generator

Ed. action: add the above definitions and acronyms to the appropriate clauses.

From: Mark E Schrader

James,

Jay an I request two editorial changes in the current Pre0 draft:

1. Clause 7.4.10

The second row, and second column of the Table 64 , on printed page 109, line 33-34

CHANGE:

"Next GTS slot start time"

TO:

SFNext

Ed. action: Changed as indicated, plus had to change an occurance in the channel time grant command., new text is below:

The grant status field format shall be formatted as illustrated in Figure 44.

bits: b0-b15	b16-b9	b20-b23
SFNext	Reason code	Reserved

Figure 44—Grant status field format

...

The SFNext indicates the start time of the next GTS. The definition of this field is the same as that of the slot start time in 7.4.10.

REASON FOR CHANGE: This will make Table 64 consistent with the current text.

2. Clause 7.4.10

REMOVE: the following text from page 109, lines 48-49 "If the slot location field is to be interpreted as the SFNext field, then the field contains the least significant two bytes of a beacon count corresponding to the superframe in which the next AWAKE slot will be allo-cated."

INSERT: the following text at page 109, at the end of line 13:

SFNext field is defined for both ACTIVE CTA elments and EPS CTA elements, as the least significant two bytes of a beacon count value. For ACTIVE CTA, the field contains the least significant two bytes of a beacon count corresponding to next superframe in which an actual time slot will be allocated. For EPS CTA, it is equal to EPSNext, which corresponds to the next WAKE superframe.

REASON FOR CHANGE: The definition of SFNext is not correct and not in the right place. SFNext is first mentioned below figure 30, printed page 109, line 13. We do not have any definition there. There is an incorrect definiton that appears in lines 48-49 of the same page 109.

Ed. action: Old text removed. I had to fix a few things, new text looks like this:

... It shall be set to 1 if the slot location field is to be interpreted as the SFNext.

The SFNext field is defined for both ACTIVE CTA elments and EPS CTA elements. For ACTIVE CTA, the field contains the least significant two octets of a beacon number corresponding to next superframe in which an actual time slot will be allocated. For EPS CTA, it is equal to the least significant two octets of EPSNext, which corresponds to the next WAKE superframe.

Thanks,

Ed. note: Your welcome.

Mark and Jay

Ed. note: Your friendly neighborhood technical editor.