802.15.3c™

IEEE Standard for
Information technology—
Telecommunications and information exchange between systems—
Local and metropolitan area networks—
Specific requirements

Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs)

Amendment 2: Millimeter-wave-based Alternative Physical Layer Extension

IEEE Computer Society

Sponsored by the
LAN/MAN Standards Committee

IEEE
3 Park Avenue
New York, NY 10016-5997, USA

12 October 2009

IEEE Std 802.15.3c™-2009
(Amendment to
IEEE Std 802.15.3™-2003)

IEEE Std 802.15.3c™-2009
(Amendment to
IEEE Std 802.15.3™-2003)

IEEE Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements

Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs)

Amendment 2: Millimeter-wave-based Alternative Physical Layer Extension

Sponsor

LAN/MAN Standards Committee
of the
IEEE Computer Society

Approved 11 September 2009

IEEE-SA Standards Board

Abstract: This amendment defines an alternative physical layer (PHY) for IEEE Std 802.15.3-2003. Three PHY modes have been defined that enable data rates in excess of 5 Gb/s using the 60 GHz band. A beam-forming protocol has been defined to improve the range of communicating devices. Aggregation and block acknowledgment have been defined to improve the medium access control (MAC) efficiency at the high data rates provided for by the PHY.

Keywords: 60 GHz, aggregation, beam forming, block acknowledgment, millimeter wave, uncompressed video, wireless personal area network, WPAN

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

PDF: ISBN 978-0-7381-6050-4 STD95963
Print: ISBN 978-0-7381-6051-1 STDPD95963

Introduction

This introduction is not part of IEEE Std 802.15.3c-2009, IEEE Standard for Information technology Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs)—Amendment 2: Millimeter-wave-based Alternative Physical Layer Extension.

IEEE Std 802.15.3c-2009 is an amendment to IEEE Std 802.15.3-2003 (reaffirmed in 2008) that defines an alternative physical layer operating in the millimiter wave band along with the necessary MAC changes to support this PHY. Some of the key features and additions are as follows:

- Operation in the 60 GHz band.
- New data rates, with the highest greater than 5 Gb/s.
- Beamforming negotiation for the transmitter to increase the communication range.
- The ability to aggregate incoming data into single packets for improve MAC efficiency.
- Acknowledgment of invidual subpackets in a packet to reduce retransmission overhead.

The PHY specifices three modes and one common mode. The three PHY modes are as follows:

- Single carrier (SC) mode optimized for low power and low complexity.
- High-speed interface (HSI) mode optimized for low-latency bidirectional data transfer.
- Qudio/video (AV) mode optimized for the delivery of uncompressed, high-defintion video and audio.

Also defined as a part of the alternate PHY is common-mode signaling, which is a PHY mode that allows devices using different PHY modes to communicate.

Interest in developing a millimeter wave alternative PHY began in the July 2003 meeting in San Francisco with the formation of an interest group. A study group was formally created in the March 2004 IEEE 802 plenary meeting in Orlando and developed a project authorization request that was approved in March 2005. The first meeting as a task group was in May 2005 in Cairns, Australia and the group worked steadily developing channel models and evaluation documents. The PHY modes were selected in November 2007 at the Atlanta meeting and draft progressed rapidly, entering working group letter ballot in June 2008. After three working group recirculation ballots, sponsor ballot started in March 2009. A total of three sponsor recirculation ballots were held, leading to approval of IEEE Std 802.15.3c-2009 by the IEEE-SA Standards Board on 11 September 2009.

Notice to users

Laws and regulations

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

Copyrights

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

Updating of IEEE documents

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

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

Errata

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

Interpretations

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

Patents

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

Participants

At the time this amendment was submitted to the IEEE-SA for approval, the IEEE 802.15 Working Group had the following membership:

Robert F. Heile, *Chair*
Richard Alfvin, *Vice-Chair*
Patrick Kinney, *Vice-Chair*
James P. K. Gilb, *Technical Editor*

Reed Fisher, *IEEE 802.15.3c Chair*
Shu Kato, *IEEE 802.15.3c Vice-Chair*
Abbie Matthew, *IEEE 802.15.3c Vice-Chair*
Tuncer Baykas, *IEEE 802.15.3c Secretary*
Chin-Sean Sum, *IEEE 802.15.3c Secretary*
James P. K. Gilb, *IEEE 802.15.3c Technical Editor*

Tuncer Baykas, *IEEE 802.15.3c Assistant Editor*
Brian Gaffney, *IEEE 802.15.3c Assistant Editor*
Zhou Lan, *IEEE 802.15.3c Assistant Editor*
Chin-Sean Sum, *IEEE 802.15.3c Assistant Editor*
Chiu Ngo, *IEEE 802.15.3c Assistant Editor*

Gahng-Seop Ahn
Xizhi An
Takashi Arita
Arthur Astrin
Taehan Bae
Michael Bahr
Jay Bain
Vered Bar
John Barr
Anuj Batra
Phil Beecher
Ghulam Bhatti
Bruce Bosco
Nancy Bravin
Sverre Brubk
John Buffington
Kiran Bynam
Philippe Chambelin
Kuor-Hsin Chang
Soo-Young Chang
Clint Chaplin
Hind Chebbo
Chihong Cho
Sangsung Choi
Chia Chin Chong
Ciaran Connell
David Davenport
Hendricus De Ruijter
Gang Ding
Paul Dixon
Guido Dolmans
Igor Dotlic
Christophe Dugas
Dietmar Eggert
Amal Ekbal
Jason Ellis
Frederick Enns
Robert Fanfelle
Charles Farlow
John Farserotu
Will Filippo
George Flammer
Noriyasu Fukatsu
Kiyoshi Fukui
Shigeru Fukunaga
Ryuhei Funada
Uhland Goebel
Paul Gorday
Vivek Gupta
Shinsuke Hara
Hiroshi Harada
David Hart
Marco Hernandez
Jin-Meng Ho
Seungeun Hong
Wei Hong
Srinath Hosur
David Howard
Jung-Hwan Hwang
Ichirou Ida
Tetsushi Ikegami
Akio Iso
Wuncheol Jeong
Zhanfeng Jia
Seong-Soon Joo
Chol Su Kang
Noh-Gyoung Kang
Sungweon Kang
Tae-Gyu Kang
Yasunao Katayama
Yasutaka Kawamoto
Jeritt Kent
Dae Ho Kim
Jaehwan Kim
Kyeongpyo Kim
Yongsun Kim
Ryuji Kohno
Fumihide Kojima
Keitarou Kondou
Bruce Kraemer
Kyung Sup Kwak
Hyoungjin Kwon
Hyukchoon Kwon
Youngmi Kwon
Ismail Lakkis
John Lampe
Khanh Le
Myung Lee
Daniel Lewis
Huan-Bang Li
Liang Li
Sheung Li
Sang-Kyu Lim
Noam Livneh
Frederick Martin
Jeff McCullough
Michael Mcinnis
Michael McLaughlin
Klaus Meyer
Dino Miniutti
Emmanuel Monnerie
Rajendra Moorti
Theodore Myers
Yukimasa Nagai
Ken Naganuma
Tatsuo Nakagawa
Chiu Ngo
Paul Nikolich
Hirohito Nishiyama
Makoto Noda
Dominic O'Brien
Okundu Omeni
Ryoji Ono
Laurent Ouvry
Satoshi Oyama
James Pace
Hyung-Il Park

Jahng Park
Seung-Hoon Park
Tae Rim Park
Taejoon Park
Maulin Patel
Ranjeet Patro
Al Petrick
Frank Poegel
Stephen Pope
Clinton Powell
Jim Raab
Mohammad Rahman
Sridhar Rajagopal
Jayaram Ramasastry
Arthur Redfern
Ivan Reede
Edward Reuss
Richard Roberts
June Chul Roh
Benjamin Rolfe
John Rouse
Olivier Rousseaux
Seung-Moon Ryu
Didier Sagan
Zafer Sahinoglu
Katsuyoshi Sato
Hirokazu Sawada
Kamran Sayrafian
Donald Schultz
Jean Schwoerer
Cristina Seibert
Huairong Shao
Stephen Shellhammer
Nobuhiko Shibagaki
Shusaku Shimada
Chang Sub Shin
Jaeseung Son
Myunghee Son
Marinus Struik
Shuji Suzuki
Kazuaki Takahashi
Kenichi Takizawa
James Taylor
James Tomcik
Ichihiko Toyoda
Jason Trachewsky
Sana Ullah
Masahiro Umehira
Jana van Greunen
Joachim Walewski
Chao-Chun Wang
Junyi Wang
Qin Wang
Scott Weikel
Magnus Wiklund
Ludwig Winkel
Eun Tae Won
Pyo Chang Woo
Hiroyuki Yamagishi
Wen-Bin Yang
Kamya Yazdandoost
Kaoru Yokoo
Su Yong
Yang Moon Yoon
Mu Zhao
Bin Zhen

Major technical contributions were received from the following individuals:

Vered Bar
Tuncer Baykas
Troy Beukema
André Bourdoux
Bruce Bosco
Chang-Soon Choi
Ryuhei Funada
Brian Gaffney
Ian Gifford
James P. K. Gilb
Eckhard Grass
Hiroshi Harada
Tian-Wei Huang
Kyoichi Iigusa
Akio Iso
Yasunao Katayama
Shuzu Kato
Seongsoo Kim
Ryota Kimura
Fumihide Kojima
Keitarou Kondou
Edwin Chang Yeul Kwon
Zhiguo Lai
Ismail Lakkis
Zhou Lan
Ming Lei
Wei Li
Nadine Malhouroux-Gaffet
Abbie Matthew
Michael McLaughlin
Hiroyuki Nakase
Chiu Ngo
Yoshinori Nishiguchi
Makoto Noda
Hiroyo Ogawa
Jisung Oh
Pascal Pagani
Chang-woo Pyo
Yu Zhan Raymond
Mohammad Azizur Rahman
Rick Roberts
Katsuyoshi Sato
Hirokazu Sawada
Alireza Seyedi
Harkirat Singh
Huai-Rong Shao
Masashi Shinagawa
Yozo Shoji
Stan Skafidas
Chin-Sean Sum
Ichihiko Toyoda
Masahiro Umehira
Alberto Valdes-Garcia
Junyi Wang
Pengfei Xia
Sandra Xiang Ping Qin
Hiroyouki Yamagishi
Su-Khiong Yong

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

Richard Alfvin
Butch Anton
Yoshio Aoki
Taehan Bae
Jay Bain
Vered Bar
John Barr
Tuncer Baykas
Stephen Bennett
H. Stephen Berger
Harry Bims
Bruce Bosco
Nancy S. Bravin
Vern Brethour
William Byrd
James Carlo
Juan Carreon
Clint Chaplin
Wei-Peng Chen
Jaeweon Cho
Hokyu Choi
Keith Chow
Charles Cook
Todor Cooklev
Joseph Decuir
Yohannes Demessie
Russell Dietz
Thomas Dineen
John Dorsey
Paul Eastman
Richard Eckard
Darwin Engwer
Stanislav Filin
Michael Fischer
Reed Fisher
C. Fitzgerald
Mark Forbes
Dengwei Fu
Kiyoshi Fukui
Shigeru Fukunaga
Ryuhei Funada
Devon Gayle
Theodore Georgantas
James P. K. Gilb
Jeffrey Gilbert
Uhland Goebel
Michael Gordon
Randall Groves
Xiang Guan
C. Guy
Christopher Hansen
Hiroshi Harada
Robert F. Heile
Marco Hernandez
Keang Ho
Oliver Hoffmann
Robert Huang
Chan-Soo Hwang
Akio Iso
Atsushi Ito
Stephen Jackson
Raj Jain
Jaehyuk Jang
Bobby Jose
Woochul Jung
Tal Kaitz
Naveen Kakani
Shinkyo Kaku
Hyunjeong Kang
Noh-Gyoung Kang
Haideh Karkhanechi
Assaf Kasher
Yasunao Katayama
Isao Katsuno
Stuart J. Kerry
Jaehwa Kim
Sangbum Kim
Yongbum Kim
Youngdoo Kim
Youngsoo Kim
Patrick Kinney
Cees Klik
Fumihide Kojima
Keitarou Kondou
Bruce Kraemer
Shoji Kuriki
Edwin Kwon
Zhou Lan
Jeremy Landt
Michael McLaughlin
Insun Lee
Jae Min Lee
Jeong Ho Lee
Myung Lee
Tae Hoon Lee
John LeMoncheck
Jan-Ray Liao
Arthur Light
Chiwoo Lim
Hyoung-Kyu Lim
Jianhan Liu
Noam Livneh
Steve Lo
William Lumpkins
John Marshall
David Mazzarese
Patrick Mcelwee
Apurva Mody
Hiroyuki Nakase
Michael S. Newman
Chiu Ngo
Makoto Noda
John Notor
Satoshi Obara
Knut Odman
Jisung Oh
Okundu Omeni
Chris Osterloh
Satoshi Oyama
James R. Parker
Eldad Perahia
Venkatesha Prasad
Mohammad Rahman
Jayaram Ramasastry
Ivan Reede
Robert Robinson
Benjamin Rolfe
Jaeyoung Ryu
Randall Safier
Osman Sakr
John Santhoff
Shigenobu Sasaki
Katsuyoshi Sato
Hirokazu Sawada
Bartien Sayogo
Cheng Shan
Huairong Shao
Jaejeong Shim
Changyong Shin
Gil Shultz
Harkirat Singh
Jaeseung Son
Guocong Song
Amjad Soomro
Thomas Starai
Walter Struppler
Mark Sturza
Chin-Sean Sum
Jun Ichi Takada
David Thompson
Geoffrey Thompson
Karim Nassiri Toussi
Ichihiko Toyoda
Solomon Trainin
Masahiro Umehira
Alberto Valdes-Garcia
Dmitri Varsanofiev
Prabodh Varshney
Junyi Wang
Stanley Wang
Fujio Watanabe
Robert Wiser
Andreas Wolf
Pyo Chang Woo
Harry Worstell
Xuyong Wu
Hiroyuki Yamagishi
James Yee
Su Khiong Yong
Oren Yuen
Chunhui Zhu

When the IEEE-SA Standards Board approved this standard on 11 September 2009, it had the following membership:

Contents

IEEE Standard for
Information technology—
Telecommunications and information exchange between systems—
Local and metropolitan area networks—
Specific requirements

Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs)

Amendment 2: Millimeter-wave-based Alternative Physical Layer Extension

IMPORTANT NOTICE: This standard is not intended to ensure safety, security, health, or environmental protection in all circumstances. Implementers of the standard are responsible for determining appropriate safety, security, environmental, and health practices or regulatory requirements.

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

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

3. Definitions

Insert the following definitions in alphabetical order:

3.45 beacon: A broadcast frame sent by the piconet coordinator operating in either omni mode or quasi-omni mode for piconet synchronization and management.

3.46 dBm: Decibels relative to a milliwatt.

3.47 guard interval: A time period placed at the front of a symbol for inter-symbol interference avoidance.

4. Acronyms and abbreviations

Insert the following acronyms in alphabetical order:

AAS	asymmetric antenna system
AV	audio/visual
BBIFS	beam-forming beam-level interframe spacing
BCRD	bidirectional channel time allocation relinquish duration
Blk-ACK	block acknowledgment
BPSK	binary phase-shift keying
BSIFS	beam-forming sector-level interframe spacing
BST	beam switching/steering and tracking
CE	channel estimation
CES	channel estimation sequence
CMS	common mode signaling
DAMI	dual alternate mark inversion
EEP	equal error protection
EIRP	equivalent isotropic radiated power
GI	guard interval
GMSK	Gaussian filtered minimum shift keying
HR	high rate
HRP	high-rate physical layer
HRPDU	high-rate protocol data unit
HRS	high resolution
HSI	high-speed interface
LDPC	low-density parity check
LFSR	linear feedback shift register
LR	low rate
LRP	low-rate physical layer
LRPDU	low-rate protocol data unit
MCS	modulation and coding scheme
MPR	mandatory PHY rate
MR	medium rate
mmWave	millimeter wave
MSK	minimum shift keying
NLOS	non line of sight

OFDM	orthogonal frequency-division multiplexing
OOK	on-off keying
PCES	pilot channel estimation sequence
PET	pattern estimation and tracking
PRBS	pseudo-random bit stream
PSD	power spectral density
QT	quasi-omni training
RS	Reed Solomon
SAS	symmetric antenna system
SC	single carrier
S-CAP	sub-contention access period
SINR	signal to interference plus noise ratio
SSB	single side band
ST	sector training
SYNC	synchronization
TSD	transmit switched diversity
UEP	unequal error protection

5. General description

5.3 Overview of MAC functionality

After 5.3.11, insert the following new subclauses:

5.3.12 Superfame structure using quasi-omni mode

A mmWave WPAN operates in either omni mode or quasi-omni mode. The superframe structure for omni mode is illustrated in Figure 2. The superframe structure for quasi-omni mode is illustrated in Figure 2a. In quasi-omni mode, the same beacon frame is transmitted in different quasi-omni directions in a round-robin way in order to allow devices (DEVs) located in different directional coverage to join the piconet, as described in 8.6.6.

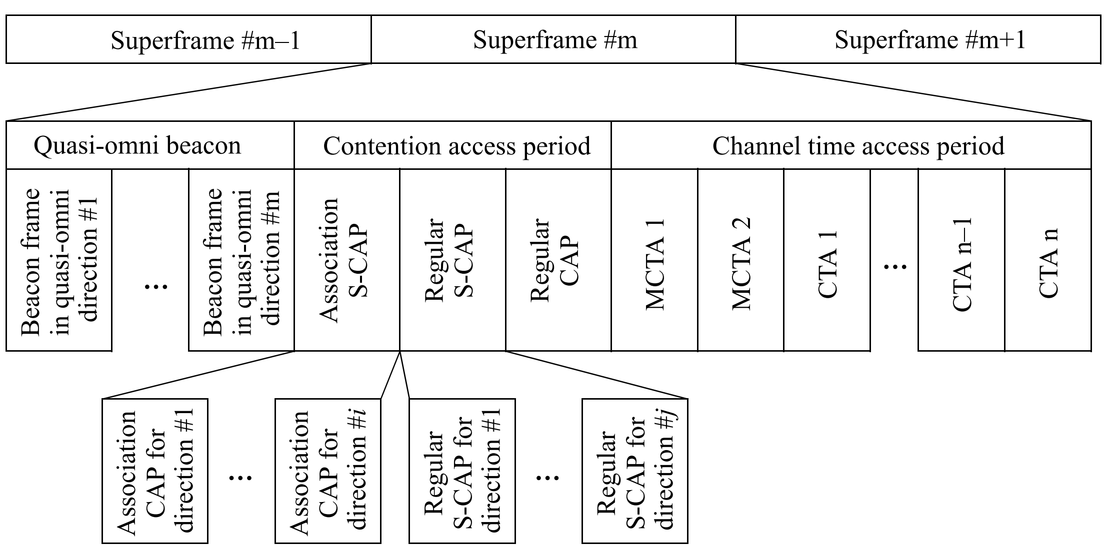

Figure 2a—Piconet superframe structure in quasi-omni mode

5.3.13 Frame aggregation

Frame aggregation, as described in 8.7a, is supported for the purpose of high throughput. With the high data rate provided by the mmWave PHY, throughput increases if the payload length is increased. Two aggregation methods are provided, one is suitable for normal data and A/V streaming and the other is optimized for low-latency communications.

5.3.14 Beam forming

The beam forming procedure allows a DEV that is capable of beam forming to increase the antenna gain for supporting high data rate transmission. The beam forming procedures are described in Clause 13.

5.3.15 Channel probing

Channel probing, as described in 8.9.7, is an optional capability that is used by a DEV to determine the best MCS to use with the target DEV under current channel condition.

5.3.16 Unequal error protection (UEP)

For flexible error protection that accounts for data that has more emphasis on the msbs than the lsbs, e.g., video, UEP is provided as an option. Three types of UEP are defined, as described in 8.16, for use with the three mmWave PHYs.

After 5.4, insert the following subclause as 5.5:

5.5 Characteristics of the mmWave PHY

5.5.1 mmWave PHY characteristics

The mmWave PHY, as described in Clause 12, is defined for the frequency band of 57.0–66.0 GHz, as allocated by the regulatory agencies in Europe, Japan, Canada, and the United States as well as any other areas where the regulatory bodies have allocated this band. While four channels are defined for the PHY, regulatory requirements allow fewer in some regions.

A total of three PHYs are defined for the mmWave PHY. They are as follows:

a) Single Carrier mode in mmWave PHY (SC PHY), as described in 12.2.
b) High Speed Interface mode in mmWave PHY (HSI PHY), as described in 12.3.
c) Audio/Visual mode in mmWave PHY (AV PHY), as described in 12.4.

For DEVs that implement the mmWave PHY, at least one of the three PHYs is required. In addition, to promote coexistence and interoperability, a common mode signaling (CMS) is defined based on a low data rate SC PHY mode, as described in 12.1.12.

The SC PHY supports a variety of modulation and coding schemes (MCSs) that support up to 5 Gb/s. The SC PHY supports a wide range of modulations, $\pi/2$ BPSK, $\pi/2$ QPSK, $\pi/2$ 8-PSK, $\pi/2$ 16-QAM, pre-coded MSK, pre-coded GMSK, on-off keying (OOK), and dual alternate mark inversion (DAMI). The coding schemes included are Reed-Solomon (RS) coding with low-complexity implementation and low-density parity check (LDPC) coding with high error-correcting capability. Code spreading using either linear feedback shift register (LFSR) code or Golay sequence is also applied to increase robustness of the system.

The HSI PHY, as described in 12.3, is designed for NLOS operation and uses OFDM with an FEC based on LDPC.

The AV PHY, as described in 12.4, is designed for NLOS operation and the transport of uncompressed, high definition video and audio. It uses OFDM modulation with convolutional inner code and a Reed Solomon outer code. The AV mode supports omni-directional coverage via the low-rate PHY (LRP) for the purpose of setting up high-throughput connections using the high-rate PHY (HRP).

Different PHYs are a result of demands of different market segments, which were based on the development of the usage models for this standard. For example one usage model is for kiosk applications. This usage model requires 1.5 Gb/s at a 1 m range. The SC-PHY can provide such a data rate at that short range with less complexity thus lower cost than an OFDM PHY. Another usage model required the streaming of uncompressed video. Due to the nature of uncompressed video signals a special PHY, the AV PHY, was selected to provide high throughput. A third usage model involves an ad-hoc system to connect computers and devices around a conference table. In this usage model, all of the devices in the WPAN will have bi-directional, NLOS high speed, low-latency communication, which is provided for by the HSI PHY. Mandatory data rates of all those PHYs are selected according to specific usage models. In addition, higher data rates are provided to give options to the implementors so that they can best address the different market segments.

5.5.2 Piconets using mmWave PHY modes

When a mmWave PHY PNC capable DEV starts a piconet, the type of piconet it starts depends on the PHY modes that are supported. For example, if the PNC capable DEV supports only the AV PHY mode, it would start an AV piconet in which the beacon is sent with the AV PHY mode and the contention periods (CPs) would use the AV PHY mode. The AV PNC would also send the sync frame to improve coexistence with other piconets. DEVs that support only the AV PHY mode are able to find and join the piconet using the AV PHY mode.

The same process is used for a mmWave PHY PNC capable DEV that supports only HSI PHY mode or SC PHY mode, with the exception that an SC PHY mode PNC does not need to send a sync frame as its beacon is sent in CMS mode already.

If a PNC capable DEV supports more than one mmWave PHY mode, then it is able to select the type of piconet it starts, potentially starting more than one piconet, each with a different PHY mode, with the additional piconets as dependent piconets. Alternatively, the multi-mode PNC is able to have a single piconet in which DEVs use any MCS in a CTA that is supported by both of the DEVs.

The complete rules are described in 12.1.

6. Layer management

6.3 MLME SAP interface

Insert the following row in Table 3 as shown. The other rows are not changed and are not shown.

Table 3—Summary of MLME primitives

Name	Request	Confirm	Indication	Response
MLME-TXDIV	6.3.18.1	6.3.18.2	—	—

6.3.2 Scanning for piconets

Insert the following row to Table 3c in 6.3.2 as shown. All other rows are unchanged and are not shown.

Table 3c—Elements of PiconetDescription

Name	Type	Valid range	Description
PHYMode	Enumeration	2.4_GHZ, SC_MMWAVE, HSI_MMWAVE, AV_MMWAVE	The PHY mode that is being used in the piconet that was found.

6.3.3 Starting a piconet

Insert the following row to Table 3e in 6.3.3 as shown. All other rows are unchanged and are not shown.

Table 3e—MLME-START primitive parameters

Name	Type	Valid range	Description
PHYMode	Enumeration	2.4_GHZ, SC_MMWAVE, HSI_MMWAVE, AV_MMWAVE	The PHY that will be used for the beacons and the CP(s) in the piconet that will be started.

6.3.3.1 MLME-START.request

Change the primitive definition in 6.3.3.1 as follows:

```
MLME-START.request            (
                              BSIDLength,
                              BSID,
                              SECMode,
                              MinDepSuperframePercent,
                              DesiredDepSuperframePercent,
                              PHYMode
                              )
```

Insert the following subclause after 6.3.17 from IEEE Std 802.15.3b as 6.3.18:

6.3.18 Transmit switched diversity

These primitives are used to exchange the transmit switched diversity capability between a source and a destination, as described in 13.8. The parameters used for these primitives are defined in Table 3x.

Table 3x—MLME-TXDIV primitive parameters

Name	Type	Valid range	Description
TrgtID	Integer	Any valid DEVID, as defined in 7.2.3	Specifies the DEVID of the target DEV for exchanging transmit Switched Diversity information.
OrigID	Integer	Any valid DEVID, as defined in 7.2.3	The DEVID of the DEV that initiated the MLME request.
SupportedNumAntennas	Integer	0–15	The number of antennas supported by the DEV.
Timeout	Integer	0–65535	The time in milliseconds allowed for the primitive to complete.
TXDiversityThresholdType	Enumeration	LQI type, as defined in 7.4.11.	Specifies the type of LQI measurement to use.
TXDiversityTheshold	Integer	Any valid LQI value, as defined in 7.4.27.	Specifies the value of the threshold for antenna switching.
AntennaIndex	Integer	0–15	Specifies antenna index to be used.
ResultCode	Enumeration	SUCCESS, FAILURE	Indicates the result of the MLME request.
ReasonCode	Enumeration	NOT_SUPPORTED, OTHER	Indicates the reason for a ResultCode of FAILURE.

6.3.18.1 MLME-TXDIV.request

This primitive is used to request to use transmit switched diversity. The semantics of this primitive are as follows:

```
MLME-TXDIV.request          (
                            TrgtID,
                            TXDiversityThresholdType,
                            TXDiversityTheshold,
                            Timeout
                            )
```

The primitive parameters are defined in Table 3x.

6.3.18.2 MLME-TXDIV.confirm

This primitive is used to report the result of the request for transmit switched diversity. The semantics of this primitive are as follows:

```
MLME-TXDIV.confirm          (
                            AntennaIndex,
                            ResultCode,
                            ReasonCode
                            )
```

The primitive parameters are defined in Table 3x.

6.5 MAC management

6.5.1 MAC PIB PNC group

Change and insert the following rows in Table 33 as shown. The other rows are unchanged and are not shown.

Table 33—MAC PIB PNC group parameters

Managed object	Octets	Definition	Access
MACPIB_CAPStartTime	2	The time at which the CAP starts when quasi-omni beacons are used.	Read only

6.5.2 MAC PIB characteristics group

Insert the following new row in Table 34 as shown. The other rows are unchanged and are not shown.

Table 34—MAC PIB characteristics group parameters

Managed object	Octets	Definition	Access
MACPIB_CTARelinquishDuration	2	The CTA Relinquish Duration field value sent in the CTA Relinquish Duration IE	Read/write

6.6 MAC SAP

Insert the following new rows to Table 36 as shown. The other rows are unchanged and are not shown.

Table 35—MAC-ISOCH-DATA and MAC-ASYNC-DATA primitive parameters

Name	Type	Range	Description
UncompressedVideo	Boolean	TRUE, FALSE	Set to TRUE if the Data parameter contains uncompressed video data.
UEPAllowed	Boolean	TRUE, FALSE	Indicates if UEP is allowed for the transmission of the data.
ErrorsFreeData	Boolean	TRUE, FALSE	Indicates if the Data parameter that was received contains errors or is free from errors.
InterlacedFieldIndication	Enumeration	TOP_FIELD, BOTTOM_FIELD, NOT_INTERLACED	If UncompressedVideo is TRUE, the parameter indicates if the pixels belong to the top or bottom field of an interlaced image or that the image is not interlaced.
VideoFrameNumber	Integer	As defined in 7.2.9.1.4	If UncompressedVideo is TRUE, the parameter contains a sequential numbering of the video frames that are being transferred.
HPosition	Integer	As defined in 7.2.9.1.4	If UncompressedVideo is TRUE, the parameter contains the horizontal position of the first pixel in the MSDU.
VPosition	Integer	As defined in 7.2.9.1.4	If UncompressedVideo is TRUE, the parameter contains the vertical position of the first pixel in the MSDU.

Insert the following paragraph to the end of 6.6:

The parameters InterlacedFieldIndication, VideoFrameNumber, HPosition, and VPosition are used by the AV PHY in the Video Subheader field, as described in 7.2.9.1.4. These fields are sent in the MAC header because it has lower error rate than the subframes carrying video data. For uncompressed video applications, video data that is in error can still be displayed, but only if the position of the pixel data is known.

6.6.4 MAC-ISOCH-DATA.request

Change the primitive definition as shown:

```
MAC-ISOCH-DATA.request        (
                              RequestID,
                              StreamIndex,
                              TransmitTimeout,
                              MaxRetries,
                              SNAPHeaderPresent,
                              ACKRequested,
                              ConfirmRequested,
                              Length,
                              Data,
                              DataType,
                              UEPAllowed,
                              InterlacedFieldIndication,
                              VideoFrameNumber,
                              HPosition,
                              VPosition
                              )
```

6.6.5 MAC-ISOCH-DATA.indication

Change the primitive definition as shown:

```
MAC-ISOCH-DATA.indication     (
                              TrgtID,
                              OrigID,
                              StreamIndex,
                              SNAPHeaderPresent,
                              Length,
                              Data,
                              DataType,
                              ErrorFreeData,
                              InterlacedFieldIndication,
                              VideoFrameNumber,
                              HPosition,
                              VPosition
                              )
```

Insert the following paragraph after the primitive definition in 6.6.5:

Data identified as video by the source DEV may be passed by the destination DEV to the higher layers even if the data was received in error. The error status of the data passed up by this primitive is indicated in the ErrorFreeData parameter.

7. MAC frame format

7.2 General frame format

Change the second paragraph in 7.2 as shown:

The figures in this subclause are a representation of the MAC header and MAC frame body. The HCS is not shown since this is calculated and verified by the PHY. The MAC frame shall be formatted as illustrated in Figure 6. The maximum size of the MAC frame body, pMaxFrameBodySize, is a PHY dependent parameter that includes the frame payload and FCS, but not the PHY preamble, PHY header, MAC header, MAC subheader, or MAC Header validation. ~~For the 2.4 GHz PHY, this parameter is defined in 11.2.8.1.~~ The parameter pMaxFrameBodySize is defined in the following:

— 11.2.8.1 for the 2.4 GHz PHY
— 12.2.7.1 for the SC PHY mode
— 12.3.6.3 for the HSI PHY mode
— 12.4.1.3.1 for the AV PHY mode

7.2.1 Frame control

Replace Figure 9 with the following:

bits: b15	b14	b13	b12	b11	b10	b9	b8-b7	b6	b5-b3	b2-b0
Reserved	Blk-ACK	CTA relinquish	Imp-ACK NACK	Imp-ACK request	More data	Retry	ACK policy	SEC	Frame type	Protocol version

Figure 9—Frame control field format

7.2.1.2 Frame type

Change Table 39 as follows. The other rows are kept unchanged.

Table 39—Valid frame type values (numeric values in this table are shown in binary)

Type value b5 b4 b3	Frame type description	Subclause
110	Sync frame	7.3.6
~~110~~ 111	Reserved	—

Change the title of 7.2.1.4 as follows:

7.2.1.4 ACK policy, ~~and~~ implied ACK (Imp-ACK) request and Blk-ACK

Change the first paragraph as shown:

The ACK Policy field, ~~and~~ Imp-ACK Request field~~s~~, and Blk-ACK field are used to indicate the type of acknowledgment procedure that the addressed recipient is required or allowed to perform. The use of the ACK procedures is described in 8.8. The allowed values of the ACK Policy field, ~~and~~ the Imp-ACK Request field, and the Blk-ACK field are defined in Table 40. The ACK policy of a frame is determined by the combination of the ACK Policy field, ~~and~~ the Imp-ACK Request field, and Blk-ACK field.

Delete Table 40 and insert the following table in its place:

Table 40—Valid ACK policy field type values

Blk-ACK field b14	Imp-ACK request field b11	ACK policy field b8 b7	ACK policy type	Description
0	0	00	No ACK	The recipient(s) does not acknowledge the transmission, and the sender treats the transmission as successful without regard for the result, as described 8.8.1
0	0	01	Immediate ACK (Imm-ACK)	The addressed recipient returns an Imm-ACK frame after successful reception, as described in 8.8.2.
0	0	10	Delayed ACK (Dly-ACK)	The addressed recipient keeps track of the frames received with this policy until requested to respond with a Dly-ACK frame, as described in 8.8.3.
0	0	11	Dly-ACK request	The addressed recipient returns either an Imm-ACK or a Dly-ACK frame after successful reception, as described in 8.8.3.
0	1	01	Imp-ACK	The addressed recipient returns an Imm-ACK frame, a data frame, or a command frame after successful reception, as described in 8.8.4.
1	0	01	Blk-ACK	The addressed recipient returns a frame with the MAC sub-header including the Blk-ACK Bitmap field after successful reception of the MAC header, as described in 8.8.3b.

7.2.4 Fragmentation control

Change the second paragraph in 7.2.4 as shown:

The three octets that compose the Fragmentation Control field may be used for reporting PHY-dependent receive status information to the transmitting DEV in Imm-ACK, ~~and~~ Dly-ACK, and data frames with ACK policy set to Blk-ACK. If the source DEV is not reporting PHY-dependent receive status information in an Imm-ACK or Dly-ACK frame, it shall set the fragmentation field of the frame to all zeros, i.e., 0x000000. All other values are PHY-dependent. The receive status for the 2.4 GHz PHY is defined in 11.7. The receive status for the mmWave PHYs is defined in 12.1.8.3.

7.2.5 Stream index

Change the dashed list in 7.2.5 as shown:

The Stream Index field reserved values are as follows:

— 0x00 reserved for asynchronous data
— 0xFB reserved for beam forming starting from sector level training
— 0xFC reserved for beam forming staring from beam level training
— 0xFD reserved for MCTA traffic
— 0xFE reserved for unassigned streams
— 0xFF reserved for future use

7.2.6 MAC header validation

Change the paragraph in 7.2.6 as shown:

When the PHY receives a frame it validates the received frame's MAC header before passing the MAC header and its associated MAC frame body to the MAC. The protection mechanism used to ~~validiate~~ validate the MAC header is PHY dependent. In addition, the bit order and the length of the protection mechanism, pLengthHCS, are also PHY dependent. ~~For the 2.4 GHz PHY, the~~ The MAC header protection mechanism is defined in the following:~~11.2.9.~~

— 11.2.9 for the 2.4 GHz PHY
— 12.2.3.2.2 for the SC PHY mode
— 12.3.3.4 for the HSI PHY mode
— 12.4.1.4 for the AV PHY mode

After 7.2.7, insert the following subclauses as 7.2.8 and 7.2.9:

7.2.8 SC and HSI aggregated frame format

Figure 10a illustrates the SC and HSI aggregated frame format.

MAC frame body	MAC subheader

Figure 10a—SC and HSI aggregated frame format

The MAC frame body and MAC subheader for standard and low-latency aggregation are defined in 7.2.8.1 and 7.2.8.2.

7.2.8.1 Standard aggregation format

7.2.8.1.1 Non-secure standard aggregation format

To use standard aggregation, the Aggregation field and Low-latency Mode field in PHY header shall be set as described in 12.2.3.2.1.

The MAC subheader for standard aggregation shall be formatted as illustrated in Figure 10b.

octet: 1	5	...	5	2
RX buffer size	Subheader 8	...	Subheader 1	Blk-ACK bitmap

Figure 10b—MAC subheader format for standard aggregation

The Blk-ACK Bitmap field shall be formatted as illustrated in Figure 10c.

bits:8	8
ACK or lsb ACK	ACK or msb ACK

Figure 10c—Blk-ACK bitmap field format

In EEP mode, the ACK or msb ACK field is a bitmap that indicates if a subframe was correctly received. The bit position zero, which is the first bit from right in Figure 10c, corresponds to the first subframe of the frame that is being ACKed. If a subframe was correctly received, then the bit for that subframe is set to one and shall be set to zero otherwise. The ACK or lsb ACK field shall be set to the same value as the ACK or msb ACK field.

To use UEP mode, the UEP field in PHY header shall be set as described in 12.2.3.2.1. In UEP mode, the ACK or msb ACK field is a bitmap that indicates if the msbs of a subframe were correctly received. The bit position zero, which is the first bit from right in Figure 10c, corresponds to the first subframe of the frame that is being ACKed. If the msbs of the subframe were correctly received, then the bit for that subframe is set to one and shall be set to zero otherwise. The ACK or lsb ACK field is a bitmap that indicates if the lsbs of a subframe were correctly received. The bit position zero of the ACK or lsb ACK field corresponds to the first subframe of the frame that is being ACKed. If the lsbs of the subframe were correctly received, then the bit for that subframe is set to one and shall be set to zero otherwise.

The Subheader field shall be formatted as illustrated in Figure 10d.

bits: 1	1	7	9	1	2	11	1	1	1	5
Reserved	Last fragment	Fragment number	MSDU number	Skewed constellation	Subframe information	Subframe length	Resolution indication	Retry	FCS present	MCS information

Figure 10d—Subheader field format

In UEP mode, the MCS Information field shall be set to the UEP MCS (as defined in 12.2.2.9 for SC PHY and 12.3.2.1 for HSI PHY) that is used for the subframe.

In EEP mode, the MCS Information field shall be set to zero. All the subframes use the MCS indicated in the PHY header.

The FCS Present field shall be set to one if the subframe uses an FCS and shall be set to zero otherwise. The FCS is optional for subframes for the cases in which the upper layer will handle checking the data integrity. If the FCS present field is set to zero, then the subframe is considered correctly received only if the PHY header, MAC header, and MAC subheader were all correctly received.

The Retry field shall be set to one if the subframe is a retransmission and shall be set to zero otherwise.

The Resolution Indication field shall be set to zero if the resolution of the Subframe Length field is 1 octet and shall be set to one if the resolution is 512 octets.

The Subframe Length field is used to determine the length of the subframe before coding, not including the FCS. If the resolution of the Subframe Length field is 1 octet, then this field contains the length of the subframe in octets. If the resolution of the Subframe Length field is 512 octets, then the length of the subframe in octets is 512 times the value of this field. If the Subframe length field is set to zero, the corresponding subframe is not present in the aggregated data frame.

The Subframe Information field indicates if the subframe contains msb, lsb, or msb and lsb combined data. Valid values for the Subframe Information field are as follows:

- 0 → The subframe contains only msb data
- 1 → The subframe contains only lsb data
- 2 → The subframe contains msb and lsb data
- 3 → Reserved

The Skewed Constellation field shall be set to one if a skewed constellation is used in the subframe and shall be set to zero otherwise.

The MSDU Number field indicates the MSDU number of the subframe. Each subframe that contains a fragment from the same MSDU shall have the same MSDU number.

The Fragment Number field indicates the fragment sequence number of the subframe within the current MSDU, if the subframe contains a MSDU fragment. This field shall be set to zero if the corresponding subframe contains an unfragmented MSDU.

The Last Fragment bit shall be set to one if the subframe contains the last fragment of the MSDU and shall be set to zero otherwise.

The RX Buffer Size field indicates the free buffer space at the target DEV as a multiple of the preferred fragment size, as defined in 7.4.11.

The MAC frame body for standard aggregation shall be formatted as illustrated in Figure 10e.

octets: 4 or 8	variable	…	4 or 8	variable
FCS or combined FCS	Subframe payload *n*	…	FCS or combined FCS	Subframe payload 1

Figure 10e—MAC frame body format for standard aggregation

The maximum number of subframes that are aggregated in one frame shall be mMaxSubframeSize, as defined in 8.15.

In EEP mode, the subframe contains the FCS field, as defined in 7.2.7.6. In UEP mode, the subframe contains the Combined FCS field, as defined in Figure 10ag. The Subframe Payload field includes an LLC/SNAP header as the first octets in the payload, as defined in A.1.

7.2.8.1.2 Secure standard aggregation format

The secure MAC subheader for standard aggregation shall be formatted as illustrated in Figure 10f.

octet: 1	5	...	5	2	5
RX buffer size	Subheader 8	...	Subheader 1	Blk-ACK bitmap	Security header

Figure 10f—Secure MAC subheader format for standard aggregation

The Security header shall be formatted as illustrated in Figure 10g.

octets: 1	2	2
Subframe security	SFC	SECID

Figure 10g—Security header format

The SECID field is used to identify the key set that is used to encrypt and/or authenticate the data in the frame, as defined in 7.2.7.2.

The SFC field contains a counter that is used to ensure the uniqueness of the nonce of a secure frame, as defined in 7.2.7.3.

The Subframe Security field contains a bitmap that indicate if a subframe applies security or not. The bit position zero, which is the first bit from right in Figure 10g, corresponds to the first subframe. The bit shall be set to one if the subframe applies security and shall be set to zero otherwise.

The secure MAC frame body for standard aggregation shall be formatted as illustrated in Figure 10h.

octets: 4 or 8	8	variable	…	4 or 8	8	variable
FCS or combined FCS	Integrity code	Subframe payload n	…	FCS or combined FCS	Integrity code	Subframe payload1

Figure 10h—Secure MAC frame body format for standard aggregation

The Integrity Code field is used to cryptographically protect the integrity of the header and payload as defined in 7.2.7.5.

7.2.8.2 Low-latency aggregation formats

7.2.8.2.1 Non-secure low-latency aggregation format

To use low-latency aggregation mode, the Aggregation field and the Low-latency Mode field in PHY header shall be set as described in 12.2.3.2.1.

The MAC subheader for low-latency aggregation shall be formatted as illustrated in Figure 10i.

octets: 64	2	2	1	1
Blk-ACK bitmap	MSDU response number	MSDU request number	RX buffer size	UEP

Figure 10i—MAC subheader field format for low-latency aggregation

The UEP field shall be formatted as illustrated in Figure 10j.

bits: 3	5
Reserved	UEP MCS

Figure 10j—UEP field format

In EEP mode, the UEP MCS field shall be set to zero and all subframes use the MCS indicated in the PHY header.

In UEP mode, the UEP MCS field indicates the MCS used for UEP mode for the subframes, as defined in 12.2.2.9.

The RX Buffer size field indicates the free buffer space at the destination DEV as a multiple of the preferred fragment size, as defined in 7.4.11.

The MSDU Request Number field and the MSDU Response Number field shall be formatted as illustrated in Figure 10k.

bits: 7	9
Reserved	MSDU sequence number

Figure 10k—MSDU request/response number field format

The MSDU Sequence Number field is used as the basis for the Blk-ACK bitmap.

The MSDU Request Number field indicates the most recent MSDU sequence number acknowledged at the transmitter.

The MSDU Response Number field indicates the first MSDU sequence number of the transmitted Blk-ACK bitmap-field. The MSDU Response Number field is used to generate the offset for the Blk-ACK bitmap field. The MSDU Response Number field shall be a copy from the received MSDU Request Number, upon reception of a valid MAC Subheader.

For EEP mode the Blk-ACK Bitmap field shall be formatted as illustrated in Figure 10l.

bits: 256	1	…	1	1
Reserved	Subframe ACK #256	…	Subframe ACK #2	Subframe ACK #1

Figure 10l—EEP mode Blk-ACK bitmap field format

The Subframe ACK field is an offset from the MSDU sequence number of the received MSDU Response Number field. It shall be set to one if the corresponding subframe was correctly received and shall be set to zero otherwise.

For UEP mode the Blk-ACK Bitmap field shall be formatted as illustrated in Figure 10m.

bit: 1	1	…	1	1	1	1
Subframe lsb ACK #256	Subframe msb ACK #256	…	Subframe lsb ACK #2	Subframe msb ACK #2	Subframe lsb ACK #1	Subframe msb ACK #1

Figure 10m—UEP mode Blk-ACK bitmap field format

The Subframe msb ACK field shall be set to one if the msbs of the corresponding subframe were correctly received and shall be set to zero otherwise.

The Subframe lsb ACK field shall be set to one if the lsbs of the corresponding subframe were correctly received and shall be set to zero otherwise.

The MAC frame body for low-latency aggregation shall be formatted as illustrated in Figure 10n.

octets: 4 or 8	variable	3	…	4 or 8	variable	3
FCS or combined FCS	Subframe payload n	MSDU subheader n	…	FCS or combined FCS	Subframe payload 1	MSDU subheader 1

Figure 10n—MAC frame body format for low-latency aggregation

The maximum number of subframes that are be aggregated in one frame shall be mMaxSubframeSize, as defined in 8.15.

The MSDU Subheader field shall be formatted as illustrated in Figure 10o.

bits: 8	1	6	9
MSDU subheader HCS	Reserved	Subframe length	MSDU number

Figure 10o—MSDU subheader format

The MSDU Number field indicates the sequence number of the MSDU that is aggregated in the subframe. The MSDU number is unique in a window of 256 MSDUs, allowing retransmission of an out-of-order MSDU over several frames.

Subframe Length field contains the length of each subframe in unit of four octets. This field may be set to zero to allow idle transmission of data if data is not present at the FCSL. The HCS field shall be inverted in the case of a zero length MSDU, so that the MSDU HCS check fails.

The MSDU subheader HCS is an 8 bit CRC defined as the ones-complement of the remainder of the division of the 16 bits of the header by the polynomial $x^8 + x^2 + x + 1$. A serial implementation is illustrated in Figure 10p.

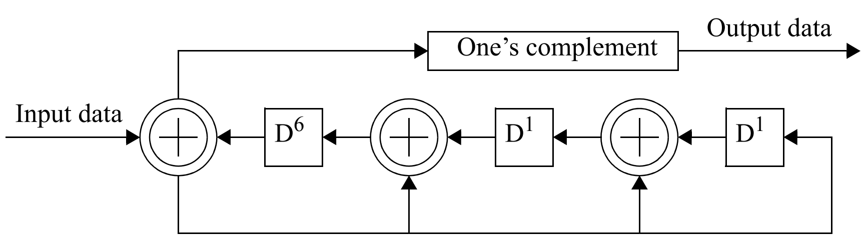

Figure 10p—HCS implementation example for MSDU subheader

In EEP mode, the subframe contains the FCS field, as defined in 7.2.7.6. In UEP mode, the subframe contains the Combined FCS field, as defined in Figure 10ag.

7.2.8.2.2 Secure low-latency aggregation format

The Secure MAC subheader for low-latency aggregation shall be formatted as illustrated in Figure 10q.

octets: 32/64	2	2	1	1	2	2
Blk-ACK bitmap	MSDU response number	MSDU request number	RX buffer size	UEP	SFC	SECID

Figure 10q—Secure MAC subheader field format for low-latency aggregation

The SECID field is used to identify the key set that is used to encrypt and/or authenticate the data in the frame, as defined in 7.2.7.2.

The SFC field contains a counter that is used to ensure the uniqueness of the nonce of a secure frame, as defined in 7.2.7.3.

The secure MAC frame body for low-latency aggregation shall be formatted as illustrated in Figure 10r.

octets: 4 or 8	8	variable	3	…	4 or 8	8	variable	3
FCS or combined FCS	Integrity code	Subframe payload 1	Secure MSDU subheader n	…	FCS or combined FCS	Integrity code	Subframe payload 1	Secure MSDU subheader n

Figure 10r—Secure MAC frame body format for low-latency aggregation

The Secure MSDU Subheader field shall be formatted as illustrated in Figure 10s.

bits: 8	1	6	9
MSDU subheader HCS	Subframe security	Subframe length	MSDU number

Figure 10s—Secure MSDU subheader format

The Subframe Security bit shall be set to one if the subframe applies security and shall be set to zero otherwise.

The Integrity Code field is used to cryptographically protect the integrity of the header and payload as defined in 7.2.7.5.

7.2.9 AV aggregated frame format

The AV aggregated frame format is optimized to carry uncompressed audio and video in an efficient manner. The AV aggregated frame format is used instead of the standard aggregation or low-latency aggregation formats.

Figure 10t illustrates the AV aggregated MAC frame body format.

MAC frame body				
Subframe 7	Subframe 6	…	Subframe 2	Subframe 1

Figure 10t—AV aggregated MAC frame body format

The collection of subframes in a single frame is referred to as the Frame Body field. The Frame Body field of a MAC frame may have one to seven, inclusive, subframes of varying sizes.

7.2.9.1 Extended MAC header format

The Extended MAC header is used to describe the contents of an AV aggregated frame, which typically is used to transport uncompressed audio and video.

The HRP Extended MAC Header shall be formatted as illustrated in Figure 10u.

octets: 16	24	5	5	2	10
Reserved	Video header	Security header	MAC extension header	Extended control header	MAC header

Figure 10u—HRP Extended MAC header format

The LRP Extended MAC Header shall be formatted as illustrated in Figure 10v.

octets: 5	5	2	10
Security header	MAC extension header	Extended control header	MAC header

Figure 10v—LRP extended MAC header format

The MAC extension header, Security header, and Video header are present in LRP frames only if their corresponding control bits are set in the Extended control header.

The Extended MAC header for HRP frames shall have the space for the MAC extension Header, Security header, and Video header. Therefore, the HRP MAC header always has a fixed size.

7.2.9.1.1 Extended Control header

The Extended Control header field shall be formatted as illustrated in Figure 10w.

bits: 7	3	1	1	1	3
Reserved	Frame type	Video header present	Security header present	MAC extension header present	Frame class

Figure 10w—Extended Control header

Valid values for the Frame Class field are given in Table 40a.

Table 40a—Frame class field values

Field value	Frame class
0b000	Regular
0b0001	AV aggregated
0b010	Omni-ACK
0b011	Beacon
0b100–0b111	Reserved

Regular frames have a single payload field without MAC level aggregation while AV aggregated frames have one or more subframes as a part of the frame.

The MAC Extension Header Present field shall be set to one if the MAC Extension Header is in the Extended MAC Header field and shall be set to zero otherwise.

The Security Header Present field shall be set to one if the Security header is in the Extended MAC Header field and shall be set to zero otherwise.

The Video Header Present field shall be set to one if the Video Header is in the Extended MAC Header field and shall be set to zero otherwise.

For HRP frames, the MAC Extension Header Present field, Security Header Present field and Video Header Present field indicate the validity of the associated field. Because the HRP header is a fixed length, space for these fields is always present in the Extended MAC Header field. If a header in the HRP MAC header is present but not valid, it may be set to any value and shall be ignored upon reception. For LRP frames, unused headers, as indicated by the appropriate header present bit, are not present in the MAC header.

The Frame Type field is only defined for regular frame class, it shall be set to zero in other frames. AV aggregated frames use the Type field in the MAC Extension Header field while the Omni-ACK and Beacon frames do not require the Frame Type field. Valid values for the Frame Type field are as follows:

- 0 → MAC commands
- 1 → Data
- 2 → Audio
- 3–7 → Reserved

7.2.9.1.2 MAC Extension header

The MAC Extension Header field is illustrated in Figure 10x.

bits: 8	**4**	**4**	**…**	**4**
ACK groups	Reserved	Type 7	…	Type 1

Figure 10x—MAC extension header format

The Type field indicates the type of data that is contained in the subframe. Valid values for the Type field are as follows:

- 0x0 → MAC commands
- 0x1 → Data
- 0x2 → Audio
- 0x3 → Video
- 0x4–0xF → Reserved

The ACK Groups field shall be formatted as illustrated in Figure 10y.

bits: 1	**1**	**…**	**1**	**1**
lsb FCS	Subframe 7	…	Subframe 1	Subframe 1

Figure 10y—ACK groups field format

The bit for a subframe shall be set to one if the subframe is in the same ACK group as the previous (i.e., lower numbered subframe). Otherwise, it is the first subframe in an ACK group and its bit shall be set to zero. The first bit, corresponding to subframe 1, shall always be set to zero as it is always the start an ACK group. No more than 5 ACK groups shall be defined, therefore, the number of bits set to zero among the subframe bits shall not exceed five.

The lsb FCS field shall be set to one if the lsb FCS is part of the calculation to determine if a subframe was correctly received, as defined in 8.1. It shall be set to zero if the lsb FCS is ignored in determining if a subframe was correctly received. The setting of this field applies only to those subframes that are sent with a UEP HRP mode. For all other subframes, the lsb FCS field shall be set to zero.

7.2.9.1.3 Security header

The Security header shall be formatted as illustrated in Figure 10z.

octets: 2	3
SFC	Security control

Figure 10z—Security header format

The Security Control field shall be formatted as illustrated in Figure 10aa.

bits: 2	…	2	2	8
Subframe 1 security	…	Subframe 7 security	Reserved	SECID

Figure 10aa—Security Control field format

The SECID field is used to identify the key set that is used to encrypt and/or authenticate the data in the frame, as defined in 7.2.7.2.

The Subframe Security field indicates the type of security that is applied to a subframe. Valid values are as follows:

— 0b00 → No security applied
— 0b01 → Encryption and integrity code
— 0b10–0b11 → Reserved

The SFC field is defined in 7.2.7.3. The SFC shall be incremented for each subframe in the frame, even if security is not applied to the subframe.

7.2.9.1.4 Video header

The Video Header shall be formatted as illustrated in Figure 10ab.

octets: 4	5	5	5	5
Reserved	Video control 4	Video control 3	Video control 2	Video control 1

Figure 10ab—Format of the Video header

Unless otherwise stated, the numbers for all of the fields begin with zero, e.g., the first video frame is number zero, then number one, etc. One of the first video subframes in the first frame sent in a stream should have the Video Frame Number field, the H-position field, and V-position field set to zero.

The Video Subheader is included as part of the MAC header because the error rate for the MAC header is lower than that of the subframes.

The Video Control field shall be formatted as illustrated in Figure 10ac.

bits: 4	3	1	16	16
Reserved	Video frame number	Interlaced field indication	H-position	V-position

Figure 10ac—Video control field format

The Interlaced Field Indication field shall be set to one if the video subframes carry pixels for the bottom field. It shall be set to zero if the video subframes carry pixels for the top field or if the video subframes carry pixels for non-interlaced video modes.

The Video Frame Number field contains a counter that keeps track of the video frame to which the pixels in the subframe belong. The video frame number is calculated follows:

— For progressive video, the Video Frame Number field shall be incremented sequentially. After reaching the max value of 0x7, the next value shall be zero. All frames belonging to the same video frame have identical Video Frame Number values.
— For interlaced video, the Video Frame Number field shall be incremented in a step of two. Thus, each video frame has two frame numbers. All frames belonging to the first field have even Video Frame Numbers and all frames belonging to the second field have odd Video Frame Numbers. For example, for the first uncompressed video frame, the frames belonging to the first field have a Video Frame Number set to zero, and the frames belonging to the second field have a Video Frame Number set to one. Therefore, the same video frame has two Video Frame Numbers.

The H-position field contains the horizontal position of the first pixel in the subframe where zero is on the left side of the screen.

The V-position field contains the vertical position of the first pixel in the subframe where zero corresponds to the top of the screen. For interlaced formats, the V-position of the lines range from 0-539 independent of status as even or odd frame number.

7.2.9.2 Subframe format

The subframes with Type Video in the MAC frame shall be formatted as illustrated in Figure 10ad.

octets: 8	variable
Combined FCS	Subframe payload

Figure 10ad—Subframe format

Subframes other than those with Type Video shall be formatted as illustrated in Figure 10ae.

octets: 4	variable
FCS	Subframe payload

Figure 10ae—LRP data subframe format

The Subframe Payload field includes an LLC/SNAP header as the first octets in the payload, as defined in A.1, followed by the data to be transmitted. If the subframe has an integrity code, as indicated by the

Subframe Security field, then Subframe Payload field shall be formatted as illustrated in Figure 10af, where the Payload Data field includes an LLC SNAP header as the first octets in the payload, as defined in A.1.

octets: 8	variable
Integrity code	Payload data

Figure 10af—Subframe payload with integrity code

The Integrity Code field is defined in 7.2.7.5.

The Combined FCS field shall be formatted as illustrated in Figure 10ag.

bits:4	4	4	…	4	4	4	4	…	4	4	4
msb FCS 4	lsb FCS 4	msb FCS 3	…	lsb FCS 2	msb FCS 1	lsb FCS 1	msb FCS 8	…	lsb FCS 6	msb FCS 5	lsb FCS 5

Figure 10ag—Combined FCS field format

The lsb FCS field contains the FCS, as defined in 7.2.7.6, calculated over the four lsbs of each octet in the Subframe Payload field. The lsb FCS field occupies the lsbs of the Combined FCS field.

The msb FCS field contains the FCS, as defined in 7.2.7.6, calculated over the four msbs of each, if present, and Subframe Payload field. The msb FCS field occupies the msbs of the Combined FCS field.

The FCS field contains the FCS, as defined in 7.2.7.6, calculated over Subframe Payload field.

Subframes with frame type other than data do not support aggregation or fragmentation.

Subframes with frame type data support the aggregation of MSDUs or the fragmentation of MSDUs in their payloads. The format of a Subframe Payload field for a subframe with Frame Type data is illustrated in Figure 10ah.

octets: variable	…	variable	variable
Sub-payload N	…	Sub-payload 2	Sub-payload 1

Figure 10ah—Subframe payload field with aggregation

Each sub-payload field shall be formatted as shown in Figure 10ai.

bits: variable	1	1	10	20
MSDU	First fragment	Last fragment	Sequence number	Length

Figure 10ai—Sub-payload field format

The Length field indicate the length of the MSDU field in octets.

The Sequence Number field is incremented for each fragment, regardless if it is of the same or different MSDU.

The fragment fields shall be set as indicated in Table 40b

Table 40b—Fragment field settings

Fragment type	Last fragment field	First fragment field
First fragment	0	1
Last fragment	1	0
Middle fragment	0	0
Complete MSDU	1	1

The MSDU field for subframes with frame type data shall be formatted as defined in 7.3.5.

7.2.9.3 AV aggregated

The AV aggregated frame shall be formatted as illustrated in Figure 10aj. The AV aggregated frame shall not be used for LRP frames.

octets: variable	…	variable	variable	66
Subframe N	…	Subframe 2	Subframe 1	Extended MAC header

Figure 10aj—AV aggregated frame format

The AV aggregated frame has one to seven, inclusive, subframes. The subframe format is defined in 7.2.9.2.

7.2.9.4 Regular

The HRP regular frame shall be formatted as illustrated in Figure 10ak.

variable	octets: 66
Subframe	Extended MAC header

Figure 10ak—HRP regular frame format

The LRP regular frame shall be formatted as illustrated in Figure 10al.

variable	octets: variable
Subframe	Extended MAC header

Figure 10al—LRP regular frame format

The subframe format is defined in 7.2.9.2.

7.2.9.5 Directional ACK

The Directional ACK frame format is defined in 12.4.3.8. The Directional ACK frame is used to acknowledge HRP frames and beam formed LRP frames. The ACK policy for a directional ACK shall be Imm-ACK. The directional ACK shall be only used with the AV PHY HRP frames

After 7.3.5, insert the following subclause as 7.3.6:

7.3.6 Sync frame

The Sync frame shall be formatted as illustrated in Figure 23a.

octets: 4	4	…	4	4	10
FCS	CTA block-n	…	CTA block-1	Synchronization parameters	MAC header

Figure 23a—Sync frame format

The Synchronization Parameters field shall be formatted as illustrated in Figure 23b.

octets: 2	2
Frame start time	Superframe duration

Figure 23b—Synchronization parameters field format

The Superframe Duration field indicates the duration of the current superframe, as described in 7.3.1.1.

The Frame Start Time field indicates the time stamp for the Sync frame, which is the start time of the preamble of a Sync frame, measured from the start of the superframe.

The CTA Block field shall be formatted as illustrated in Figure 23c.

octets: 2	2
CTA duration	CTA location

Figure 23c—CTA block field format

The CTA Location field indicates the start time of the allocation, measured from the start of the superframe, as described in 7.3.1.1.

The CTA Duration field specifies the duration of the CTA, as described in 7.3.1.1.

7.4 Information elements

Change and insert the following rows in Table 48 as shown. The other rows are not changed and are not shown.

Table 48—Information elements

Element ID hex value	Element	Subclause	Present in beacon
0x14	Synchronization	7.4.22	As needed
0x15	Transmit switched diversity	7.4.23	As needed
0x16	UEP specific	7.4.24	As needed
0x17	IFS	7.4.25	As needed
0x18	CTA relinquish duration	7.4.26	As needed

Table 48—Information elements *(continued)*

Element ID hex value	Element	Subclause	Present in beacon
0x19	Feedback	7.4.27	As needed
0x1A	Mapping	7.4.28	As needed
0x1B	BST clustering	7.4.29	As needed
0x1C	Pattern estimation and tracking (PET) clustering	7.4.30	As needed
0x1D	Beam PET	7.4.31	As needed
0x1E	HRS beam PET	7.4.32	As needed
0x1F	PET amplitude	7.4.33	As needed
0x20	PET phase	7.4.34	As needed
0x21	Sync frame frequency	7.4.35	As needed
0x22	Directional peer	7.4.36	As needed
0x~~14~~23–0xFF	Reserved		

7.4.6 Piconet parameter change

Change Table 49 as shown. The other rows are not changed and are not shown.

Table 49—Description of field contents for change type values

Change type	Interpretation	Field to decode	Description of field contents
4	CHANNEL	New channel index	The channel index of the PHY channel that the piconet will begin using the beacon that has the beacon number equal to the Change Beacon Number field. The mapping of the channel number is PHY dependent. For the 2.4 GHz PHY the mapping is defined in 11.2.3. For the SC PHY and HSI PHY the mapping is defined in 12.1.5. For the AV PHY the mapping is defined in 12.4.1.1.

7.4.11 Capability

After Figure 38, insert the following paragraphs and Figure 38a:

For the mmWave PHYs, the Capability IE shall be formatted as illustrated in Figure 38a.

octets: 13	1	1
Overall capabilities	Length	Element ID

Figure 38a—Capability information element format for mmWave PHYs

After Figure 39, insert the following paragraphs and Figure 39a:

For the mmWave PHYs, the overall Capabilities field shall be formatted as illustrated in Figure 39a.

octets: 3	6	4
Beam forming capabilities	DEV capabilities	PNC capabilities

Figure 39a—Overall capabilities field format for mmWave PHYs

After Figure 42, insert the following paragraphs and Figure 42a:

For the mmWave PHYs, the DEV Capabilities field shall be formatted as illustrated in Figure 42a.

bits: b7	b6	b5	b4	b3	b2	b1	b0
Supported MCSs							

bits: b15	b14	b13	b12	b11	b10	b9	b8
Imp-ACK	Dly-ACK	Listen to Multicast	Listen to Source	Always AWAKE	Preferred fragment size		

bits: b23	b22	b21	b20	b19	b18	b17	b16
Blk-ACK	DAMI capable	OOK capable	AV capable	HSI capable	SC capable	STP	CTA relinquish

bits: b31	b30	b29	b28	b27	b26	b25	b24
TSD support	HRP TX capable	HRP RX capable	UEP type			UEP capable	

bits: b39	b38	b37	b36	b35	b34	b33	b32
PCES capability	Pilot word capability	Supported aggregation		Supported IFS			

bits: b47	b46	b45	b44	b43	b42	b41	b40
Reserved							Sync frame capable

Figure 42a—DEV capabilities field format for mmWave PHYs

Change the fourteenth and fifteenth paragraphs as shown:

The Supported Data Rates <u>or Supported MCSs</u> field is a PHY dependent mapping that indicates the data rates that the DEV is capable of using. For the 2.4 GHz PHY<u>, this field is the Supported Data Rates field, it shall be formatted as illustrated in Figure 42, and</u> the mapping of a field value to a set of data rates is defined in Table 89. For the mmWave PHYs, this field is the Supported MCSs field, it shall be formatted as illustrated in Figure 42a, and the mapping of a field value to a set of MCSs is defined in 12.1.8.1.

The Preferred Fragment Size field is a PHY dependent mapping that indicates the maximum MAC frame size preferred to be received by the DEV when fragmentation is used. For the 2.4 GHz PHY, this field shall be formatted as illustrated in Figure 42 and the mapping of a field of value to a preferred fragment size is defined in Table 90. For the mmWave PHYs, this field shall be formatted as illustrated in Figure 42a and the mapping of a field value to a preferred fragment size is defined in 12.1.8.2.

After the last paragraph, insert the following paragraphs, tables, and figures.

For the mmWave PHYs, the SC Capable field shall be set to one if the DEV supports the SC PHY, as defined in 12.2. It shall be set to zero otherwise.

The HSI Capable field shall be set to one if the DEV supports the HSI PHY, as defined in 12.3, and shall be set to zero otherwise.

The AV Capable field shall be set to one if the DEV supports the AV PHY, as defined in 12.4, and shall be set to zero otherwise.

The OOK Capable field shall be set to one if the DEV supports the OOK mode, as defined in Annex D2, and shall be set to zero otherwise.

The DAMI Capable field shall be set to one if the DEV supports the DAMI mode, as defined in Annex D2, and shall be set to zero otherwise.

The Blk-ACK field shall be set to one if the DEV is capable of performing Blk-ACK as defined in 8.8.3b. Otherwise the field shall be set to zero.

The UEP Capable field indicates if the DEV supports UEP in the PHY. The valid values of the UEP Capable field are given in Table 49a.

Table 49a—UEP capable field values

Value	UEP support
0	No UEP
1	SC UEP
2	HSI UEP
3	AV UEP

The UEP Type field indicates the type of UEP that is supported based on the value of the UEP capable field. The valid values of the UEP Type field are given in Table 49b.

The HRP RX Capable field shall be set to one if the DEV supports HRP RX for the AV PHY, as defined in 12.4, and shall be set to zero otherwise.

The HRP TX Capable field shall be set to one if the DEV supports HRP TX for the AV PHY, as defined in 12.4, and shall be set to zero otherwise.

Table 49b—UEP type field values

Value	PHY UEP type of mmWave PHYs
0	No UEP support
1	UEP type 1 using different FECs (SC PHY)
2	UEP type 2 using different MCSs (SC PHY)
3	UEP type 3 using different MCSs (All PHYs)
4	UEP type 3 using skewed constellation (All PHYs)
5	UEP type 3 using different MCSs and skewed constellation (SC and HSI PHYs)
6	All UEP types (SC PHY)
7	Reserved

The TSD Support field shall be set to one if the DEV is capable of transmit switched diversity (TSD) as defined in 13.8, it shall be set to zero otherwise.

The Supported IFS field indicates the minimum value of the IFS supported by a DEV. The valid values of the field are given in Table 49c.

Table 49c—Supported IFS encoding

Field value	SIFS and MIFS duration
0	0.2 µs
1	0.4 µs
2	0.6 µs
3	0.8 µs
4	1.0 µs
5	2.0 µs
6	2.5 µs
7–15	Reserved

The minimum allowed IFS for either SIFS or MIFS may also be constrained by the PHY mode in use. A source DEV may use the shortest SIFS or MIFS supported by the destination DEV. The SIFS and MIFS used may be different.

The Supported Aggregation field indicates the type of aggregation that is supported by the DEV. The valid values for the Supported Aggregation field are as follows:

0 → No aggregation support
1 → Standard aggregation support
2 → Low-latency aggregation support
3 → Standard and low-latency aggregation support

The Pilot Word Capability field shall be set to one if the DEV supports a pilot word of length 8. It shall be set to zero otherwise.

The PCES Capability field shall be set to one if the DEV supports the use of PCES and shall be set to zero otherwise.

The Sync Frame Capable field shall be set to one if the DEV supports Sync frame transmission, as defined in 8.17, and shall be set to zero otherwise.

The Beam Forming Capabilities field shall be formatted as illustrated in Figure 42b.

bits:6	2	2	1	3	4	4	2
Reserved	Number RX quasi-omni directions	Number TX quasi-omni directions	PET	Antenna type	Number RX sectors	Number TX sectors	LQI type

Figure 42b—Beam forming capabilities field format

The LQI Type field indicate the type of LQI used in beam forming procedure. Valid values of the LQI type fields are

0 → RSSIr
1 → SNR
2→ SINR
3→ Reserved

The Number TX Sectors field indicates the number of TX sectors supported by the DEV.

The Number RX Sectors field indicates the number of RX sectors supported by the DEV.

The Antenna Type field indicate the supported antenna type of the DEV. Valid values of the Antenna Type field are as follows:

0 → No beam forming capability
1 → Beam forming antenna capable
2–7 → Reserved

The PET field shall be set to one if the DEV supports pattern estimation and tracking and shall be set to zero otherwise.

The Number TX Quasi-omni Directions field indicates the number of TX quasi-omni directions supported by the DEV.

The Number of RX Quasi-omni Directions field indicates the number of RX quasi-omni directions supported by the DEV.

Insert the following subclauses as 7.4.22 through 7.4.35:

7.4.22 Synchronization

The Synchronization IE shall be formatted as illustrated in Figure 48e. The Synchronization IE shall be supported by DEVs that support the SC PHY or HSI PHY.

octets: 5	5	1	4	1	1
Regular S-CAP info	Association S-CAP info	Section indications	Quasi-omni beacon info	Length	Element ID

Figure 48e—Synchronization information element format

The Quasi-omni Beacon Info field shall be formatted as illustrated in Figure 48f.

bits: 6	4	3	3	16
Reserved	PNC directional antenna capabilities	Number beacon frames	Beacon index	Beacon offset time

Figure 48f—Quasi-omni beacon info field format

The Beacon Offset Time field contains the time in microseconds that is the start of this beacon frame delayed from the start of the superframe.

The Beacon Index field indicates the index of the current beacon frame.

The Number Beacon Frames field indicates the number of beacon frames that will be sent as part of the quasi-omni beacon.

The PNC Directional Antenna Capabilities field shall be formatted as illustrated in Figure 48g.

bits: 1	1	1	1
BST support	Sectors only	PNC beam forming capable	Antenna symmetry

Figure 48g—PNC directional capabilities field format

The Antenna Symmetry field shall be set to one if the PNC has a symmetric antenna system (SAS) and shall be set to zero otherwise, i.e., the PNC has an asymmetric antenna system (AAS).

The PNC Beam Forming Capable bit shall be set to one if the PNC is capable of beam forming and shall be set to zero otherwise.

The Sectors Only field shall be set to one if the PNC supports only sectorized antenna and shall be set to zero otherwise.

The BST Support field shall be set to one if the PNC supports beam switching/steering and tracking, as described in Clause 13, and shall be set to zero otherwise.

The Section Indication field shall be formatted as illustrated in Figure 48h.

bits: 4	1	1	1	1
Reserved	Regular S-CAP present	Association S-CAP present	Sector training present	PNC quasi-omni tracking present

Figure 48h—Section indication field format

The PNC Quasi-Omni Tracking Present field shall be set to one if the PNC quasi-omni tracking section exists in the beacon and shall be set to zero otherwise.

The Sector Training Present field shall be set to one if the sector training section exists in the beacon and shall be set to zero otherwise.

The Association S-CAP Present field shall be set to one if association S-CAP sections exist in the CAP and shall be set to zero otherwise.

The Regular S-CAP Present field shall be set to one if regular S-CAP sections exist in the CAP and shall be set to zero otherwise.

The Association S-CAP info field shall be formatted as illustrated in Figure 48i.

bits: 2	2	2	2	16	16
Reserved	First PNC quasi-omni RX index	Number association S-CAPs in superframe	Total number association S-CAPs	S-CAP duration	Association S-CAP start time

Figure 48i—Association S-CAP info field format

The Association S-CAP Start Time field specifies the start time offset of the association S-CAPs from the start of the superframe in microseconds.

The S-CAP Duration field specifies the duration of the association S-CAP, as described in 8.6.6.2, in microseconds.

The Total Number of Association S-CAPs is the same as the total number of PNC quasi-omni RX directions.

The Number of Association S-CAPs in Superframe field specifies the number of association S-CAPs in the current superframe.

The First PNC Quasi-Omni RX Index field specifies the PNC’s quasi-omni RX index that will be used in the first association S-CAP in the current superframe.

The Regular S-CAP info field shall be formatted as illustrated in Figure 48j.

bits: 2	2	2	2	16	16
Reserved	First PNC quasi-omni RX index	Number regular S-CAPs in superframe	Total number regular S-CAPs	S-CAP duration	Regular S-CAP start time

Figure 48j—Regular S-CAP info field format

The Regular S-CAP Start Time field specifies the start time offset of the regular S-CAP from the start of the superframe in microseconds.

The S-CAP Duration field specifies the duration of the regular S-CAPs in microseconds.

The Total Number of Regular S-CAPs field is set to be the same as the total number of PNC quasi-omni RX directions.

The Number of Regular S-CAPs field in Superframe field specifies the number of regular S-CAPs in the current superframe.

The First PNC Quasi-Omni RX Index field specifies the PNC's quasi-omni RX index that will be used in the first regular S-CAP in the current superframe.

7.4.23 Transmit switched diversity

The Transmit Switched Diversity IE is used for information exchange in TSD procedure as described 13.8. The Transmit Switched Diversity IE shall be formatted as illustrated in Figure 48k.

octets: 1	1	1	1	1	1
Transmit direction index	TSD feedback period	Number of transmit directions	Mode	Length	Element ID

Figure 48k—Transmit switched diversity information element format

The Mode field is encoded as indicated in Table 49d.

Table 49d—Mode field encoding

Field value	Mode	Description
0	Announce	Announce TSD IE sent from PNC to DEV
1	Request to switch	DEV requests PNC to switch to the next transmit direction
2	Request to stay	DEV requests PNC to stay at the current transmit direction
3	Response	In response of a request
4-255	Reserved	

The Number Of Transmit Directions field indicates the number of TX directions supported by the PNC.

The TSD Feedback Period field indicates the time interval between the TSD IEs sent from the DEV to the PNC for feedback.

The Transmit Direction Index field indicates the index of the transmit direction of the PNC. A field value of zero indicates that the PNC shall select one of the transmit directions that was not used previously.

7.4.24 UEP specific

The UEP specific IE is used to indicate the separating position of msb and lsb data. For example, it is used to handle different components of a video signal, e.g., RGB, YCbCr, etc. This IE shall be only used in UEP type 2. The UEP specific IE shall be formatted as illustrated in Figure 48l.

octets: 4	**1**	**1**
msb/lsb separation	Length	Element ID

Figure 48l—UEP specific information element format

The msb/lsb Separation field shall be formatted as illustrated in Figure 48m.

octets: 1	**1**	**1**	**1**
Reserved	Component 3	Component 2	Component 1

Figure 48m—msb/lsb separation field format

The Component fields contain the information to indicate the separating position of msb and lsb data. Valid values of the Component field 1,2 and 3 are as follows:

0 → all bits are msb
1 → b_0 is lsb, b_1–b_7 are msb
2 → b_0–b_1 is lsb, b_2–b_7 are msb
3 → b_0–b_2 is lsb, b_3–b_7 are msb
4 → b_0–b_3 is lsb, b_4–b_7 are msb
5 → b_0–b_4 is lsb, b_5–b_7 are msb
6 → b_0–b_5 is lsb, b_6–b_7 are msb
7 → b_0–b_6 is lsb, b_7 is msb
8 → all bits are lsb
9–255 → Reserved

7.4.25 IFS

The IFS IE shall be formatted as illustrated in Figure 48n.

octets: 1	**1**	**1**
IFS value	Length	Element ID

Figure 48n—IFS information element format

The IFS Value field shall be formatted as illustrated in Figure 48o.

bits: 4	**4**
MIFS	SIFS

Figure 48o—IFS value field format

The mapping of a field value to a specific IFS value is defined in Table 49c.

7.4.26 CTA relinquish duration

The CTA Relinquish Duration IE shall be formatted as illustrated in Figure 48p.

octets: 2	1	1
CTA relinquish duration	Length	Element ID

Figure 48p—CTA relinquish duration information element format

The CTA Relinquish Duration field contains the time in microseconds for which the time in the CTA is relinquished.

7.4.27 Feedback

The Feedback IE shall be formatted as illustrated in Figure 48q.

bits: 4	4	3	1	6	4	6	4	8	8
Index of current feedback	Total number of feedbacks	Reserved	End of sector training	Second best sector/beam LQI	Second best sector/beam	Best sector/beam LQI	Best sector/beam	Length	Element ID

Figure 48q—Feedback information element format

The Best Sector field shall be set to the index of the best transmit sector for the originating DEV. In the SAS case, this also represents the index of the best receiver sectors for the target DEV.

The Best Sector LQI field shall be set to the value of LQI measured by the Target DEV for the best sector. The Best Sector LQI field is encoded in dB. The corresponding LQI values in the case of SNR/SINR and RSSIr are given in Table 49e and Table 49f. The LQI can be measured using the CES field in the preamble.

The Second Best Sectors field shall be set to the index of the second best transmit sector for the originating DEV. In the SAS case, this also represents the index of the second best receiver sectors for the target DEV. If the second best beam or sector is not supported, then this field shall be set to zero.

The Second Best Sector LQI field shall be set to the value of LQI measured by the Target DEV for the second best sector. The Second Best Sector LQI is encoded in the same manner as the Best Sector LQI. If the second best beam or sector is not supported, then this field shall be set to zero.

The encoding of LQI for SNR/SINR is defined in Table 49e. The SNR value is equal to E_c/N_0 (dB)+20 dB.

The encoding of LQI for RSSIr is defined in Table 49f where 0 dB indicates that the received signal level is equal to receiver sensitivity for the respective base rate.

The End of Sector Training field is used to indicate that a DEV wants to end the training stage. Any DEV who wants to terminate the training stage at the end of sector level training may set this field to one during sector level training period. However if one of DEVs, DEV1, sets the field to one but the other DEV, DEV2, does not, then DEV1 shall help DEV2 to finish beam level training by using DEV1's best transmit and receive sectors.

Table 49e—SNR/SINR value for LQI field

Field Value	SNR/SINR value
000000	≤ 0 dB
000001	0.5 dB
000010	1 dB
...	...
111110	31 dB
111111	> 31.5 dB

Table 49f—RSSIr value for LQI field

Field Value	RSSIr value
000000	≤ 0 dB
000001	1 dB
000010	2 dB
...	...
111110	62 dB
111111	> 63 dB

The Total Number of Feedbacks field shall be set to the number of Announce commands that will be sent as part of the feedback process.

The Index of Current Feedback field shall be set to number of the current Feedback IE in the series that is being sent. The first Feedback IE that is sent shall have the Index of the Current Feedback field set to zero.

7.4.28 Mapping

The Mapping IE shall be formatted as illustrated in Figure 48r.

bits: 2	4	4	6	2	6	8	8
Reserved	Second best cluster tracking period	Best cluster tracking period	Number of RX beams	Sync mode	Number TX beams	Length	Element ID

Figure 48r—Mapping information element format

When sent at the end of sector level training, the Number of TX Beams field and Number of RX Beams field shall be set to one less than the number of beams that the sending device will be using during the beam level training. When sent at the end of the beam level training, these fields shall be set to the number of HRS beams to be used during tracking.

If the system is SAS, the Number of RX Beams field shall be set to zero.

The sync mode field shall encode the length of the sync sequence to be used for subsequent training sequences. The value of the field shall be the same as that used in the PHY header, as described in 12.2.3.1.1 and 12.3.3.2.

The Best Cluster Tracking Period and Second Best Cluster Tracking Period fields shall be set to the tracking period of the best cluster and the second best cluster, respectively. The Tracking Period fields shall be encoded as defined in Table 49g.

Table 49g—Tracking period encoding

Filed Value	Tracking period value
0000	Reserved
0001	0.004 ms
0010	0.008 ms
0011	0.016 ms
0100	0.032 ms
0101	0.064 ms
0110	0.128 ms
0111	0.256 ms
1000	0.512 ms
1001	1.024 ms
1010	2.048 ms
1011	4.096 ms
1100	8.192 ms
1101	16.384 ms
1110	32.768 ms
1111	Reserved

7.4.29 BST clustering

The BST Clustering IE shall be formatted as illustrated in Figure 48s.

octets: variable	variable	1	1
BST cluster mapping (RX)	BST cluster mapping (TX)	Length	Element ID

Figure 48s—BST Clustering information element format

The BST Clustering IE is used for AAS sector and beam level training as defined in 13.5.1.1.1 and 13.5.1.2.1, respectively. It is also used for SAS sector and beam level training, as defined in 13.5.1.1.2 and 13.5.1.2.2, respectively.

The BST Cluster Mapping field shall be formatted as illustrated in Figure 48t. In the case of a SAS system, the TX and RX clusters are identical, and only one BST Cluster Mapping field shall be included in the IE.

bits: 4/0	4	…	4	4	4
Reserved	Number of beams in cluster *N*	…	Number of beams in cluster 1	Number of beams in cluster 0	Number of clusters

Figure 48t—BST cluster mapping field format

The Number of Clusters field contains one less than the number of TX or RX clusters, C_T and C_R, respectively, as defined in 13.2.4.

The Number of Beams in Cluster field shall be set to one less than the number of beams that make up that cluster.

The Reserved field shall be included if there is an even number of clusters and shall be omitted otherwise.

7.4.30 Pattern estimation and tracking (PET) clustering

The PET Clustering IE shall be formatted is illustrated in Figure 48u.

octets: variable	variable	1	1
Cluster mapping (RX)	Cluster mapping (TX)	Length	Element ID

Figure 48u—PET clustering information element

The PET Clustering IE is used for AAS sector and beam level training as defined in 13.5.1.1.1 and 13.5.1.2.1, respectively. It is also used for SAS sector and beam level training, as defined in 13.5.1.1.2 and 13.5.1.2.2, respectively.

The Cluster Mapping fields shall be formatted as illustrated in Figure 48v. In the case of a SAS system, the transmit and receive clusters are identical, and only one Cluster Mapping field shall be included in the IE.

octets: 2	…	2	2	1
Cluster *N* descriptor	…	Cluster 1 descriptor	Cluster 0 descriptor	Number of clusters

Figure 48v—PET cluster mapping field format

The Number of Clusters field shall be set to one less than the number of TX or RX clusters, C_T and C_R, respectively, as defined in 13.2.4.

The Cluster Descriptor fields shall be for the TX or RX arrays as identified by the position of the field in the PET Clustering IE. The Cluster Descriptor field shall be formatted as illustrated in Figure 48w.

octets: 1	1
Cluster encoding	Center beam index

Figure 48w—Cluster descriptor field format

The Cluster Encoding field describes the geometry of the cluster and shall be as described in 13.2.4.

The Center Beam Index field shall be the index of the beam or HRS beam around which the cluster is formed.

7.4.31 Beam PET

The Beam PET IE shall be formatted as illustrated in Figure 48x.

octets: 4	4	1	1
PET configuration (RX)	PET configuration (TX)	Length	Element ID

Figure 48x—Beam PET Information element format

For the SAS case, the transmit and receive PET configurations are the same, so the PET Configuration (RX) field shall be omitted.

The PET Configuration field for RX or TX shall be formatted as shown in Figure 48y.

bits: 2	3	3	8	8	4	4
Reserved	Phase resolution	Amplitude resolution	Beam codebook ID, *x*-axis	Beam codebook ID, *z*-axis	Number of antennas, *x*-axis	Number of antennas, *z*-axis

Figure 48y—PET Configuration field format

The Number of Antennas, *x*-axis and Number of Antennas, *z*-axis fields shall be set to one less than the number of antennas along the *x*-axis and *z*-axis, respectively, as described in 13.2.3. The x-axis and z-axis are only applied here as an example to define an antenna pattern for a linear two-dimensional antenna array, as describe in 13.2.3. The reference to x-axis and z-axis are changeable according to the implementation requirement. These values shall be for the TX or RX arrays as identified by the position of the field in the PET Clustering IE.

The Beam Codebook ID fields shall identify the codebooks, as described in 13.3.1, to be used for the respective axes.

The Amplitude Resolution field shall indicate the number of discrete values for amplitude that can result from pattern estimation. The field shall be coded as N, where there are 2^N possible resulting amplitude values. The value of this field shall be limited to the range 0 to 4.

Similarly, the Phase Resolution field shall indicate the number of discrete phase values that can result from pattern estimation. This field shall also be coded as N, where there are 2^N possible resulting phase values. The value of this field shall be limited to the range 0 to 4.

7.4.32 HRS beam PET

The HRS Beam PET IE shall be formatted as illustrated in Figure 48z.

octets: 2	2	1	1
PET HRS configuration (RX)	PET HRS configuration (TX)	Length	Element ID

Figure 48z—HRS beam PET information element

For the SAS case, the PET HRS Configuration fields for RX and TX are the same, so the PET HRS Configuration (RX) field shall be omitted.

Each PET HRS configuration field shall be formatted as illustrated in Figure 48aa.

octets: 1	1
HRS beam codebook ID, *x*-axis	HRS beam codebook ID, *z*-axis

Figure 48aa—PET HRS configuration field format

The HRS beam codebook ID fields shall identify the codebooks, as described in 13.3.1, to be used for their respective axes.

7.4.33 PET amplitude

The PET Amplitude IE shall be formatted as illustrated in Figure 48ab.

bits: 4 or 0	4	…	4	4	8	8
Reserved	Element *M* amplitude	…	Element 1 amplitude	Element 0 amplitude	Length	Element ID

Figure 48ab—PET amplitude information element format

The Element Amplitude fields are sent in the predefined antenna element order, as defined in 13.2.3. If F represents the amplitude relative to the element with the highest amplitude, the value of the Element Amplitude field shall be the one less than the numerator, N, of a fraction $N/D = F$, where D is the number of possible values as specified by the Amplitude Resolution field of the Beam PET IE. For example, if the Amplitude Resolution field value is 3, then there are $2^3 = 8$ possible values and $D = 8$. In this case, one of the antenna element amplitudes will be $N = 8$, and the others will have values in the range of $N = 1$ to $N = 8$. For this case, one Element Amplitude value shall be seven while the others will have values in the range zero to seven.

The Reserved field shall be present if there is an odd number of antenna elements, it shall be omitted otherwise.

7.4.34 PET phase

The PET Phase IE shall be formatted as illustrated in Figure 48ac.

bits: 4 or 0	4	…	4	4	8	8
Reserved	Element *M* phase	…	Element 1 phase	Element 0 phase	Length	Element ID

Figure 48ac—PET phase information element format

The Element Phase fields are sent in the predefined antenna element order, as defined in 13.2.3. If F represents the phase as a fraction of a full circle, the Element Phase field shall set to the numerator, N, of a fraction $N/D = F$, where D is the number of possible values as specified by the Phase Resolution field of the Beam PET IE. For example, if the phase resolution field value is 4, then there are $2^4 = 16$ possible values and $D = 16$. In this case, each lsb of the phase value is equal to 22.5°, and the values shall be in the range of $N = 0$ to $N = 15$, which corresponds to phases of 0° to 342.5°.

The Reserved field shall be present if there is an odd number of antenna elements, it shall be omitted otherwise.

7.4.35 Sync frame frequency

The Sync Frame Frequency IE shall be formatted as illustrated in Figure 48ad.

bits: 2	1	5	8	8
Reserved	Sync frame direction	Sync frame frequency	Length	Element ID

Figure 48ad—Sync frame frequency information element format

The Sync Frame TX Frequency field is set to the number of superframes between transmission of two Sync frames, as defined in 7.3.6, requested by the PNC. If the Sync Frame TX Frequency field is set to zero, then the PNC is requesting the DEV cease sending Sync frames.

The Sync Frame Direction field shall be set to zero if the PNC is requesting omni-directional transmission of the Sync frame and shall be set to one if the PNC is requesting directional transmission of the Sync frame. If the request is for directional transmission, then the Sync frame direction is a round robin of the DEV's available directions.

7.4.36 Directional peer

The Directional Peer IE shall be formatted as illustrated in Figure 48ae.

Octets: 2	1	1	1	8	8
Allocated superframes	Configuration	DestID	SrcID	Length	Element ID

Figure 48ae—Directional peer information element format

The SrcID field contains the DEVID of the source for the directional peer communication in the regular CAP and/or regular S-CAP.

The DestID field contains the DEVID of the destination for the directional peer communication in the regular CAP and/or regular S-CAP.

The Configuration field shall be formatted as illustrated in Figure 48af.

bits: 6	1	1
Reserved	Unidirectional/Bidirectional allowance	Request/Release

Figure 48af—configuration field format

The Request/Release bit shall be set to one to request directional communication and shall be set to zero to release the directional communication.

The Unidirectional/Bidirectional Allowance bit shall be set to one for unidirectional communication and shall be set to zero otherwise.

The Allocated Superframes field indicates the number of superframes allocated for directional communication between the source and destination.

7.5 MAC command types

7.5.1 Association and disassociation commands

7.5.1.1 Association request

Replace Figure 49 with the following:

octets: 1	2	As defined in 7.4.11	8	2	2
DEV utility	ATP	Overall capabilities	DEV address	Length	Command type

Figure 49—Association request command format

Replace Figure 50 with the following:

bits: b7-b5	b4-b2	b1	b0
Reserved	Best PNC TX quasi-omni pattern	Neighbor PNC	Piconet services inquiry

Figure 50—DEV utility field format

Insert the following paragraphs after the seventh paragraph in 7.5.1.1:

The Best PNC TX Quasi-Omni Pattern field contains the PNC TX quasi-omni pattern, as determined by the DEV from receiving beacons, that the PNC should use for further quasi-omni transmissions to the DEV.

If the PNC uses AAS, as determined from the received beacon, the DEV may set the Response TX sector field to the index of the beacon that was initially heard by the DEV. It shall be set to zero otherwise.

7.5.4 Information request commands

7.5.4.2 PNC information command

Replace Figure 62 with the following:

octets: 1	As defined in 7.4.11	1	1	6
System wake beacon interval	Capability	Membership status	DEVID	DEV address

Figure 62—Format of a DEV info field in a PNC information command

7.5.6 Channel time allocation request, modification, and termination commands

7.5.6.1 Channel time request

Insert the following sentence to the end of the seventh paragraph in 7.5.6.1 as shown.

The Stream Index field is defined in 7.2.5. In the case where the DEV is requesting the creation of an isochronous stream, it is set to the unassigned stream value, as described in 7.2.5, by the originating DEV. In the case where the DEV is requesting the reservation or termination of an asynchronous channel time, it is set to the asynchronous stream value, as described in 7.2.5. When the stream index is other than the unassigned stream index or asynchronous stream index value, this CTRq is a request to modify or terminate an existing CTA. In the case where the DEV is requesting a specific MCTA interval, as described in 8.4.3.3, the stream index shall be set to the MCTA stream value, as described in 7.2.5. In the case where the DEV is requesting an allocation for beam forming, the stream index shall be set to beam forming stream value as described in 7.2.5.

8. MAC functional description

8.1 Introduction

Insert the following paragraph after the fifth paragraph in 8.1:

The aggregation is described in 8.7a. Channel probing is described in 8.9.7. UEP is described in 8.16.

8.2 Starting, maintaining, and stopping piconets

8.2.1 Scanning through channels

Change first paragraph in 8.2.1 as shown:

All DEVs shall use passive scanning to detect an active piconet. That is, DEVs shall be in receive mode for a period of time in a channel no less than mMinChannelScan~~, as specified in the MLME-SCAN.request,~~ to look for beacon frames or, if supported, sync frames from a PNC. If a particular BSID, PNID, or PNC address to scan for is not specified with an MLME-SCAN.request, ~~open scan is specified in the MLME-SCAN.request,~~ the DEV searches for any beacon frame or, if supported, a sync frame. If a particular BSID, PNID, or PNC address to scan for is ~~open scan is not~~ specified, the DEV shall ignore all received frames not matching the ~~PNID and BSID~~ parameter or parameters contained in the request.

Change the fourth paragraph in 8.2.1 as shown:

While searching, if any frame type other than a beacon frame is received, the searching DEV shall stay in the channel for a minimum of mMinChannelScan from the time of reception of first frame and look for a beacon from the PNC. The DEV shall scan all indicated channels to find piconets before returning the scan information via the MLME-SCAN.confirm primitive. The DEV shall only report piconets found due to the reception of a beacon frame or, if supported, a sync frame as a part of the MLME-SCAN.confirm primitive.

8.2.3 PNC handover

Change the first paragraph in 8.2.3 as shown:

When the PNC leaves the piconet or when it transfers its PNC functionality to another DEV, the PNC may ~~shall attempt to~~ choose a DEV that is capable of being a PNC as its successor. PNC handover is optional for a PNC using a mmWave PHY. The PNC Capable bit in the PNC Capabilities field, as described in 7.4.11, is used to indicate that a DEV is capable of being a PNC. The PNC shall use the information in the PNC Capabilities field of the other DEVs in the piconet with the evaluation criteria defined in Table 55 to select the most qualified PNC capable DEV that is currently a member of the piconet to be the new PNC. The PNC shall send a PNC Handover Request command, as described in 7.5.3.1, to its chosen DEV with the parameters specified in 7.5.3.1. If the piconet is not a dependent piconet, the DEV shall accept the nomination and be prepared to receive the piconet information records. If the DEV is currently the PNC of a dependent piconet, it may refuse the request by sending a PNC Handover Response command to the PNC with the Reason Code field set to 'Handover refused, unable to act as PNC for more than one piconet'. If both the current and the new PNC are members of the same dependent piconet, then the DEV shall accept the handover request unless it is unable to join the parent piconet as either a regular DEV or a neighbor PNC.

In the case where the DEV is unable to join the parent piconet, the DEV sends the PNC Handover Response command to the PNC with the Reason Code field set to 'Handover refused, unable to join parent piconet'.

8.4 Channel access

8.4.1 Interframe space (IFS)

Change the first paragraph in 8.4.1 as shown:

There are four IFSs that are defined; the minimum interframe space (MIFS), the short interframe space (SIFS), the backoff interframe space (BIFS) and the retransmission interframe space (RIFS). The actual values of the MIFS, SIFS, BIFS, and RIFS are PHY dependent. For the 2.4 GHz PHY, they are listed in 11.2.7.1. <u>For the SC PHY, they are listed in 12.2.6. For the HSI PHY, they are listed in 12.3.5.5. For the AV PHY, they are listed in 12.4.1.2.</u>

8.4.2 Carrier sense multiple access with collision avoidance (CSMA/CA)

Change the third item in the dashed list in 8.4.2 as shown:

— pBackoffSlot: 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 11.2.7.1. <u>For the SC PHY, this is defined in 12.2.6.1. For the HSI PHY, this is defined in 12.3.5.4. For the AV PHY this is defined in 12.4.1.2.</u>

Insert one sentence at the end of 8.4.2:

For DEVs supporting mmWave PHY, Blk-ACK is allowed in a CP. In a CP, the frame sent in response to a Blk-ACK request shall have a zero length MAC frame body. The directional use of CP is defined in 8.6.6.2.

8.4.3 Channel time allocation period channel access

8.4.3.7 Calculating channel time requests

Insert the following sentence to the end fifth paragraph:

The calculation method in Figure 113 is preferred for better CTA efficiency.

8.5 Channel time management

8.5.1 Isochronous stream management

8.5.1.1 Isochronous stream creation

Insert the following sentences after the seventh paragraph in 8.5.1.1 as shown.

If the stream index is set to beam forming stream value then the request is for a CTA for beam forming. If the PNC grants this request, it shall assign the stream index value to be the beam forming stream index.

8.6 Synchronization

8.6.2 Beacon generation

Change the first paragraph in 8.6.2 as shown:

The PNC shall send a beacon at the beginning of each superframe using the beacon frame as described in 7.3.1. The PNC may transmit quasi-omni beacons, as described in 8.6.6.1, between the PNC and a DEV and to enable DEVs located in different directional antenna coverage areas to join the same piconet if the PHY does not require support of omni-directional modes.

After 8.6.5, insert the following new subclause:

8.6.6 Superframe support for directional PHYs

If a PHY does not support omni-directional modes of operation or if it allows DEVs that do not support omni-directional modes of operation, then the superframe, beacon and CAP need additional features to enable theses DEVs to operate. In particular, the superframe structure for DEVs using these PHY modes will be different, as illustrated in Figure 125a.

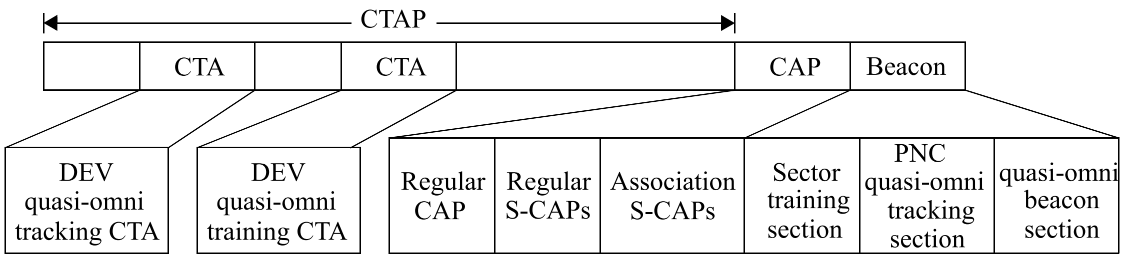

Figure 125a—Superframe structure for quasi-omni capable PHYs

This subclause defines optional procedures to enable DEVs to associate and operate with PHYs that do not have omni-directional modes. A summary of the types of CTAs used in the standard is given in D1.1.1.

8.6.6.1 Quasi-omni beacon

The quasi-omni beacon shall be structured as illustrated in Figure 125b.

Quasi-omni beacon		
Sector training section ($J^{(1,t)}$ STs)	PNC quasi-omni tracking section ($I^{(1,t)}$ QTs)	Quasi-omni beacon section ($I^{(1,t)}$ frames)

Figure 125b—Beacon structure for quasi-omni capable PHYs

The quasi-omni beacon section shall be formatted as illustrated in Figure 125c.

BSIFS	Beacon frame TX direction # $I^{(1,t)}$–1, $Q^{(1,t)}_{I^{(1,t)}-1}$	…	BSIFS	Beacon frame TX direction # $i^{(1,t)}$, $Q^{(1,t)}_{i^{(1,t)}}$	…	BSIFS	Beacon frame TX direction # 0, $Q^{(1,t)}_{0}$

Figure 125c—Quasi-omni beacon section structure

The PNC quasi-omni tracking section shall be formatted as illustrated in Figure 125d.

BSIFS	QT # $I^{(1,t)}$–1, $Q^{(1,t)}_{I^{(1,t)}-1}$	…	BSIFS	QT # $i^{(1,t)}$, $Q^{(1,t)}_{i^{(1,t)}}$	…	BSIFS	QT # 0, $Q^{(1,t)}_{0}$

Figure 125d—PNC quasi-omni tracking section structure

The sector training section shall be formatted as illustrated in Figure 125e.

BSIFS	ST # $J^{(1,t)}-1$, $S^{(1,t)}_{J^{(1,t)}-1}$	…	BSIFS	ST # $j^{(1,t)}$, $S^{(1,t)}_{j^{(1,t)}}$	…	BSIFS	ST # 0, $S^{(1,t)}_0$

Figure 125e—Sector training section structure

Quasi-omni coverage during beaconing shall be supported by repeating the beacon frame $I^{(1,t)}$ times and sending each repetition with a different TX antenna, quasi-omni direction, or array pattern. Each repetition is followed by a BSIFS, as defined in 13.1. The $I^{(1,t)}$ antenna directions or array patterns shall be identified by indices zero through $I^{(1,t)}$ –1. The number of beacon frame repetitions $I^{(1,t)}$ and the index of the current beacon transmit direction are parameters in the Synchronization IE, as described in 7.4.22. In the following, the term *direction* shall be used to refer to an antenna direction or an array pattern.

When the PNC Quasi-omni Tracking Present field in the Synchronization IE is set to one, then a PNC quasi-omni tracking section shall follow the quasi-omni beacon section to allow devices in the piconet to track the PNC quasi-omni directions. The PNC quasi-omni tracking section consists of $I^{(1,t)}$ quasi-omni training (QT) sequences transmitted in the $I^{(1,t)}$ PNC quasi-omni directions. The $I^{(1,t)}$ QT sequences may be transmitted at once in a superframe or distributed over multiple superframes. The QT sequence shall be identical to the long preamble.

When the Sector Training field in the Synchronization IE is set to one, then a sector training section shall follow the PNC quasi-omni tracking section to enable pro-active beam forming, as described in 13.7. The sector training section consists of $J^{(1,t)}$ sector training (ST) sequences transmitted in the $J^{(1,t)}$ PNC sector directions. The $J^{(1,t)}$ ST sequences may be transmitted once in a superframe or distributed over multiple superframes. The ST sequences shall be identical to the long preamble.

8.6.6.2 Directional CAP

The CAP period for directional communication may be divided into three sections, an association section, a regular S-CAP section, and a regular CAP section as shown in Figure 125f. If PNC allows new association requests, an association section exists at the beginning of the CAP period. The association section may be further divided into a set of $I^{(1,r1)}$ equal size association sub CAPs (S-CAPs) corresponding to some of the $I^{(1,r)}$ PNC different quasi-omni receive directions within one superframe or distributed over multiple superframes ($r1 \le r$). The regular S-CAP section is divided into the $I^{(1,r1)}$ equal size S-CAPs corresponding to some of the $I^{(1,r)}$ PNC different quasi-omni receive directions within one superframe or distributed over multiple superframes. Each S-CAP is received by the PNC using a different antenna receive direction.

<table>
<tr><th colspan="9">CAP</th></tr>
<tr><td rowspan="2">Regular CAP</td><td colspan="4">Regular S-CAP</td><td colspan="4">Association S-CAP</td></tr>
<tr><td>S-CAP #$I^{(1,r1)}$–1</td><td>…</td><td>S-CAP #1</td><td>S-CAP #0</td><td>Association S-CAP #$I^{(1,r1)}$–1</td><td>…</td><td>Association S-CAP #1</td><td>Association S-CAP #0</td></tr>
</table>

Figure 125f—Quasi-omni CAP structure

The association CAP shall be used solely for devices to send Association Request commands to the PNC and for the Imm-ACK from the PNC to the Association Request command. The regular CAP and regular S-CAP may be used for all other command and data exchanges. It is up to the implementor to determine the method used to achieve omni-directional communications for the SC and HSI PHY modes in the regular CAP.

If the AAS field in the piconet synchronization parameters field is set to zero, indicating that the PNC is SAS, the number of S-CAPs shall be one-to-one with the number of beacons ($I^{(1,t)} = I^{(1,r)}$). That is, the PNC transmit direction used for the i^{th} (i = 0, 1, ,2, …, $I^{(1,r)} - 1$) beacon frame transmission shall be used for the i^{th} S-CAP respectively.

If the AAS field in the piconet synchronization parameters field is set to one, indicating that the PNC is AAS, the number of S-CAPs shall be equal to $I^{(1,r)}$, which is specified in the Synchronization IE, as described in 7.4.22. The special case where $I^{(1,r)} = 1$ indicates that the PNC is omni capable on reception.

Before two peer DEVs communicate in the regular CAP and/or regular S-CAPs, the two DEVs may perform beam forming, as described in 13.5, if they do not know the antenna directions to point to each other.

To support directional peer communication in the regular CAP and/or regular S-CAPs, the source DEV may send the PNC an Announce command with Directional Peer IE with the Request/Release bit set to one, as described in 7.4.36. If the PNC allows the directional peer communication, the PNC shall include the Directional Peer IE in the beacon to announce that the two devices will use regular CAP and/or regular S-CAPs for directional peer communication.

After the source and destination DEVs receive the Directional Peer IE in the beacon from the PNC, the two DEVs may switch their antenna directions and communicate with each other in the regular CAP and/or regular S-CAPs. After the two devices complete directional peer communication, the source device may send the PNC an Announce command with Directional Peer IE with the Request/Release bit set to zero, and the PNC shall remove Directional Peer IE from its beacon.

Directional peer-to-peer communication between two non-PNC DEVs is not bound by the S-CAP boundaries.

8.6.6.3 Directional association

A DEV that is not omni capable on reception and supports multiple receive directions shall implement directional association. Let $I^{(1,t)}$ and $I^{(1,r)}$ be the number of PNC quasi-omni transmit and receive directions respectively, and let $I^{(2,t)}$ and $I^{(2,r)}$ be the number of DEV quasi-omni transmit and receive directions, respectively, of a DEV that wants to find a PNC and associate with that PNC.

The best and second best pair of PNC transmit and DEV receive or DEV transmit and PNC receive directions are referred as the best antenna direction pair and second best antenna direction pair, respectively

While searching for a PNC, a DEV shall listen to quasi-omni beacons at all $I^{(2,r)}$ quasi-omni receive directions to find the best and second best antenna direction pairs based on LQI measurement. The DEV shall use the best DEV quasi-omni receive direction to receive further transmissions from the PNC when quasi-omni transmission is used. The DEV shall include the information of the best PNC quasi-omni transmit direction in its Association Request commands, as defined in 7.5.1.1, to inform the PNC the best PNC quasi-omni transmit direction for further quasi-omni transmissions to the DEV.

The DEV shall track the best and second best antenna direction pairs during the quasi-omni beacon section and PNC quasi-omni tracking section of beacons based on LQI measurement. If the beacon quality in the second best antenna direction pair is better than the best antenna direction pair, the PNC and the DEV shall switch to the second best antenna direction pair, which becomes the new best antenna direction pair.

The association procedure of a DEV depends on the antenna types at both the PNC and the DEV. The DEV shall transmit one or multiple Association Request commands, as described in 7.5.1.1. If both the DEV itself and the PNC utilize SAS antennas, this command shall be transmitted on DEV antenna quasi-omni direction $I^{(2,t)} = I^{(2,r)}$, during the S-CAP with index, $I^{(1,t)}$.

If either the DEV or the PNC is AAS, the DEV does not know which quasi-omni transmit direction to use, nor which S-CAP is the best. In this case, the DEV shall transmit the Association Request command at different antenna direction pairs during association S-CAPs in one or multiple superframes until it receives an Association Response command successfully or association timeout. The Association Request command shall be sent individually instead of back to back, with random backoff applied to each one. The backoff window used by the DEV to send the Association Request command shall be the same for the cycle of $I^{(2,t)}$ quasi-omni transmit directions. If a complete cycle of quasi-omni transmit direction fails, then the backoff window shall be increased as defined in 8.4.2. The Association Response command shall include the information of the antenna direction pair from which it receives the Association Request command. The Association Response command may be sent in a CTA.

Once an Association Response command or an Imm-ACK to the Association Request command is received successfully, as defined in 8.3.1, the DEV shall cease the transmission of Association Request command in association S-CAPs and ignore other copies of the same Association Response command from PNC. The DEV shall use the regular S-CAP of the PNC quasi-omni receive direction carried in the Association Response command for all further CAP transactions with the PNC before the completion of the best antenna direction pair searching. In addition, the DEV shall use the DEV transmit quasi-omni direction carried in the Association Response command before the completion of the best antenna direction pair searching.

8.6.6.4 DEV quasi-omni transmit direction training and tracking

Since the antenna direction pair found at the association stage may not be the best pair due to possible collisions in CAP, the PNC shall reserve a CTA to search for the best and the second best antenna direction pairs when channel has free time. The CTA structure for DEV quasi-omni transmit direction training is illustrated in Figure 125g.

CTA of DEV quasi-omni transmit direction training					
SIFS	DEV –› PNC Imm-ACK $Q_{i^{(2,t)}}^{(2,t)} \rightarrow Q_{i^{(1,r)}}^{(1,r)}$	SIFS	PNC –› DEV Announce command with Feedback IE $Q_{i^{(1,t)}}^{(1,t)} \rightarrow Q_{i^{(2,r)}}^{(2,r)}$	SIFS	Quasi-omni training DEV –› PNC $I^{(2,t)}$ cycles

Figure 125g—CTA structure for DEV quasi-omni transmit direction training

The structure of the quasi-omni training from DEV to PNC is illustrated in Figure 125h.

Quasi-omni training cycle #$I^{(2,t)}-1$, $I^{(2,t)}$ cycles	…	Quasi-omni training cycle #$i^{(2,t)}$, $I^{(2,t)}$ cycles	…	Quasi-omni training cycle #0, $I^{(2,r)}$ cycles

Figure 125h—Quasi-omni training DEV –› PNC structure

The structure of the quasi-omni training cycle is illustrated in Figure 125g.

SIFS	QT #$I^{(2,t)}-1$ $Q_{i^{(2,t)}}^{(2,t)} \rightarrow Q_{i^{(1,r)}-1}^{(1,r)}$	…	QT #1 $Q_{i^{(2,t)}}^{(2,t)} \rightarrow Q_1^{(1,r)}$	QT #0 $Q_{i^{(2,t)}}^{(2,t)} \rightarrow Q_0^{(1,r)}$

Figure 125i—Quasi-omni training cycle structure

The quasi-omni training consists of $I^{(2,t)}$ cycles. During each cycle, the DEV shall send $I^{(1,r)}$ repetitions of a QT sequence, as defined in 8.6.6.1, in the same direction. Each cycle except the last one shall end with a BSIFS. The $I^{(2,t)}$ cycles shall be sent in $I^{(2,t)}$ different directions, namely $[Q_0^{(2,t)}, Q_1^{(2,t)}, \ldots, Q_{I^{(2,t)}-1}^{(2,t)}]$.

During a cycle, the PNC shall attempt to receive each of the $I^{(1,r)}$ quasi-omni training sequences using a different direction. The $I^{(1,r)}$ different directions, $[Q_0^{(1,r)}, Q_1^{(1,r)}, \ldots, Q_{I^{(1,r)}-1}^{(1,r)}]$, during a cycle shall correspond to the PNC's quasi-omni receive directions.

At the completion of the full $I^{(2,t)}$ cycles, the PNC will have had an opportunity to receive a QT sequence using each combination of DEV2 transmit quasi-omni direction (0 to $I^{(2,t)} - 1$) and PNC receive quasi-omni direction (0 to $I^{(1,r)} - 1$). Based on this information, the PNC selects the best quasi-omni pair, i.e., DEV2's optimal transmit quasi-omni direction, $I_{i^{(2,t)}}^{(2,t)}$, and the PNC optimal transmit and receive sector, $I_{i^{(1,r)}}^{(1,r)}$.

Following the quasi-omni training, the PNC shall transmit its quasi-omni feedback in a Feedback IE by sending an Announce command with Imm-ACK requested. The Announce command shall be sent in the optimal transmit quasi-omni direction, $Q_{i^{(2,t)}}^{(1,t)}$, and DEV shall listen on its optimal receive direction, $Q_{i^{(2,r)}}^{(1,r)}$. The Feedback IE informs DEV of its optimal transmit quasi-omni direction, $Q_{i^{(2,t)}}^{(2,t)}$, second best quasi-omni direction, and the corresponding LQIs.

The PNC shall track the best and second best antenna direction pairs based on LQI measurement by allocating a tracking CTA periodically when channel has free time. The CTA structure for DEV quasi-omni transmit direction tracking is illustrated in Figure 125j. If the signal quality in the second best pair is better than the best pair, the PNC and the DEV shall switch to the second best pair, which becomes the new best antenna direction pair thereafter.

CTA of DEV quasi-omni transmit direction tracking					
SIFS	DEV → PNC Imm-ACK	SIFS	PNC → DEV Announce command with Feedback IE	SIFS	Quasi-omni training DEV → PNC

Figure 125j—CTA structure for DEV quasi-omni transmit direction tracking

The structure for quasi-omni tracking DEV → PNC is illustrated in Figure 125k.

QT #j, $Q_{j^{(2,t)}}^{(2,t)}$	BSIFS	QT #i, $Q_{i^{(2,t)}}^{(2,t)}$

Figure 125k—Quasi-omni direction training cycle structure

After 8.7, insert the following new subclause as 8.7a:

8.7a Aggregation

Aggregation may be performed for high-speed data/video transmission or low-latency bidirectional data transmission. Accordingly, there are two aggregation methods defined, standard aggregation and low-latency aggregation.

8.7a.1 Standard aggregation

Figure 125l illustrates the aggregation process. The originating DEV, upon receiving an MSDU, maps it into a subframe payload. If the length of the MSDU exceeds the predetermined value indicated in the Preferred Fragment Size field in Capability IE as defined in 7.4.11, the MSDU shall be fragmented and mapped into multiple subframe payloads. Each MSDU is assigned an unique MSDU number for identification. If fragmentation is adopted, each fragment is assigned a fragment number for identification within the MSDU. All the fragments of the same MSDU shall have the same MSDU number.

A subheader is created and configured, as defined in 7.2.8.1, for each subframe to contain the necessary information that helps the target DEV to retrieve the original data. The ACK Policy field in MAC header shall be set to Blk-ACK as described in 7.2.1.4.

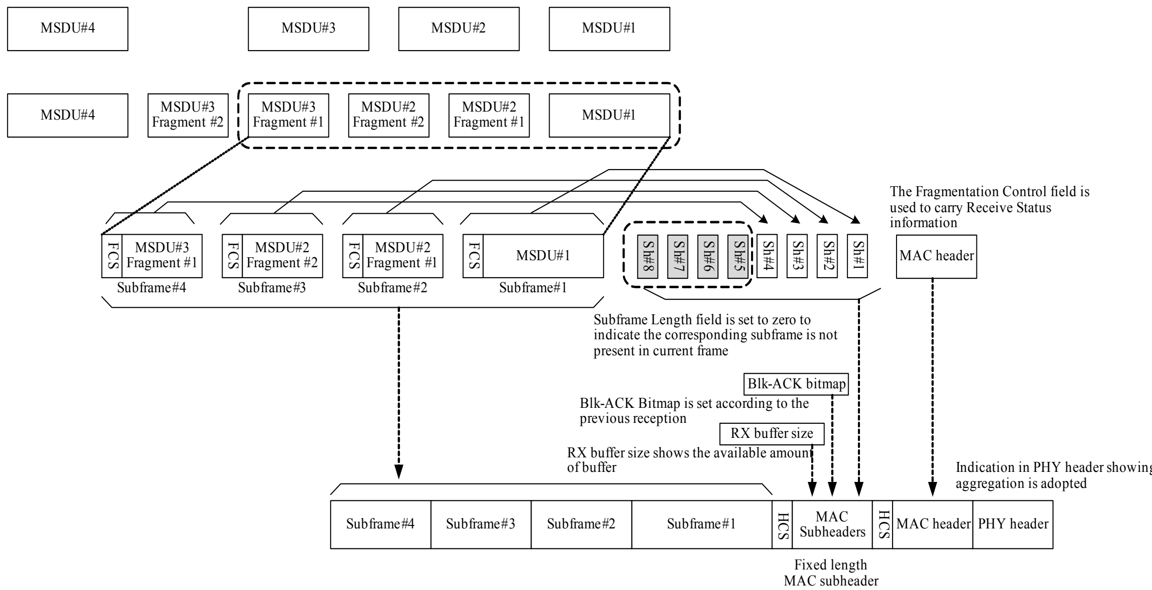

Figure 125l—Aggregation at originating DEV

All the subheaders are combined together to form the MAC subheader. As specified in 7.2.8.1, up to 8 subframes aggregated into a single frame. All unused subframes have zero length in the Subframe Length field.

Figure 125m illustrates the deaggregation process. After receiving the aggregated frame, the target DEV divides it into subframes according to the information in MAC subheader, validates each subframe by FCS. To recreate the original MSDU, the target DEV uses the MSDU Number field and the Fragment Number field in the sub-header, as defined in 7.2.8.1.

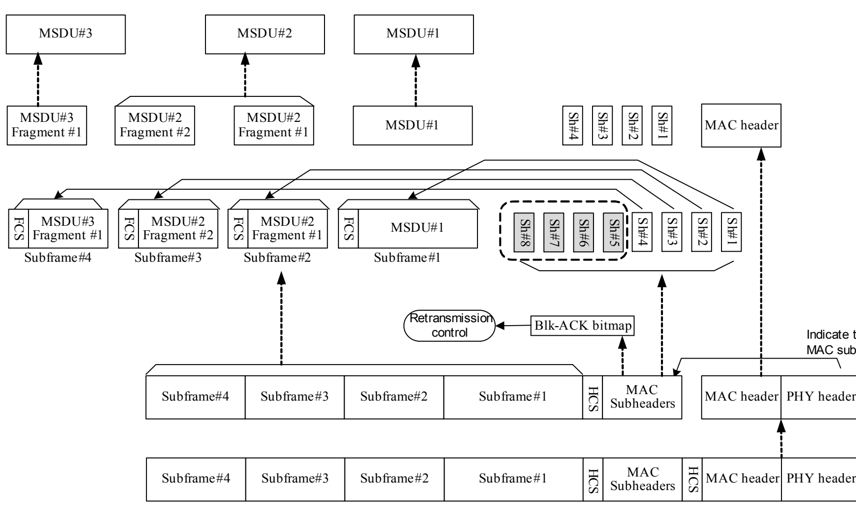

Figure 125m—Deaggregation at target DEV

The standard aggregation supports unidirectional and bidirectional data transmission by attaching ACK information with data. For unidirectional transmission, the target DEV upon receiving an aggregated frame replies an empty data frame with the Blk-ACK Bitmap field properly configured to support retransmission as described in 8.8.3b.1. For bidirectional data transmission, the target DEV sends ACK information with real data if any in the same frame to the originating DEV.

To avoid buffer overflow, before starting real data transmission, the originating and target DEV may exchange empty data frame with the RX Buffer Size field properly configured as defined in 7.2.8 to inform each other the available receiving buffer size. This information is used by the DEVs to adjust the subframe number and subframe length when sending real data.

The Fragmentation Control field in the MAC header of an aggregated frame shall always be used to carry the Receive Status field, as defined in 12.1.8.3.

8.7a.2 Low-latency aggregation

Before starting low-latency aggregation, the originating and target DEVs shall exchange the desired CTA relinquish duration value by sending CTA Relinquish Duration IE, as described in 7.4.26, in an Announce command, as defined in 7.5.5.2. Low-latency aggregation shall not be allowed in CP because the usage of relinquish operation is not allowed in CP.

The DEV chooses the smaller value among its own CTA relinquish duration and that of the target DEV to start the bidirectional CTA relinquish duration (BCRD) timer.

In low-latency aggregation mode, the originating DEV receives MSDUs from the FCSL, and aggregates them into the MAC frame body. Each MSDU is formatted as described in 7.2.8.2. The originating DEV shall also aggregate zero length MSDUs if there is no MSDU available from the FCSL until an MSDU is available, or until the BCRD timeout has expired. If the originating DEV has not received any MSDU from the FCSL until the beginning of the next BCRD period, it may transmit a zero length frame with the ACK Policy field in MAC header set to No-ACK, the Aggregation bit set to zero and the Low-latency Mode bit set to one in the PHY header (for the purpose of maintaining the link alive). Each MSDU transmitted is assigned an MSDU number on its first transmission that is used to identify it uniquely in the MSDU sequence delivered to the destination FCSL. The sequence number of zero length MSDU is assigned the most recent sequence number transmitted by the source for which it has received an ACK. The MSDU sequence number in a zero length MSDU shall be ignored by the destination. No retransmission is required for zero length MSDU.

When the target DEV receives an aggregated MAC frame, it verifies the Low-latency Aggregation field, as defined in 12.2.3.2.1.

The target DEV reads the frame body parsing each aggregated MSDU based on the received MSDU subheader. For each MSDU subheader, the destination validates its HCS. If the HCS is not valid, the destination shall continue searching for a match of a valid MSDU HCS, shifting the next expected MSDU on octet alignment until one is found. If the FCS is correct, the destination passes the MSDU in order to the FCSL, withholding any trailing MSDUs until all intermediate MSDU fragments are retransmitted and received correctly.

The target DEV sets the corresponding bit in the next transmitted Blk-ACK Bitmap field according to its order offset taking the value of MSDU Response Number field of the transmitted MAC subheader as reference. The MSDU Response Number field is updated upon reception of a valid MAC subheader, from the MSDU Request Number field. In case the MSDU order offset is not sequential to the previous MSDU order-offset, the corresponding range of bits shall be cleared in the Blk-ACK Bitmap field.

The target DEV shall inform the originating DEV the available receiving buffer size using RX buffer size field as defined in 7.2.8. The originating DEV adjusts the maximum number of MSDUs to send in the next frame to avoid buffer overflow.

If the MSDU is incorrectly received, the corresponding bit is set to zero in the next transmitted Blk-ACK Bitmap field according to its order offset in the current aggregated frame.

The source and destination interchange roles during the CTA, every BCRD or less, utilizing the CTA relinquish mechanism described in 8.4.3.8, to enable minimum latency retransmission of incorrectly received MSDUs.

8.8 Acknowledgment and retransmission

Change the paragraph in 8.8 as shown:

The~~re are four~~ acknowledgment types defined for this standard are as follows:~~; no acknowledgment (no-ACK), immediate acknowledgment (Imm-ACK), delayed acknowledgment (Dly-ACK), and implied acknowledgment (Imp-ACK).~~

— No acknowledgment (no-ACK),
— Immediate acknowledgment (Imm-ACK),
— Delayed acknowledgment (Dly-ACK),
— Implied acknowledgment (Imp-ACK), and
— Block acknowledgment (Blk-ACK).

After 8.8.3, insert the following new subclause as 8.8.3b:

8.8.3b Block ACK

Blk-ACK shall only be used with an aggregated frame. The destination, upon receiving an aggregated frame, checks each subframe. Based on the status of the subframe, either correctly or incorrectly received, the corresponding ACK bit or msb/lsb ACK bit of the Blk-ACK Bitmap field of the MAC subheader will be set as described in 7.2.8.1. The originating DEV, after reading the Blk-ACK Bitmap field in MAC subheader, handles subframe retransmission.

For example, if the destination correctly receives subframes 0–4, 6, and 8–10, then the destination would set bits 0–4, 6, and 8–10 to one and the rest of the bits in the Blk-ACK Bitmap fields to zero.

The control of retransmission is different according to the different aggregation mode, standard or low-latency.

8.8.3b.1 Blk-ACK for standard aggregation

For standard aggregation, the rules in this subclause apply.

If the MAC subheader is incorrectly received by the target DEV, all the subframes in the frame are considered invalid.

If the MAC subheader is correctly received, but some of the subframes are incorrectly received, then the target DEV responds with an data frame with the Blk-ACK Bitmap field in the MAC subheader set to indicate the subframes that were incorrectly received, as described in 7.2.8. If the data transmission is unidirectional from originating DEV to target DEV, the data frame sent back by the target DEV shall be an empty data frame.

In any frame sent by the originating DEV, any retransmitted subframes should be put in the original order. The MSDU Number field in MAC header shall be set to the value of the first subframe, no matter it is a retransmission or a newly aggregated subframe.

If the originating DEV wants to drop certain retransmissions, it advances the MSDU number in MAC header to inform the target DEV that any subframe with the MSDU number older than this will not be retransmitted anymore.

In case that the Blk-ACK is lost, the originating DEV may retransmit the same frame after RIFS. The subframes contained will be the same as those of the previous transmitted frame. The retry bit in the MAC header shall be set to one to indicate to the target DEV that the same frame is retransmitted. For any frame that contains different subframes from previous transmission, the Retry bit in MAC header shall be set to zero.

If the subframe contains only msb information, the lsb ACK in the Blk-ACK Bitmap field in the MAC header shall be set to zero and shall be ignored upon reception. Likewise, if the a subframe contains only lsb information, the lsb ACK in the Blk-ACK Bitmap field shall be set to zero and shall be ignored upon reception.

8.8.3b.2 Blk-ACK for low-latency aggregation

For low-latency aggregation, the rules in this subclause apply.

When the source receives the information in the Blk-ACK Bitmap field, it inserts the incorrectly received MSDUs for retransmission in the next frame to be sent.

The originating DEV of a Blk-ACK shall maintain the Blk-ACK Bitmap field according the rules in this subclause.

The originator maintains a list of recently transmitted MSDU numbers for retransmissions.

The list of recently transmitted MSDUs shall be updated per transmission frame, and with each Blk-ACK received, to reflect any outstanding MSDU not acknowledged.

The originating DEV may abort retransmissions of a specific MSDU by advancing the value of the MSDU Request Number field, as defined in 7.2.8.2, beyond the specific MSDU sequence number. When the MSDU Request Number is advanced beyond a missing MSDU sequence number the receiver updates its MSDU Response Number accordingly and assumes the skipped MSDU sequence number is discarded.

For each newly transmitted MSDU, its MSDU number is appended to the end of the list. The number of MSDUs in the list may grow up to the limit of the mMaxSubframeNum, as defined in 8.15.

If the MAC subheader is correctly received, the Blk-ACK Bitmap field is parsed. For each bit that is set, the entry of the MSDU number is removed from the list, updating the relative order of all remaining MSDUs (that were not acknowledged) and subsequently their offset order in the following retransmission attempt on the next transmission frame.

If the MAC subheader is not correctly received, the Blk-ACK Bitmap field is considered as corrupt. In this case, the source DEV may retransmit all the MSDUs from the previous frame.

If the subframe contains only msb information, the lsb ACK in the Blk-ACK Bitmap field in the MAC header shall be set to zero and shall be ignored upon reception. Likewise, if the a subframe contains only lsb information, the lsb ACK in the Blk-ACK Bitmap field shall be set to zero and shall be ignored upon reception.

8.9 Peer discovery

After 8.9.6, insert the following new subclause as 8.9.7:

8.9.7 Channel probing

Channel probing is an optional function to allow the source DEV choose the best MCS for the current channel. A DEV may use the Channel Status Request, as described in 7.5.7.1, and Channel Status Response, as described in 7.5.7.2, commands to gather information of the link quality with another DEV. In addition, when the DEVs are capable of beam forming, the quality of the link may be obtained by beam forming procedure, as described in 13.5. During data streaming, the destination DEV reports the channel status information, as defined in 12.1.8.3, to the source DEV. After learning the channel status, the source DEV may switch to an MCS that is best adapted to the channel condition. For the SC/HSI PHY, the channel status information may be written in the fragmentation field of the MAC header of a Imm-ACK or Dly-ACK, as defined in 7.3.4. For AV PHY, it may be written in the fragmentation field of the MAC header of an Imm-ACK or Dly-ACK frame.

8.12 Multi-rate support

Change the first paragraph in 8.12 as shown:

A compliant PHY may support more than one data rate. In each PHY there is one mandatory base rate specified for the purposes described in this subclause. In addition to the base rate, the PHY may support rates that are both faster and slower than the base rate. A DEV shall send a frame with a particular data rate to a destination DEV only when the destination DEV is known to support that rate. The allowed data rates and the mandatory base rate are PHY dependent and are defined~~. For the 2.4 GHz PHY, the supported rates are defined in 11.3.~~

— In 11.3 for the 2.4 GHz PHY,
— In 12.2.2.1 for the SC PHY mode,
— In 12.3.2.1 for the HSI PHY mode, and
— In 12.4 for the AV PHY mode.

In order to determine the rates that are supported by a target DEV in the piconet, the DEV shall use one of three methods:

After Table 58, insert the following paragraph to 8.12:

HRP frames in the AV PHY with the ACK policy set to Imm-ACK shall be ACKed with the directional ACK, as described in 7.2.9.5. The data rate for the payload in a directional ACK with payload is selected by the DEV sending the directional ACK. Blk-ACK policy shall not be used with HRP AV PHY frames.

8.15 MAC sublayer parameters

Insert the following row at the end of Table 60. All other rows are unchanged and are not shown.

Insert the following text and table after Table 61 in 8.15:

Additional characteristics that are PHY dependent are listed in Table 61a for the SC PHY.

Additional characteristics that are PHY dependent are listed in Table 61b for the HSI PHY.

Table 60—MAC sublayer parameter

Parameter	Value
mMaxSubframeNum	For standard aggregation mode = 8, for low-latency mode = 256
mMaxSubframeSize	For standard aggregation mode = 1,0478,575, for low-latency mode = 256

Table 61a—MAC sublayer parameters—SC PHY dependent

Parameter	Subclause
SIFS	12.2.6.1
MIFS	12.2.6.1
BIFS	12.2.6.1
RIFS	12.2.6.1
pBackoffSlot	12.2.6.1
pMaxTransferUnitSize	12.2.7.2
pMaxFrameBodySize	12.2.7.1
pPHYClockAccuracy	12.2.4.3
pLengthHCS	12.2.3.2.2
pMinFragmentSize	12.2.7.3

Table 61b—MAC sublayer parameters—HSI PHY dependent

Parameter	Subclause
SIFS	12.3.5.5
MIFS	12.3.5.8
BIFS	12.3.5.6
RIFS	12.3.5.6
pBackoffSlot	12.3.5.6
pMaxTransferUnitSize	12.3.6.4
pMaxFrameBodySize	12.3.6.3
pPHYClockAccuracy	12.3.4.2
pLengthHCS	12.3.3.4.1
pMinFragmentSize	12.3.6.5

Additional characteristics that are PHY dependent are listed in Table 61c for the AV PHY.

Table 61c—MAC sublayer parameters—AV PHY dependent

Parameter	Subclause
SIFS	12.4.1.2.2
MIFS	12.4.1.2.4
BIFS	12.4.1.2
RIFS	12.4.1.2
pBackoffSlot	12.4.1.2
pMaxTransferUnitSize	12.4.1.3.2
pMaxFrameBodySize	12.4.1.3.1
pPHYClockAccuracy	12.4.4.3
pLengthHCS	12.4.1.4
pMinFragmentSize	12.4.1.3.3

After 8.15, insert the following new subclauses as 8.16 and 8.17:

8.16 UEP

Three types of UEP may be used for flexible error protection of a frame consisting of msb and lsb subframes.

— UEP Type 1 protects an aggregated frame that consists of msb subframes, lsb subframes, or both msb and lsb subframes by using the different FECs.
— UEP Type 2 protects an aggregated frame that consists of msb subframes, lsb subframes, or both msb and lsb subframes by using different MCSs. The UEP Specific IE, as defined in 7.4.24, is used in UEP Type 2 only for the originating DEV to inform the target DEV of the separating position of the msbs and lsbs. If necessary, a DEV using UEP Type 2 may send an Announce command with the UEP Specific IE to dynamically change the separating position.
— UEP Type 3 protects msb and lsb in a subframe unequally by either applying different FEC coding rates to msbs and lsbs or mapping msbs and lsbs to a skewed constellation. The capability of supporting UEP by FEC coding and/or by the skewed constellation is indicated in the Capability IE, as defined in 7.4.11.

All UEP types, UEP Type 1, UEP Type 2 and UEP Type 3, may be used in SC PHY as described in 12.2.2.9. UEP Type 3 may be used in HSI PHY, as described in 12.3.2.1, and AV PHY, as described in 12.4.

It is required that originating and target DEVs understand the UEP type and MCS information of each other before sending frames that use UEP. This is accomplished by the DEVs reporting their UEP capability in the DEV Capability IE in the association process. If the source and destination do not support the same UEP type, then only EEP frames shall be exchanged between the DEVs.

8.17 Sync frame transmission and virtually dependent piconet

Sync Frame Transmission is a function that mitigates co-channel interference due to a hidden PNC node. It also provides a method to obtain synchronization among independent piconets. Sync frame transmission is mandatory for PNC-capable DEVs when operating as the PNC, as defined in 12.1.9, and optional for non-

PNC capable DEVs. A DEV indicates if it is capable of sending Sync frames by setting the Sync Frame Capable bit in the Capabilities IE.

The PNC controls the transmission of Sync frames by DEVs in the piconet by sending a Sync Frame Frequency IE, as defined in 7.4.35, in a Announce command to a DEV that is capable of sending Sync frames. The PNC sets the frequency with which the frames will be sent by setting the Sync Frame TX Frequency field to the appropriate value.

When a DEV that is capable of sending Sync frames receives the Sync Frame Frequency IE from the PNC, it sends a Sync frame in the first transmission opportunity in the next superframe. The DEV continues sending the Sync frame every Sync Frame TX Frequency superframes. DEVs may use the response frame in an Imp-ACK exchange to send the Sync frame. DEVs may use either data or Imp-ACK transmit opportunities for sync frame transmission.

A DEV that receives a Sync frame from another piconet may utilize the information in the frame to obtain network synchronization and mitigate interference. A PNC may use Sync frames that it receives from DEVs in another piconet to synchronize to PNC timing and forming a virtually dependent piconet. The PNC may then adjust the CTAs in its piconet to avoid interference with the other piconet.

10. Security specifications

10.2 Symmetric cryptography building blocks

10.2.4 Nonce value

After Figure 154, insert Figure 154a and the following sentences:

Figure 154a specifies the format of the nonce for aggregated frame.

octets: 1	**1**	**2**	**6**	**1**	**1**
Padding bits	Secure subframe counter	Secure frame counter	Time token	DestID	SrcID

Figure 154a—CCM nonce format for aggregated frame

The Secure subframe counter shall be included to generate an unique nonce for each subframe that is aggregated. The counter shall be incremented for each subframe in a frame.

The Padding bits shall be set to 0b11001111.

Insert the following clause as Clause 12:

12. PHY specification for millimeter wave

12.1 General requirements

A compliant mmWave PHY shall implement at least one of the following PHY modes:

a) Single Carrier mode in mmWave PHY (SC PHY), as defined in 12.2,.
b) High Speed Interface mode in mmWave PHY (HSI PHY), as defined in 12.3.

c) Audio/Visual mode in mmWave PHY (AV PHY), as defined in 12.4.

Unless otherwise specified, all reserved fields shall be set to zero on transmission and shall be ignored upon reception.

Unless otherwise stated, in all figures in this clause the ordering of the octets and bits as they are presented to the PHY for modulation is the same as defined in 7.1.

12.1.1 Regulatory information

The mmWave PHY operating frequency is within the 57.0–66.0 GHz range as allocated by the regulatory agencies in Europe, Japan, Canada, and the United States. This band will also be available in other areas where allocated by the regulatory bodies.

The documents listed in Table 94 are provided as a reference for various geographic regulatory regions. The list is neither exhaustive nor complete. It is the responsibility of the implementor to verify that the DEV complies with all regulatory requirements in the geographic region where the device is deployed or sold.

Table 94—Documents for selected geographic regulatory regions

Region	Regulatory document
Australia	Radio communications class license 2000
Canada	RSS-210, Issue 6, September 2005
Japan	Regulations for the enforcement of radio law, 6-4.2 specified low power radio station (17) 59-66 GHz band
USA	47 CFR 15.255

The maximum allowable output power, as measured in accordance with practices specified by the appropriate regulatory bodies, is shown in Table 95. A compliant DEV may use any transmit power level up to the applicable limits in the geographical region.

Table 95—Maximum transmit power levels in selected geographical regions

Geographical Region	Power limit	EIRP limit	Regulatory document
USA	—	Maximum indoor EIRP: 27 dBi Maximum outdoor EIRP: 40 dBi	47 CFR 15.255
Japan	Maximum output power: 10dBm Maximum bandwidth: 2.5 GHz	Maximum EIRP: 57 dBi	ARIB STD-T69, ARIB STD-T74
Australia	Maximum output power: 10dBm	Maximum EIRP: 51.8 dBi	Radiocommunications Class License 2000

12.1.2 RF power measurements

Unless otherwise stated, all RF power measurements for the purpose of this standard, either transmit or receive, shall be made based on EIRP and any radiated measurements shall be corrected to compensate for the antenna gain in the implementation. The gain of the antenna is the maximum estimated gain by the manufacturer.

12.1.3 Unwanted emissions

Conformant implementations shall comply with the in-band and out-of-band emissions for all operational modes as set by the applicable regulatory bodies.

12.1.4 Operating temperature range

A conformant implementation shall meet all of the specifications in this standard for ambient temperatures from 0 °C to 40 °C.

12.1.5 RF channelization

The mmWave PHYs use the channels defined in Table 96.

Table 96—mmWave PHY channelization

CHNL_ID	Start frequency[a]	Center frequency	Stop frequency[a]
1	57.240 GHz	58.320 GHz	59.400 GHz
2	59.400 GHz	60.480 GHz	61.560 GHz
3	61.560 GHz	62.640 GHz	63.720 GHz
4	63.720 GHz	64.800 GHz	65.880 GHz

[a]The start and stop frequencies are nominal values. The frequency spectrum of the transmitted signal needs to conform to the transmit power spectral density (PSD) mask for the PHY mode as well as any regulatory requirement

12.1.6 Transmit PSD mask

The transmitted spectrum shall adhere to the transmit spectrum mask shown in Figure 155. For the transmit mask measurements, the resolution bandwidth is set to 3 MHz and video bandwidth to 300 kHz. During OOK modulation, transmitters shall meet the same PSD mask, except for the single line spectra of 40 dB above the 0 dB line in Figure 155 within the frequency band of [–6 MHz,+6 MHz] from the carrier frequency.

12.1.7 Error vector magnitude calculation

12.1.7.1 SC PHY

The error vector measurement shall be made on baseband I and Q data after recovery through an ideal reference receiver system. An ideal receiver is a receiver that is capable of converting the transmitted signal into a stream of complex samples at sufficient rate or more, with sufficient accuracy in terms of I/Q arm amplitude and phase balance, DC offsets, and phase noise. It shall perform carrier lock, symbol timing

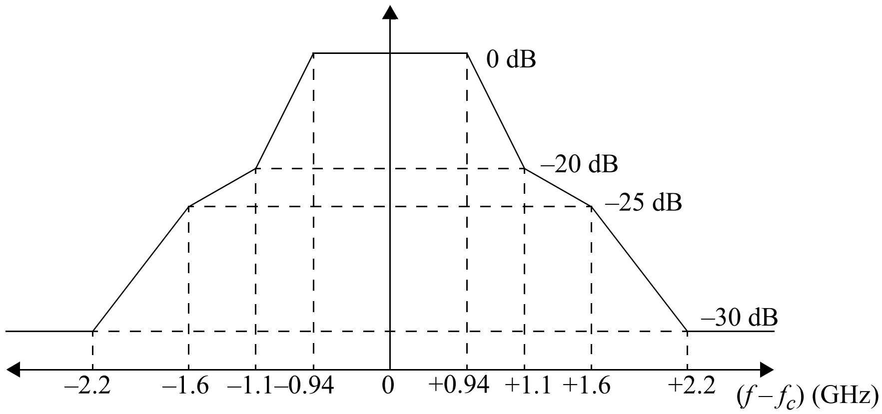

Figure 155—Transmit spectral mask

recovery and amplitude adjustment while making the measurements. For SC PHY EVM, measuring 1000 chips at the chip rate is recommended as shown in Equation (12):

$$EVM \equiv \sqrt{\frac{1}{1000P_{avg}}\sum_{i=1}^{1000}[(I_i - I^*_i)^2 + (Q_i - Q^*_i)^2]} \tag{12}$$

where P_{avg} is the average power of the constellation, (I_i*, Q_i*) is the complex coordinates of the i^{th} measured chip, and (I_i, Q_i) is the complex coordinates of the nearest constellation point for the i^{th} measured chip.

The measuring device should have the accuracy of at least 20 dB better than the EVM value to be measured. The test equipment should use a root-raised cosine filter with roll-off factor of 0.25 for the pulse shaping filter when conducting EVM measurement.

12.1.7.2 AV and HSI PHYs

The EVM of an OFDM mmWave transmitter is measured by calculating the Euclidean distance between the received symbol and the closest constellation point. The EVM measurement has the following requirements:

a) The measurement shall be made over the OFDM data symbols starting with the PHY header.
b) The phase and frequency for the data subcarriers shall be corrected for using the pilot subcarriers.
c) Channel estimation shall have been performed for each of the data subcarriers.
d) The error vector magnitude calculation for the HSI and AV PHY modes shall be averaged over a minimum of 20 frames.

The EVM is then the average the RMS errors in the frame over a number of frames. In equation form, it is given by Equation (13):

$$EVM_{rms} \equiv \frac{1}{N_p}\left[\sum_{i=1}^{N_p}\sqrt{\frac{1}{P_{avg}N_sN_{dsc}}\sum_{j=1}^{N_s}\left\{\sum_{k=1}^{N_{dsc}}[(I_{ijk} - I^*_{ijk})^2 + (Q_{ijk} - Q^*_{ijk})^2]\right\}}\right] \tag{13}$$

where

N_p is the number of frames in the measurement

N_s is the number of symbols per frame in the measurement
N_{dsc} is the number of data subcarriers in the OFDM modulation
P_{avg} is the average power of the constellation
(I^*_{ijk}, Q^*_{ijk}) is the complex coordinates of the j^{th} measured symbol in the k^{th} subcarrier of the i^{th} frame
(I_{ijk}, Q_{ijk}) is the complex coordinates of the nearest constellation point for the j^{th} measured symbol of the k^{th} subcarrier of the i^{th} frame

12.1.8 Common PHY management for mmWave PHY modes

12.1.8.1 Supported MCSs

The Supported Data Rates field in the Capability IE, as described in 7.4.11, shall be formatted as illustrated in Figure 156.

bits:1	1	1	1	1	1	1	1
SC spreading	SC QPSK	SC 8-PSK	SC 16-QAM	SC LDPC 1	SC LDPC 2	AV 16-QAM	HSI QAM

Figure 156—Supported data rates field format for mmWave PHY modes

The SC spreading field shall be set to one if spreading factors 2, 4, and 6 are supported by the SC PHY DEV and shall be set to zero otherwise.

The SC QPSK field shall be set to one if QPSK modulation is supported by the SC PHY DEV and shall be set to zero otherwise.

The SC 8-PSK field shall be set to one if 8-PSK modulation is supported by the SC PHY DEV and shall be set to zero otherwise.

The SC 16-QAM field shall be set to one if 16-QAM modulation is supported by the SC PHY DEV and shall be set to zero otherwise.

The SC LDPC 1 field shall be set to one if the optional FECs LDPC(672, 336), LDPC(672, 504), and LDPC(672, 588) are supported by the SC PHY DEV and shall be set to zero otherwise.

The SC LDPC 2 field shall be set to one if the optional FEC LDPC(1440, 1344) is supported by the SC PHY DEV and shall be set to zero otherwise.

The AV 16-QAM field shall be set to one if 16-QAM modulation is supported by the AV PHY DEV and shall be set to zero otherwise.

The HSI QAM field shall be set to one if the 16-QAM and 64-QAM modulations are supported by the HSI PHY DEV and shall be set to zero otherwise.

12.1.8.2 Preferred fragment size

The encoding of the Preferred Fragment Size field, as described in 7.4.11, is given in Table 97.

Table 97—Preferred fragment size field encoding

Field value	Preferred fragment size (octets)
000	1048576
001	262144
010	65536
011	16384
100	4096
101	2048
110	512
111	Reserved

12.1.8.3 Receive Status field

The Receive Status field is used to send information about the received frame to the transmitter, as described in 7.2.4. The Receive Status field shall be formatted as illustrated in Figure 157.

bits: 4	2	1	2	4	2	4	4	1
Reserved	Suggested MCS status	Suggested PCES	Suggested pilot word type	FER	Suggested preamble type	SINR	RSSIr	Valid

Figure 157—Receive Status field format for mmWave PHY

The Valid field shall be set to one if the information in the Receive Status field is to be accessed by the receiver and shall be set to zero otherwise.

The RSSIr field (dB) contains the difference of the received signal power in dBm and the sensitivity in dBm of the selected MCS. The range of the RSSIr field is from 0 dB to 28 dB in 2 dB steps with 0b0000 corresponding to less than or equal to 0 dB, and 0b1111 corresponding to greater than 28 dB. For example, an RSSIr value that is greater than or equal to 8 dB but less than 10 dB would be encoded as 0b0101.

The SINR field contains the estimated signal to noise and interference ratio of the received frame, and is defined to be $E_b/(I + N)$ in dB where E_b is the energy per bit, I is the interference power and N is the noise power over the bandwidth. The range of the SINR field is from 2 dB to 28 dB in 2 dB steps with 0b0001 corresponding to less than or equal to 2 dB, 0b1111 corresponding to greater than 28 dB, and 0b0000 reserved. For example, an SINR value that is greater than or equal to 18 dB but less than 20 dB would be encoded as 0b1010.

Valid values of the Suggested Preamble Type field are as follows:

— 0b00 → Suggest CMS preamble
— 0b01 → Suggest SC preamble
— 0b10–0b11 → Reserved

Valid values of the Suggested Pilot Word Type field are as follows:

0b00 → Suggest SC pilot word length of 64

0b01 –› Suggest SC pilot word length of 8
0b10 –› Suggest SC pilot word length of 0
0b11 –› Reserved

The Suggested PCES Type field shall be set to zero to suggest that no PCES should be sent and shall be set to one to suggest the use of PCES.

The FER field contains the exponent of the estimate of the FER ranging from 10^{-1} to 10^{-10} in steps of 10^{-1} with 0b0000 corresponding to an FER exponent of less than or equal to –1 (i.e., an FER greater than or equal to 10^{-1}), 0b1010 corresponding to an FER exponent of greater than –10 (i.e., an FER of less than 10^{-10}), and 0b1011–0b1111 reserved. For example, an FER field value of 0b0110 indicates an exponent of –6 and that the FER was less than or equal to 10^{-6} but was greater than 10^{-7}.

Valid values of the Suggested MCS Status field are as follows:

— 0b00 –› Use an MCS with lower required SNR
— 0b01 –› Use current MCS
— 0b10 –› Use an MCS with higher required SNR
— 0b11 –› Reserved

12.1.9 Requirements for mmWave PNCs

In order to promote coexistence among DEVs using different PHYs, the following MAC rules have defined.

— An AV PNC-capable DEV, when operating as a PNC, shall transmit an AV beacon and a CMS sync frame in every superframe.
— An HSI PNC-capable DEV, when operating as a PNC, shall transmit an HSI beacon and a CMS sync frame in every superframe.
— An AV PNC-capable DEV shall be able to receive the CMS sync frame and command frames.
— An HSI PNC-capable DEV shall be able to receive the CMS sync frame and command frames.
— A DEV capable of transmitting a sync frame may do so in the first granted CTA in a superframe, and in every pre-defined number of superframes for the directions indicated in the Sync Frame Frequency IE as indicated in 7.4.35.

A DEV is defined as being able to receive a CMS frame if it can at least successfully perform SYNC and SFD detection in the preamble when the signal power is greater than the receiver sensitivity.

12.1.10 CP operation

A PNC-capable DEV compliant to this standard shall allow the use of the CAP for contention based access for association, data and commands when using the mmWave PHY, as described in 7.3.1. A DEV compliant to this standard shall support the use of the CAP. CSMA/CA shall be used for all CPs in a mmWave PHY piconet.

The CPs in a piconet shall all be conducted using the same mmWave PHY mode, one of SC, HSI or AV, as the beacon. Any MCS that is supported by both the source and destination and that is in the same PHY mode as the beacon may be used in a CP, with the exception of the AV HRP modes, which shall not be used in a CP. The usage of directional CAPs is described in 8.6.6.2.

For medium access in the CP, the following rules apply:

— A DEV attempting to access the medium in an SC CAP for data transmission, shall always use frames consisting of the CMS preamble, as described in 12.1.12.4, the MR header, as described in

12.2.2.2, and any SC MCS, as described in 12.2.2.1, using the best TX direction over regular CAP and S-CAP.

— A DEV attempting to access the medium in an HSI CAP for data transmission, shall always do so by using frames consisting of the HSI MCS0 long preamble, as described in 12.3.3.1, and any HSI MCS, as described in 12.3.2.1, using the best TX direction over regular CAP and S-CAP.
— A DEV attempting to access the medium in an AV CP, shall always do so by using omni-LRPDU, as described in 12.4.3.

12.1.11 mmWave PHY mode usage in CTA

In a CTA, any PHY mode may be used that is supported by both the source and destination DEVs.

Examples of how this is used in a mmWave PHY piconet are given in D1.4.

12.1.12 Common Mode Signaling (CMS)

Common mode signaling (CMS) is a low data rate SC PHY mode specified to enable interoperability among different PHY modes. The CMS is used for transmission of the beacon frame defined in 8.6.2, and, if supported, the sync frame defined in 7.3.6 and 12.1.9. The CMS is also used for transmission of command frame and training sequence in the beam forming procedure, as defined in Clause 13, for the SC and HSI PHYs.

The CMS frame shall be formatted as illustrated in Figure 158.

CMS PHY payload	CMS frame header	CMS PHY preamble

Figure 158—CMS frame format

The CMS PHY preamble is defined in 12.1.12.4.

The CMS frame header shall be formatted as illustrated in Figure 159. The CMS frame header shall be constructed as described in 12.1.12.5 and 12.1.12.6.

RS parity bits	HCS	MAC header	PHY header

Figure 159—CMS frame header format

The PHY Payload field of a CMS frame shall be constructed as defined in 12.1.12.7.

The chip rate of CMS is 1760 Mchip/s. The entire CMS frame shall be modulated with π/2 BPSK/precoded (G)MSK as specified in 12.2.2.5.1. The FEC for the CMS frame header and MAC frame body shall be as specified in 12.1.12.1. The CMS frame header and MAC frame body shall be spread as specified in 12.1.12.2. The CMS preamble shall be excluded from the spreading process. The scrambling process shall be specified in 12.1.12.3.

The header rate shall be as defined in Table 105 while the PHY payload rate shall be as defined in Table 103.

12.1.12.1 Forward error correction for CMS

The FEC scheme for CMS shall be RS coding. The RS(255,239), which is the mother code, shall be used for encoding the MAC frame body of CMS. The RS(n+16, n), a shortened version of RS(255,239) where n is the number of octets in the frame header, shall be used for encoding the frame header of CMS. Details of the coding are provided in 12.2.2.6.1.

12.1.12.2 Code spreading for CMS

To increase robustness in the frame header and MAC frame body of the CMS, code spreading shall be applied using Golay sequences. The code spreading factor shall be 64, and the Golay sequence specified in Table 98 shall be used. The frame header and the MAC frame body shall be spread according to Figure 160. Note that in each hexadecimal-equivalent 4-binary-digit group, the leftmost bit shall be the msb, and the rightmost bit, the lsb. For example, 3 is denoted as 0011. The order of the octets and bits over the air is the same as defined in 7.1.

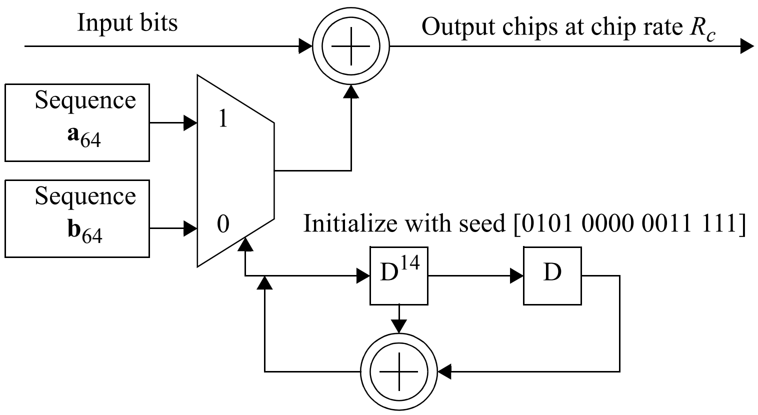

Figure 160—Realization of the CMS code spreading

Table 98—Golay sequences with length 64

Sequence name	Sequence value
$\mathbf{a}_{64}$	63AF05C963500536
$\mathbf{b}_{64}$	6CA00AC66C5F0A39

As shown in Figure 160, a pseudo-random bit sequence (PRBS) generated using an linear-feedback shift register (LFSR) is used to vary the spreading code from one bit to another, i.e., each bit is spread with Golay code $\mathbf{a}_{64}$ or Golay code $\mathbf{b}_{64}$ depending on the LFSR output. The LFSR shall be run at a rate equals to 1/64 the bit rate and shall be initialized with the following 15 bit seed value: $[x_{-1}, x_{-2}, \ldots, x_{-15}]$ =[0101 0000 0011 111]. Each input bit shall be held for 64 chips as it is XORed with Golay code $\mathbf{a}_{64}$ or Golay code $\mathbf{b}_{64}$ operating at the chip rate.

12.1.12.3 Scrambling for CMS

Scrambling shall be employed to whiten the CMS frames. Scrambling shall be used for the MAC header, HCS, and the entire PHY Payload field as detailed in 12.2.2.10.

12.1.12.4 PHY preamble for CMS

A PHY preamble is used to aid receiver algorithms related to AGC setting, timing acquisition, frame synchronization and channel estimation. The CMS preamble shall be formatted as illustrated in Figure 161.

The SYNC field is used primarily for frame detection and shall consist of 48 repetitions of $\mathbf{b}_{128}$.

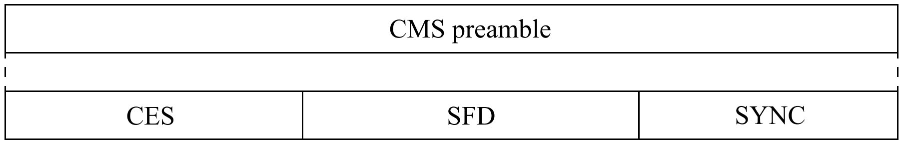

Figure 161—PHY preamble structure for CMS

The SFD field is used to establish frame timing and shall consist of [+1 –1 +1 +1 –1 –1 –1] spread by $\mathbf{b}_{128}$.

The CES field is used for channel estimation and shall consist of $\mathbf{b}_{128}$ $\mathbf{b}_{256}$ $\mathbf{a}_{256}$ $\mathbf{b}_{256}$ $\mathbf{a}_{256}$ $\mathbf{a}_{128}$ with $\mathbf{a}_{128}$ first in time.

The Golay complimentary sequences of length 128, denoted by $\mathbf{a}_{128}$ and $\mathbf{b}_{128}$, are shown in Table 99. The Golay code complimentary sequences of length 256, denoted by $\mathbf{a}_{256}$ and $\mathbf{b}_{256}$, are defined as:

$$\mathbf{a}_{256} = [\mathbf{a}_{128}\ \mathbf{b}_{128}]$$

$$\mathbf{b}_{256} = [\mathbf{a}_{128}\ \overline{\mathbf{b}}_{128}]$$

where the number on the right ($\mathbf{b}_{128}$ and $\overline{\mathbf{b}}_{128}$) are the first in time and the binary-complement of a sequence x is denoted by an overline on x (i.e., $\overline{x}$).

Alternatively, the CMS preamble may be used by the SC PHY for data transmission in the CAP, as specified in 12.2.3.1.

Table 99—Golay sequences with length 128

Sequence name	Sequence value
$\mathbf{a}_{128}$	0x0536635005C963AFFAC99CAF05C963AF
$\mathbf{b}_{128}$	0x0A396C5F0AC66CA0F5C693A00AC66CA0

12.1.12.5 Frame Header for CMS

The frame header conveys information in the PHY and MAC headers necessary for successfully decoding the frame. The construction of the CMS header is shown in Figure 162. The detailed process of the construction is as follows:

a) Construct the PHY header as described in 12.1.12.6.
b) Compute the HCS as described in 12.2.3.2.2 over the combined PHY and MAC headers.
c) Append the HCS to the MAC header.
d) Scramble the combined MAC header and HCS as described in 12.1.12.3.
e) Compute the RS parity bits by encoding the concatenation of the PHY header, scrambled MAC header, and scrambled HCS into a shortened RS block code as described in 12.1.12.1.
f) Form the base frame header by concatenating the PHY header, scrambled MAC header, scrambled HCS and RS parity bits.
g) Spread the frame header as described in 12.1.12.2.

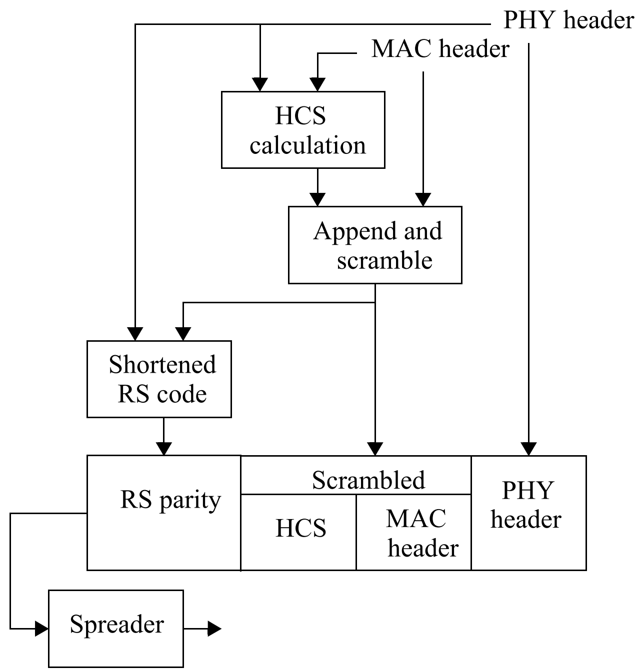

Figure 162—Frame header construction process for CMS

12.1.12.6 PHY header for CMS

The CMS PHY header shall be formatted as illustrated in Figure 176. The description of each field is provided in 12.2.3.2.1. In this subclause, the field values for the CMS PHY header are specified.

The Scrambler Seed ID field contains the scrambler seed identifier value, as defined in 12.1.12.3.

The Aggregation bit shall be set to zero.

The UEP bit shall be set to zero.

The MCS field shall be set to 0b00000.

The Frame Length field shall be an unsigned integer that indicates the number of octets in the MAC frame body, excluding the FCS.

The Preamble Type field shall be set to 0b00.

The Beam Tracking field shall be set to one if the training sequence for beam tracking is following the current frame, and shall be set to zero otherwise.

The Low-latency mode bit shall be set to zero.

The Pilot Word Length field shall be set to 0b10.

The PCES field shall be set to zero.

12.1.12.7 PHY Payload field for CMS

The PHY Payload field is the last component of the CMS frame, and is constructed as shown in Figure 163. The PHY Payload field of the CMS shall be constructed as follows:

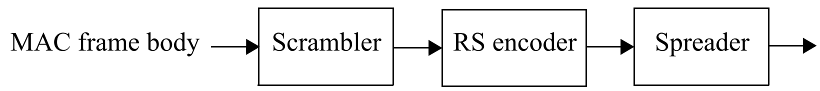

Figure 163—PHY Payload field construction process for CMS

a) Scramble the MAC Frame Body field according to 12.1.12.3.
b) Encode the scrambled MAC Frame Body field as specified in 12.1.12.1.
c) Spread the encoded and scrambled MAC Frame Body field using the spreading code as detailed in 12.1.12.2.

12.1.12.8 Receiver clear channel assessment performance for CMS

The start of a valid preamble sequence at a receive level equal to or greater than the minimum sensitivity for the CMS, as described in 12.2.5.2, shall cause CCA to indicate medium busy with a probability of > 90 % within pCCADetectTime for the SC PHY mode. The receiver CCA function shall in all circumstances report medium busy with any signal 20 dB above the minimum sensitivity for the CMS. The CCA shall be maintained as busy until the end of the frame.

12.1.13 mmWave PHY PIB

The PHY dependent PIB values for the mmWave PHY are given in Table 100 and Table 101. The PHY PIB characteristics group, Table 100, contains information that is common to most implementations.

Table 100—PHY PIB characteristics group parameters

Managed Object	Octets	Definition	Access
PHYPIB_Type	1	0x01 = mmWave PHY	Read/write
PHYPIB_Mode	1	bit 1 = SC PHY bit 2 = HSI PHY bit 3 = AV PHY bit 4 = OOK mode bit 5 = DAMI mode bit 6-8 = Reserved A bit is set to one if the associated PHY is supported, and is set to zero otherwise.	Read/write
PHYPIB_RegDomainsSupported	Variable	One octet for each regulatory domain supported, as defined for PHYPIB_CurrentRegDomain.	Read/write
PHYPIB_CurrentRegDomain	1	0x00 = European Telecommunications Standards Institute (ETSI) 0x01 = Federal Communications Commission (FCC) 0x02 = Industry Canada (IC) 0x03 = Association of Radio Industries and Businesses (ARIB)	Read/write

Table 100—PHY PIB characteristics group parameters *(continued)*

Managed Object	Octets	Definition	Access
PHYPIB_DataRateVector	Variable	One octet for each supported MCS. The three msbs indicate the mmWave PHY mode, as in PHYPIB_Mode, and the last five lsbs contain the MCS supported for that mode using the encoding for that PHY mode.	Read/write
PHYPIB_NumChannelsSupported	Variable	Value = 0x04, as defined in 12.1.5.	Read/write
PHYPIB_CurrentChannel	1	Indicates the channel that is currently being used, as defined in 12.1.5.	Read/write
PHYPIB_CCAThreshold		The CCA threshold in dBm, encoded in two's complement format. The value is implementation dependent but no larger than the value listed in 12.1.12.8.	Read/write
PHYPIB_FrameLengthMax	2	pMaxFrameBodySize.	Read/write

The PHY PIB implementation group, Table 101 contains information that is more characteristic of a particular PHY implementation than of the PHY as a whole.

Table 101—PHY PIB Implementation group parameters

Managed Object	Octets	Definition	Access
PHYPIB_DiversitySupported	1	Numeric entry that indicates the number of antennas that are available.	Read/write
PHYPIB_MaxTXPower	1	The maximum TX power that the DEV is capable of using, 7.4.11, implementation dependent.	Read/write
PHYPIB_TXPowerStepSize	1	The step size for power control supported by the DEV, 7.4.12, implementation dependent.	Read/write
PHYPIB_NumPMLevels	1	Number of power management levels supported. The range is 1 to 8 and the value is implementation dependent.	Read/write
PHYPIB_PMLevelReturn	Variable	Table of vectors with number of entries given by PHYPIB_NumPMLevels. Each vector is the time required to change between power saving states of the PHY. Vector number 0 is the time required to change the PHY from the off state to a state where it is ready to receive commands. Other values are implementation dependent.	Read/write

12.2 Single Carrier Mode of mmWave PHY

The Single Carrier Mode in mmWave PHY (SC PHY) provides three classes of modulation and coding schemes (MCSs) targeting different wireless connectivity applications. Class 1 is specified to address the low-power low-cost mobile market while maintaining a relatively high data rate of up to 1.5 Gb/s. Class 2 is specified to achieve data rates up to 3 Gb/s. Class 3 is specified to support high performance applications with data rates in excess of 5 Gb/s. Table 102 summarizes the MCS classes. There are two mandatory MCSs for all SC DEVs except for the optional OOK/DAMI mode DEVs, the common mode signaling (CMS) and the mandatory PHY rate (MPR). The optional OOK/DAMI modes as described in 12.2.8 are allowed for low complexity SC DEVs. The CMS shall be the base rate for the SC PHY. The CMS is used for transmission of beacon and command frame in the association procedure as well as for transmission of command frame and training sequence in the beam forming procedure.

Table 102—MCS categorization for the SC PHY

Class	Categorization
Class 1	Data rates < 1.5[a] Gb/s
Class 2	1.5 Gb/s < Data rates < 3 Gb/s
Class 3	Data rates >3 Gb/s

[a]The data rates are based on subblock length of 512 and pilot word length of 64.

The SC PHY is specified with a high degree of flexibility in order to allow implementors the ability to optimize for different applications. The SC PHY supports operation in NLOS as well as LOS, with or without equalization. The data is divided into blocks, each block is divided into subblocks as described in 12.2.3.4.1, and each subblock may be equalized in time, frequency or in hybrid time-frequency domain. A subblock consists of pilot word and data.

The SC PHY supports $\pi/2$ BPSK, $\pi/2$ QPSK, $\pi/2$ 8-PSK and $\pi/2$ 16-QAM modulations. Two main FEC schemes are specified, RS block codes and LDPC block codes. The RS(255,239) and the shortened RS(33,17) block codes are mandatory, whereas all the other FECs are optional.

12.2.1 PHY operating specifications of SC PHY

12.2.1.1 Channelization

The RF channels are defined in Table 96. A compliant implementation shall support at least 1 channel from the channels allocated for operation by its corresponding regulatory body.

The PHYPIB_CurrentChannel is the CHNL_ID of the current channel. For the purpose of the Remote Scan Request and Remote Scan Response commands, as described in 7.5.7.3 and 7.5.7.4, respectively, the Channel Index field is the CHNL_ID in Table 96.

12.2.1.2 Scanning channels

When a DEV is scanning to start a piconet, it should scan all channels it supports in respective regions to decrease the probability of choosing an occupied channel.

12.2.2 Modulation, forward error correction and spreading

12.2.2.1 MCS dependent parameters

The chip rate for all SC PHY MCS is given in Table 107. The MCS dependent parameters shall be set according to Table 103. The data rates in the table are approximate and are calculated to three significant figures. The CMS and MPR are part of Class 1, and are listed as MCS identifier 0 and 3, respectively.

Table 103—MCS dependent parameters

MCS class	MCS identifier	Data rate (Mb/s) with pilot word length = 0	Data rate (Mb/s) with pilot word length = 64	Modulation	Spreading factor, L_{SF}	FEC type
Class1	0	25.8 (CMS)	—	π/2 BPSK/(G)MSK[a]	64	RS(255,239)
	1	412	361		4	
	2	825	722		2	
	3	1650 (MPR)	1440		1	
	4	1320	1160	π/2 BPSK/(G)MSK	1	LDPC(672,504)
	5	440	385	π/2 BPSK/(G)MSK	2	LDPC(672,336)
	6	880	770		1	
Class2	7	1760	1540	π/2 QPSK	1	LDPC(672,336)
	8	2640	2310	π/2 QPSK	1	LDPC(672,504)
	9	3080	2700	π/2 QPSK	1	LDPC(672,588)
	10	3290	2870	π/2 QPSK	1	LDPC(1440,1344)
	11	3300	2890	π/2 QPSK	1	RS(255,239)
Class3	12	3960	3470	π/2 8-PSK	1	LDPC(672,504)
	13	5280	4620	π/2 16-QAM	1	LDPC(672,504)

[a]The standard supports the use of MSK/GMSK modulations with appropriate filtering and pre-coding as an alternative way of generating π/2 BPSK waveform signals for the SC PHY. The π/2 BPSK is equivalent to pre-coded (G)MSK as described in 12.2.2.5.1.

The FEC rate, R_{FEC}, is for an RS(n, m) code or an LDPC(n, m) code is m/n. The subblock length $L_{subblock}$ shall be 512 chips. The pilot word length L_{PW} shall be 0, 8, or 64 chips. For subblocks with L_{PW} = 8, the effective length of the pilot word and data is equivalent to that with L_{PW} = 64, as described in 12.2.3.4.1.

The MCS dependent parameters for the optional OOK/DAMI modes are given in Table 104.

12.2.2.2 Header rate dependent parameters

The base header rate dependent parameters shall be set according to Table 105. The base headers use a shortened RS code, as defined in 12.2.3.2.3.

For the MR and HR headers, the header subblock is divided into eight sub-subblocks. The first sub-subblock is a pilot word with L_{PW} = 64. This is followed by six sub-subblocks, each with a prepended pilot word L_{PW} = 8. The last sub-subblock consists of eight pilot words of L_{PW} = 8. This gives the effective pilot word length of L_{PW} = 176.

There are three types of MAC subheaders, the standard aggregation subheader, the low-latency EEP subheader, and the low-latency UEP subheader, as defined in 7.2.8. When the MAC subheader is present, the subheader shall be modulated with π/2 BPSK/(G)MSK and the header rate-dependent parameters shall be set according to Table 106. The subheader subblock is the same as for the base header, giving the same effective pilot word length of 176 chips. The MAC subheaders for standard aggregation use a shortened RS code for the FEC, as defined in the 12.2.3.2.5. If the base header uses either the CMS or MR, the MAC

Table 104—MCS dependent parameters for optional OOK/DAMI modes

Device Type	MCS identifier	Data rate[a] (Mb/s)	Modulation	Spreading factor	FEC type	Support for CMS
PNC-capable DEVs	OOK	25.8 (CMS)	π/2 BPSK/ (G)MSK	64	RS(255,239)	Mandatory
		818	OOK	2		
		1640		1		
	DAMI	25.8 (CMS)	π/2 BPSK/ (G)MSK	64		
		3270	DAMI	1		
Non PNC-capable DEVs	OOK	818	OOK	2	RS(255,239)	Not mandatory
		1640		1		
	DAMI	3270	DAMI	1		

[a]The data rate of CMS is calculated in a manner similar to Table 103. The data rates of two OOK modes are calculated based on pilot word design as in D2.5. The data rate of DAMI mode is calculated based on subblock length of 512 and pilot word length of 4. The data rates are approximate and are rounded to three significant figures.

Table 105—Base header rate dependent parameters

Header class	Header rate (Mb/s)	Modulation scheme	Spreading factor, L_{SF}	Pilot word length (chips), L_{PW}	Coded bits per subblock, L_{CBPS}	Number of occupied subblocks, $N_{subblock\_hdr}$	Number of stuff bits, L_{STUFF}
CMS rate	12.5	π/2 BPSK/ (G)MSK	64	0	8	33	0
Medium rate (MR)	82.5	π/2 BPSK/ (G)MSK	6	176 (effective)	56	5	16
High rate (HR)	206	π/2 BPSK/ (G)MSK	2		168	2	72

subheader shall be sent using the subheader MR. If the base header uses HR, the MAC subheader shall be sent using subheader HR. For the standard aggregation frames, each subframe may use a different MCS. For the low-latency frames, the MAC subheaders and the MSDU subheaders shall use the same MCS for the MAC frame body, thus the MCS remains the same within a low-latency aggregation frame.

12.2.2.3 Timing-related parameters

Table 107 lists the general timing parameters associated with the SC PHY.

12.2.2.4 Frame-related parameters

The frame parameters associated with the PHY are listed in Table 108 where CEIL is the ceiling function, which returns the smallest integer value greater than or equal to its argument. The maximum frame duration occurs when the number of octets in the PHY Payload field is 1048576.

Table 106—MAC subheader rate dependent parameters for standard aggregation

Frame type	Subheader class	Header rate[a] (Mb/s)	Spreading factor, L_{SF}	Coded bits per subblock, L_{CBPS}	Number of occupied subblocks	Number of stuff bits, L_{STUFF}
Non-secure frame	Subheader MR	131	6	56	9	16
	Subheader HR	394	2	168	3	16
Secure frame	Subheader MR	132	6	56	10	32
	Subheader HR	330	2	168	4	144

[a]Only the parameters for the MAC subheader for standard aggregation is presented in the table. The MAC subheader for low-latency aggregation uses the same MCS as for the MAC frame body and therefore is not listed in this table.

Table 107—Timing-related parameters

Parameter	Description	Value			Unit	Formula
R_c	Chip rate	1760			Mchip/s	
T_c	Chip duration	~0.568			ns	$1/R_c$
$L_{subblock}$	Subblock length	512			chips	
L_{PW}	Pilot word length	0	8[a]	64	chips	
T_{PW}	Pilot word duration	0	4.5	~37.0	ns	
L_{DC}	Length of data chips per subblock	512	56	448	chips	
$T_{subblock}$	Subblock duration	~290.9			ns	$L_{subblock} \times T_c$
$R_{subblock}$	Subblock rate	~3.44			MHz	$1/T_{subblock}$

[a]Details of the subblock with pilot word length 8 are given in 12.2.2.1 and 12.2.2.2.

Table 108—Frame-related parameters

Parameter	Description	Value
N_{SYNC}	Number of code repetitions[a] in the SYNC sequence	14
T_{SYNC}	Duration of the SYNC sequence	~1 µs
N_{SFD}	Number of code repetitions in the SFD	4
T_{SFD}	Duration of the SFD	~0.29 µs
N_{CES}	Number of code repetitions in the CES	9
T_{CES}	Duration of the CES	~0.65 µs
N_{pre}	Number of code repetitions in the PHY preamble	27

Table 108—Frame-related parameters *(continued)*

Parameter	Description	Value
T_{pre}	Duration of the PHY preamble	~1.96 µs
L_{hdr}	Length of the base header in octets	33
$N_{subblock\_hdr}$	Number of subblocks in the base frame header	$CEIL[L_{hdr} \times 8 \times L_{SF} / (L_{subblock} - L_{PW}{}^{b})]$
T_{hdr}	Duration of the base frame header	$N_{subblock\_hdr} \times T_{subblock}$
$L_{payload}$	Length of frame payload in octets	variable
L_{FCS}	Length of FCS in octets	4
L_{MFB}	Length of the MAC frame body in octets	$L_{payload} + L_{FCS}$
N_{PCES}	Number of code repetitions in the PCES	9
T_{PCES}	PCES duration	0.65 µs
N_{PCES_frame}	Number of PCESs per frame	$CEIL[(L_{MFB} \times 8) / (L_{subblock} \times 64)]–1$
$T_{PCES\_interval}$	Interval of PCES insertion	$T_{subblock} \times 64 + T_{PW}$
L_{CBPS}	Number of coded bits per subblock in the MAC frame body	$(L_{subblock} - L_{PW}{}^{c}) / L_{SF}$
$N_{subblock\_MFB}$	Number of subblocks in the MAC frame body	$CEIL[(L_{MFB} \times 8) / (R_{FEC} \times L_{CBPS})]$
T_{MFB}	Duration of the MAC frame body	$N_{subblock\_MFB} \times T_{subblock}$
$T_{datafield}$	Duration of the PHY Payload field	$T_{MFB} + (N_{PCES\_frame} + 1) \times T_{PW} + N_{PCES\_frame} \times T_{PCES}$
T_{frame}	Duration of the frame	$T_{pre} + T_{hdr} + T_{datafield}$

[a]A code repetition is defined as a code with length of 128 chips.
[b]The value for L_{PW} given here is the effective pilot word length, as defined in 12.2.2.2.
[c]The value for L_{PW} given here is the effective pilot word length, as defined in 12.2.2.1.

NOTE—Regardless of the maximum frame length, no frame is allowed to exceed the timing boundaries, e.g., CAP end time or CTA end time.[1]

12.2.2.5 Modulation

After channel encoding and spreading, the bits shall be inserted into the constellation mapper. The constellations used for the SC PHY are illustrated in Figure 164. The mapping rules and diagrams for skewed constellation are given in 12.3.2.6.

12.2.2.5.1 π/2 BPSK/(G)MSK

The π/2-shift BPSK (π/2 BPSK) modulation is a binary phase modulation with π/2 phase shift counter-clockwise. Figure 164(a) shows the signal constellation of π/2 BPSK signals, and Figure 165(a) shows the π/2 BPSK modulator. The data chips, g_n, at the output of PCES inserter shown in Figure 177 are mapped to $d_1, d_2, \ldots, d_N$, where $d_N = 2 \times g_n - 1$. The d_n values are mapped onto constellation points z_n as follows:

$$z_n = j^n \times d_n,\ n = 1, 2, \ldots, N$$

[1]Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard.

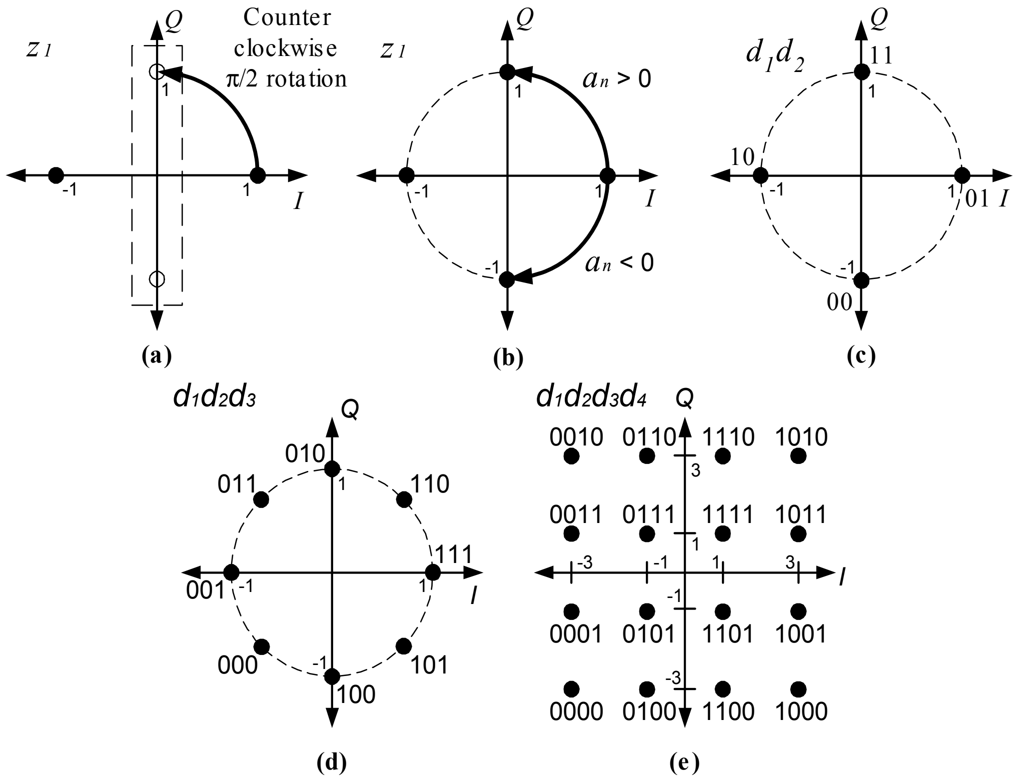

Figure 164—Constellation maps for modulations: (a) π/2 BPSK, (b) pre-coded (G)MSK, (c) π/2 QPSK, (d) π/2 8-PSK, (e) π/2 16-QAM

where j denotes π/2 phase rotation. c_n in Figure 165(a) denotes a complex envelope of filtered π/2 BPSK signal.

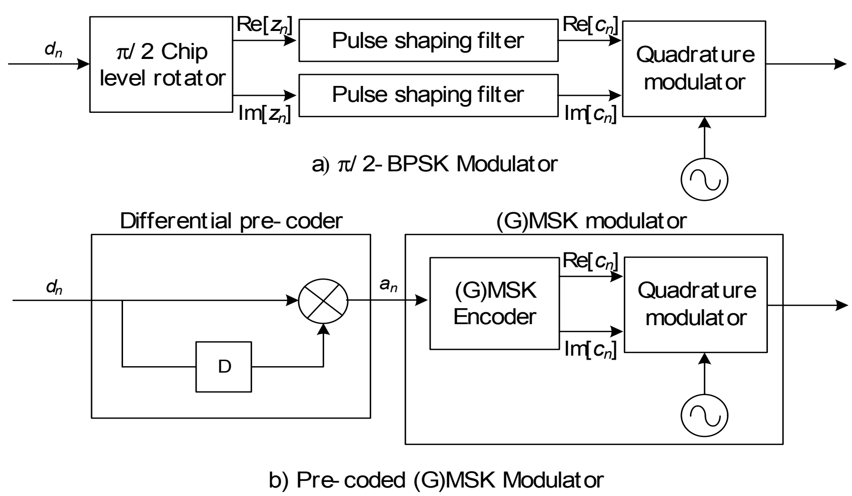

Figure 165—Possible π/2 BPSK realizations with (a) π/2 BPSK modulator and (b) pre-coded (G)MSK modulator

Pre-coded MSK/GMSK ((G)MSK) modulation is a continuous phase modulation by applying differential pre-coding before the (G)MSK modulation. The differential pre-coding is defined by the following:

$z_n = d_n \times d_{n-1}$, $n = 1, 2, \ldots, N$

where $d_0 = 1$.

The differential pre-coded bits are then fed to the (G)MSK encoder. The waveform of (G)MSK shown in Figure 164(b), which is the output of the (G)MSK encoder c_n, is approximately equivalent to that of filtered π/2 BPSK, if an appropriate pulse shaping filter for π/2 BPSK is employed. This is further illustrated in Figure 165(b). The filtered waveform of each modulation signal shall satisfy transmit PSD mask as in 12.1.6.

12.2.2.5.2 π/2 QPSK

The π/2 QPSK constellation diagram is shown in Figure 164(c), with four points equally spaced on a circle of radius one, representing four phases. QPSK shall encode 2 bits per symbol, with input bit d_1 being the earliest in the stream. The π/2 shift is employed to obtain a simple implementation aligning with the π/2 BPSK. The π/2 rotation is performed in the same manner as in 12.2.2.5.1. As illustrated in Figure 164(c), Gray encoding shall be employed. The normalization factor, K_{MOD} is 1.

12.2.2.5.3 π/2 8-PSK

The π/2 8-PSK constellation diagram is shown in Figure 164(d), equally spaced on a circle of radius one, representing eight phases. The π/2 8-PSK shall encode 3 bits per symbol, with input bit d_1 being the earliest in the stream. The π/2 rotation is performed in the same manner as in 12.2.2.5.1. Gray encoding shall be employed in the mapping of π/2 8-PSK. The normalization factor, K_{MOD} is 1.

12.2.2.5.4 π/2 16-QAM

The π/2 16-QAM constellation diagram is shown in Figure 164(e). The serial bit stream shall be divided into groups of four bits with input bit d_1 being the earliest in the stream. The π/2 rotation is performed in the same manner as in 12.2.2.5.1. The normalization factor for π/2 16-QAM constellation is $1/\sqrt{10}$. An approximate value of the normalization factor may be used, as long as the device conforms to the modulation accuracy requirements.

12.2.2.6 Forward Error Correction

The forward error correction (FEC) schemes are specified in this subclause. Support for RS block codes is mandatory, whereas support for LDPC block codes is optional.

12.2.2.6.1 Reed-Solomon block codes in GF(2^8)

The RS(255,239), which is the mother code, is used in payloads of CMS, MPR and MCS identifier 1 MCSs in Table 103. A shortened version of RS(255,239) is used for the base frame header and MAC subheader, as defined in 12.2.2.2.

The systematic RS code shall use the following generator polynomial [Equation (14)]:

$$g(x) = \prod_{k=1}^{16} (x + \alpha^k) \tag{14}$$

where $\alpha = 0x02$ is a root of the binary primitive polynomial $p(x)=1+x^2+x^3+x^4+x^8$. As notation, the element $M = b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x^1 + b_0$, has the binary representation $b_7b_6b_5b_4b_3b_2b_1b_0$, where b_7 is the msb and b_0 is the lsb.

The mapping of the information octets $\boldsymbol{m} = (m_{238}, m_{237}, \ldots, m_0)$ to codeword octets $\boldsymbol{c} = (m_{238}, m_{237}, \ldots, m_0, r_{15}, r_{14}, \ldots, r_0)$ is achieved by computing the remainder polynomial *r(x)* [Equation (15)]:

$$r(x) = \sum_{k=0}^{15} r_k x^k = x^{16} m(x) \bmod g(x) \tag{15}$$

where $m(x)$ is the information polynomial [Equation (16)]:

$$m(x) = \sum_{k=0}^{238} m_k x^k \tag{16}$$

and r_k, $k = 0, \ldots, 15$, and m_k, $k = 0, \ldots, 238$, are elements of GF(2^8). The message order is as follows: m_{238} is the first octet of the message and m_0 is the last octet of the message.

For a shortened RS(L_{inf} + 16, L_{inf}), 239-L_{inf} zero elements are appended to the incoming L_{inf} octet message as follows:

$m_k = 0$, $k = L_{inf}, \ldots, 238$

These inserted zero elements are not transmitted. A shift-register implementation of the RS encoder RS(L_{inf}+16, L_{inf}) is shown in Figure 166, with additions and multiplications over GF(2^8). After m_0 has been inserted into the shift register, the switch shall be moved from the message polynomial input connection to the shift register output connection (right-to-left).

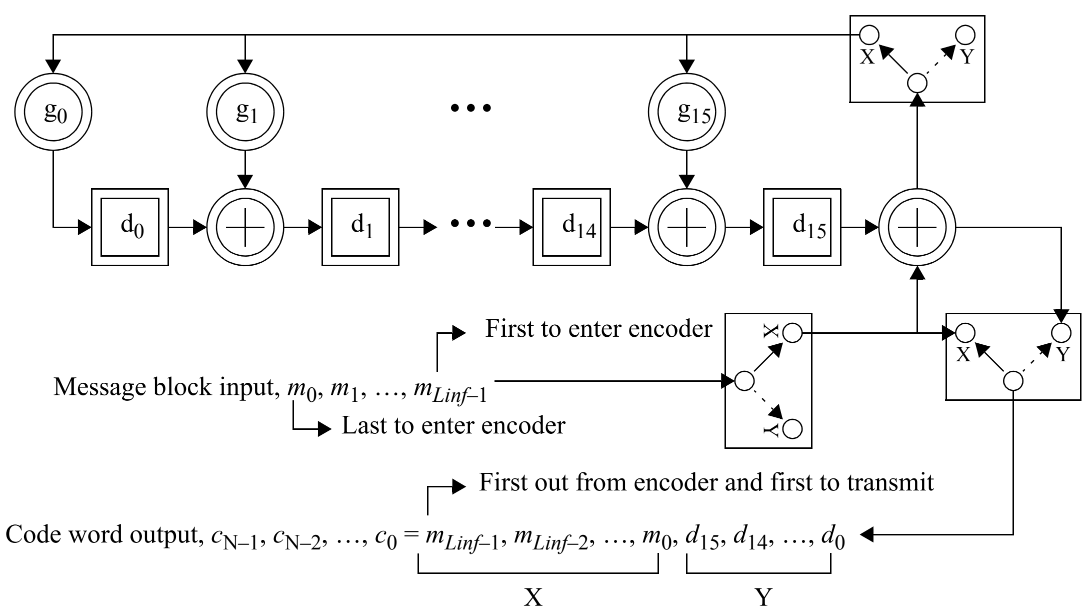

Figure 166—Reed Solomon encoder GF(2^8)

12.2.2.6.2 Irregular LDPC codes

The irregular LDPC codes are used as a high-performance error correction coding technique. The supported FEC rates, information block lengths, and codeword block lengths are described in Table 109.

Table 109—Irregular LDPC parameters

FEC rate, R_{FEC}	LDPC information block length (bits), L_{inf}	LDPC codeword block length (bits), L_{FEC}
1/2	336	672
3/4	504	672
7/8	588	672

The LDPC encoder is systematic, i.e., it encodes an information block of size k, $i = (i_0, i_1, \ldots, i_{(k-1)})$, into a codeword **c** of size n, $\mathbf{c} = (i_0, i_1, \ldots, i_{(k-1)}, p_0, p_1, \ldots, p_{(n-k-1)})$, by adding n-k parity bits obtained so that $\mathbf{Hc}^\mathbf{T} = 0$, where $\mathbf{H}$ is an $(n-k) \times n$ parity check matrix.

Each of the parity-check matrices can be partitioned into square subblocks (submatrices) of size $z \times z$ ($z = 21$). These submatrices are either cyclic-permutations of the identity matrix or null (all-zero) submatrices.

The cyclic-permutation matrix $\mathbf{p}^i$ is obtained from the $z \times z$ identity matrix by cyclically shifting the columns to the left by i elements. The matrix $\mathbf{p}^0$ is the $z \times z$ identity matrix.

In the following, an example of cyclic-permutation matrices with $z = 21$ is shown. The matrix $\mathbf{p}^1$ and $\mathbf{p}^2$ are produced by cyclically shifting the columns of the identity matrix $\mathbf{I}_{21\times21}$ to the left by 1 and 2 places, respectively.

$$p^0 = \begin{bmatrix} 1 & 0 & \ldots & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \ldots & 0 & \ldots & 0 & \ldots \\ 0 & \ldots & 0 & 1 & 0 \\ 0 & \ldots & \ldots & 0 & 1 \end{bmatrix}, p^1 = \begin{bmatrix} 0 & \ldots & \ldots & 0 & 1 \\ 1 & 0 & \ldots & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \ldots & 0 & 1 & 0 & 0 \\ 0 & \ldots & 0 & 1 & 0 \end{bmatrix}, p^2 = \begin{bmatrix} 0 & \ldots & 0 & 1 & 0 \\ 0 & \ldots & \ldots & 0 & 1 \\ 1 & 0 & \ldots & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

where all the above matrices have dimension 21 × 21.

Due to the cyclic permutation, $\mathbf{p}^{21} = \mathbf{p}^0 = \mathbf{I}_{21\times21}$.

Figure 167 displays the matrix permutation indices of parity-check matrices for all three FEC rates at block length $n = 672$ bits. The integer i denotes the cyclic-permutation matrix $\mathbf{p}^i$, as explained in above example. The '–' entries in the table denote null (all zero) submatrices.

For shortened LDPC operation, the k–l zero elements are appended to the incoming l message bits as follows: $r_i = 0$ for $i = l, l+1, \ldots, k-1$. These inserted zero elements are not transmitted.

12.2.2.6.3 Rate 14/15 LDPC code

The rate 14/15 LDPC(1440,1344) code is also systematic, i.e., the LDPC encoder encodes an information block of size k, $\mathbf{i} = (i_0, i_1, \ldots, i_{(k-1)})$, into a codeword **c** of size n, $\mathbf{c} = (i_0, i_1, \ldots, i_{(k-1)}, p_0, p_1, \ldots, p_{(n-k-1)})$, by adding $n - k$ parity bits obtained so that $\mathbf{Hc}^\mathrm{T} = 0$, where $\mathbf{H}$ is an $(n - k) \times n$ parity check matrix. Denote the 96×1440 parity check matrix as $\mathbf{H} = (h_{i,j})$, where $h_{i,j}$ consists of $\{0,1\}$, $0 \le i < 96$ and $0 \le j < 1440$. Table 110 shows the matrix elements whose values are '1' in the first 15 columns of parity check matrix.

(672,336), Code rate: 1/2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	-	-	-	5	-	18	-	-	-	-	3	-	10	-	-	-	-	-	-	5	-	-	-	-	-	-	-	5	-	7	-	-
2	0	-	-	-	-	-	16	-	-	-	-	6	-	-	-	0	-	7	-	-	-	-	-	-	-	10	-	-	-	-	-	19
3	-	-	6	-	7	-	-	-	-	2	-	-	-	-	9	-	20	-	-	-	-	-	-	-	-	-	19	-	10	-	-	-
4	-	18	-	-	-	-	-	0	10	-	-	-	-	16	-	-	-	-	9	-	-	-	-	-	4	-	-	-	-	-	17	-
5	5	-	-	-	-	-	18	-	-	-	-	3	-	10	-	-	5	-	-	-	-	-	-	-	-	-	-	-	-	-	7	-
6	-	0	-	-	-	-	-	16	6	-	-	-	0	-	-	-	-	-	7	-	-	-	-	-	-	-	-	-	19	-	-	-
7	-	-	-	6	-	7	-	-	-	-	2	-	-	-	-	9	-	20	-	-	-	-	-	-	-	-	-	-	-	10	-	-
8	-	-	18	-	0	-	-	-	-	10	-	-	-	-	16	-	-	-	-	9	-	-	-	-	-	-	-	-	-	-	-	17
9	-	5	-	-	-	-	-	18	3	-	-	-	-	-	10	-	-	5	-	-	4	-	-	-	-	5	-	-	-	-	-	7
10	-	-	0	-	16	-	-	-	-	6	-	-	-	0	-	-	-	-	-	7	-	4	-	-	-	-	-	10	-	19	-	-
11	6	-	-	-	-	-	7	-	-	-	-	2	9	-	-	-	-	-	20	-	-	-	4	-	19	-	-	-	-	-	10	-
12	-	-	-	18	-	0	-	-	-	-	10	-	-	-	-	16	9	-	-	-	-	-	-	12	-	-	4	-	17	-	-	-
13	-	-	5	-	18	-	-	-	-	3	-	-	-	-	-	10	-	-	5	-	-	-	-	-	-	-	5	-	-	-	-	-
14	-	-	-	0	-	16	-	-	-	-	6	-	-	-	0	-	7	-	-	-	-	-	-	-	10	-	-	-	-	-	-	-
15	-	6	-	-	-	-	-	7	2	-	-	-	-	9	-	-	-	-	-	20	-	-	-	-	-	19	-	-	-	-	-	-
16	18	-	-	-	-	-	0	-	-	-	-	10	16	-	-	-	-	9	-	-	-	-	-	-	-	-	-	4	-	-	-	-

(672,504), Code rate: 3/4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	0	-	-	5	-	18	16	-	-	-	3	6	10	-	-	0	-	7	-	5	-	-	4	4	-	10	-	5	-	-	-	-
2	-	18	6	-	7	-	-	0	10	2	-	-	-	16	9	-	20	-	9	-	4	12	-	-	4	-	19	-	-	-	-	-
3	5	0	-	-	-	-	18	16	6	-	-	3	0	10	-	-	5	-	7	-	4	-	-	4	5	-	10	-	19	-	-	-
4	-	-	18	6	0	7	-	-	-	10	2	-	-	-	16	9	-	20	-	9	-	4	12	-	-	4	-	19	-	10	-	-
5	-	5	0	-	16	-	-	18	3	6	-	-	-	0	10	-	-	5	-	7	4	4	-	-	-	5	-	-	-	-	-	-
6	6	-	-	18	-	0	7	-	-	-	10	2	9	-	-	16	9	-	20	-	-	-	4	12	19	-	-	-	-	-	-	-
7	-	-	5	0	18	16	-	-	-	3	6	-	-	-	0	10	7	-	5	-	-	4	4	-	10	-	5	-	7	-	19	-
8	18	6	-	-	-	-	0	7	2	-	-	10	16	9	-	-	-	9	-	20	12	-	-	4	-	19	-	4	-	17	-	10

(672,588), Code rate: 7/8

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	0	18	6	5	7	18	16	0	10	2	3	6	10	16	9	0	20	7	9	5	4	12	4	4	4	10	19	5	10	-	-	-
2	5	0	18	6	0	7	18	16	6	10	2	3	0	10	16	9	5	20	7	9	4	4	12	4	5	4	10	19	19	10	-	-
3	6	5	0	18	16	0	7	18	3	6	10	2	9	0	10	16	9	5	20	7	4	4	4	12	19	5	4	10	17	19	10	-
4	18	6	5	0	18	16	0	7	2	3	6	10	16	9	0	10	7	9	5	20	12	4	4	4	10	19	5	4	7	17	19	10

Figure 167—Matrix permutation indices of structured parity check matrices

Table 110—Positions of ‘1’s in the first 15 columns of parity check matrix H (codeword block length L_{FEC} = 1440)

$h_{0,0}$ $h_{1,0}$ $h_{4,0}$
$h_{32,1}$ $h_{34,1}$ $h_{39,1}$
$h_{64,2}$ $h_{70,2}$ $h_{78,2}$
$h_{8,3}$ $h_{18,3}$ $h_{95,3}$
$h_{31,4}$ $h_{42,4}$ $h_{54,4}$
$h_{63,5}$ $h_{76,5}$ $h_{91,5}$

Table 110—Positions of '1's in the first 15 columns of parity check matrix H (codeword block length L_{FEC} = 1440) *(continued)*

$h_{14,6}$ $h_{45,6}$ $h_{94,6}$
$h_{30,7}$ $h_{47,7}$ $h_{83,7}$
$h_{17,8}$ $h_{62,8}$ $h_{80,8}$
$h_{28,9}$ $h_{48,9}$ $h_{82,9}$
$h_{22,10}$ $h_{60,10}$ $h_{81,10}$
$h_{27,11}$ $h_{49,11}$ $h_{84,11}$
$h_{7,12}$ $h_{53,12}$ $h_{77,12}$
$h_{19,13}$ $h_{44,13}$ $h_{85,13}$
$h_{6,14}$ $h_{46,14}$ $h_{75,14}$

For $15 \le j$, the matrix element can be obtained by using Equation (17).

$$h_{i,j} = h_{\mathrm{mod}(i + \lfloor j/15 \rfloor,\ 96),\ \mathrm{mod}(j,\ 15)} \tag{17}$$

where mod(x, y) is the modulo function and is defined as $x - n \times y$ where n is the nearest integer less than or equal to x/y.

The LDPC(1440, 1344) code is a quasi-cyclic code such that every cyclic shift of a codeword by 15 symbols yields another codeword.

For shortened LDPC operation, the $k–l$ zero elements are appended to the incoming l message bits as follows: $r_i = 0$ for $i = l, l+1, \ldots, k–1$. The message order is $r_{k–1}$ as the first bit of the message with r_0 as the last bit of the message. These inserted zero elements are not transmitted.

12.2.2.7 Stuff bits

Stuff bits shall be added to the end of the encoded MAC frame body if the number of the encoded data bits is not an integer multiple of the length of the data portion in the subblock. The number of stuff bits is computed for each subframe if standard aggregation is employed. The calculation of stuff bits is as follows.

In the encoded MAC frame body, the number of FEC codewords, N_{FEC} is given by Equation (18).

$$N_{FEC} = \mathrm{CEIL}[(L_{MFB} \times 8)/(L_{FEC} \times R_{FEC})] \tag{18}$$

where L_{FEC} is the FEC codeword length, L_{MFB} is the length of the MAC frame body in octets, and R_{FEC} is the FEC rate. The FEC codeword length, L_{FEC}, is 2040 for the RS FEC specified in 12.2.2.6.1, 672 for the irregular LDPC specified in 12.2.2.6.2, and 1440 for the rate 14/15 LDPC specified in 12.2.2.6.3.

The encoded MAC frame body shall be concatenated with stuff bits of length L_{STUFF} so that the resulting MAC frame body is aligned on the subblock symbol boundary. The stuff bits shall be set to zero and then scrambled using the continuation of the scrambler sequence that scrambled the MAC frame body in 12.2.2.10. The length of bits in the encoded MAC frame body, L_{ebits} is given by Equation (19).

$$L_{ebits} = 8 \times L_{MFB} + N_{FEC} \times (1 - R_{FEC}) \times L_{FEC} \tag{19}$$

The number of subblocks in the encoded MAC frame body, $N_{subblock\text{-}encMFB}$, and the length of stuff bits, L_{STUFF}, are given by Equation (20) and Equation (21).

$$N_{subblock–encMFB} = \mathrm{CEIL}(L_{ebits}/L_{CBPS}) \tag{20}$$

$$L_{STUFF} = N_{subblock–encMFB} \times L_{CBPS} - L_{ebits} \tag{21}$$

where L_{CBPS} is the number of coded bits per subblock as given in Table 111 for each MCS.

Table 111—Rate dependent bits per symbol

MCS identifier	Coded bits per subblock, L_{CBPS} (pilot word length = 0)	Coded bits per subblock, L_{CBPS} (pilot word length = 64)
0	8	--
1	128	112
2	256	224
3	512	448
4	512	448
5	256	224
6	512	448
7	1024	896
8	1024	896
9	1024	896
10	1024	896
11	1024	896
12	1536	1344
13	2048	1792

For the stuff bits in the frame headers, the values are given in Table 105 and Table 106.

12.2.2.8 Code spreading

To increase robustness in header and MAC frame body, Golay and pseudo random binary sequence (PRBS) codes by linear feedback shift register (LFSR) are applied for code spreading. The following two categories of spreading are defined:

1) For spreading factor of 64, Golay sequences shall be used.
2) For Class 1 MCSs with spreading factor of 2 and 4 and for headers with spreading factor of 2 and 6, the LFSR shall be used.

12.2.2.8.1 Golay sequences

In the base rate mode, the frame header and MAC frame body shall be spread as shown in Figure 160. The Golay sequences for spreading factor 64 are given in Table 98.

12.2.2.8.2 PRBS generation with LFSR

For a spreading factor of length 2, 4 or 6, the data bits shall be spread with a PRBS generated using an LFSR, as shown in Figure 168. Since the output of the spreader is a factor of L_{SF} larger than the input, the input shall hold while the feedback and output clock.

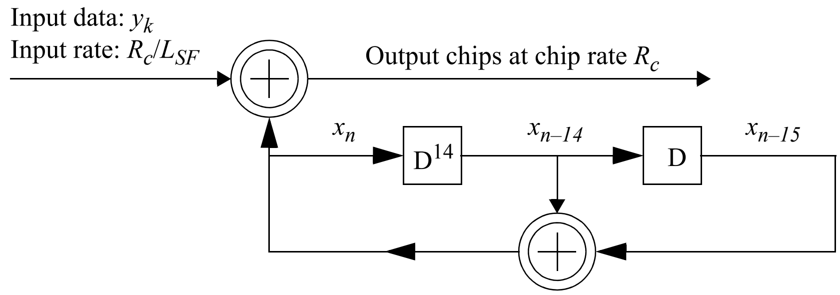

Figure 168—PRBS generation by LFSR

The 15 bit seed value of the LFSR shall be: $[x_{-1}, x_{-2}, \ldots, x_{-15}]$ = [0101 0000 0011 111].

12.2.2.9 Unequal Error Protection

The UEP MCSs for SC PHY are shown in Table 112. UEP is an optional function. The data rate listed in the table is approximate and is rounded to three significant figures.

Table 112—UEP MCS for SC PHY

UEP MCS	Modulation	FEC		Data rate (Mb/s)	Supported UEP Type
		msb	lsb		
0b00001	π/2 BPSK	RS (255,239)		1420	UEP Type 1 and UEP Type 2
0b00010		LDPC (672,336)		756	UEP Type 1 and UEP Type 2
0b00011		LDPC (672,504)		1130	UEP Type 1 and UEP Type 2
0b00100	π/2 QPSK	LDPC (672,336)		1510	UEP Type 1 and UEP Type 2
0b00101		LDPC (672,504)		2270	UEP Type 1 and UEP Type 2
0b00110		LDPC (672,588)		2650	UEP Type 1 and UEP Type 2
0b00111	π/2 QPSK	LDPC(672,336)	LDPC(672,504)	2040	UEP Type 3
0b01000		LDPC(672,504)	LDPC(672,588)	2650	UEP Type 3

In UEP Type 3, when the transmitter chooses one of the UEP MCS types, the transmitted data are divided into two groups in octet, msb and lsb, as shown in Figure 169. The msb and lsb groups are processed in parallel to adopt different LDPC coding rates. The outputs of the two LDPC encoders are multiplexed to generate a single bit stream for the symbol mapping.

The bit multiplexing/interleaving method depends on the msb and lsb coding rates as illustrated in Figure 170. When LDPC(672,336) is used for msb group and LDPC(672,504) is used for lsb group, the encoded bits shall be multiplexed every 10 bits with 6 bits as the encoded msbs and 4 bits as the encoded

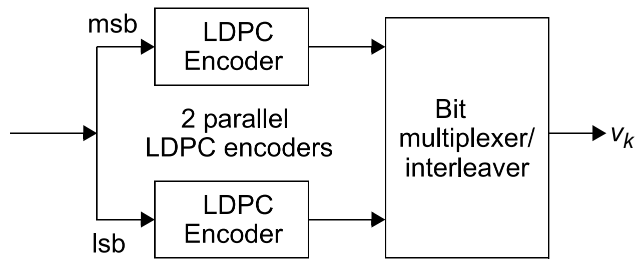

Figure 169—Parallel encoders for UEP

lsbs. A1, A2, A3, A4, A5, A6 are used in an increasing order in time to label the 6 encoded msbs from encoder LDPC(672,336), while B1, B2, B3, B4 are used in an increasing order in time to label the 4 encoded lsbs from encoder LDPC(672,504). The bit multiplexing/interleaving shall be performed such that the output pattern is A1, B1, A2, B2, A3, A4, B3, A5, B4, A6 with A1 being the earliest bit at the input of the symbol mapper while A6 being the latest, as illustrated in the upper part of Figure 170.

When LDPC(672,504) is used for msb group and LDPC(672,588) is used for lsb group, the encoded bits shall be multiplexed every 13 bits with 7 bits as the encoded msbs and 6 bits as the encoded lsbs. A1, A2, A3, A4, A5, A6, A7 are used in an increasing order in time to label the 7 encoded msbs from encoder LDPC(672,504), while B1, B2, B3, B4, B5, B6 are used in an increasing order in time to label the 6 encoded lsbs from encoder LDPC(672,588). The bit multiplexing/interleaving shall be performed such that the output pattern is A1, B1, A2, B2, A3, B3, A4, B4, A5, B5, A6, B6, A7 with A1 being the earliest bit at the input of the symbol mapper while A7 being the latest, as illustrated in the lower part of Figure 170.

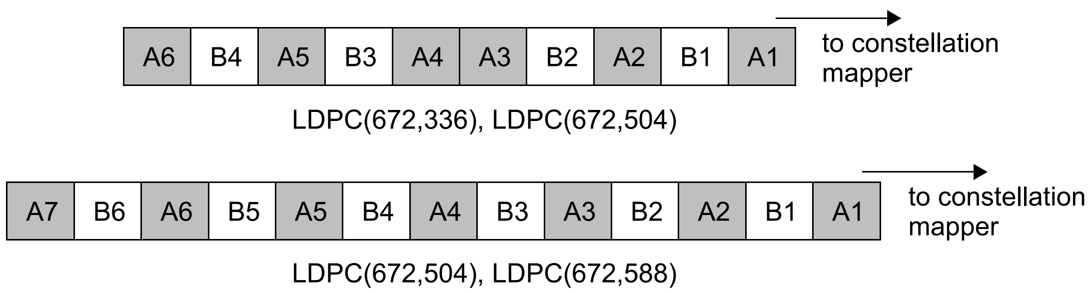

Figure 170—UEP bit multiplexing/interleaving

The effect of unequal error protection can also be obtained by using a skewed constellation, which is shown in Figure 183. A longer distance in x-axis than in y-axis is given between the two symbols in the skewed constellation, so that more energy is given to the msb group. The mapping rules and the constellation are given in 12.3.2.6.

To apply the skewed constellation in the symbol mapping, both LDPC encoders in Figure 169 shall have the same coding rates among LDPC (672,336), LDPC (672,504), and LDPC (672,588), and the bit multiplexing/interleaving is 1:1 alternating, as illustrated in Figure 171.

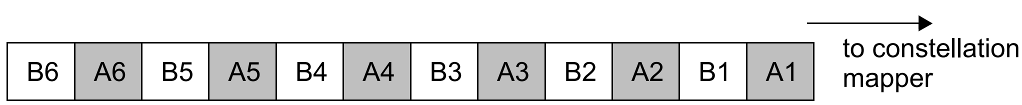

Figure 171—UEP bit multiplexing/interleaving for skewed constellations

12.2.2.10 Scrambling

The frames shall be scrambled by modulo-2 addition of the data with the output of a PRBS generator, as illustrated in Figure 168 with $L_{SF} = 1$.

The scrambler shall be used for the MAC header, HCS, MAC subheader, HCS (for subheader), and MAC frame body. The PHY preamble, PHY header, and RS bits shall not be scrambled. The polynomial for the PRBS generator used by the scrambler shall be as shown in Equation (22).

$$g(D) = 1 + D^{14} + D^{15} \tag{22}$$

where D is a single bit delay element. The polynomial forms not only a maximal length sequence, but also is a primitive polynomial. By the given generator polynomial, the corresponding PRBS, is generated as shown in Equation (23).

$$x_n = x_{n-14} \oplus x_{n-15},\ n = 0, 1, 2, \ldots \tag{23}$$

Equation (24) defines the initialization sequence.

$$x_{init} = [x_{-1}x_{-2}x_{-3}x_{-4}x_{-5}x_{-6}x_{-7}x_{-8}x_{-9}x_{-10}x_{-11}x_{-12}x_{-13}x_{-14}x_{-15}] \tag{24}$$

The scrambled data bits, s_n, are obtained as follows in Equation (25):

$$s_n = b_n \oplus x_n \tag{25}$$

where b_n represents the unscrambled data bits. The side-stream de-scrambler at the receiver shall be initialized with the same initialization vector, x_{init}, used in the transmitter scrambler. The initialization vector is determined from the Scrambler Seed ID field contained in the PHY header of the received frame.

The 15 bit seed value chosen shall be computed from the Scrambler Seed ID field as follows in Equation (26):

$$[x_{-1}x_{-2}\ldots x_{-15}] = [11010000101\ S1\ S2\ S3\ S4] \tag{26}$$

The seed identifier value is set to 0000 when the PHY is initialized and is incremented in a 4-bit rollover counter for each frame that is sent by the PHY. The value of the seed identifier that is used for the frame is sent in the PHY header.

For a Scrambler Seed ID field set to all zero, the first 16 bits should be as shown in Equation (27).

$$[x_0x_1\ldots x_{15}] = [0001111000111010] \tag{27}$$

The 15-bit seed value is configured as follows. At the beginning of each PHY frame, the register is cleared, the seed value is loaded, and the first scrambler bit is calculated. The first bit of the data of the MAC header is modulo-2 added with the first scrambler bit, followed by the rest of the bits in the MAC header, MAC subheader, and MAC frame body. The pilot word and pilot channel estimation sequences shall be excluded from the scrambling process.

12.2.3 SC PHY frame format

The SC PHY frame shall be formatted as illustrated in Figure 172.

PHY Payload field	Frame header	PHY preamble

Figure 172—SC PHY frame format

The Frame Header field for the PHY frame shall be formatted as illustrated in Figure 173.

Optional frame header				Base frame header				
Stuff bits	RS parity bits	HCS	MAC subheader	Stuff bits	RS parity bits	Base HCS	MAC header	PHY header

Figure 173— Frame header format

The PHY preamble is described in 12.1.12.4. The MAC header is defined in 7.2 and the MAC subheader is defined in 7.2.8. The PHY header is defined in 12.2.3.2.1, and the HCS is defined in 12.2.3.2.2. The PHY Payload field consisting of the MAC frame body, the PCES and stuff bits, is described in 12.2.3.3. The PCES is described in 12.2.3.4.2. The stuff bits are described in 12.2.2.7.

When transmitting a frame, the PHY preamble is sent first, followed by the base frame header, and then followed by the optional frame header, and finally the PHY Payload field.

12.2.3.1 PHY preamble

A PHY preamble shall be added prior to the frame header to aid receiver algorithms related to AGC setting, antenna diversity selection, timing acquisition, frequency offset estimation, frame synchronization, and channel estimation.

The PHY preamble shall be transmitted at the chip rate defined in Table 107.

In the CTAP, the CMS may be used. Transmission using the CMS in a CTAP shall be done using the frame consisting of the CMS preamble described in 12.1.12.4, the CMS header described in 12.2.2.2, and the CMS payload described in 12.2.2.1.

In the CAP, the CMS may be used. For this purpose, the SC data frame shall be specified as in 12.1.12, except that a different pattern, [+1 +1 +1 –1 –1 +1 –1] spread by $\mathbf{b}_{128}$, shall be used in the SFD field.

Figure 174 shows the structure of the PHY preamble.

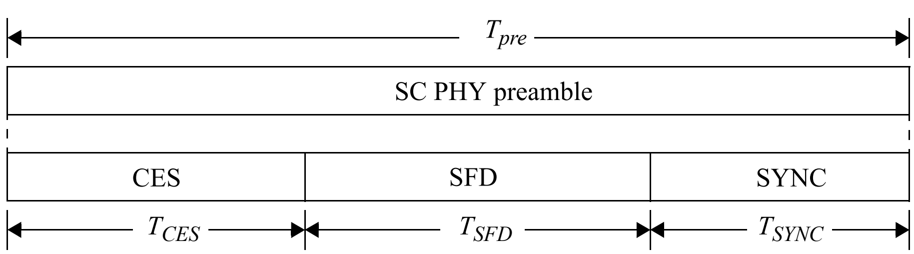

Figure 174—SC PHY preamble structure

12.2.3.1.1 Frame synchronization (SYNC)

The SYNC field is used for frame detection and uses a repetition of codes for a higher of robustness. The SYNC field shall consist of 14 code repetitions of $\mathbf{a}_{128}$. Table 99 shows the sequence for $\mathbf{a}_{128}$ used for the SYNC field.

12.2.3.1.2 Start frame delimiter (SFD)

The SFD is used to establish frame timing as well as the header rate, either MR or HR. The SFD for the two header rates are as follows:

— The MR header shall use an SFD with [+1 –1 +1 –1] spread by $\mathbf{a}_{128}$.
— The HR header shall use an SFD with [+1 +1 –1 –1] spread by $\mathbf{a}_{128}$.

12.2.3.1.3 Channel estimation sequence (CES)

The CES field, used for channel estimation, shall consist of [$\mathbf{b}_{128}$ $\mathbf{b}_{256}$ $\mathbf{a}_{256}$ $\mathbf{b}_{256}$ $\mathbf{a}_{256}$] where the right most sequence, $\mathbf{a}_{256}$, is first in time. The sequences $\mathbf{a}_{128}$, $\mathbf{a}_{256}$, and $\mathbf{b}_{256}$ are specified in 12.1.12.4.

12.2.3.2 Frame Header

A frame header shall be added after the PHY preamble. The frame header conveys information in the PHY and MAC headers necessary for successfully decoding the frame. The frame header consists of a base frame header followed by an optional frame header. The construction of the frame header is shown in Figure 175. The detailed process of the construction is as follows:

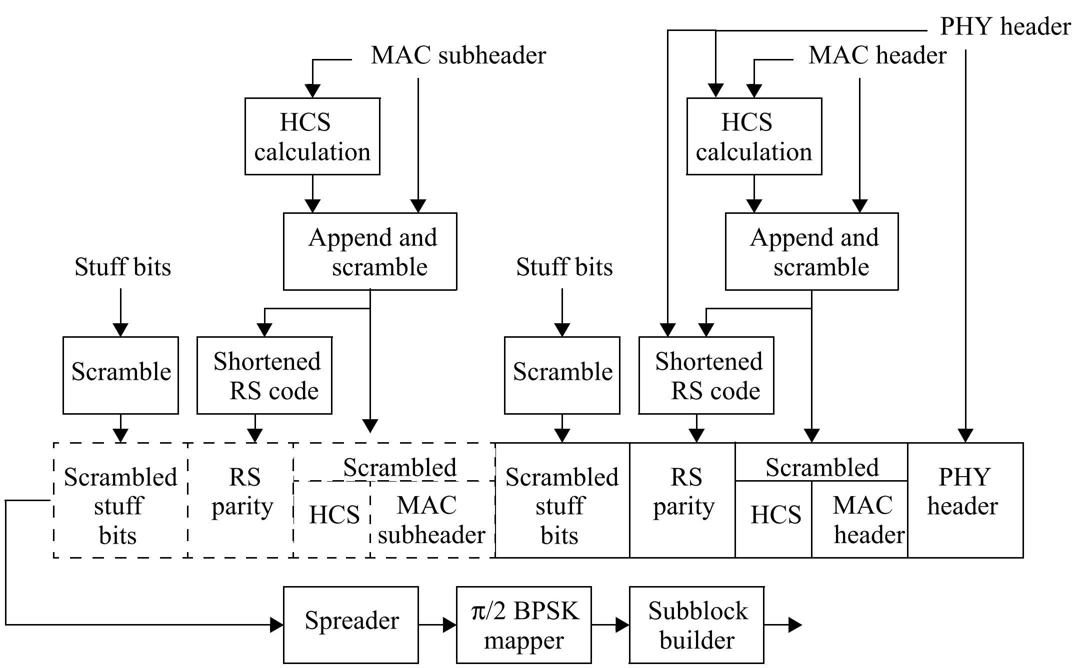

Figure 175—Frame header construction process

a) Form the base frame header as follows:
   1) Construct the PHY header based on information provided by the MAC
   2) Compute the HCS over the combined PHY and MAC headers
   3) Append the HCS to the MAC header

4) Scramble the combined MAC header and HCS, as described in 12.2.2.10
5) Compute the RS parity bits by encoding the concatenation of the PHY header, scrambled MAC header and scrambled HCS into a shortened RS block code, as described in 12.2.2.6.1
6) Form the base frame header by concatenating the PHY header, scrambled MAC header, scrambled HCS, RS parity bits, and scrambled stuff bits

b) Form the optional frame header as follows:
1) Compute the HCS over the MAC subheader
2) Append the HCS to the MAC subheader
3) Scramble the combined MAC subheader and HCS, as described in 12.2.2.10
4) Compute the RS parity bits by encoding the concatenation of scrambled MAC subheader and scrambled HCS into a shortened RS block code, as defined in 12.2.2.6.1
5) Form the optional frame header by concatenating the scrambled MAC subheader, scrambled HCS, RS parity bits, and scrambled stuff bits.

c) Form the frame header by concatenating the base frame header and optional frame header.

The resulting frame header shall be modulated as shown in Figure 175.

d) Spread the frame header, as described in 12.1.12.2.
e) Map the frame header onto π/2 BPSK/(G)MSK, as described in 12.2.2.5.1.
f) Build subblocks from the resulting frame header, as described in 12.2.3.4.1.

The LFSR for the spreader is reset between the header and payload.

12.2.3.2.1 SC PHY header

The SC PHY header shall be formatted as illustrated in Figure 176.

bits: 2	1	2	1	1	2	20	5	1	1	4
Reserved	PCES	Pilot word length	Low-latency mode	Beam tracking	Preamble type	Frame length	MCS	UEP	Aggregation	Scrambler seed ID

Figure 176—PHY header format for CMS and SC PHY

The Scrambler Seed ID field contains the scrambler seed identifier value, as defined in 12.2.2.10.

The Aggregation field shall be set to one if aggregation is used, it shall be set to zero otherwise.

The UEP field shall be set to one if UEP is used, it shall be set to zero if otherwise.

The MCS field shall be set according to the values in Table 113.

The Frame Length field shall be an unsigned integer equal to the number of octets in the MAC frame body of a regular frame, excluding the FCS, it shall be set to zero for an aggregated frame.

The Preamble Type field indicates the type of the PHY preamble used in the next frame as defined in Table 114.

The Beam Tracking field shall be set to one if training sequences for beam tracking are present following the current frame, it shall be set to zero otherwise.

Table 113—Modulation and coding scheme

MCS	MCS identifier
00000	0
00001	1
00010	2
00011	3
00100	4
00101	5
00110	6
00111	7
01000	8
01001	9
01010	10
01011	11
01100	12
01101	13
01110–11111	Reserved

Table 114—Preamble type field definition

Preamble type	Type of preamble used for next frame
00	CMS preamble
01	SC preamble
10–11	Reserved

The Low-latency Mode field shall be set to one if the frame is using the low-latency aggregation mode and it shall be set to zero otherwise. If the Low-latency Mode field is set to one, then the Aggregation field shall also be set to one.

The Pilot Word Length indicates the length of the pilot word used in the current frame and shall be encoded as defined Table 115.

The PCES field shall be set to one if the frame includes PCES and it shall be set to zero if otherwise.

12.2.3.2.2 Base header HCS

The combination of the PHY header and MAC header shall be protected with an ITU-T CRC-16 base HCS. The ITU-T CRC-16 is described in 11.2.9.

Table 115—Pilot word length field definition

Pilot word length	Pilot word length in subblock
00	64 (mandatory)
01	8
10	0 (mandatory)
11	Reserved

12.2.3.2.3 Base header FEC

The concatenation of the PHY header, scrambled MAC header and scrambled HCS shall use shortened systematic RS(n+16,n), for the FEC, where n is the number of octets in the combined PHY header, MAC header and HCS. The 128 RS parity bits are appended after the scrambled HCS as shown in Figure 175.

12.2.3.2.4 MAC subheader HCS

The MAC subheader shall be protected with an ITU-T CRC-16 HCS. This shall be computed using the method specified in 12.2.3.2.2.

12.2.3.2.5 MAC subheader FEC

The scrambled MAC subheader and scrambled HCS shall be encoded using a shortened RS(n+16,n), for the FEC, where n is the number of octets in the MAC subheader and HCS. The 128 RS parity bits are appended after the scrambled HCS, as shown in Figure 175.

NOTE—The length of the MAC subheader is different for secure and non-secure frames and hence the length of the shortened RS code will be different as well.

12.2.3.3 SC PHY Payload field

The SC PHY Payload field is the last component of the frame, and is constructed as shown in Figure 177.

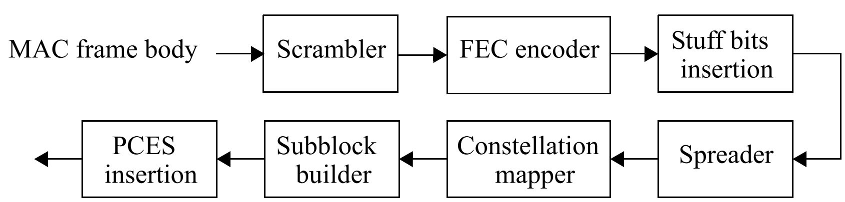

Figure 177—SC PHY Payload field construction process

The PHY Payload field shall be constructed as follows:

a) Scramble the MAC frame body according to 12.2.2.10.
b) Encode the scrambled MAC frame body as specified in 12.2.2.6.
c) Add stuff bits to the encoded and scrambled MAC frame body according to 12.2.2.7.
d) spread the encoded and scrambled MAC frame body using the spreading code as detailed in 12.1.12.2.
e) Map the resulting MAC frame body onto the appropriate constellation as described in 12.2.2.5.
f) Build subblocks from the resulting MAC frame body according to 12.2.3.4.1.
g) Insert PCES periodically as described in 12.2.3.4.2.

h) Apply a chip-level π/2 continuous rotation to the resulting MAC frame body as described in 12.2.2.5.

12.2.3.3.1 SC PHY Payload scrambling

The SC PHY payload shall use the scrambling process defined in 12.2.2.10.

12.2.3.3.2 Modulation

Modulation for the MAC frame body is defined in 12.2.2.5.

12.2.3.3.3 FEC

FEC for the MAC frame body is defined in 12.2.2.6.

12.2.3.3.4 Code spreading

Code spreading for the MAC frame body is defined in 12.1.12.2.

12.2.3.4 Pilot word and PCES

12.2.3.4.1 Subblocks and pilot word

Pilot words are used in SC PHY for timing tracking, compensation for clock drift and compensation for frequency offset error. Furthermore, pilot words act as a known cyclic prefix and enables frequency domain equalization if desired. In frequency domain equalization, the data is handled in the unit of subblocks. The building of the data blocks and subblocks is illustrated by Figure 178.

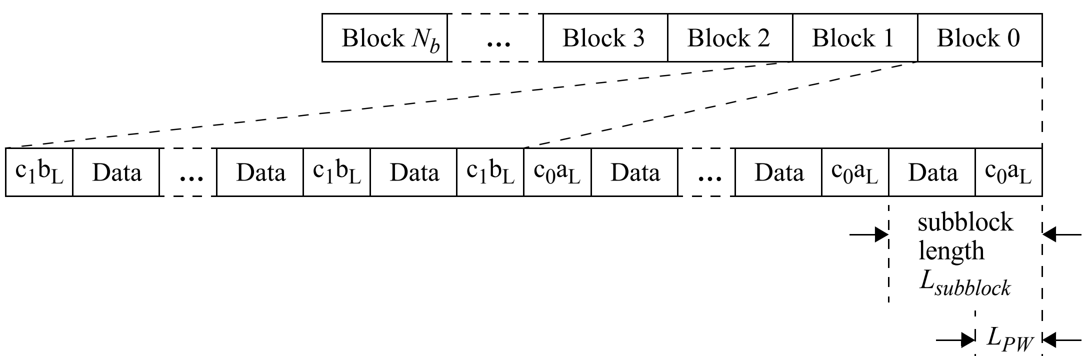

Figure 178—Frame format with pilot word

A block shall contain 64 subblocks with the exception of the last block (i.e., the N_b-th block). A subblock is formed by appending a pilot word to the data. The possible pilot word lengths are 0, 8 and 64. For pilot word lengths 0 and 64, the length of the data is $L_{DC} = L_{subblock} – L_{PW}$ symbols. For $L_{PW} = 8$, every subblock consists of 8 sub-subblocks each with $L_{PW} = 8$ and $L_{DC} = L_{subblock}/8 – L_{PW}$, thus giving the effective length of pilot word and data that is equivalent to that with $L_{PW} = 64$.

The Golay sequence for pilot word length 8 are given in Table 116 while for pilot word of length 64 they are shown in Table 98.

Even number blocks shall use pilot word sequences of type **a**. Odd number blocks shall use pilot word sequences of type **b**. Furthermore, an LFSR shall be used to change the polarity of pilot word from one block

Table 116—Golay sequences of length 8

Sequence name	Sequence value
$\mathbf{a}_8$	0xEB
$\mathbf{b}_8$	0xD8

to another. The LFSR used shall be the same as the LFSR described in 12.2.2.8.2 with the same initial state, but shall be run at the appropriate rate, i.e., one LFSR output per block. The last subblock of the block shall be followed by a pilot word as well. The pilot word is modulated with π/2 BPSK.

12.2.3.4.2 PCES

Pilot channel estimation sequence (PCES) insertion is an optional feature that allows a DEV to periodically re-acquire the channel. To add the PCES, the scrambled, encoded, spread and modulated MAC frame body is divided into data blocks. Each data block, as shown in Figure 178, shall be preceded by a PCES, with the exception of the first block. The duration and insertion interval of PCES are specified in Table 108. The PCES field shall be the CES field in the preamble defined in 12.2.3.1.3. Similarly, the PCES is modulated with π/2 BPSK.

12.2.4 Transmitter specifications

12.2.4.1 Error Vector Magnitude

A compliant transmitter shall have EVM values of less than those given in Table 117 for the MCS classes listed.

Table 117—EVM for SC PHY MCS classes

MCS	EVM (dB)
Class 1	–7
Class 2	–14
Class 3	–21

12.2.4.2 Transmit center frequency tolerance

The transmitted center frequency tolerance shall be ±25 μHz/Hz maximum.

12.2.4.3 Symbol rate

The SC PHY shall be capable of transmitting at the chip rate, as defined in Table 107, to within ±25 μs/s. The MAC parameter, pPHYClockAccuracy, shall be ±25 μs/s. The transmit center frequency and symbol clock frequency shall be derived from the same reference oscillator (locked).

12.2.4.4 Transmit power-on and power-down ramp

The transmit power-on ramp is defined as the time it takes for the RF power emitted by the compliant DEV to rise from less than 10% to greater than 90% of the maximum power to be transmitted in the frame.

The transmit power-on ramp shall be less than 9.3 ns.

The transmit power-down ramp is defined as the time it takes for the RF power emitted by the compliant DEV to fall from greater than 90% to less than 10% of the maximum power to be transmitted in the frame.

The transmit power-down ramp shall be less than 9.3 ns.

The transmit power ramps shall be constructed such that the emissions conform to the unwanted emissions specification defined in 12.1.3.

12.2.5 Receiver specifications

12.2.5.1 Error rate criterion

The error rate criterion shall be a frame error rate (FER) of less than 8% with a frame payload length of 2048 octets. The error rate should be determined at the PHY SAP interface after any error correction methods (excluding retransmission) required in the proposed device has been applied. The measurement shall be performed in AWGN channel.

12.2.5.2 Receiver sensitivity

The receiver sensitivity is the minimum power level of the incoming signal, in dBm, present at the input of the receiver for which the error rate criterion in 12.2.5.1 is met. The error ratio shall be determined after any error correction has been applied. A compliant DEV that implements the SC PHY shall achieve at least the reference sensitivity listed in Table 118.

Table 118—Reference sensitivity levels for MCS

MCS Identifier	Receiver sensitivity
0	–70 dBm
1	–61 dBm
2	–58 dBm
3	–55 dBm
4	–59 dBm
5	–65 dBm
6	–62 dBm
7	–58 dBm
8	–56 dBm
9	–54 dBm
10	–53 dBm
11	–52 dBm
12	–50 dBm
13	–46 dBm

12.2.5.3 Receiver maximum input level

The receiver maximum input level is the maximum power level of the incoming signal, in dBm, present at the input of the receiver for which the error rate criterion in 12.2.5.1 is met. A compliant receiver shall have a receiver maximum input level of at least –10 dBm for each of the modulation formats that the DEV supports.

12.2.5.4 Receiver clear channel assessment performance

A compliant receiver provides CCA capability by performing energy detection in the received signal bandwidth. The start of a valid preamble sequence at a receive level equal to or greater than the minimum sensitivity for the CMS, as described in 12.2.5.2, shall cause CCA to indicate medium busy with a probability of >90% within pCCADetectTime. The receiver CCA function shall in all circumstances report medium busy with any signal 20 dB above the minimum sensitivity for the CMS.

The CCA detection time shall be equal to pCCADetectTime. The CCA shall be maintained as busy until the end of the frame.

12.2.6 PHY layer timing

The values for the PHY layer timing parameters are defined Table 119.

Table 119—PHY layer timing parameters

PHY parameter	Value	Subclause
pPHYMIFSTime	0.2 µs, 0.5 µs (default), 2.0 µs	12.2.6.4
pPHYSIFSTime	0.2 µs, 2.0 µs, 2.5 µs (default)	12.2.6.3
pCCADetectTime	4 µs	12.2.5.4
pPHYChannelSwitchTime	100 µs	12.2.6.5

12.2.6.1 Interframe space

A conformant implementation shall support the IFS parameters, as described in 8.4.1, given in Table 120.

Table 120—IFS parameters

MAC parameter	Corresponding PHY parameter	Definition
MIFS	pPHYMIFSTime	12.2.6.4
SIFS	pPHYSIFSTime	12.2.6.3
pBackoffslot	pPHYSIFSTime + pCCADetectTime	12.2.6.3
BIFS	pPHYSIFSTime + pCCADetectTime	12.2.6.3, 12.2.5.4
RIFS	2 × pPHYSIFSTime + pCCADetectTime	12.2.6.3, 12.2.5.4

12.2.6.2 Receive-to-transmit turnaround time

The receive to transmit turnaround time shall be pPHYSIFSTime, including the power-up ramp specified in 12.2.4.4. The receive to transmit turnaround time shall be measured at the air interface from the trailing edge of the last symbol received until the first symbol of the PHY preamble is present at the air interface.

12.2.6.3 Transmit-to-receive turnaround-time

The transmit to receive turnaround time shall be less than pPHYSIFSTime, including the power-down ramp specified in 12.2.4.4.

12.2.6.4 Time between successive transmissions

The minimum time between successive transmissions shall be pPHYMIFSTime, including the power-up ramp specified in 12.2.4.4. The pPHYMIFSTime shall be measured at the air interface from the trailing edge of the last symbol transmitted until the first symbol of the PHY preamble is present at the air interface.

12.2.6.5 Channel switch

The channel switch time is defined as the time from the last valid bit is received at the antenna on one channel until the DEV is ready to transmit or receive on a new channel. The channel switch time shall be less than pPHYChannelSwitchTime.

12.2.7 PHY management for SC PHY

The PHY PIB comprises the managed objects, attributes, actions, and notifications required to manage the SC PHY layer of a DEV.

12.2.7.1 Maximum frame size

The maximum frame length allowed, pMAXFrameBodySize, shall be 8388608 octets. This total includes the MAC subheader and the MAC frame body, but not the PHY preamble, base header, (PHY header, MAC header and HCS). The maximum frame length also does not include the stuff bits.

12.2.7.2 Maximum transfer unit size

The maximum size data frame passed from the upper layers, pMaxTransferUnitSize, shall be 8388576 octets. If security is enabled for the data connection, the upper layers should limit data frames to 8388576 octets minus the security overhead as defined in 7.3.4.2, 7.2.8.1.2, or 7.2.8.2.2.

12.2.7.3 Minimum fragment size

The minimum fragment size, pMinFragmentSize, allowed with the SC PHY shall be 512 octets.

12.2.8 Optional OOK/DAMI modes

Besides the MCS classes in 12.2.2.1, optional low complexity and low power consumption MCSs, which are important for SC applications, may be employed within child piconets. As optional modes, OOK and DAMI may be employed for these applications. This subclause describes the OOK and DAMI MCSs should an implementor chose to implement these optional modes.

All PNC-capable OOK/DAMI DEVs shall be able to transmit and receive CMS signals.

If a PNC-capable OOK/DAMI DEV starts an independent piconet, it shall start the piconet in SC mode. OOK/DAMI PNC-capable DEVs may use OOK/DAMI signals in CTAs allocated for child piconet to support communication with non-PNC-capable OOK/DAMI DEVs. Details on child piconet creation and usage are described in D2.2.

The summary of the MCS for OOK and DAMI is given in Table 104. All PNC-capable OOK/DAMI DEVs shall transmit CMS beacons and conduct CP in CMS, with π/2 BPSK and RS(255,239). These PNC-capable OOK/DAMI DEVs may create child piconet for respective non-PNC-capable DEVs by using respective MCS-formatted signals in Table 104. For OOK non-PNC-capable DEVs, OOK modulation and RS(255,239) shall be used. For DAMI non-PNC-capable devices, DAMI modulation and RS(255,239) shall be used.

12.2.8.1 OOK

The OOK modulation shall use variable amplitudes to represent the data. As shown in Figure 179(a), OOK shall be represented by two points in the constellation map. The simplest form of OOK represents a binary '1' with the presence of the signal, and a binary '0' with the absence of it. The normalization factor, K_{MOD} shall be sqrt(2).

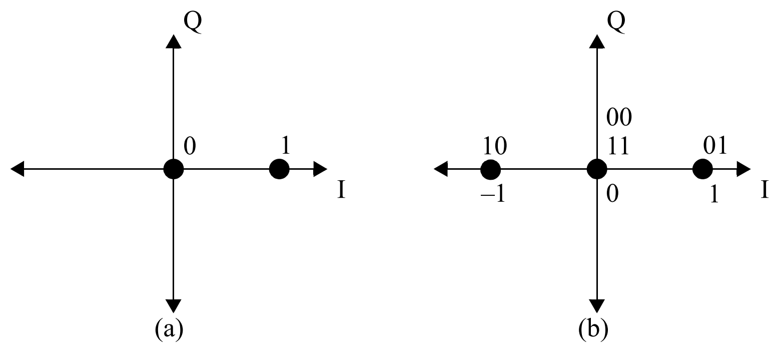

Figure 179—Constellation diagram for (a) OOK and (b) DAMI

12.2.8.2 DAMI

DAMI modulation is shown in Figure 179(b). The transmitted RF signal for a DAMI system shall be a single-sideband (SSB) modulated signal accompanied by two low-power pilot tones, as described in D2.3.

12.2.8.3 FEC

The forward error correction scheme for OOK and DAMI shall be RS(255,239), as described in 12.2.2.6.

12.2.8.4 OOK/DAMI spreading

The spreading scheme for OOK with spreading factor of 2 shall use a simple bit repetition in which each bit shall be repeated twice.

12.3 High Speed Interface mode of mmWave PHY

The High Speed Interface mode of mmWave PHY (HSI PHY) is designed for devices with low-latency, bidirectional high-speed data and uses orthogonal frequency domain multiplexing (OFDM). HSI PHY supports a variety of modulation and coding schemes (MCSs) using different frequency-domain spreading factors, modulations, and LDPC block codes.

12.3.1 General operating specifications

12.3.1.1 Operating frequency bands

The set of operating channels is defined in Table 98. A compliant IEEE 802.15.3c implementation that implements the HSI PHY shall support at least CHNL_ID 2 or CHNL_ID 3.

12.3.2 HSI PHY modulation, forward error correction, and spreading

12.3.2.1 MCS dependent parameters

The HSI PHY MCS dependent parameters are listed in Table 121. For the FEC rates of 1/2, 5/8, 3/4 and 7/8, LDPC(672,336), LDPC(672,504), LDPC(672,420) and LDPC(672,588) codes are used respectively. For the UEP MCSs (MCS index 8-11), different coding schemes are applied to the msb octets and lsb octets. The data rates specified in Table 121 assume a cyclic prefix length of 64 chips.

Table 121—HSI PHY MCS dependent parameters

MCS index	Data rate (Mb/s)	Modulation scheme	Spreading factor (L_f)	Coding mode	FEC rate (R_{FEC})	
					msb 8b	lsb 8b
0	32.1	QPSK	48	EEP	1/2	
1	1540	QPSK	1		1/2	
2	2310	QPSK	1		3/4	
3	2695	QPSK	1		7/8	
4	3080	16-QAM	1		1/2	
5	4620	16-QAM	1		3/4	
6	5390	16-QAM	1		7/8	
7	5775	64-QAM	1		5/8	
8	1925	QPSK	1	UEP	1/2	3/4
9	2503	QPSK	1		3/4	7/8
10	3850	16-QAM	1		1/2	3/4
11	5005	16-QAM	1		3/4	7/8

The number of spread, coded and data information bits per OFDM symbol are listed in Table 122 for the various MCSs.

The HSI PHY frame header rate-dependent modulation parameters are listed in Table 123. The frame header is QPSK modulated and encoded with shortened LDPC(672,336) code.

The base rate shall be MCS index 0 for HSI PHY and the corresponding header rates shall be as indicated in Table 123.

A DEV that supports the HSI PHY shall support HSI MCS index 1 and either CMS or HSI MCS index 0, as defined in 12.1.12.

Table 122—HSI PHY MCS dependent bits per OFDM symbol

MCS index	Spread and coded bits/OFDM symbol (N_{SCBPOS})	Coded bits/OFDM symbol (N_{CBPOS})	Information bits/OFDM symbol (N_{IBPOS})	
			msb 8b	lsb 8b
0	672	14	7	
1	672	672	336	
2	672	672	504	
3	672	672	588	
4	1344	1344	672	
5	1344	1344	1008	
6	1344	1344	1176	
7	2016	2016	1260	
8	672	672	336	504
9	672	672	504	588
10	1344	1344	672	1008
11	1344	1344	1008	1176

Table 123—HSI PHY frame header rate-dependent parameters

Header rate type	Header rate (Mb/s)		Spreading factor	Coded bits /OFDM symbol	Number of occupied OFDM symbols		Number of stuff bits[a]	
	Main	Optional			Main	Optional	Main	Optional
MCS 0	16.8	29.6	48	14	35	47	10	2
MCS 1–11	587	1363	1	672	1	1	192	16

[a]Stuff bits are inserted after the LDPC encoding and before constellation mapping.

12.3.2.2 HSI PHY timing-related parameters

Table 124 lists the timing-related parameters.

Table 124—Timing-related parameters

Parameters	Description	Value	Formula
f_s	Reference sampling rate/chip rate	2640 MHz	
T_C	Sample/chip duration	~0.38 ns	$1/f_s$
N_{sc}	Number of subcarriers/FFT size	512	
N_{dsc}	Number of data subcarriers	336	

Table 124—Timing-related parameters *(continued)*

Parameters	Description	Value	Formula
N_P	Number of pilot subcarriers	16	
N_G	Number of guard subcarriers	141	
N_{DC}	Number of DC subcarriers	3	
N_R	Number of reserved subcarriers	16	
N_U	Number of used subcarriers	352	$N_{dsc} + N_P$
N_{GI}	Guard interval length in samples	64	
Δf_{sc}	Subcarrier frequency spacing	5.15625 MHz	f_s/N_{sc}
BW	Nominal used bandwidth	1815 MHz	$N_U \times \Delta f_{sc}$
T_{FFT}	IFFT and FFT period	~193.94 ns	$1/\Delta f_{sc}$
T_{GI}	Guard interval duration	~24.24 ns	$N_{GI} \times T_C$
T_S	OFDM Symbol duration	~218.18 ns	$T_{FFT} + T_{GI}$
F_S	OFDM Symbol rate	~4.583 MHz	$1/T_S$
N_{CPS}	Number of samples per OFDM symbol	576	$N_{sc} + N_{GI}$

12.3.2.3 HSI PHY frame-related-parameters

The frame-related parameters are listed in Table 125.

Table 125—OFDM frame-related parameters

Parameter	Description	Value	
N_{pre}	Number of symbols in the PHY preamble (A preamble symbol is 512 chips long)	Long Preamble	16
		Short Preamble	6.75
T_{pre}	Duration of the PHY preamble	Long Preamble	~3.15 µs
		Short Preamble	~1.31 µs
T_{HDR}	Duration of the header	Main header only for MCS 0	~7.64 µs
		Main header only for MCS 1–11	~0.22 µs
		Main header and optional header for MCS 0	~17.89 µs
		Main header and optional header for MCS 1–11	~0.44 µs

Table 125—OFDM frame-related parameters *(continued)*

Parameter	Description	Value
N_{OSMF}	Number of OFDM symbols in the MAC frame body	variable
T_{OSMF}	Duration of the MAC frame body	$N_{OSMF} \times T_S$
N_{frame}	Number of OFDM symbols in the frame	$N_{pre} + N_{HDR} + N_{OSMF}$
T_{frame}	Duration of the frame	$T_{pre} + T_{HDR} + T_{OSMF}$

12.3.2.4 HSI PHY FEC

The HSI PHY FEC process is illustrated in Figure 180. In the figure, the bit interleaver block is drawn with a dashed line because it is an optional part of the HSI PHY FEC process.

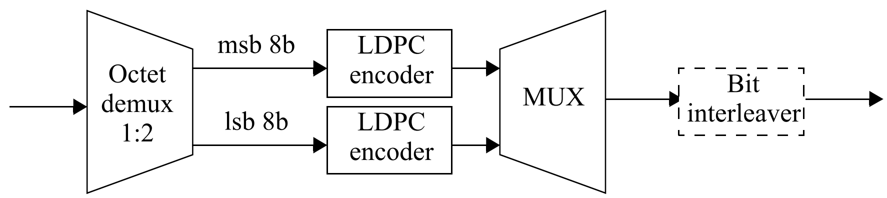

Figure 180—FEC process for the HSI PHY

12.3.2.4.1 LDPC block code

The supported LDPC block FEC rates, information block lengths, L_{INF}, and codeword block lengths, L_{FEC}, are described in Table 126.

Table 126—LDPC parameters

R_{FEC}	L_{INF} (bits)	L_{FEC} (bits)
1/2	336	672
5/8	420	
3/4	504	
7/8	588	

The LDPC encoder with rates 1/2, 3/4, and 7/8 is described in 12.2.2.6.2. For rate 5/8, the matrix permutation indices of parity-check matrix is given in Figure 167.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	0	-	-	5	-	18	16	-	-	-	3	6	10	-	-	0	-	7	-	5	-	-	4	4	-	10	-	5	-	-	-	-
2	-	-	6	-	7	-	-	-	-	2	-	-	-	-	9	-	20	-	-	-	4	-	-	-	-	-	19	-	-	-	-	-
3	-	18	-	-	-	-	-	0	10	-	-	-	-	16	-	-	-	-	9	-	-	12	-	-	4	-	-	-	-	-	17	-
4	5	0	-	-	-	-	18	16	6	-	-	3	0	10	-	-	5	-	7	-	4	-	-	-	-	-	-	-	-	-	-	-
5	-	-	-	6	-	7	-	-	-	-	2	-	-	-	-	9	-	20	-	-	-	4	-	-	-	-	-	-	-	-	-	-
6	-	-	18	-	0	-	-	-	-	10	-	-	-	-	16	-	-	-	-	9	-	-	12	-	-	-	-	-	-	-	-	-
7	-	5	0	-	16	-	-	18	3	6	-	-	-	0	10	-	-	5	-	7	4	4	-	-	-	5	-	-	-	-	-	-
8	6	-	-	-	-	-	7	-	-	-	-	2	9	-	-	-	-	-	20	-	-	-	4	-	19	-	-	-	-	-	-	-
9	-	-	-	18	-	0	-	-	-	-	10	-	-	-	-	16	9	-	-	-	-	-	-	12	-	-	-	-	-	-	-	-
10	-	-	5	0	18	16	-	-	-	3	6	-	-	-	0	10	7	-	5	-	-	4	4	-	10	-	5	-	7	-	-	-
11	-	6	-	-	-	-	-	7	2	-	-	-	-	9	-	-	-	-	-	20	-	-	-	4	-	19	-	-	-	-	-	10
12	18	-	-	-	-	-	0	-	-	-	-	10	16	-	-	-	-	9	-	-	12	-	-	-	-	-	-	4	-	17	-	-

Figure 181—Matrix permutation indices of parity check matrix of rate-5/8 LDPC code

12.3.2.4.2 EEP data multiplexer

For the EEP MCSs, as defined in Table 121, the two LDPC encoders use the same rate, and the outputs of the LDPC encoders shall be multiplexed to form a single data stream. Let a_n and b_n be the outputs bits of the msb and lsb encoders respectively. The multiplexer output shall be a_0,b_0,a_1,b_1...

12.3.2.4.3 UEP data multiplexer

For UEP Type 3 MCSs, as defined in Table 121, the two LDPC encoders have different rates. The method used to multiplex the encoded bits is dependent on the LDPC msb and lsb rates.

For UEP MCSs with msb encoder rate 1/2 and lsb encoder rate 3/4, the encoded bits shall be multiplexed every 5 bits. During the length 5 multiplexing cycle, a group multiplexer shall be used with group size three for the msb encoder and group size two for the lsb encoder. a_0,a_1,a_2 are used to label the three encoded bits (in increasing order in time) from the msb encoder, and b_0,b_1 are used to label the two encoded bits from the lsb encoder. The multiplexer output shall be: a_0,b_0,a_1,b_1,a_2.

For UEP MCSs with msb encoder rate 3/4 and lsb encoder rate 7/8, the encoded bits shall be multiplexed every 13 bits. During the length 13 multiplexing cycle, a group multiplexer shall be used with group size 7 for the msb encoder and group size 6 for the lsb encoder. $a_0,a_1,a_2,...a_7$ are used to label the 7 encoded bits (in increasing order in time) from the msb encoder, and $b_0,b_1,b_2,...b_6$ are used to label the 6 encoded bits from the lsb encoder. The multiplexer output shall be: $a_0,b_0,a_1,b_1,...a_6,b_6,a_7$.

12.3.2.4.4 Bit interleaver

After the data multiplexer, the bits shall be interleaved by a block interleaver if the Bit Interleaver field in the PHY header is set to one. The interleaving is performed upon encoded bits included within an interleaving depth covering 4 LDPC codewords, i.e., more than 2688 bits.

The block interleaving process is performed using a permutation rule $L(k)$. That is, the k^{th} output, written to location k in the output vector, is read from location $L(k)$ in the input vector.

The block interleaving algorithm $L(k) = I^j_{p,q}(k)$ is described by four parameters: the block size $K_B = 2688$, an integer parameter p setting the partition size, an integer parameter q and the iteration j governing the interleaving spreading. The relationship between the block of K_B coded bits, $a_0, a_1, \ldots, a_{K-1}$, and the block of K_B interleaved bits, $b_0, b_1, \ldots, b_{K-1}$, is given by Equation (28).

$$b(k) = a[I^j_{p,q}(k)] \tag{28}$$

To realize the interleaver stage, it is recommended to implement a lookup table that contains the interleaving rule.

The interleaving rule is based on an iterative structure, as illustrated in Figure 182, in order to increase the scalability of the interleaver.

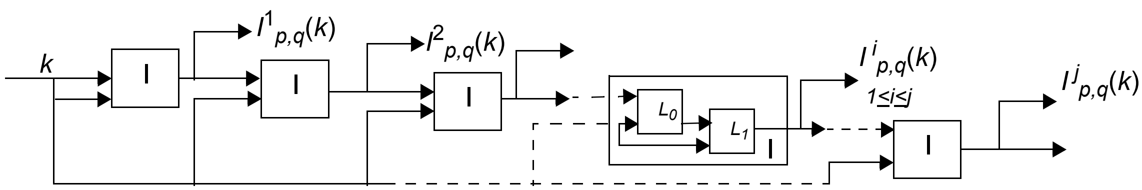

Figure 182—Turbo-based bit interleaver structure

The interleaving rule for the 1st and the jth iteration is defined as shown in Equation (29) and Equation (30).

$$I^1_{p,q}(k) = \mathrm{mod}[K_B - p + k + q \times p \times \mathrm{mod}(-k - p \times k, K_B), K_B] \tag{29}$$

$$I^j_{p,q}(k) = \mathrm{mod}[K_B - p + k + q \times p \times \mathrm{mod}(-k - p \times I^{j-1}_{p,q}(k), K_B), K_B] \tag{30}$$

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

The interleaver parameters are selected in order to optimize the interleaving spreading between successive samples. The interleaving spreading $\Delta L(s)$ is defined as the minimum distance between interleaved bits separated by a distance $s - 1$, and is expressed as shown in Equation (31).

$$\Delta L(s) = \min_k[\ |I^j_{p,q}(k + s) - I^j_{p,q}(k)|\ ] \tag{31}$$

The interleaving spreading is calculated in an algebraic way and allows the selection of interleaving parameters $\{p,q,j\}$ for each interleaving block size K_B. The binary interleaving parameters shall be: $p = 24$, $q = 2$, $j = 1$ and $\Delta L(s = 1, 2, 4, 6) = \{1199, 290, 580, 870\}$.

12.3.2.5 Stuff bits

Stuff bits shall be appended in the MAC frame body after scrambling, encoding and interleaving so that the resulting MAC frame body is aligned with the boundaries of an OFDM symbol. The stuff bits shall be set to zero and then scrambled using the continuation of the scrambler sequence that scrambled the MAC Frame Body field. To calculate the number of stuff bits, L_{STUFF}, that shall be inserted, first the number of codewords should be calculated as follows in Equation (32):

$$N_{FEC} = \mathrm{CEIL}[(L_{MFB} \times 8)/(L_{FEC} \times R_{FEC})] \tag{32}$$

where L_{MFB} is the length of the uncoded MAC frame body in octets. FEC codeword length is denoted by L_{FEC} and R_{FEC} is the coding rate. Number of the encoded bits L_{EBITS} in the MAC frame body is given by Equation (33).

$$L_{EBITS} = L_{MFB} \times 8 + N_{FEC} \times (1 - R_{FEC}) \times L_{FEC} \tag{33}$$

Number of OFDM symbols N_{OSMF} in MAC frame body is equal to Equation (34).

$$N_{OSMF} = \text{CEIL}(N_{EBITS}/N_{SCBPOS}) \tag{34}$$

where N_{SCBPOS} is the number of spread and coded bits per OFDM symbol. N_{SCBPOS} for different MCSs is given in Table 122. The number of stuff bits for a MAC frame body is equal to Equation (35).

$$L_{PAD} = N_{sOSMF} \times N_{SCBPOS} - N_{EBITS} \tag{35}$$

Stuff bits shall be added to main and optional headers as well. The number of stuff bits for each header is given in Table 123.

12.3.2.6 Constellation mapping

The coded and interleaved binary serial input data, b_i, where $i = 0, 1, 2, \ldots$, shall be modulated using QPSK, 16-QAM or 64-QAM modulation, depending on the MCS requested. The binary serial stream shall be divided into groups of N_{BPSC} (2, 4, or 6) bits and converted into complex numbers representing QPSK, 16-QAM or 64-QAM constellation points. The conversion shall be performed according to Gray-coded constellation mappings, illustrated in Figure 183 with the input bit, b_0, being the earliest in the stream. The output values, a_k, where $k = 0, 1, 2, \ldots$, are formed by multiplying the resulting value $(I_k + jQ_k)$ by a normalization factor K_{MOD}, as described in Equation (36).

$$a_k = (I_k + jQ_k) \times K_{MOD} \tag{36}$$

The normalization factor, K_{MOD}, depends on the modulation, as prescribed in Table 127. The purpose of the normalization factor is to achieve the same average power for all mappings. In practical implementations, an approximate value of the normalization factor can be used, as long as the device conforms to the modulation accuracy requirements described in 12.3.4.1.

An optional skewed constellation is also specified in Figure 183. A parameter d is introduced to distinguish between normal and skewed constellation. Its value is given by Equation (37).

$$d = \begin{cases} 1 & \text{normal constellation} \\ 1.25 & \text{skewed constellation} \end{cases} \tag{37}$$

Table 127—Modulation dependent normalization factor

Modulation	K_{MOD}
QPSK	$1/\sqrt{1+d^2}$
16-QAM	$1/\sqrt{5(1+d^2)}$
64-QAM	$1/\sqrt{21(1+d^2)}$

12.3.2.7 HSI spreader

The spreading factors in relationship to the data rate in the PHY Payload field and Frame Header field are specified in Table 121 and Table 123, respectively. The spreading rules shall be the same for all three fields.

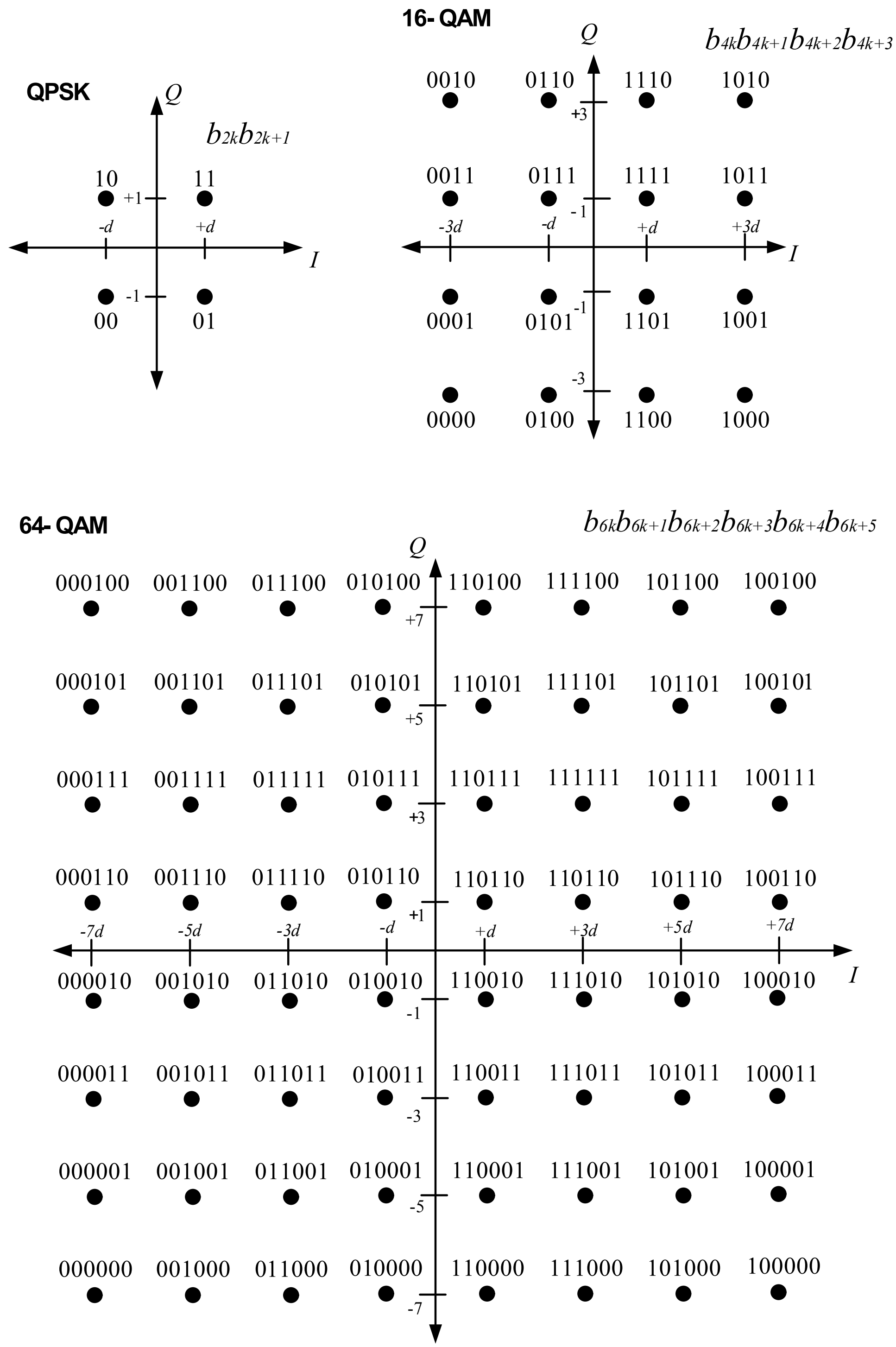

Figure 183—QPSK,16-QAM and 64-QAM normal and skewed constellation bit encoding

12.3.2.7.1 Spreader for spreading factor of 1

For a spreading factor of 1, the modulated QPSK and QAM complex values a_k, where $k = 0, 1, 2, \ldots$ at the output of the constellation mapper shall be grouped into sets of 336 complex numbers. Each group shall be assigned to an OFDM symbol. This is denoted by writing the complex number $b_{k,n}$ as shown in Equation (38).

$$b_{k,n} = a_{k+n\times 336}, \text{ for } k = 0,1,...,335,\ n = 0, 1, 2, \ldots \tag{38}$$

where n is the OFDM symbol number. Each group shall be passed to the tone interleaver before being modulated by the OFDM modulator into an OFDM symbol.

12.3.2.7.2 Spreader for spreading factor of 48

For a spreading factor of 48, the modulated QPSK complex values $a(k)$, where $k = 0, 1, 2, \ldots$ at the output of the constellation mapper shall be grouped into sets of seven complex numbers. This is denoted by writing the complex number $a_{k,n}$ as shown in Equation (39).

$$a_{k,n} = a_{k+n\times 28}, \text{ for } k = 0{:}6,\ n = 0, 1, 2, \ldots \tag{39}$$

where n is the group number. Each group shall be spread by a factor of 48 to generate a block of 336 complex numbers as follows in Equation (40) and Equation (41).

$$b_{k,n} = q_{\text{floor}(k/28)} a_{k,n}, \text{ for } k = 0{:}167, \text{ and} \tag{40}$$

$$b_{k,n} = b^*_{335-k,n}, \text{ for } k = 168{:}335 \tag{41}$$

where q is a length 12 sequence given by Equation (42).

$$q = [+1\ +j\ -1\ +j\ +j\ +1\ -1\ +j\ -j\ +j\ -1\ -j\ -1\ +1\ +1\ +1\ +j\ -j\ -1\ -1\ -1\ +j\ -j\ +j] \tag{42}$$

and the function floor() rounds its argument to the nearest integer toward minus infinity. Each spread group shall be passed to the tone interleaver before being modulated into an OFDM symbol.

12.3.2.8 Tone interleaver

All bits shall be interleaved by a block interleaver with a block size corresponding to the size of FFT in a single OFDM symbol, N_{sc}. The interleaver is used so that the adjacent data symbols are mapped onto separate subcarriers.

At the transmitter side, the interleaver permutation shall be defined as follows: Let k be the index of the tones (including data tones, pilot tones, DC tones and null tones) before permutation ranging between 0 and N_{sc}–1. Let i be the index of the interleaved tones over the same range (including data tones, pilot tones, DC tones and null tones) after permutation. Let

$$k = \sum_{j=0}^{L} a_j 2^j \tag{43}$$

where $L = \log_2(N_{sc}) - 1$, with $[a_L, \ldots, a_0]$ being the binary representation of integer k. Then the binary representation of integer i can be written as $[a_0, \ldots, a_L]$, i.e., see Equation (44).

$$i = \sum_{j=0}^{L} a_j 2^{L-j} \tag{44}$$

DC, null, and pilot tones shall be inserted in the bit-reversal position before the tone interleaver. This ensure that after permutation, the DC, null and pilot tones appear in the pre-specified positions.

12.3.2.9 HSI PHY OFDM modulator

The stream of interleaved complex numbers is divided into groups of N_{dsc} data complex numbers. This is denoted by writing the complex number $d_{k,n}$, which corresponds to data subcarrier k of OFDM symbol n, as in Equation (45).

$$d_{k,n} = d_{k+n\times N_D}, \quad k = 0, 1, 2, \ldots, N_D - 1, \quad n = 0, 1, \ldots, N_{hdr} + N_{OSMF} - 1 \tag{45}$$

where N_{hdr} + N_{OSMF} is the number of OFDM symbols occupied by the header and PHY Payload field.

The discrete-time signal during the n^{th} OFDM symbol is given by Equation (46).

$$s_{k,n} = \frac{1}{\sqrt{N_{sc}}}\left[\sum_{m=0}^{N_D-1} d_{m,n} e^{j2\pi\frac{k\times M_D(m)}{N_{sc}}} + x_n \sum_{m=0}^{N_P-1} p_{m,n} e^{j2\pi\frac{k\times M_P(m)}{N_{sc}}} + \sum_{m=0}^{N_D-1} g_{m,n} e^{j2\pi\frac{k\times M_G(m)}{N_{sc}}}\right] \tag{46}$$

where $k \in [0{:}N_{FFT}-1]$, N_{dsc} is the number of data subcarriers, N_P is the number of pilot subcarriers, N_R is the number of reserved subcarriers, N_G is the number of guard subcarriers, N_{sc} is the number of total subcarriers, and $d_{m,n}$, $p_{m,n}$, and $g_{m,n}$, are the complex numbers placed on the m^{th} data, pilot, and guard subcarriers of the n^{th} OFDM symbol, respectively.

The functions $M_{dsc}(m)$, $M_P(m)$, and $M_G(m)$ define a mapping between the indices [0:N_{dsc} – 1], [0:N_P – 1], [0:N_R – 1], and [0:N_G – 1] into the logical frequency offset index [–N_{sc}/2:N_{sc}/2 – 1]. The definition for the mapping functions are given in Equation (47) through Equation (49):

$$M_{dsc}(m) = \begin{cases} m - 177 + round(m/21) & 0 \le m \le 167 \\ m - 174 + round[(m+1)/21] & 168 \le m \le 335 \end{cases} \tag{47}$$

$$M_P(m) = \begin{cases} -166 + m \times 22 & 0 \le m \le 7 \\ 12 + (m - 8 \times 22) & 8 \le m \le 15 \end{cases} \tag{48}$$

$$M_G(m) = \begin{cases} -185 + m & 0 \le m \le 7 \\ 170 + m & 8 \le m \le 15 \end{cases} \tag{49}$$

where the function round(), rounds the input argument to the nearest integer.

The mapping of data and pilot subcarriers within an OFDM symbol is illustrated in Figure 184. The mapping is further summarized in Table 128. As shown in Figure 184, there are 16 groups of subcarriers where each group is constituted of 21 data subcarriers and one pilot subcarrier.

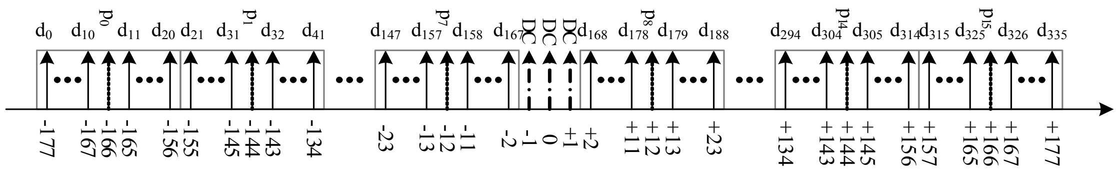

Figure 184—Subcarrier frequency allocation

Table 128—Subcarrier frequency allocation

Subcarriers type	Number of subcarriers	Logical subcarriers indexes
Null subcarriers	141	$[-256:-186] \cup [186:255]$
DC subcarriers	3	–1, 0, 1
Pilot subcarriers	16	$[-166:22:-12] \cup [12:22:166]$
Guard subcarriers	16	$[-185:-178] \cup [178:185]$
Data subcarriers	336	All others

The common way to implement the inverse Fourier transform is by an inverse Fast Fourier Transform (IFFT) algorithm. If, for example, a 512-point IFFT is used, the logical subcarriers 2 to 185 are mapped to the same numbered IFFT inputs, while the logical frequency subcarriers –185 to –2 are copied into IFFT inputs 327 to 510. The rest of the inputs, 186 to 326 and the 0 (DC) input, 1, and 511 are set to zero. The subcarriers 0, 1, and 511 are set to zero to avoid difficulties in D/A and A/D converter offsets and carrier feed through in the RF system. This mapping is illustrated in Figure 185. After performing an IFFT, the output is cyclically extended to the desired length.

12.3.2.9.1 Pilot subcarriers

In all OFDM symbols following the frame preamble, sixteen of the subcarriers shall be dedicated to pilot signals in order to allow for coherent detection and to provide robustness against frequency offsets and phase noise. These pilot signals shall be placed into logical frequency subcarriers –166, –144, –122, –100, –78, –56, –34, –12, 12, 34, 56, 78, 100, 122, 144, and 166. The information for the m^{th} pilot subcarrier of the n^{th} OFDM symbol shall be defined as follows in Equation (50):

$$p_m = \begin{cases} (1+j)/\sqrt{2} & \text{for} \quad m = 0, 3, 5, 7, 9, 13, 15 \\ (1-j)/\sqrt{2} & \text{for} \quad m = 1, 2, 4, 6, 8, 10, 11, 12, 14 \end{cases} \tag{50}$$

The polarity of the pilot subcarriers is controlled by the sequence, x_n, generated by the linear feedback shift register described in 12.3.2.11.

The 15 bit seed value chosen is $x_{-1:-15}$ = [1101 0000 1010 000]. The first 16 output values should be $x_{0:15}$ = [0001 1110 0011 1010].

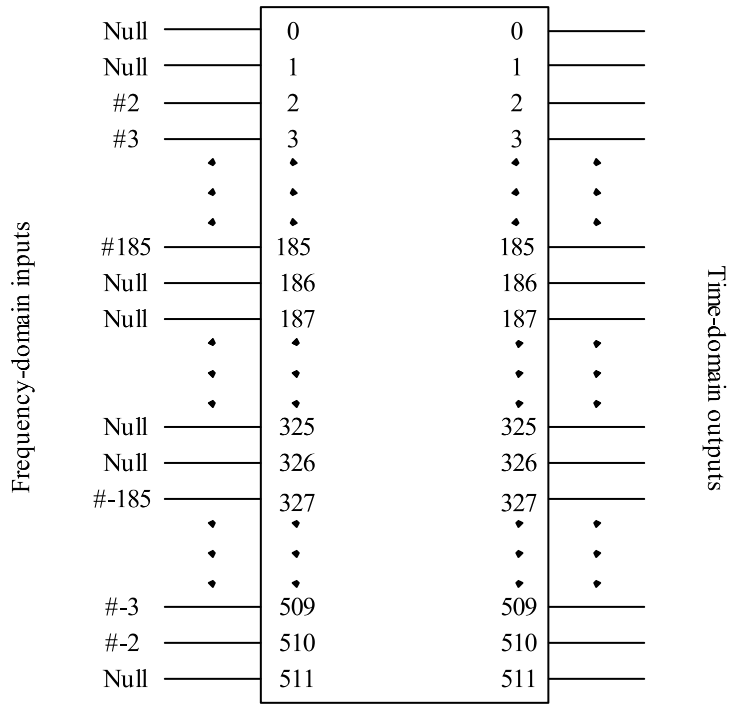

Figure 185—Input and output relationship of the IFFT

12.3.2.9.2 Guard subcarriers

In all OFDM symbols following the frame preamble, there shall be 16 guard subcarriers, 8 on each edge of the occupied frequency band, at logical frequency subcarriers –185, –184, …¦, –178 and 178, 179, …¦, 185. The data on these subcarriers shall be left to the implementor. Individual implementations may exploit these guard subcarriers for various purposes, including relaxing the specifications on analog transmit and analog receive filters, and possibly peak to average power ratio reduction.

12.3.2.10 PCES insertion

The pilot channel estimation sequence (PCES) symbols are used for channel re-acquisition or tracking. This is an optional field which content is identical to the CES of the short preamble prepended by $\mathbf{c}_{128}$.

If PCES is used then the PCES symbols shall be inserted periodically in the PHY payload field as shown in Figure 186. The value of the exact period N_{PCES} shall be 96 OFDM symbols.

N_{PCES} OFDM symbols	PCES	N_{PCES} OFDM symbols	…	PCES	N_{PCES} OFDM symbols	PCES	N_{PCES} OFDM symbols

Figure 186—PCES positions in the PHY Payload field

12.3.2.11 HSI PHY scrambling

HSI PHY uses same scrambling method of SC PHY, which is explained in 12.2.2.10.

12.3.3 HSI PHY frame format

The HSI PHY frame shall be formatted as illustrated in Figure 172.

PHY payload	Frame header	PHY preamble

Figure 187—HSI PHY frame format

The PHY preamble is defined in 12.3.3.1.

The Frame Header field for the HSI PHY frame shall be formatted as illustrated in Figure 173. The goal of the frame header is to convey necessary information about both the PHY and the MAC to aid decoding of the PHY Payload at the receiver.

Optional frame header				Main frame header				
Stuff bits	Parity bits	HCS	MAC subheader	Stuff bits	Parity bits	HCS	MAC header	PHY header

Figure 188—HSI frame header format

The construction of main frame header and optional frame header is explained in 12.3.3.2. PHY header is defined in 12.3.3.3. MAC header is defined in 7.2 and the MAC subheader is defined in 7.2.8.

The PHY Payload field is formed by adding any necessary stuff bits to the MAC Frame Body field, as defined in 7.2 to align the data stream on the boundary of an OFDM symbol. The optional PCES may added to the PHY Payload field.

The frame header is sent at the appropriate header rate as defined in Table 123. The PHY Payload field is transmitted at the desired data rate as defined in Table 121.

12.3.3.1 PHY preamble

A PHY preamble shall be added prior to the frame header to aid receiver algorithms related to frame detection, AGC setting, timing acquisition, frequency recovery, frame synchronization, and channel estimation.

The PHY preamble shall be transmitted at the rate equal to the subcarrier frequency spacing, as defined in Table 124, i.e., $R_S = \Delta f_{sc}$. A preamble symbol is defined as a sequence of length 512 chips that corresponds to the FFT length.

Two preambles are defined for the HSI PHY mode: the long preamble and optional short preamble. The Preamble Type field in the PHY header, as described in 12.3.3.3, indicates the type of preamble that shall be used in the next frame. The long preamble is used with MCS index 0 and is the same structure as defined for the CMS in Figure 161 using the chip rate of the HSI PHY mode. The short preamble is the same structure as defined for the SC PHY mode in Figure 174 using the chip rate of the HSI PHY mode and the SYNC defined in 12.2.3.1.1, the SFD defined in 12.2.3.1.2, and the CES defined in 12.2.3.1.3. The durations of the parts of the preambles are provided in Table 124.

12.3.3.2 Frame header

A frame header shall be added after the PHY preamble. It conveys information in the PHY and the MAC headers necessary for a successful decoding of the frame. The frame header is constructed from a main frame header followed by an optional frame header.

The frame header shall be constructed as follows:

a) Form the main frame header as follows:
   1) Construct the PHY header based on information provided by the MAC,
   2) Compute the HCS over the combined PHY and MAC headers as described in 12.3.3.4.1,
   3) Append the HCS to the MAC header,
   4) Scramble the combined MAC header and HCS as described in 12.3.2.5,
   5) Form the main frame header by concatenating the PHY header, scrambled MAC header, scrambled HCS, and
   6) The resulting main frame header shall be modulated as follows:
      i) The main frame header is further encoded with shortened LDPC encoder of rate 1/2, as described in 12.3.2.4.1,
      ii) The resulting main frame header has stuff bits added as indicated in Table 123,
      iii) The resulting main frame header is then mapped to a QPSK constellation as described in 12.3.2.6, and
      iv) The resulting complex values are passed to the OFDM modulator as described in 12.3.2.9. The cyclic prefix during the header is fixed to 128 chips.
b) Form the optional frame header as follows:
   1) Compute the HCS over the MAC subheader as described in 12.3.3.4.1,
   2) Append the HCS to the MAC subheader,
   3) Scramble the combined MAC subheader and HCS as described in 12.3.2.5,
   4) Form the optional frame header by concatenating the scrambled MAC subheader, scrambled HCS,
   5) The resulting optional frame header shall be modulated as follows:
      i) The optional frame header is further encoded with shortened LDPC encoder of rate 1/2, as described in 12.3.2.4.1,
      ii) The resulting optional frame header has stuff bits added as indicated in Table 123,
      iii) The resulting optional frame header is then mapped to a QPSK constellation as described in 12.3.2.6, and
      iv) The resulting complex values are passed to the OFDM modulator as described in 12.3.2.9. The cyclic prefix during the header is fixed to 128 chips.
c) Form the frame header by concatenating the main frame header and optional frame header.

The construction and modulation of the main frame header for HSI PHY is illustrated in Figure 189.

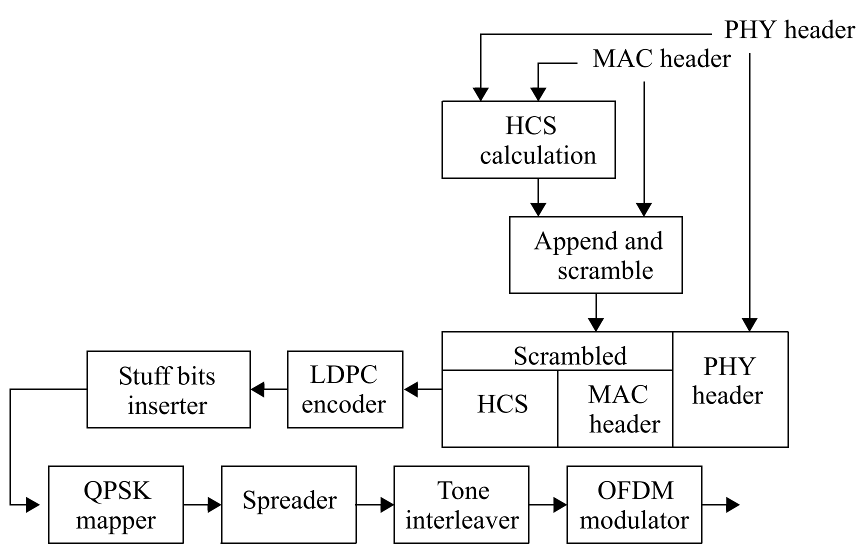

Figure 189—HSI main frame header encoding process

The construction and modulation of the optional header for HSI is illustrated in Figure 190.

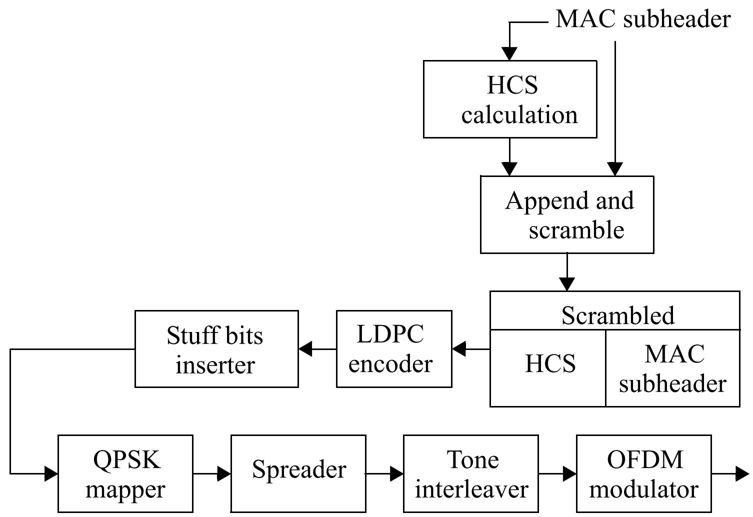

Figure 190—HSI optional frame header encoding process

12.3.3.3 HSI PHY header

The HSI PHY header field shall be formatted as illustrated in Figure 191.

bits: 2	20	5	1	1	4
Preamble type	Frame length	MCS	UEP	Aggregation	Scrambler ID

10	1	1	1	1	1
Reserved	Skewed constellation	PCES mode	Bit interleaver	Low-latency mode	Beam tracking

Figure 191—HSI PHY header format

The scrambling process is described in section 12.3.2.5. The MAC shall set the bits defined in section 12.3.2.11 according to the scrambler seed identifier value.

The Aggregation field shall be set to one if aggregation is used, it shall be set to zero otherwise.

The UEP field shall be set to one if any type of UEP is used, it shall be set to zero otherwise.

The MCS field shall be set, dependent on MCS used in the MAC frame body, according to the values in Table 129. MCS field also identifies which header rate should be used, as defined in Table 123. The MCS identifiers in relation to the data rate, modulation, spreading code length, FEC, pilot word and burst length are provided in Table 121.

The Frame Length field shall be an unsigned integer number that indicates the number of octets in the Frame Payload field (which does not include the FCS).

The Preamble Type field indicates the type of the PHY preamble (long or short) used in the next frame as defined in Table 130.

Table 129—MCS field

MCS identifier	MCS index
00000	0
00001	1
00010	2
00011	3
00100	4
00101	5
00110	6
00111	7
01000	8
01001	9
01010	10
01011	11
01100–0b11111	Reserved

Table 130—Preamble type field

Preamble type	Type of preamble used for next frame
0b00	Long preamble
0b01	Short preamble
0b10-0b11	Reserved

The Beam Tracking field shall be set to one if training sequences for beam tracking are present following the current frame. It shall be set to zero otherwise.

The Low-latency Mode field shall be set to one if the frame is using the low-latency aggregation mode and it shall be set to zero otherwise.

If the Low-latency Mode field is set to one, then the Aggregation field shall also be set to one.

The Bit Interleaver field shall be set to one to indicate that bit interleaving is used in the payload of the frame and shall be set to zero otherwise. Bit interleaving shall not be used for header bits.

The PCES Mode field shall be set to one to indicate that PCES symbols are added periodically to the PHY payload field and it shall be set to zero otherwise.

The Skewed Constellation field is optional. When used, this field shall be set to one if UEP uses constellation mapping and shall be set to zero to indicate that UEP uses coding. If the field is not used, it shall be set to zero.

12.3.3.4 Header Check sequences

12.3.3.4.1 Main Header HCS

The combination of the PHY header and the MAC header shall be protected with an ITU-T CRC-16 header check sequence (HCS). The ITU-T CRC-16 is described in 11.2.9.

12.3.3.4.2 Optional Header HCS

The MAC subheader shall be protected with an ITU-T CRC-16 header check sequence (HCS). This shall be computed in the same way as specified in 11.2.9.

12.3.3.5 PHY payload field

The PHY Payload field is the last component of the frame, and is constructed as shown in Figure 192.

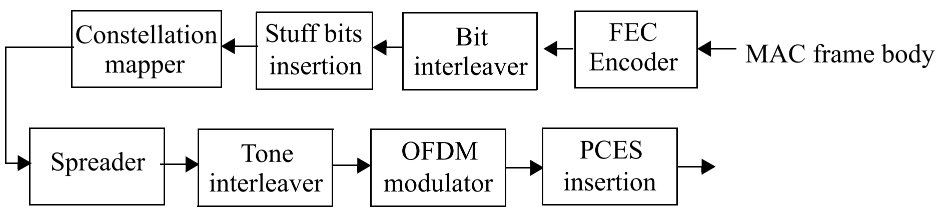

Figure 192—PHY Payload field construction process

The PHY Payload field shall be encoded as follows:

d) Scramble the MAC frame body according to 12.3.2.11.
e) Encode and interleave the scrambled MAC frame body as specified in 12.3.2.4.
f) If necessary, add stuff bits, as defined in 12.3.2.5.
g) Map the MAC frame body onto the appropriate constellation as described in 12.3.2.6.
h) Spread the modulated signal as described in Table 12.3.2.7.
i) Interleave the MAC frame body with the tone interleaver as described in 12.3.2.8.
j) Modulate the resulting complex values with the OFDM modulator described in 12.3.2.9.
k) Insert PCES periodically as described in 12.3.2.10.

Note that when the frame payload length is zero, the length of the PHY Payload field shall also be zero.

12.3.4 Transmitter specifications

12.3.4.1 EVM requirement

The EVM of a compliant transmitter shall be measured and calculated as defined in 12.1.7 and shall not exceed the values given in Table 131.

The relative constellation RMS error calculation shall be performed using a device capable of converting the transmitted signal into a stream of complex samples at the sampling rate, f_s, as defined in Table 124, or higher, with sufficient accuracy in the I/Q imbalance, DC offset, phase noise, etc. The sampled signal shall then be processed in a manner similar to that of an ideal receiver including adding the 64 samples of the cyclic prefix to the received OFDM symbol.

Table 131—Allowed relative constellation error versus data rate

Data rate	Relative constellation RMS error
Up to 1.5 Gb/s	–7 dB
2.1 Gb/s to 2.7 Gb/s	–14 dB
2.8 Gb/s to 5.3 Gb/s	–21 dB
Above 5.4 Gb/s	–23 dB

12.3.4.2 Chip rate and clock alignment

The transmitted center frequency and chip clock frequency tolerances shall be ±20 μHz/Hz maximum. The transmit center frequency and symbol clock frequency shall be derived from the same reference oscillator (locked).

12.3.5 Receiver specifications

12.3.5.1 Receiver sensitivity

For a bit error rate (BER) of 10^{-6}, the minimum receiver sensitivity shall be –50 dBm for MCS index 1 and shall be -70 dBm for MCS index 0. The BER shall be measured at the MAC/PHY interface after PHY level FEC. A PN sequence defined by $x^{55} + x^{24} + 1$ shall be used for the BER measurement.

12.3.5.2 Receiver CCA performance

The start of a valid transmission at a receiver equal to or greater than the minimum sensitivity for MCS index 0, if MCS index 0 is implemented, shall cause CCA to indicate busy with a probability > 90% within pCCADetectTime, as defined in Table 119. The receiver CCA function shall in all circumstances report the medium busy with any signal 20 dB above the minimum sensitivity for MCS index 0.

If CMS is supported, then the receiver CCA performance is described in 12.1.12.8.

12.3.5.3 Receiver maximum input level

The receiver maximum input level is the maximum power level of the incoming signal present at the input of the receiver for which the error rate criterion is met. A compliant receiver shall have a receiver maximum input level of at least –25 dBm for each of the modulation formats that the device supports.

12.3.5.4 PHY layer timing

The values for the PHY layer timing parameters are given in Table 119.

12.3.5.5 Interframe spacing

The interframe spacing parameters are given in Table 133.

12.3.5.6 Receive-to-transmit turnaround time

The RX-to-TX turnaround time shall be less than pPHYSIFSTime. The RX-to-TX turnaround time shall be measured at the air interface from the trailing edge of the last symbol received until the first symbol of the PHY preamble is present at the air interface.

Table 132—PHY layer timing parameters

PHY parameter	Value	Subclause
pPHYMIFSTime	0.2 μs, 0.5 μs (default), 2.0 μs	12.3.5.8
pPHYSIFSTime	0.2 μs, 2.0 μs, 2.5 μs (default)	12.3.5.6
pCCADetectTime	2.5 μs	12.3.5.2
pPHYChannelSwitchTime	100 μs	12.3.5.9

Table 133—HSI interframe spacing parameters

MAC parameter	PHY parameter	Definition
MIFS	pPHYMIFSTime	11.2.7.4
SIFS	pPHYSIFSTime	11.2.7.2
pBackoffSlot	pPHYSIFSTime + pCCADetectTime	11.2.7.1
BIFS	pPHYSIFSTime + pCCADetectTime	11.2.7.1
RIFS	2 × pPHYSIFSTime + pCCADetectTime	11.2.7.1

12.3.5.7 Transmit-to-receive turnaround time

The TX-to-RX turnaround time shall be less than pPHYSIFSTime. The TX-to-RX turnaround time shall be measured at the air interface from the trailing edge of the last transmitted symbol until the receiver is ready to begin the reception of the next PHY frame.

12.3.5.8 Time between successive transmissions

The time between successive transmissions shall be pPHYMIFSTime, including the power-up ramp specified in 11.5.7. The pPHYMIFSTime shall be measured at the air interface from the trailing edge of the last symbol transmitted until the first symbol of the PHY preamble is present at the air interface.

12.3.5.9 Channel switch time

The channel switch time is defined as the time from when the last valid bit is received at the antenna on one channel until the DEV is ready to transmit or receive on a new channel. The channel switch time shall be less than pPHYChannelSwitchTime.

12.3.6 HSI PHY management

The PHY PIB comprises the managed objects, attributes and notifications required to manage the PHY layer of a DEV.

12.3.6.1 PHY supported data rate encoding

The encoding of the PHY data rates used in the Supported Data Rates field in the Capability IE, as described in 7.4.11, is given in 12.1.8.1.

12.3.6.2 HSI PHY fragment size encoding

The encoding of the preferred fragment size used in the Capability IE, as described in 7.4.11, is given in Table 97.

The PHY definitions create restrictions on the maximum frame size, maximum transfer unit size and minimum fragmentation size that will be supported.

12.3.6.3 Maximum frame length

The maximum frame length allowed, pMaxFrameBodySize, shall be $2^{20} – 1$ octets. This total includes the frame body and FCS but not the PHY preamble, PHY header, or MAC header. The maximum frame length also does not include the tail symbols or the stuff bits.

12.3.6.4 Maximum transfer unit size

The maximum size data frame passed from the upper layers, pMaxTransferUnitSize, shall be the same as pMaxFrameBodySize, as defined in 12.3.6.3. If security is enabled for the data connection, the upper layers should limit data frames to pMaxFrameBodySize minus the security overhead as defined in 7.3.4.2.

12.3.6.5 Minimum fragment size

The minimum fragment size, pMinFragmentSize, shall be 512 octets.

12.4 Audio/Visual mode of mmWave PHY

The Audio/Visual (AV) PHY is implemented with two PHY modes, the high-rate PHY (HRP) and low-rate PHY (LRP), both of which use orthogonal frequency domain multiplexing (OFDM). The data rates supported by the HRP are defined in Table 134.

Table 134—HRP data rates and coding

HRP mode index	Coding mode	Modulation	Inner code rate		Data rate (Gb/s)
			MSB	LSB	
			[7] [6] [5] [4]	[3] [2] [1] [0]	
0	EEP	QPSK	1/3		0.952
1		QPSK	2/3		1.904
2		16-QAM	2/3		3.807
3	UEP	QPSK	4/7	4/5	1.904
4		16-QAM	4/7	4/5	3.807
5	MSB-only retransmission	QPSK	1/3	N/A	0.952
6		QPSK	2/3	N/A	1.904

If a DEV supports the use of the HRP, it shall support the use of HRP mode index 0 and HRP mode index 1.

HRP mode index 5 and HRP mode index 6 transmit only the 4 msbs of each octet. In this case the data rate refers to msbs only. The 4 lsbs in each octet are discarded by the transmitter and only the msb portions of the baseband, as shown in Figure 187, shall be used.

Typical video and audio consumer electronics are configured either as a source of data, e.g., a video disc player, or as a sink, e.g., a display. For these applications, the data flow is highly asymmetric in the same direction. Thus, AV PHY DEVs shall implement one of the following configurations:

— HR0: The DEV implements LRP transmit and receive functions.
— HRRX: The DEV implements HRP receive, LRP transmit and LRP receive functions.
— HRTX: The DEV implements HRP transmit, LRP transmit and LRP receive functions.
— HRTR: The DEV implements HRP transmit, HRP receive, LRP transmit and LRP receive functions.

The data rates supported by the LRP are defined in Table 135.

Table 135—LRP data rates and coding

LRP mode index	Modulation	FEC	Data rate (Mb/s)	Repetition
0	BPSK	1/3	2.5	8×
1		1/2	3.8	8×
2		2/3	5.1	8×
3		2/3	10.2	4×

A DEV that supports the AV PHY mode shall support LRP modes 0, 1, and 2. All broadcast and multi-cast frames using the AV PHY shall be sent with an LRP mode.

12.4.1 General requirements

12.4.1.1 AV PHY channelization

The HRP mode uses the channels defined in Table 96. A compliant IEEE 802.15.3c implementation that implements the AV PHY shall support at least channel number 2.

In each of the HRP channels, three LRP channels are defined. In a piconet, only one HRP channel and one LRP channel is used at a time. Each of the LRP channels is defined relative to the center frequency of the current HRP channel, $f_{c(HRP)}$, as defined in Table 96. The LRP channels are defined in Table 136.

Table 136—LRP channelization

LRP channel index	Start frequency[a]	Center frequency	Stop frequency[a]
1	$f_{c(HRP)}$ – 207.625 MHz	$f_{c(HRP)}$ – 158.625 MHz	$f_{c(HRP)}$ – 109.625 MHz
2	$f_{c(HRP)}$ – 49 MHz	$f_{c(HRP)}$	$f_{c(HRP)}$ + 49 MHz
3	$f_{c(HRP)}$ + 109.625 MHz	$f_{c(HRP)}$ + 158.625 MHz	$f_{c(HRP)}$ + 207.625 MHz

[a]The start and stop frequencies are nominal values. The TX mask requirements for the LRP is defined in 12.4.4.1.

The PHYPIB_Current_Channel is the CHNL_ID of the current channel. For the purpose of the Remote Scan Request and Remote Scan Response commands, 7.5.7.3 and 7.5.7.4, respectively, the Channel Index field is the CHNL_ID in Table 137.

Table 137—Mapping HRP/LRP channel index to CHNL_ID

CHNL_ID	HRP channel index	LRP channel index
1	1	1
2	2	1
3	3	1
4	4	1
5	1	2
6	2	2
7	3	2
8	4	2
9	1	3
10	2	3
11	3	3
12	4	3

12.4.1.2 PHY layer timing

The values for the AV PHY layer timing are defined in Table 138.

Table 138—AV PHY layer timing parameters

PHY parameter	Value	Subclause
pPHYMIFSTime	pPHYSIFSTime	12.4.1.2.4
pPHYSIFSTime	2 µs ± 24 samples at LRP reference sampling rate	12.4.1.2.2
pCCADetectTime	9 µs	12.4.1.2.6
pPHYChannelSwitchTime	1000 µs	12.4.1.2.5

12.4.1.2.1 Interframe space

A conformant AV PHY implementation shall support the IFS parameters, as described in 8.4.1, given in Table 139.

Table 139—AV PHY IFS parameters

802.15.3 MAC parameter	Corresponding AV PHY parameter	Definition
MIFS	pPHYMIFSTime	12.4.1.2.4
SIFS	pPHYSIFSTime	12.4.1.2.2
pBackoffSlot	pPHYSIFSTime+pCCADetectTime	12.4.1.2.2, 12.4.1.2.6
BIFS	pPHYSIFSTime+pCCADetectTime	12.4.1.2.2, 12.4.1.2.6
RIFS	2 × pPHYSIFSTime+pCCADetectTime	12.4.1.2.2, 12.4.1.2.6

12.4.1.2.2 Receive-to-transmit turnaround time

The RX-to-TX turnaround time shall be pPHYSIFSTime. The RX-to-TX turnaround time shall be measured at the air interface from the trailing edge of the last symbol received until the first symbol of the PHY preamble is present at the air interface.

12.4.1.2.3 Transmit-to-receive turnaround time

The TX-to-RX turnaround time shall be less than pPHYSIFSTime. The TX-to-RX turnaround time shall be measured at the air interface from the trailing edge of the last transmitted symbol until the receiver is ready to begin the reception of the next PHY frame.

12.4.1.2.4 Time between successive transmissions

The time between successive transmissions shall be pPHYMIFSTime. The pPHYMIFSTime shall be measured at the air interface from the trailing edge of the last symbol transmitted until the first symbol of the PHY preamble is present at the air interface.

12.4.1.2.5 Channel switch time

The channel switch time is defined as the time from when the last valid bit is received at the antenna on one channel until the DEV is ready to transmit or receive on a new channel. The channel switch time shall be less than pPHYChannelSwitchTime.

12.4.1.2.6 CCA detect time

The LRP shall be able to detect the presence of an LRP preamble at an input power level equivalent to the sensitivity of the LRP mode 1 within pCCADetectTime. CCA detection is not used for the HRP mode because only the LRP is used in CPs.

12.4.1.3 Data size restrictions

The PHY definitions creates restrictions on the maximum frame size, maximum transfer unit size and minimum fragmentation size that will be supported. These parameters are defined in this subclause.

12.4.1.3.1 Maximum frame length

The maximum frame length allowed, pMaxFrameBodySize, shall be $2^{20} - 1$ octets for HRP subframes and $2^{12} - 1$ octets for LRP frames. This total includes the frame payload and FCS but not the PHY preamble, PHY header, MAC header or HCS. The maximum frame length also does not include the tail symbols or the stuff bits.

12.4.1.3.2 Maximum transfer unit size

The maximum size data frame passed from the upper layers, pMaxTransferUnitSize, shall be the same as pMaxFrameBodySize, as defined in 12.4.1.3.1. If security is enabled for the data connection, the upper layers should limit data frames to pMaxFrameBodySize minus the security overhead as defined in 7.3.4.2.

12.4.1.3.3 Minimum fragment size

The minimum fragment size, pMinFragmentSize, that is allowed with the AV PHY shall be 512 octets.

12.4.1.4 Header check sequence

The HCS shall be a 32-bit CRC that is equivalent to the one used for the FCS, as defined 7.2.7.6. The MAC parameter pLengthHCS shall be four for the AV PHY. The HCS shall be transmitted in the order specified in 7.1. At both the receiver and transmitter, the initial state of the remainder shall be set to all ones.

12.4.2 AV PHY modulation and forward error correction

The modulation parameters for the HRP are given in Table 140.

Table 140—HRP modulation parameters

Parameter	Value	Symbol
Occupied bandwidth	1.76 GHz	N/A
Reference sampling rate	2.538 GHz	$f_{s(HR)}$
Number of subcarriers	512	$N_{sc(HR)}$
FFT period	$N_{sc(HR)}/f_{s(HR)}$ ~ 202 ns	$T_{FFT(HR)}$
Subcarrier spacing	$1/T_{FFT(HR)}$ ~ 4.96 MHz	$\Delta f_{SC(HR)}$
Guard interval	$64/f_{s(HR)}$ ~ 25.2 ns	$T_{GI(HR)}$
Symbol duration	$T_{FFT(HR)} + T_{GI(HR)}$ ~ 227 ns	$T_{S(HR)}$
Number of data subcarriers	336	$N_{dsc(HR)}$

The modulation parameters for the LRP are given in Table 141.

Table 141—LRP modulation parameters

Parameter	Value	Symbol
Occupied bandwidth	92 MHz	N/A
Reference sampling rate	317.25 MHz	$f_{s(LR)}$
Number of subcarriers	128	$N_{sc(LR)}$
FFT period	$N_{sc(LR)}/f_{s(LR)}$ ~ 403 ns	$T_{FFT(LR)}$

Table 141—LRP modulation parameters *(continued)*

Parameter	Value	Symbol
Subcarrier spacing	$1/T_{FFT(LR)} \sim 2.48$ MHz	$\Delta f_{SC(LR)}$
Guard interval	$28/f_{s(LR)} \sim 88.3$ ns	$T_{GI(LR)}$
Symbol duration	$T_{FFT(LR)} + T_{GI(LR)} \sim 492$ ns	$T_{S(LR)}$
Number of data subcarriers	30	$N_{dsc(LR)}$

Unless otherwise specified, the terms sampling rate or sample are in terms of the reference sampling rate for that mode.

A reference implementation of the HRP baseband is illustrated in Figure 187.

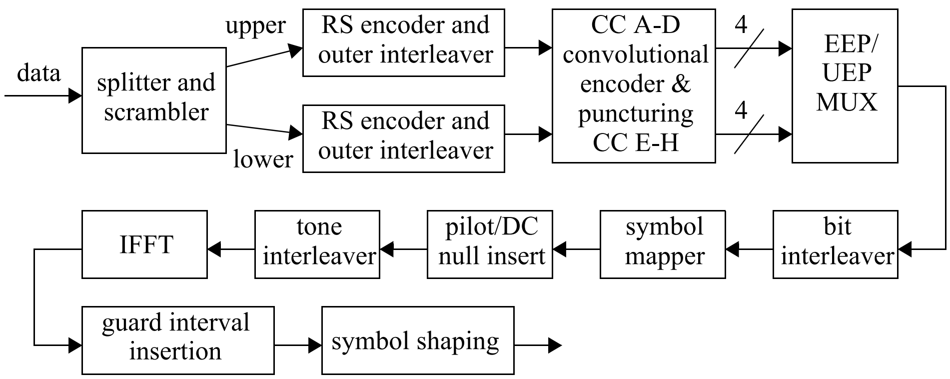

Figure 187—HRP reference implementation block diagram

A reference implementation of the LRP baseband is illustrated in Figure 188.

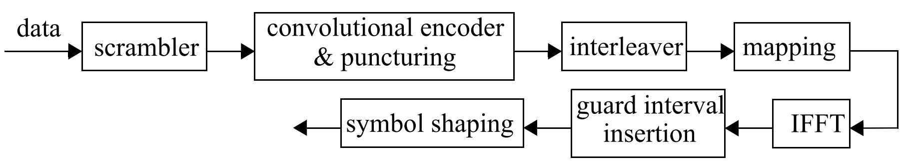

Figure 188—LRP reference implementation block diagram

The purpose of the reference implementation is to provide a reference for defining the encoding of the incoming bit stream into the appropriate RF signal. Implementations may use different architectures as long as the resulting RF signal matches the one that would be generated by a reference implementation.

12.4.2.1 AV PHY base rate

The base data rate of the AV PHY shall be LRP mode 0, 1, or 2.

12.4.2.2 Repetition coding and spatial diversity

The LRP transmitter utilizes spatial diversity and repetition coding for the omni-directional and directional modes. Up to eight antenna directions or phased array patterns may be used that shall be identified by indices zero through seven.

For omni LRPDUs, the repetition coding is implemented by repeating the OFDM symbol and its associated cyclic prefix four or eight times as indicated by the LRP mode index and sending each repetition with a different TX antenna direction or pattern. For the case of eight times repetition coding, each OFDM symbol in the Channel Estimation field and the OFDM symbols that follow in the frame are repeated using patterns zero through seven. For the four times repetition coding each OFDM symbol in the MAC header, HCS and MAC Frame Body field is repeated using patterns zero through three, where patterns zero through three are the first four patterns used in the Channel Estimation field of the omni LRP preamble. The assignment of indices to the patterns is implementation dependent and arbitrary. However, for all omni LRPDUs that use the short preamble, with the exception of LRP mode 3, the assignment of indices shall remain unchanged after the system is powered on. For omni LRPDUs that use the long omni LRP preamble, or omni LRPDUs that use the short preamble with LRP mode 3, the assignment of indices may be different from frame to frame. However, in these cases the assignment of indices shall remain unchanged for the duration of a frame. These TX antenna directions or patterns are selected such that the transmission covers the region of space that is of interest. The patterns that are used do not need to be unique as long as the same set of patterns is used for all omni LRPDUs. For example, for implementations with less than eight independently controllable elements, some of the patterns are repeated during the eight times repetition. In the case of a single TX antenna, a single direction or pattern is used for all of the repetitions.

In the Directional mode, the preamble is used to perform channel estimation and receiver training for only one optimum TX antenna direction, or phase array pattern. The same TX optimum pattern is also used for the header and payload. This optimum direction is selected while an omni LRPDU with short preamble is received. The Directional LRPDU Header field shall use the eight times directional repetition coding while the directional repetition coding of the data symbols is either eight times or four times as determined by the LRP mode index. The directional repetition coding for Directional LRPDUs is performed by adding one additional OFDM symbol as cyclic prefix to the eight or four times repeated 128-sample OFDM symbols depending on the mode index. Therefore, for eight times directional repetition, the 128-sample OFDM symbol is repeated nine times, while it is repeated five times for four times directional repetition. In this case, no other cyclic prefix is used.

12.4.2.3 Stuff bits

In order that an integer number of OFDM symbols are created and, for the HRP, complete the outer interleaver units, the PHY adds additional bits to the bit stream, called stuff bits, prior to performing any operations on the incoming data. Stuff bits for the HRP shall be set to zero prior to adding them to the end of the bit stream. Stuff bits for the LRP shall be set equal to a sequence generated by the scrambler, defined in 12.4.2.5, initialized to an arbitrary state. In LRPDUs, stuff bits only occur at the end of a frame, i.e., after the Frame Body field or the Directional LRP Payload field.

The PHY shall add the minimum number of stuff bits necessary to create an integer number of OFDM symbols and, for HRP modes, complete the outer interleaver unit for the combination of the PHY Header field, MAC Header field and HCS field. In addition, the HRP PHY shall add the minimum number of stuff bits necessary to create an integer number of OFDM symbols and complete the outer interleaver unit for each combination of subframes that have the same HRP mode index and for the last subframe in the frame. These additional bits shall be discarded by the receiver upon reception. The last outer interleaver unit for each of the subframes that end on a HRP mode change and for the last subframe may be shortened as specified in 12.4.2.7.

12.4.2.4 HRP splitter and scrambler

For the HRP header and msb only mode, no re-ordering is applied to the input octet stream before it is sent to the scrambler. For all other HRP modes, prior to scrambling, the input octets are re-ordered into upper and lower branches. If the input to the splitter is a an array of octets, $in(i, b)$, where i is the index of the input octets, $i = 0, 1, \ldots, N - 1$, where N is the number of octets in the input stream and b is the index of the bit in the octet where $b = 0$ is the lsb and $b = 7$ is the msb. The output of the scrambler is two arrays of octets, *upper*(n, b) and *lower*(n, b), where $n = 0, 1, \ldots, N-1$. Note that adding the stuff bits guarantees that the input octet stream is an even number of octets.

In EEP mode, the output arrays are constructed according to Equation (51) and Equation (52).

$$lower(n, b) = in(\{\text{floor}[b/4] + \text{floor}[n/2] \times 4\}, \{\text{mod}[b, 4] + \text{mod}[n, 2] \times 4\}) \quad (51)$$

$$upper(n, b) = in(\{\text{floor}[b/4] + \text{floor}[n/2] \times 4 + 2\}, \{\text{mod}[b, 4] + \text{mod}[n, 2] \times 4\}) \quad (52)$$

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

In UEP mode, the output arrays are constructed according to Equation (53) and Equation (54).

$$lower(n, b) = in(\{\text{floor}[b/4] + n \times 2\}, \text{mod}[b, 4]) \quad (53)$$

$$upper(n, b) = in(\{\text{floor}[b/4] + n \times 2\}, \{\text{mod}[b, 4] + 4\}) \quad (54)$$

When the octets are re-ordered, the first 8 bits of the scrambler are applied to the lower branch and the next 8 bits to the upper branch, switching every 8 bits between the two branches

The HRP scrambler shall use the generator polynomial $P(x) = x^{15} + x^{14} + 1$. A reference implementation is illustrated in Figure 168.

The initial value of scrambler shall be set by the four variable seeds, S_0, S_1, S_2 and S_3, and the 11 fixed seeds as $[x_{-1}, x_{-2}, \ldots, x_{-15}]$ = [1101 0000 101 S_3 S_2 S_1 S_0]. All fields in the HRP header, MAC header and HCS, as illustrated in 12.4.3, shall be scrambled as with the variable seeds set to $S_0 = 0$, $S_1 = 1$, $S_2 = 0$, $S_3 = 1$. The entire data stream following the HCS, including the stuff bits, is scrambled using the seed bits specified in the PHY Control field, as illustrated in Figure 203.

12.4.2.5 LRP scrambler

Scrambling applies to the MAC header, HCS, and Frame Body field only. A scrambler with six state bits, s0 through s5, and wired according to the polynomial $x^6 + x + 1$ is initialized with a four bit random value combined with 0b01 as illustrated in Figure 189. Bit s5 from successive states of the scrambler are exclusive-ORed with each data bit in sequence to create the scrambled data. The four bit random value that is used to initialize the scrambler is placed in the fields S_0, S_1, S_2, and S_3 in Figure 208.

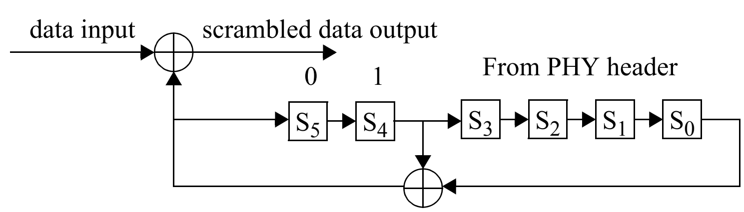

Figure 189—LRP scrambler reference implementation

NOTE—The directional LRPDU, which does not contain the MAC Header, HCS, or MAC Frame Body field, is not scrambled.

12.4.2.6 HRP outer code

For HRP modes, the output from the scrambler is split into two branches, msb and lsb. The data in each branch shall have an outer code using a Reed-Solomon code with parameters (224, 216, t = 4). The last portion of the data shall be coded by a further shortened Reed-Solomon code. The tail bits for the inner convolutional code are inserted in the outer interleaver by further shortening of the Reed-Solomon code. The parameters of these shortened codes are defined in 12.4.2.7.

The field generator polynomial is as in Equation (55).

$$g_{field}(x) = x^8 + x^4 + x^3 + x^2 + 1 \tag{55}$$

and the code generator polynomial is as in Equation (56).

$$g_{RS}(x) = (x+\lambda)(x+\lambda^2)(x+\lambda^3)(x+\lambda^4)(x+\lambda^5)(x+\lambda^6)(x+\lambda^7)(x+\lambda^8), \tag{56}$$

where λ = 0x02.

The combination of PHY Header field, MAC Header field and HCS shall use only the msb branch.

The input and output order for the Reed-Solomon encoder is specified in 12.4.2.7 in combination with the outer interleaver.

12.4.2.7 HRP outer interleaver

The outer interleaver shall output the octets from $i = 0$, $k = 0$ first to $i = depth–1$, $k = N–1$ last, where *depth* is the depth of the outer interleaver and N is the length of RS code. With M parallel convolutional inner encoders for each RS codeword and $b(n, m)$ is the output of the RS encoder, the outer interleaver shall give the RS octets of b(0,0), …, b(*depth*–1,0) to the first convolutional encoder with lsb first. All octets of $b(i,k \times M + m)$, $i = 0, \ldots, depth–1$, $k = 0, 1, \ldots, N/M–1$, shall be output to the m^{th} convolutional encoder. The number of parallel convolutional encoders is specified in 12.4.2.8.

The outer block interleaver shall be operated with a $depth = 4$ for HRP data and a $depth = 2$ for the combination of the HRP Header, MAC Header and HCS fields. LRP modes do not use an outer block interleaver.

The combination of HRP header, MAC header and HCS field has 92 octets that are encoded into 112 octets by adding 16 parity octets for error protection and 4 tail octets to terminate the convolutional code.

For the combination of the HRP Header, MAC Header and HCS fields, the first 48 octets are encoded using RS(56, 48, $t = 4$) while the next 44 octets are encoded using RS(52, 44, $t = 4$). The second codeword is followed by 4 tail octets set to zero. The transmitted order of those octets, the method to insert the tail octets, and the method of interleaving for the outer interleaver are the same as those used for data.

The outer interleaver inserts the tail bits for the convolutional encoder. For the outer interleaver with tail bits, to improve the efficiency, the number of rows of the outer interleaver may be reduced to a minimum number that is an integer multiple of 28. At the columns of $i = 0$ to $i = depth – 2$, a shortened RS($28 \times n$, $28 \times n – 8$, $t = 4$) code may be used, where $n = 1$ to 8.

12.4.2.8 Convolutional encoder

Scrambled data, tail bits and stuff bits shall be encoded with a convolutional encoder of code rate 1/3. The convolutional encoder shall use constraint length K = 7, delay memory 6, generator polynomial g0 = 133o, g1 = 171o, g2 = 165o, mother code rate 1/3. A detailed schematic diagram of the convolutional encoder is shown in Figure 190.

The initial value of the delay register shall be zero at the beginning of every HRP header and at the beginning of the frame body.

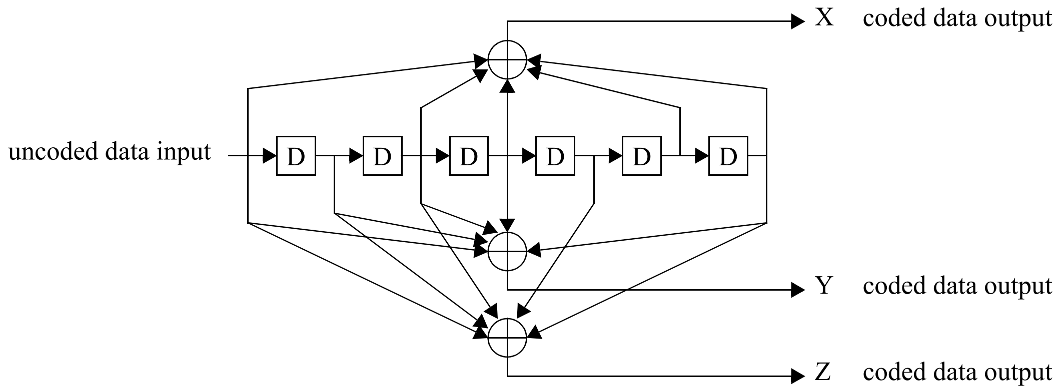

Figure 190—Convolutional encoder reference implementation

The HRP transmitter uses eight parallel convolutional encoders, labeled A through H. The first four encoders, labeled A through D, are for the first outer Reed-Solomon coding branch and the last four encoders, labeled E through H are for the second outer Reed-Solomon coding branch.

The HRP header shall use the same convolutional encoder with a code rate of 1/3. However, only four parallel convolutional encoders, labeled A through D, are used.

For all LRP Frame Body fields other than with the directional LRP payload, 6 tail bits with value zero are added. For LRP PHY header, the 1/2 and 1/3 rate coding with tail-biting are used. In the tail biting method, the initial encoder state is set equal to the last six information bits, and no tail bits are appended. Therefore, the initial and final encoder states are equal. The directional LRPDU payload shall use either tail-biting or tail bits, as specified in 12.4.3.8.

12.4.2.9 Puncturing

Convolutional encoded data is punctured to make the desired code rate using the puncturing pattern indicated in Table 142. In Table 142, puncturing pattern '1' means to send the bit while a '0' means to omit (or do not transmit) the corresponding bit.

The output of puncturing block shall be serialized with sequential order as shown in Table 142.

12.4.2.10 HRP data multiplexer and bit interleaver

The output of the 8 encoders, labeled A through H, shall be multiplexed to form a single data stream prior to the bit interleaver, defined in 12.4.2.11, as illustrated in Figure 187. The method used to multiplex the encoded bits is dependent on the type of HRP mode, either EEP or UEP.

12.4.2.10.1 EEP data multiplexer

In the EEP mode, all the 8 encoders shall use the same inner code rate. The encoded bits shall be multiplexed and bit-interleaved every 48 bits.

During the length 48 multiplexing/interleaving cycle, a group multiplexer shall be used first with fixed group size 6 for all eight encoders. A1, A2, A3, A4, A5, A6 are used to label the 6 encoded bits (in increasing order in time) from encoder A, and similarly for B1 through B6, C1 through C6, D1 through D6, E1 through E6, F1 through F6, G1 through G6, and H1 through H6 from encoders B, C, D, E, F, G, and H, respectively. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Table 143.

Table 142—Puncturing table

Code rate	Puncturing pattern	Transmitted sequence
1/3	X: 1 Y: 1 Z: 1	X1 Y1 Z1
1/2	X: 1 Y: 1 Z: 0	X1 Y1
4/7	X: 1 1 1 1 Y: 1 0 1 1 Z: 0 0 0 0	X1 Y1 X2 X3 Y3 X4 Y4
2/3	X: 1 1 Y: 1 0 Z: 0 0	X1 Y1 X2
4/5	X: 1 1 1 1 Y: 1 0 0 0 Z: 0 0 0 0	X1 Y1 X2 X3 X4

Table 143—Multiplexing scheme for EEP mode

numbering	0	1	2	3	4	5	6	7	8	9	10	11
labeling	A1	A2	A3	A4	A5	A6	B1	B2	B3	B4	B5	B6
numbering	12	13	14	15	16	17	18	19	20	21	22	23
labeling	C1	C2	C3	C4	C5	C6	D1	D2	D3	D4	D5	D6
numbering	24	25	26	27	28	29	30	31	32	33	34	35
labeling	E1	E2	E3	E4	E5	E6	F1	F2	F3	F4	F5	F6
numbering	36	37	38	39	40	41	42	43	44	45	46	47
labeling	G1	G2	G3	G4	G5	G6	H1	H2	H3	H4	H5	H6

The multiplexed 48 bits are then sent to the bit interleaver. Let x = 0, …, 47 and y = 0, …, 47 be the index at the input and output of the bit interleaver, respectively. The bit interleaver in the EEP mode shall implement the relation shown in Equation (57).

$$y = \mathrm{mod}((6 \times \mathrm{floor}(x/6) - 5 \times \mathrm{mod}(x,6)), 48) \tag{57}$$

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

The overall read out order of the multiplexer and interleaver for EEP is illustrated in Figure 191.

At the receiver side, after demodulation, the received bits shall be deinterleaved. Let y = 0, …, 47 and z = 0, …, 47 be the index at the input and output of the bit deinterleaver, respectively. The bit deinterleaver in the EEP mode shall implement the relation shown in Equation (58).

$$z = \mathrm{mod}((6 \times \mathrm{floor}(y/6) - 7 \times \mathrm{mod}(y,6)), 48) \tag{58}$$

column-wise readout (from top to bottom) in this direction

A1	E5	D3	C1	G5	F3	E1	A5	H3	G1	C5	B3
B2	F6	E4	D2	H6	G4	F2	B6	A4	H2	D6	C4
C3	B1	F5	E3	D1	H5	G3	F1	B5	A3	H1	D5
D4	C2	G6	F4	E2	A6	H4	G2	C6	B4	A2	E6

Figure 191—EEP multiplexing and bit interleaving pattern

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

12.4.2.10.2 UEP coding data multiplexer

The UEP coding data multiplexer shall be implemented by DEVs that support UEP HRP modes. It shall not be implemented by DEVs that do not support any UEP HRP modes.

In the UEP coding mode, top 4 encoders (or encoders A, B, C, D) shall use rate 4/7 convolutional codes, and bottom 4 encoders (or encoders E, F, G, H) shall use rate 4/5 convolutional codes. The encoded bits shall be multiplexed and bit interleaved every 96 bits. The total length of 96 bits is divided into two half cycles, with each half cycle multiplexing and interleaving 48 bits, in a slightly different manner.

In the first half cycle, a group multiplexer with group size 7, 7, 7, 7, 5, 5, 5, 5 for all eight encoders, A through G, respectively, shall be used. In this half cycle, A1, A2, A3, A4, A5, A6, A7 are used to label the 7 encoded bits (in increasing order in time) from encoder A, and similarly B1 through B7, C1 through C7, D1 through D7, E1 through E5, F1 through F5, G1 through G5, and H1 through H5 from encoders B, C, D, E, F, G, and H, respectively. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Figure 144.

Table 144—Multiplexing scheme in the first half cycle for UEP coding mode

numbering	0	1	2	3	4	5	6	7	8	9	10	11
labeling	A1	A2	A3	A4	A5	A6	A7	B1	B2	B3	B4	B5
numbering	12	13	14	15	16	17	18	19	20	21	22	23
labeling	B6	B7	C1	C2	C3	C4	C5	C6	C7	D1	D2	D3
numbering	24	25	26	27	28	29	30	31	32	33	34	35
labeling	D4	D5	D6	D7	E1	E2	E3	E4	E5	F1	F2	F3
numbering	36	37	38	39	40	41	42	43	44	45	46	47
labeling	F4	F5	G1	G2	G3	G4	G5	H1	H2	H3	H4	H5

The multiplexed 48 bits are then sent to the bit interleaver. Let $x = 0, \ldots, 47$ be the index at the input of the bit interleaver, and $y = 0, \ldots, 47$ be the index at the output of the bit interleaver. The bit interleaver in the first half cycle of the UEP coding mode shall implement the relation shown in Equation (59).

$$y = \mathrm{mod}((6 \times \mathrm{floor}(x/6) - 5 \times \mathrm{mod}(x,6)), 48) \quad (59)$$

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

The overall read out order of the multiplexer and interleaver in the first half cycle for UEP is illustrated in Figure 192.

column-wise readout (from top to bottom) in this direction →

A1	E1	C7	B6	G3	E5	D4	A5	H2	F4	C3	B2	I msb
B1	F1	D7	C6	H5	G2	E4	B5	A4	H1	D3	C2	I lsb
C1	A7	F2	D6	C5	H4	G1	E3	B4	A3	G5	D2	Q msb
D1	B7	G4	F1	D5	A6	H3	F5	C4	B3	A2	E2	Q lsb

Figure 192—UEP coding mode first half cycle multiplexing and bit interleaving pattern

At the receiver side, after demodulation, the received bits shall be deinterleaved. Let y = 0, …, 47 and z = 0, …, 47 be the index at the input and output of the bit deinterleaver, respectively. The bit deinterleaver in the first half cycle of the UEP coding mode shall implement the relation shown in Equation (60).

$$z = \mathrm{mod}((6 \times \mathrm{floor}(y/6) - 7 \times \mathrm{mod}(y,6)), 48) \tag{60}$$

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

In the second half cycle, a group multiplexer with group size 7, 7, 7, 7, 5, 5, 5, 5 for all eight encoders shall be used. In this half cycle, A8, A9, A10, A11, A12, A13, A14 are used to label the 7 encoded bits (in increasing order in time) from encoder A, and similarly B8 through B14, C8 through C14, D8 through D14, E6 through E10, F6 through F10, G6 through G10, and H6 through H10 from encoders B, C, D, E, F, G, and H, respectively. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Table 145.

Table 145—Multiplexing scheme in the second half cycle for UEP coding mode

numbering	0	1	2	3	4	5	6	7	8	9	10	11
labeling	B8	B9	B10	B11	B12	B13	B14	C8	C9	C10	C11	C12
numbering	12	13	14	15	16	17	18	19	20	21	22	23
labeling	C13	C14	D8	D9	D10	D11	D12	D13	D14	A8	A9	A10
numbering	24	25	26	27	28	29	30	31	32	33	34	35
labeling	A11	A12	A13	A14	F6	F7	F8	F9	F10	G6	G7	G8
numbering	36	37	38	39	40	41	42	43	44	45	46	47
labeling	G9	G10	H6	H7	H8	H9	H10	E6	E7	E8	E9	E10

The multiplexed 48 bits are then sent to the bit interleaver. Let x = 0, …, 47 be the index at the input of the bit interleaver, and y = 0, …, 47 be the index at the output of the bit interleaver. The bit interleaver in the first half cycle of the UEP coding mode shall implement the relation shown in Equation (61).

$$y = \mathrm{mod}((6 \times \mathrm{floor}(x/6) - 5 \times \mathrm{mod}(x,6)), 48) \tag{61}$$

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

The overall read out order of the multiplexer and interleaver in the second half cycle for UEP is illustrated in Figure 193.

column-wise readout (from top to bottom) in this direction

B8	F6	D14	C13	H8	F10	A11	B12	E7	G9	D10	C9
C8	G8	A14	D13	E10	H7	F9	C12	B11	E6	A10	D9
D8	B14	G7	A13	D12	E9	H6	F8	C11	B10	H10	A9
A8	C14	H9	G6	A12	B13	E8	G10	D11	C10	B9	F7

Figure 193—UEP coding mode second half cycle multiplexing and bit interleaving pattern

At the receiver side, after demodulation, the received bits shall be deinterleaved. Let y = 0, …, 47 and z = 0, …, 47 be the index at the input and output of the bit deinterleaver, respectively. The bit deinterleaver in the second half cycle of the UEP coding mode shall implement the relation shown in Equation (62).

$$z = \mathrm{mod}((6 \times \mathrm{floor}(y/6) - 7 \times \mathrm{mod}(y,6)), 48) \tag{62}$$

where mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

12.4.2.10.3 UEP mapping data multiplexer

The UEP mapping data multiplexer shall be implemented by DEVs that support UEP HRP modes. It shall not be implemented by DEVs that do not support any UEP HRP modes.

In the UEP mapping mode, all eight encoders shall use the same coding rate and the encoded bits shall be multiplexed and bit interleaved every 48 bits.

During the length 48 multiplexing/interleaving cycle, a group multiplexer shall be used with fixed group size 6 for all eight encoders. A1, A2, A3, A4, A5, A6 are used to label the 6 encoded bits (in increasing order in time) from encoder A, and similarly B1 through B6, C1 through C6, D1 through D6, E1 through E6, F1 through F6, G1 through G6, and H1 through H6 from encoders B, C, D, E, F, G, H respectively. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Table 146.

Table 146—Multiplexing scheme for UEP mapping mode

numbering	0	1	2	3	4	5	6	7	8	9	10	11
labeling	A1	A2	A3	A4	A5	A6	B1	B2	B3	B4	B5	B6
numbering	12	13	14	15	16	17	18	19	20	21	22	23
labeling	C1	C2	C3	C4	C5	C6	D1	D2	D3	D4	D5	D6
numbering	24	25	26	27	28	29	30	31	32	33	34	35
labeling	E1	E2	E3	E4	E5	E6	F1	F2	F3	F4	F5	F6
numbering	36	37	38	39	40	41	42	43	44	45	46	47
labeling	G1	G2	G3	G4	G5	G6	H1	H2	H3	H4	H5	H6

The multiplexed 48 bits are then sent to the bit interleaver. Let $x = 0, \ldots, 47$ be the index at the input of the bit interleaver. At the output of the bit interleaver, 24 bits are mapped to the I branch of the constellation, and the other 24 bits are mapped to the Q branch of the constellation. Let $yI = 0, \ldots, 23$ be the index of those bits mapped to the I branch, and let $yQ = 0, \ldots, 23$ be the index of those bits mapped to the Q branch. The bit interleaver in the UEP mapping mode shall implement the relations shown in Equation (63) and Equation (64).

$$yI = \mathrm{mod}((6 \times \mathrm{floor}(x/6) - 5 \times \mathrm{mod}(\mathrm{x,6}), 24), \text{ if } 0 \le x \le 23 \tag{63}$$

$$yQ = \mathrm{mod}((6 \times \mathrm{floor}(x/6) - 5 \times \mathrm{mod}(\mathrm{x,6}), 24), \text{ if } 24 \le x \le 47 \tag{64}$$

The overall readout order of the multiplexer and bit interleaver for UEP mapping is illustrated in Figure 194. Columns are read out from left to right, while the read out order inside each column depends on the constellation used.

column-wise readout (from top to bottom) in this direction always mapped to the I branch

A1	C3	A5	B1	D3	B5	C1	A3	C5	D1	B3	D5
B2	D4	B6	C2	A4	C6	D2	B4	D6	A2	C4	A6

column-wise readout (from top to bottom) in this direction always mapped to the Q branch

E1	G3	E5	F1	H3	F5	G1	E3	G5	H1	F3	H5
F2	H4	F6	G2	E4	G6	H2	F4	H6	E2	G4	E6

Figure 194—UEP mapping mode overall multiplexing and bit interleaving pattern

In the QPSK case, the read out order inside each column is (row 1, row 3, row 2, row 4). Take the first column for example, and the readout is ordered as A1, E1, B2, F2, where A1, B2 are mapped to the Q branch, and E1, F2 are mapped to the I branch.

In the 16-QAM case, the read out order inside each column is (row 1, row 2, row 3, row 4). Take the first column for example, and the readout is ordered as A1, B2, E1, F2, where A1, B2 are mapped to the I branch, and E1, F2 are mapped to the Q branch.

At the receiver side, after demodulation, the received bits shall be deinterleaved. Let $yI = 0, \ldots, 23$, $yQ = 0, \ldots, 23$ be the index at the inputs of the deinterleaver I–branch and Q–branch, respectively, and $z = 0, \ldots, 47$ be the index at the output of the bit deinterleaver. The bit deinterleaver in the UEP mapping mode shall implement the relations shown in Equation (65) and Equation (66).

$$z = \mathrm{mod}((6 \times \mathrm{floor}((yI)/6) - 5 \times \mathrm{mod}(\mathrm{yI,6}), 24), \text{ if } 0 \le z \le 23 \tag{65}$$

$$z = \mathrm{mod}((6 \times \mathrm{floor}(yI/6) - 7 \times \mathrm{mod}(\mathrm{yQ,6}), 24), \text{ if } 24 \le z \le 47 \tag{66}$$

12.4.2.10.4 HRP header data multiplexer

The data multiplexer and bit interleaver for the HRP header is similar to that for EEP mode described in 12.4.2.10.1. At the output of the multiplexer, the 48 encoded bits shall be ordered and numbered as illustrated in Table 147.

After the data multiplexer, the bit interleaver, tone interleaver, and other operations for the PHY followed that for HRP mode index 0.

Table 147—HRP header multiplexing scheme

numbering	0	1	2	3	4	5	6	7	8	9	10	11
labeling	A1	A3	A5	A7	A9	A11	B1	B3	B5	B7	B9	B11
numbering	12	13	14	15	16	17	18	19	20	21	22	23
labeling	C1	C3	C5	C7	C9	C11	D1	D3	D5	D7	D9	D11
numbering	24	25	26	27	28	29	30	31	32	33	34	35
labeling	A2	A4	A6	A8	A10	A12	B2	B4	B6	B8	B10	B12
numbering	36	37	38	39	40	41	42	43	44	45	46	47
labeling	C2	C4	C6	C8	C10	C12	D2	D4	D6	D8	D10	D12

12.4.2.11 Bit reversal tone interleaver

All bits shall be interleaved by a block interleaver with a block size corresponding to the size of FFT in a single OFDM symbol, $N_{sc(HR)}$ or $N_{sc(LR)}$. The interleaver is used so that the adjacent data symbols are mapped onto separate subcarriers.

At the transmitter side, the interleaver permutation shall be defined as follows: Let k be the index of the tones (including data tones, pilot tones, DC tones and null tones) before permutation ranging between 0 and $N_{sc(HR)} - 1$ for HRP modes and between 0 and $N_{sc(LR)} - 1$ for LRP modes. Let i be the index of the interleaved tones over the same range (including data tones, pilot tones, DC tones and null tones) after permutation. Let

$$k = \sum_{j=0}^{L} a_j 2^j \tag{67}$$

where $L = \log_2(N_{sc(HR)}) - 1$ for HRP modes and $L = \log_2(N_{sc(LR)}) - 1$ for LRP modes, with $[a_L, \ldots, a_0]$ being the binary representation of integer k. Then the binary representation of integer i can be written as $[a_0, \ldots, a_L]$, i.e., see Equation (68).

$$i = \sum_{j=0}^{L} a_j 2^{L-j} \tag{68}$$

DC, null, and pilot tones shall be inserted in the bit-reversal position before the tone interleaver. This makes sure that after permutation, the DC, null and pilot tones appear in the pre-specified positions.

12.4.2.12 Signal constellations

The bits for the LRP mode for each subcarrier shall be mapped to –1 when the bit is zero and shall be mapped to +1 when the bit is one.

For QPSK modulation, serial bits shall be divided into groups of $N_{BPSC} = 2$ bits and converted into complex numbers representing QPSK constellation points. The subcarrier mapping for the HRP mode for QPSK modulation is illustrated in Figure 195 where the input bit b_0 is the earliest in time.

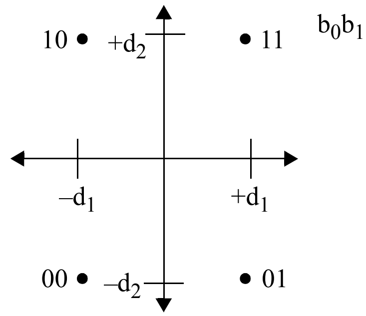

Figure 195—HRP QPSK constellation mapping

For 16-QAM modulation, serial bits shall be divided into groups of $N_{BPSC} = 4$ bits and converted into complex numbers representing 16-QAM constellation points. The subcarrier mapping for the HRP mode for 16-QAM modulation is illustrated in Figure 196 where the input bit b_0 is the earliest in time.

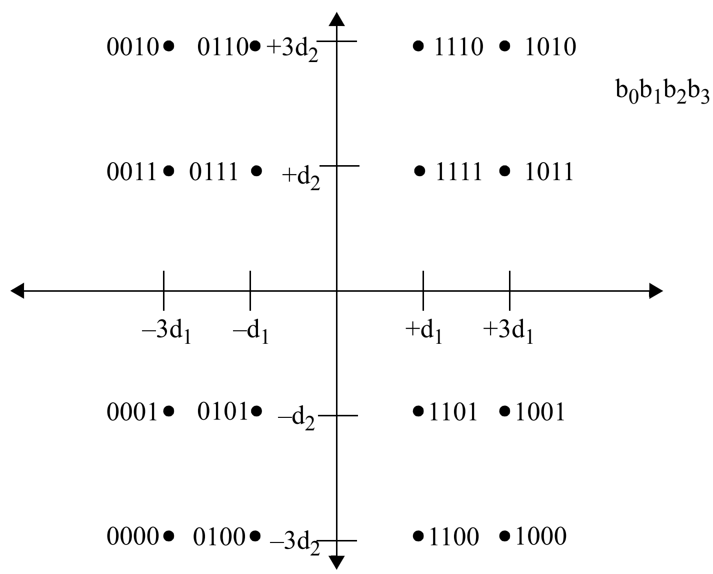

Figure 196—HRP 16-QAM constellation mapping

The constellation parameters for EEP shall be $d_1 = d_2$, while for UEP modulation mapping modes it shall be $d_1 = 1.25 \times d_2$.

The output values, d, are formed by multiplying the resulting $(I + jQ)$ value by a normalization factor K_{MOD}, calculated as $d = (I + jQ) \times K_{MOD}$. The normalization factor, K_{MOD}, is used to achieve the same average power for all mappings. The value of K_{MOD} depends on the modulation and is defined in Table 148.

12.4.2.13 AV PHY non-data subcarriers

The pilot subcarriers for the AV PHY mode shall be modulated with BPSK modulation. For HRP frames, the pilot subcarriers shall be set according to the corresponding subcarrier in the first channel estimation symbol, preamble symbol #5. For the LRP frames, the pilot subcarriers shall be set according to the corresponding subcarrier in the LRP training symbol of Table 155.

Table 148—Normalization factor for PHY modulation formats

Modulation	K_{MOD}
QPSK	$1/(\sqrt{(d_1^2 + d_2^2)})$
16-QAM	$1/(\sqrt{5 \times (d_1^2 + d_2^2)})$

A compliant AV PHY mode receiver may ignore the null and DC subcarriers.

12.4.2.14 AV PHY OFDM modulation

The subcarriers are numbered from $-N_{SC(\{HR,LR\})}/2$ to $N_{SC(\{HR,LR\})}/2 - 1$ where $N_{SC(HR)}$ is defined in Table 140 and $N_{SC(LR)}$ is defined in Table 141. The frequency offset of a subcarrier is determined by Equation (69).

$$f_{offset(sc)}(n) = n \times \Delta f_{SC(\{HR,LR\})} \tag{69}$$

where n is the number of the subcarrier, $\Delta f_{SC(HR)}$ is the HRP subcarrier spacing defined in Table 140, and $\Delta f_{SC(LR)}$ is the LRP subcarrier spacing defined in Table 141.

The HRP subcarriers shall be arranged as indicated in Table 149. In Table 149, symbol 0 is the first channel estimating symbol, symbol #5 in the preamble, as defined in 12.4.3.1. The function mod(x, y) is the modulo function and is defined in 12.2.2.6.3.

Table 149—HRP subcarrier assignment

Subcarrier type	Subcarrier number, *k*
Null	$k = (-256{:}1{:}-178)$ and $(178{:}1{:}255)$

Pilots	for *sym* in 0:*Nsymbol* – 1
	{
	$k = (-177 + \mathrm{mod}(3 \times sym, 22)){:}22{:}177$
	$k \mathrel{!=} (-1,0,1)$
	}

DC	$k = (-1,0,1)$
Data	All remaining

The LRP subcarriers shall be arranged as indicated in Table 150.

The stream of complex symbols from the modulation mapping is divided into groups of $N_{dsc(\{HR,LR\})}$ complex numbers, numbered from $n = 0$ to $n = N_{dsc(\{HR,LR\})} - 1$ where n = 0 corresponds to the first complex number received in time. Each of the complex numbers are mapped sequentially to the subcarriers, skipping the pilots and DC subcarriers. For HRP modes, the complex numbers are mapped beginning with $n = 0$ mapped to $k = -177$ and $n = N_{dsc(HR)} - 1$ mapped to $k = 177$. For LRP modes, the complex numbers are mapped beginning with $n = 0$ mapped to $k = n - 18$ and $n = N_{dsc(LR)} - 1$ mapped to $k = 18$.

Table 150—LRP subcarrier assignment

Subcarrier type	Subcarrier number
Null	–64 to –19 and 19 to 63
Pilots	–14, –6, 6, 14
DC	–1,0,1
Data	–18 to –15, –13 to –7, –5 to –2, 2 to 5, 7 to 13, 15 to 18

12.4.3 AV PHY frame formats

Unless otherwise specified, all fields, values, bits or octets defined as reserved shall be set to zero on transmission and shall be ignored on reception.

The high rate protocol data unit (HRPDU) shall be formatted as illustrated in Figure 197.

MAC frame body	HCS	MAC header	HRP header	HRP preamble

Figure 197—HRPDU frame format

The HRP Header field, MAC Header field, and HCS field, as defined in 12.4.1.4, shall be sent using HRP mode 0 modulation, as defined in Table 134.

The MAC frame body, as defined in 7.2, for HRPDUs is made up of one or more subframes. Each subframe may use a different HRP mode. The HRP mode used in a subframe shall be one that is supported by both the source and destination. Higher efficiency is available if subframes with the same HRP mode index are adjacent to each other in the MAC frame body.

The AV PHY supports the use of the beam forming protocol defined in Clause 13. However, because all AV PHY DEVs are required to support an omni-directional mode, AV PHY piconets shall not use extended beacons, as described in 8.6.2, or the superframe support for directional PHYs described in 8.6.6.

HRPDUs shall be used only in directed CTAs and not in CPs, broadcast CTAs or multicast CTAs.

Two types of low-rate protocol data units (LRPDUs) are defined based on the transmitter antenna setting: omni LRPDU and directional LRPDU. Omni LRPDUs are used for broadcast/multicast LRP frames, such as beacons and frames sent during contention periods, and may also be used for sending MAC commands and data. The directional LRPDU, with and without payload, shall be used to acknowledge HRP frames.

The first omni LRP frame in a CTA shall be sent using the short omni LRP preamble. Subsequent LRP frames sent in a CTA shall use the long omni LRP preamble. All omni ACK frames shall use the short preamble. The short omni LRP preamble shall be used for frames sent in a CP. The beacon frame shall use the long omni LRP preamble.

The omni LRPDU shall be formatted as illustrated in Figure 198.

MAC frame body	HCS	MAC header	Omni LRP header	Omni LRP preamble (long or short)

Figure 198—Omni LRPDU frame format

The directional LRPDU shall be formatted as illustrated in Figure 199.

Directional LRP payload	Directional LRP header	Directional LRP preamble

Figure 199—Directional LRPDU frame format

The MAC frame body for LRPDUs is made up of only one subframe.

12.4.3.1 HRP preamble

The first four symbols of the HRP preamble shall be derived from an 8th-order m-sequence after it is re-sampled 3/2 times. The m-sequence is generated by the polynomial of

$$x^8 + x^7 + x^2 + x + 1$$

as shown in Figure 200 with a period of 255. The re-sampled m-sequence has a reference sample rate of $f_{s(HR)}$ as specified in Table 140.

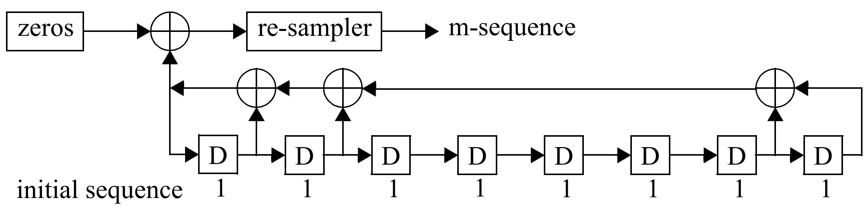

Figure 200—Reference diagram for HRP preamble m-sequence

The time domain preamble shall occupy the time interval corresponding to 4 OFDM symbols by re-sampling a sequence comprised of five repeated re-sampled m-sequences, followed by a sign flipped m-sequence, and filled with sufficient number of zeroes. The beginning of the first m-sequence shall be aligned with the OFDM boundary. The initial state of the shift registers shall be all ones. The DC components of the time domain preamble shall be zero, and the first four symbols shall have 3 dB more TX power than the remaining OFDM symbols. The DC component may be cancelled by shifting the DC-level of the m-sequence before the re-sample. Both I and Q branches of the signal shall have the same value.

The spectrum of the time domain preamble shall conform to the spectrum mask of Figure 155. The 3/2 time re-sampling is a suggestion and any method may be used as long as the transmit PSD mask is met.

The next four symbols of the preamble, 5–8, are defined in the frequency domain, as listed in Table 156. Before converting to time-domain samples the frequency domain values are multiplied by a constant whose value is 1 for symbols 5 and 6 and –1 for symbols 7 and 8. The time domain samples are obtained by taking a 512–point IFFT of the corresponding frequency-domain values. The time-domain samples for symbols 5–6 and symbols 7–8 symbols are connected with continuous-phase, then repeating the last 2×64 samples in the IFFT output for that symbol pair before the first sample, to form a 2×576 sample symbol.

12.4.3.2 HRP header

The HRP header shall be formatted as illustrated in Figure 201.

octets: 3	…	3	1
Subframe header 7	…	Subframe header 1	PHY control

Figure 201—HRP header format

The Subframe Header field shall be formatted as illustrated in Figure 202. Every HRP header shall contain all seven Subframe Header fields.

bits: 20	4
Length	HRP mode index

Figure 202—Subframe header field format

The Length field contains the length, in octets, of the subframe. If a subframe is not present in an HRP frame, then its Length field shall be set to zero.

The HRP Mode Index field shall be set to the HRP mode that is used for that subframe. The indices are defined in Table 134, all other values shall be reserved.

The PHY control field shall be formatted as illustrated in Figure 203.

bits: 2	1	1	1	1	1	1
Reserved	S3	S2	S1	S0	UEP mapping	Reserved

Figure 203—PHY control field format

The bits S0, S1, S2, and S3 are the scrambler initialization seeds as described in 12.4.2.4.

The UEP mapping field shall be set to one if the UEP modes that are used in the subframes use the UEP mapping mode, as defined in 12.4.2.10.3 and 12.4.2.12. It shall be set to zero if the UEP modes in the subframes use the UEP coding mode, as defined in 12.4.2.10.2.

12.4.3.3 LRP preamble sequences

The LRP preambles uses a variety of symbols and chip sequences that are defined in this subclause.

The sequence used in the LRP preamble is created with PN chip sequence with a chip rate of one-half of the LRP reference sampling rate, as defined in Table 141. The PN chip sequence is converted to QPSK by using the same binary PN sequence for the in-phase (I) and quadrature (Q) branches with bipolar mapping, as defined in 12.4.2.12. For the OQPSK mapping, the Q-branch signal is delayed by one-half chip, with respect to the I-branch. The sequence shall be filtered to comply with the LRP TX mask defined in Figure 213.

The Barker-13 chip sequence shall be defined as (–1 –1 –1 –1 –1 1 1 –1 –1 1 –1 1 –1) with the first number listed as the first chip in time.

The 6^{th} order M-sequences shall be generated using the polynomial $x^6 + x^5 + 1$. The LFSR that generates this 6th-order M-sequence is illustrated in Figure 204, where the seed shall be set equal to 0b010111 with the lsb representing the last register.

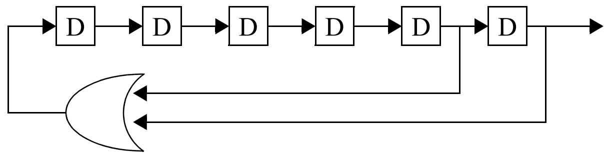

Figure 204—LFSFR for generating 6^{th} order M-sequence

The 12th order M-sequence shall be generated using the polynomial $x^{12} + x^{11} + x^8 + x^6 + 1$. The LFSR that generates this 12^{th}-order M-sequence is illustrated in Figure 205, where the seed shall be set equal to 0xB50 with the lsb representing the last register.

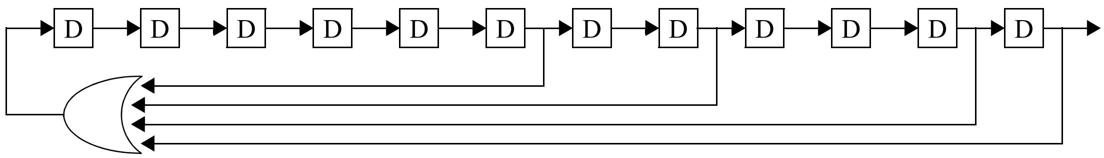

Figure 205—LFSFR for generating 12^{th} order M-sequence

12.4.3.4 Long omni LRP preamble

The long omni LRP preamble is used for frequency synchronization and blind timing recovery. It also allows DEVs receiving the beacon to adjust their transmit frequencies and symbol rates to the accuracy defined in 12.4.4.3 and 12.4.4.4, respectively. The long LRP preamble shall be formatted as illustrated in Figure 206.

Channel estimation	AGC2	RX diversity training	Fine FOE and timing recovery	Coarse FOE and timing recovery	AGC and signal detect

Figure 206—Long omni LRP preamble format

The AGC and Signal Detect field is composed of 78 symbols, with each symbol being spread by the Barker-13 chip sequence, as defined in 12.4.3.3. The symbol sequence is constructed by repeating the 3-symbol sequence (–1, –1, 1) 26 times. The symbols are modulated using the $\pi/4$ rotated QPSK-mapped or OQPSK-mapped sequence defined in 12.4.3.3.

The Coarse FOE (frequency offset estimation) and Timing Recovery field is composed of 81 symbols, with each symbols being spread by the Barker-13 chip sequence, as defined in 12.4.3.3. The symbols sequence is constructed by repeating the 9-symbol sequence (–1, 1, –1, 1, 1, 1, –1, –1, 1) 9 times. The symbols are modulated using the $\pi/4$ rotated QPSK-mapped or OQPSK-mapped sequence defined in 12.4.3.3.

The Fine FOE and Timing Recovery field consists of a 1440 chip PN sequence generated by the 12^{th} order M-sequence defined in 12.4.3.3 that is modulated using the $\pi/4$ rotated QPSK-mapped or OQPSK-mapped sequence defined in 12.4.3.3.

The RX Diversity Training field consists of a 2560 chip PN sequence generated by the 6^{th} order M-sequences defined in 12.4.3.3 that is modulated with the OQPSK sequence defined in 12.4.3.3.

The AGC2 field is an 20 times repetition of a 32-sample long OFDM training symbol equal to the IFFT of the 32-frequency tone vector described by Table 154.

The Channel Estimation field consists of 32 156-sample OFDM training symbols, where each is equal to the IFFT of the 128-frequency tone vector defined in Table 155 and preceded by a 28-sample cyclic prefix.

The TX antenna direction or phased array pattern changes during the long omni preamble fields every N_{switch} samples, where N_{switch} is defined in Table 151.

Table 151—Long omni LRP preamble antenna direction/pattern switching

Field	N_{switch}
AGC and signal detect	78
Coarse FOE	234
Fine FOE	80
RX diversity training	640
AGC2	64
Channel estimation	156

The first four fields of the long omni LRP preamble shall be transmitted with the same average transmit power, P_1. On the other hand, the last two fields of the long omni LRP preamble shall be transmitted with the same average transmit power used in the transmission of the omni LRP header and payload, P_2, which shall be within 3 dB of the P_1.

12.4.3.5 Short omni LRP preamble

The short omni LRP preamble is used for limited timing adjustment and shall be formatted as illustrated in Figure 207.

Channel estimation	AGC	RX diversity training	Signal detection and time sync	AGC 2	AGC 1

Figure 207—Short omni LRP preamble format

The AGC 1 field is 368 chips long and uses the 6th order M-sequences defined in 12.4.3.3 that is modulated with the π/4 rotated QPSK-mapped or OQPSK-mapped sequence defined in 12.4.3.3.

The AGC 2 field is 264 chips long and uses the 6th order M-sequences defined in 12.4.3.3 that is modulated with the π/4 rotated QPSK-mapped or OQPSK-mapped sequence defined in 12.4.3.3.

The Signal Detection and Time Sync field is 720 chips long and uses the 6th order M-sequences defined in 12.4.3.3 that is modulated with the π/4 rotated QPSK-mapped or OQPSK-mapped sequence defined in 12.4.3.3.

The RX Diversity Training field is defined in 12.4.3.4 and is used to select the best RX antenna or combination of antennas.

The AGC field is defined in 12.4.3.4.

The Channel estimation field is defined in 12.4.3.4.

The TX antenna direction or phased array pattern changes during the short omni preamble fields every N_{switch} samples, where N_{switch} is defined in Table 152.

Table 152—Short omni LRP preamble antenna direction/pattern switching

Field	N_{switch}
AGC and signal detect	32
Coarse FOE	48
Fine FOE	80
RX diversity training	640
AGC2	64
Channel estimation	156

The first four fields of the short omni LRP preamble shall be transmitted with the same average transmit power, P_1. On the other hand, the last two fields of the short omni LRP preamble shall be transmitted with the same average transmit power used in the transmission of the omni LRP header and payload, P_2, which shall be within 3 dB of the P_1.

12.4.3.6 Omni LRP header

The Omni LRP header shall use rate 1/3 tail-biting convolutional code, as defined in 12.4.2.8, with eight times repetition coding, defined in 12.4.2.2. The Omni LRP header shall be formatted as illustrated in Figure 208.

bits: 1	1	1	1	12	2	2
S3	S2	S1	S0	Length	Reserved	LRP mode index

Figure 208—Omni LRP header format

The fields S0, S1, S2, and S3 are used to set the initial state of the scrambler, as defined in 12.4.2.5.

The Length field contains the length, in octets of the MAC header, HCS and MAC frame body. This number shall not include the number of stuff bits that are used to create an integer number of symbols for the LRPDU, nor shall it include the number of tail bits.

The LRP Mode Index field indicates the LRP mode, as defined in Table 135, that is used for the MAC Header, HCS, and Frame Body field. The MAC Header, HCS and Frame Body field shall be sent using one of the LRP modes that is supported by both the source and destination.

12.4.3.7 Directional LRP preamble

The Directional LRP preamble consists of five repetitions of the 128-sample OFDM training symbol defined in the frequency domain by Table 155.

12.4.3.8 Directional LRP header and payload

The Directional LRPDU Header field shall be coded into one OFDM symbol using the rate 1/2 tail biting convolutional code, as defined in 12.4.2.8 and repeated as defined in 12.4.2.2. The directional LRPDU uses only one antenna direction or phased array pattern for the preamble, header and payload. This optimum TX antenna direction or pattern is regularly tracked by the LRP receiver while receiving omni-directional LRPDUs with short omni LRP preambles. This information is fed back to the LRP receiver in the header of the next HRP frame. Omni-directional LRPDUs with long omni LRP preamble shall not be used to track the optimum TX antenna direction or pattern.

The Directional LRP Payload field shall be encoded in either LRP mode index 2 or LRP mode index 3, where the mode indexes are defined in Table 135.

The Directional ACK frame uses the directional LRPDU with no Directional LRP Payload field. The Directional ACK header shall be formatted as illustrated in Figure 209.

bits: 8	5	1	1
SCS	ACK group	Reserved	0

Figure 209—Directional ACK header format

The Short Check Sequence (SCS) field shall contain the bit-wise inverse of an 8 bit CRC calculated over the first seven bits of the Imm-ACK or short LRP header and is defined by the polynomial given by $x^8 + x^2 + x + 1$.

The ACK Group n field shall be set to one if the nth ACK group was correctly received and shall be set to zero otherwise. The ACK Group n field shall be set to zero if the subframe was not present in the frame that is being acknowledged.

The ACK Group field shall be formatted as illustrated in Figure 210.

bits: 1	1	1	1	1
ACK group 5	ACK group 4	ACK group 3	ACK group 2	ACK group 1

Figure 210—ACK group field format

The first bit of the Directional ACK LRP header is set to zero to identify it as an LRPDU without any payload.

The directional LRP header for an LRPDU with a payload shall be formatted as illustrated in Figure 211.

bits: 8	4	2	1	1
SCS	Length-1	Reserved	Mode	1

Figure 211—Short LRP header format for directional data frames

The SCS field is the same as defined for the Directional ACK frame. The CRC calculation is equivalent to the one defined in ANSI X3.66-1979. Mathematically, the CRC for the SCS is defined by the following procedure:

1) All 7 bits of the header are complemented and become the coefficients of a polynomial, $M(x)$, of degree 6,
2) the remainder, $R(x)$, is calculated from $[(M(x) \times x^8 + x^7)/G(x)]$,
3) $R(x)$ is complemented to become the CRC.

The Length-1 field contains one less than the length, in octets of the Frame Body field. This number shall not include the number of tail bits or stuff bits that are used to create an integer number of symbols for the LRPDU.

The Mode field shall be set to zero if the LRP payload is transmitted with LRP mode index 2 and it shall be set to one if the LRP payload is transmitted with LRP mode index 3, as defined Table 135.

The first bit of the directional LRP header is set to one to identify the frame as a directional LRPDU with a payload.

The directional LRP payload shall be formatted as illustrated in Figure 212.

octets: 2	**3**
HCS	Receive Status field

Figure 212—Directional LRP payload format

The HCS field contains the two octet HCS, as defined in 11.2.9, calculated over the Receive Status Field.

The Receive Status field is defined in 12.1.8.3.

The number of encoded bits in the Directional LRP Payload field determines if the standard code with tail bits or the tail-biting code is used on the field. If using the tail-biting code would result in a shorter transmission (one OFDM symbol less than the full convolutional code), then the tail-biting code shall be used; otherwise, the standard code with tail bits shall be used.

12.4.4 AV PHY transmitter requirements

12.4.4.1 TX mask

The transmit PSD mask shall be measured with a 3 MHz resolution bandwidth and a 300 kHz video bandwidth. The transmit PSD mask requirement does not include any carrier leakage.

The HRP transmit PSD mask shall conform to the values illustrated in Figure 155.

The LRP transmit PSD mask shall conform to the values illustrated in Figure 213 with the following exceptions:

— For LRP channel 1, the spectral mask shall exclude the interval from f_0 + 150.625 MHz to f_0 + 166.625 MHz.
— For LRP channel 2, the spectral mask shall exclude the interval from f_0–4 MHz to f_0+4 MHz.
— For LRP channel 3, the spectral mask shall exclude the interval from f_0 – 150.625 MHz to f_0 – 166.625 MHz.

The LRP transmit spectral mask shall be measured with a fixed transmit antenna direction or phase pattern, meaning that for omni-directional LRP frames, only one antenna direction or phase pattern is used. This fixed antenna direction or phase pattern may be any of the eight transmit antenna directions or phase patterns described in 12.4.2.2.

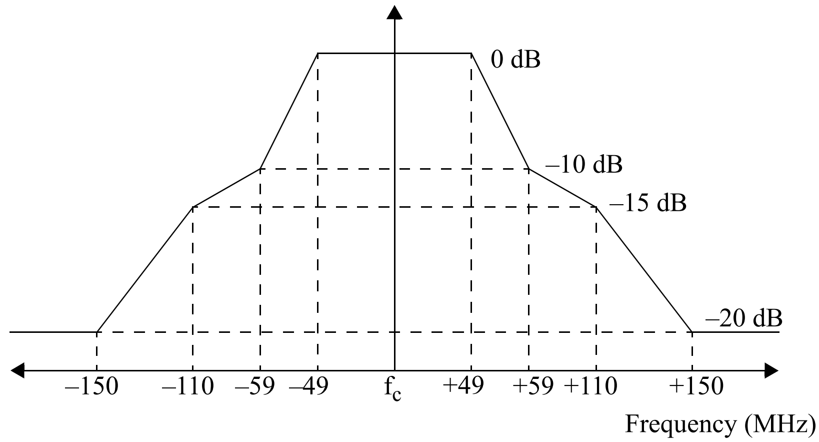

Figure 213—LRP transmit PSD mask

12.4.4.2 EVM requirement

The EVM of a compliant transmitter shall be measured and calculated as defined in 12.1.7 and shall be less than or equal to the values given in Table 153 for the indicated mode.

Table 153—Maximum allowed EVM for AV PHY transmitters

Mode index	Maximum EVM (dB)
LRP 0	–10
LRP 1	–10
LRP 2	–12
LRP 3	–12
HRP 0	–10
HRP 1	–14
HRP 2	–19

In order to measure EVM of omni-directional LRP frames, a fixed antenna direction or phase pattern may be used, where it may be any of the 8 transmit antenna directions or phase patterns described in 12.4.2.2.

12.4.4.3 Symbol timing

The HRP symbol rate shall be $1/T_{S(HR)}$, as defined in Table 140, with an accuracy of better than ±20 µs/s.

The LRP symbol rate shall be $1/T_{S(LR)}$, as defined in Table 141, with an accuracy of better than ±20 µs/s.

In addition, each DEVs in the piconet shall adjust the symbol rate of its LRP and HRP transmitter so that it is within ±1.5 µs/s of the symbol timing of the PNC. The MAC parameter pPHYClockAccuracy shall be ±1.5 µs/s for the AV PHY.

12.4.4.4 TX frequency accuracy

The frequency accuracy of the AV PHY shall be less than ±20 μHz/Hz. In addition, all DEVs in a piconet shall adjust their TX frequency so that it is within ±1.5 μHz/Hz of the frequency of the PNC. The preamble of the beacon is designed to be used by the DEVs to accurately determine the transmit frequency of the PNC. The improved frequency accuracy allows the use of shorter preambles for the LRP and HRP.

12.4.4.5 TX power ramp on and off

The TX power ramp on and off time is implementation dependent and shall be controlled to meet both the timing requirements in 12.4.1.2 and the regulatory restrictions in the appropriate geographical region.

12.4.5 AV PHY Receiver characteristics

12.4.5.1 Error rate criterion

The error rate criterion shall be a BER of less than 1×10^{-7} with data generated by a PN23 sequence as defined by $x^{n+1} = x^{n23} + x^{n18} + 1$, which conforms to CCITT O.151/ITU-T O.151. The error rate is measured at the MAC/PHY interface after any PHY level error correction has taken place.

12.4.5.2 Sensitivity

The receiver sensitivity is determined by measuring, either directly or indirectly, the minimum power input to a single receiver such that the error criterion, as defined in 12.4.5.1, is met.

A compliant LRP receiver shall have a sensitivity that is less than –70 dBm when measured with omni-directional LRP frames with short preamble sent with LRP mode index 1.

A compliant HRP receiver shall have a sensitivity that is less than –50 dBm for HRP mode index 0.

12.4.5.3 Maximum input level

The maximum input level is defined as the highest input power for a single receiver for which the error rate criterion, as defined in 12.4.5.1, is met.

A compliant LRP receiver shall have a maximum input level that is greater than or equal to –30 dBm.

A compliant HRP receiver shall have a maximum input level that is greater than or equal to –24 dBm.

12.4.6 Preambles and training symbols

The 32 sample LRP OFDM training symbol shall have the frequency domain values defined in Table 154. There are 32 tones total, all tones not listed have zero value.

The 128 sample LRP OFDM training symbol shall have the frequency domain values defined in Table 155. There are 128 tones total and the tones not listed have zero value.

The subcarrier numbers and data values for the HRP preamble for symbols 5–8 is listed in Table 156.

Table 154—32 sample LRP OFDM training symbol

OFDM tone number	Value
–4	1
–3	1
–2	1
–1	–1
0	0
1	–1
2	–1
3	1
4	–1

Table 155—128 sample LRP OFDM training symbol

OFDM tone number	Value	OFDM tone number	Value
–18	–1	1	0
–17	1	2	1
–16	1	3	–1
–15	1	4	1
–14	1	5	–1
–13	1	6	–1
–12	–1	7	1
–11	–1	8	–1
–10	1	9	1
–9	–1	10	1
–8	1	11	1
–7	–1	12	1
–6	–1	13	1
–5	–1	14	1
–4	1	15	–1
–3	–1	16	–1
–2	1	17	1
–1	0	18	1
0	0	—	—

Table 156—HRP preamble for symbols 5–8

Subcarrier	Data value	Subcarrier	Data value	Subcarrier	Data value	Subcarrier	Data value
–178	–1	–89	–1	2	–1	91	–1
–177	–1	–88	–1	3	–1	92	–1
–176	1	–87	–1	4	1	93	–1
–175	1	–86	1	5	–1	94	1
–174	–1	–85	–1	6	–1	95	–1
–173	1	–84	–1	7	–1	96	1
–172	1	–83	1	8	1	97	–1
–171	–1	–82	–1	9	1	98	–1
–170	1	–81	1	10	1	99	1
–169	–1	–80	1	11	–1	100	1
–168	–1	–79	1	12	–1	101	–1
–167	–1	–78	1	13	1	102	1
–166	–1	–77	1	14	–1	103	1
–165	1	–76	–1	15	1	104	1
–164	1	–75	–1	16	–1	105	1
–163	–1	–74	1	17	1	106	–1
–162	–1	–73	1	18	–1	107	1
–161	–1	–72	1	19	1	108	1
–160	–1	–71	–1	20	1	109	–1
–159	1	–70	1	21	–1	110	1
–158	–1	–69	–1	22	–1	111	–1
–157	1	–68	–1	23	1	112	1
–156	1	–67	1	24	–1	113	–1
–155	–1	–66	–1	25	–1	114	1
–154	–1	–65	1	26	1	115	1
–153	–1	–64	1	27	1	116	1
–152	1	–63	1	28	–1	117	1
–151	1	–62	–1	29	–1	118	1
–150	–1	–61	–1	30	–1	119	1
–149	1	–60	–1	31	–1	120	–1
–148	–1	–59	–1	32	–1	121	–1
–147	–1	–58	1	33	1	122	–1
–146	1	–57	1	34	1	123	–1

Table 156—HRP preamble for symbols 5–8 *(continued)*

Subcarrier	Data value	Subcarrier	Data value	Subcarrier	Data value	Subcarrier	Data value
–145	–1	–56	–1	35	–1	124	–1
–144	–1	–55	–1	36	–1	125	1
–143	1	–54	–1	37	1	126	–1
–142	1	–53	–1	38	1	127	–1
–141	1	–52	1	39	–1	128	1
–140	1	–51	1	40	–1	129	1
–139	–1	–50	–1	41	–1	130	1
–138	–1	–49	–1	42	1	131	1
–137	1	–48	–1	43	1	132	1
–136	–1	–47	1	44	1	133	–1
–135	1	–46	1	45	–1	134	1
–134	–1	–45	1	46	1	135	1
–133	–1	–44	–1	47	1	136	–1
–132	1	–43	1	48	1	137	1
–131	–1	–42	–1	49	1	138	–1
–130	–1	–41	1	50	1	139	1
–129	–1	–40	–1	51	–1	140	1
–128	–1	–39	1	52	1	141	–1
–127	–1	–38	1	53	–1	142	1
–126	1	–37	1	54	1	143	–1
–125	1	–36	–1	55	1	144	1
–124	–1	–35	–1	56	–1	145	1
–123	1	–34	1	57	–1	146	–1
–122	1	–33	–1	58	1	147	–1
–121	1	–32	1	59	1	148	1
–120	1	–31	1	60	–1	149	–1
–119	–1	–30	–1	61	–1	150	1
–118	1	–29	1	62	1	151	–1
–117	1	–28	1	63	–1	152	1
–116	1	–27	1	64	–1	153	1
–115	–1	–26	1	65	–1	154	1
–114	–1	–25	1	66	1	155	–1
–113	–1	–24	1	67	–1	156	1

Table 156—HRP preamble for symbols 5–8 *(continued)*

Subcarrier	Data value	Subcarrier	Data value	Subcarrier	Data value	Subcarrier	Data value
–112	1	–23	–1	68	–1	157	1
–111	1	–22	1	69	1	158	–1
–110	1	–21	–1	70	1	159	1
–109	1	–20	1	71	1	160	–1
–108	1	–19	1	72	1	161	1
–107	–1	–18	1	73	1	162	1
–106	1	–17	–1	74	1	163	1
–105	–1	–16	–1	75	–1	164	–1
–104	–1	–15	1	76	–1	165	1
–103	–1	–14	1	77	1	166	–1
–102	1	–13	1	78	–1	167	1
–101	1	–12	1	79	1	168	1
–100	1	–11	–1	80	1	169	1
–99	1	–10	–1	81	1	170	–1
–98	1	–9	1	82	–1	171	–1
–97	–1	–8	–1	83	–1	172	1
–96	1	–7	–1	84	–1	173	–1
–95	1	–6	–1	85	1	174	1
–94	1	–5	1	86	–1	175	–1
–93	1	–4	–1	87	–1	176	1
–92	1	–3	–1	88	–1	177	–1
–91	–1	–2	–1	89	1	178	–1
–90	1			90	–1	—	—

13. Beam forming

13.1 Introduction

This clause specifies an optional beam forming protocol. A multitude of antenna configurations such as single antenna element, sectored antennas, switched antennas, and one-dimensional (1-D) and two-dimensional (2-D) beam forming antenna arrays are supported.

Two types of beam forming protocols are specified: an on-demand beam forming and a pro-active beam forming.

On-demand beam forming may be used between two DEVs or between the PNC and a DEV and shall take place in the CTA allocated to the DEV for the purpose of beam forming. The CTA allocated for beam forming shall use one of the beam forming stream indices, as defined in 7.2.5. If a CTA is allocated with sector level training stream index, then the beam forming starts from sector level training. However, if the CTA is allocated with the beam level training stream index, then the beam forming starts directly from beam level training. However a DEV shall not request CTA with the beam level training stream index without first finishing sector level training.

Pro-active beam forming may be used when the PNC is the source of data to one or multiple DEVs. It allows multiple DEVs to train their own receiver antennas for optimal reception from the PNC with lower overhead. During pro-active beam forming, the sector level training from PNC to DEV shall take place in the beacon. The sector level training from DEV to PNC and the beam level training of both directions, shall take place in the CTAP, as described in 13.7.

Pro-active and on-demand beam forming are achieved using a two-level training mechanism, namely a sector (coarse) level training and a beam (fine) level training, followed by an optional high resolution (HRS) tracking phase, as detailed in 13.5.1.

Two beam forming criterion are specified: a beam switching (steering) and tracking (BST) criteria suitable for all antenna configurations, and pattern estimation and tracking (PET) criteria for 1-D linear antenna arrays and 2-D planar antenna arrays. All DEVs that support this beam forming method shall support the BST criterion. The PET criterion may be used only if the two DEVs support it. BST is based on selecting the best beam from a given set of beams whereas PET is based on finding the optimal beam former and combiner vectors (i.e., antenna weights) that do not necessarily fall into the given set of beams.

Support for beam forming is optional. However, when the beam forming in this clause is implemented, the sector level training shall be supported for switched/sectored antennas and the two-level training, as defined in 13.5.1, shall be supported for all other antenna configurations. The tracking phase is optional. SC and HSI DEVs that support beam forming need to support CMS as well, as described in 13.5.1.1.

The interframe spacings (IFSs) used in the beam forming protocol defined in this clause are as follows:

— Beam forming beam level IFS (BBIFS) = 128 SC PHY chips (~73 ns), and
— Beam forming sector level IFS (BSIFS) = 0.5 μs

13.2 Beam forming terminology

This subclause introduces the concept of patterns with increasing resolution level, namely, quasi-omni patterns, sectors, fine beams and HRS beams as illustrated in Figure 214. In addition, the clustering concept is introduced and the convention used in beams numbering and cluster encoding are clarified.

When describing beam forming between two DEVs, the following notation will be used:

a) When two DEVs are communicating, they will be referred to as DEV1 and DEV2. DEV1 may be the PNC. The DEV number, d, will be one for DEV1 (or the PNC) and 2 for DEV2;
b) The total number of transmit and receive antenna elements for DEV number d are denoted as $M^{(d,t)}$ and $M^{(d,r)}$ respectively. The corresponding transmit and receive antenna elements are denoted as $A_n^{(d,t)}$ where $n = 0{:}M^{(d,t)} - 1$ for the transmit antennas and $A_n^{(d,r)}$ where $n = 0{:}M^{(d,r)} - 1$ for the receive antennas;
c) The total number of quasi-omni transmit and receive patterns of interest for DEV number d, is denoted as $I^{(d,t)}$ and $I^{(d,r)}$ respectively. The corresponding quasi-omni transmit and receive patterns are denoted as $Q_n^{(d,t)}$ where $n = 0{:}I^{(d,t)} - 1$ for the transmit patterns and $Q_n^{(d,r)}$ where $n = 0{:}I^{(d,r)} - 1$ for the receive patterns;

d) The best pair of quasi-omni transmit and receive patterns for DEV d when communicating with the other DEV are identified by indices $i^{(d,t)}$ and $i^{(d,r)}$ respectively. The corresponding quasi-omni transmit and receive patterns are denoted as $Q^{(d,t)}_{i^{(d,t)}}$ and $Q^{(d,r)}_{i^{(d,r)}}$ respectively;

e) The total number of transmit and receive sectors of interest for DEV number d are denoted as $J^{(d,t)}$ and $J^{(d,r)}$ respectively. The corresponding transmit and receive sectors are denoted as $S^{(d,t)}_n$ where $n = 0{:}J^{(d,t)} - 1$ for the transmit sectors and $S^{(d,r)}_n$ where $n = 0{:}J^{(d,r)} - 1$ for the receive sectors;

f) The best pair of transmit and receive sectors for DEV d when communicating with the other DEV are identified by indices $j^{(d,t)}$ and $j^{(d,r)}$ respectively. The corresponding transmit and receive sectors are denoted as $S^{(d,t)}_{j^{(d,t)}}$ and $S^{(d,r)}_{j^{(d,r)}}$ respectively;

g) The total number of transmit and receive fine-beams of interest for DEV number d are denoted as $K^{(d,t)}$ and $K^{(d,r)}$ respectively. The corresponding transmit and receive fine-beams are denoted as $B^{(d,t)}_n$ where $n = 0{:}K^{(d,t)} - 1$ for the transmit fine-beams and $B^{(d,r)}_n$ where $n = 0{:}K^{(d,r)} - 1$ for the receive fine-beams;

h) The best pair of transmit and receive fine-beams for DEV d when communicating with the other DEV are identified by indices $k^{(d,t)}$ and $k^{(d,r)}$ respectively. The corresponding transmit and receive fine-beams are denoted as $B^{(d,t)}_{k^{(d,t)}}$ and $B^{(d,r)}_{k^{(d,r)}}$ respectively;

i) The total number of transmit and receive HRS beams of interest for DEV number d, are denoted as $L^{(d,t)}$ and $L^{(d,r)}$ respectively. The corresponding transmit and receive HRS beams are denoted as $H^{(d,t)}_n$ where $n = 0{:}L^{(d,t)} - 1$ for the transmit HRS beams and $H^{(d,r)}_n$ where $n = 0{:}L^{(d,r)} - 1$ for the receive HRS beams. The transmit (receive) HRS beams are grouped into two clusters. The total number of HRS beams in these two clusters are denoted as $L^{(d,t)}_1$ and $L^{(d,t)}_2$ ($L^{(d,r)}_1$ and $L^{(d,r)}_2$) respectively. HRS beams are just an additional set of fine beams;

j) The best pair of transmit and receive HRS beams for DEV d when communicating with the other DEV are identified by indices $l^{(d,t)}$ and $l^{(d,r)}$ respectively. The corresponding transmit and receive HRS beams are denoted as $H^{(d,t)}_{l^{(d,t)}}$ and $H^{(d,r)}_{l^{(d,r)}}$ respectively;

k) If both DEVs are SAS, the superscripts t and r can be dropped since they are the same.

13.2.1 Quasi-omni patterns

The term quasi-omni pattern is the lowest resolution pattern and is used to refer to an antenna pattern that covers a very broad area of the region of space of interest around a DEV (including when the DEV is acting as the PNC). A DEV covers the region of space of interest with a minimal set of, possibly overlapping, quasi-omni patterns. A set size of one indicates that the DEV is able to cover the spatial region of interest with only one quasi-omni pattern, indicating that the DEV is omni-capable.

13.2.2 Sectors

The term sector is the second level resolution pattern and is used to refer to an antenna direction or an array pattern that covers a relatively broad area of multiple beams. A sector can cover a set of consecutive or non-consecutive beams and different sectors can overlap.

13.2.3 Beams

For the pattern estimation and tracking option, beams shall be selected from the beam forming codebooks specified in 13.3.1. A transmit or receive codebook is identified by the number of transmit or receive antennas, $M^{(t)}$ or $M^{(r)}$ respectively, and the desired number of transmit or receive beams, $K^{(t)}$ or $K^{(r)}$ for fine beams, or $L^{(t)}$ or $L^{(r)}$ for HRS beams respectively. For a 2-D antenna array, separate codebooks are associated with each dimension as well as for transmit and receive.

Beam sets of different number of beams can be generated by setting different phases on the same number of antenna elements. Figure 214(c) and Figure 214(d) show an example of an 8-element linear antenna array with 8 fine beams and 16 HRS beams, respectively.

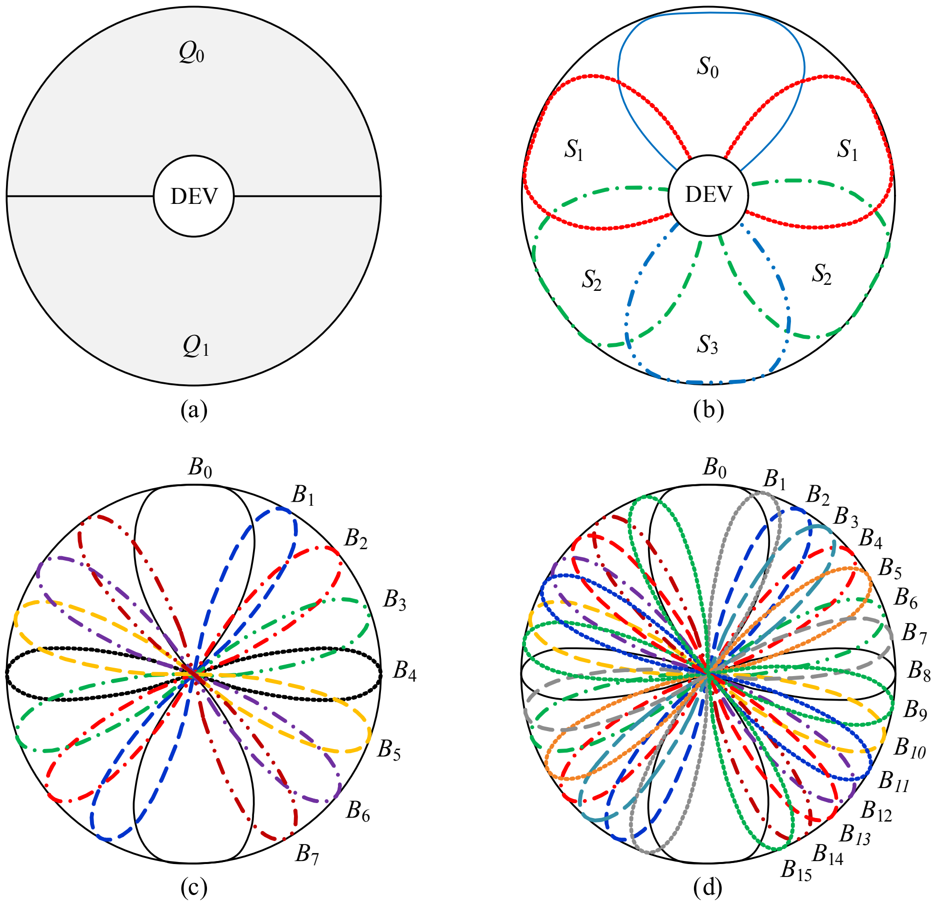

Figure 214—Beam patterns, a) quasi-omni patterns, b) sectors, c) fine beams and d) HRS beams

For a 1-D antenna array with K beams along the z-axis, beams shall be identified by indices zero through $K - 1$ in the direction of increasing polar angle as shown in Figure 214. These beams shall correspond one to one with the beam vectors zero to $K - 1$ from the selected beam forming codebook detailed in 13.3.1.

For a 2-D antenna array with K_x beams on the x-axis and K_z beams on the z-axis, the K_x beams along the x-axis shall be identified by indices zero through $K_x - 1$ in the direction of increasing polar angle and shall correspond one to one with the beam vectors zero to $K_x - 1$ from the selected x-beam forming codebook. The K_z beams along the z-axis shall be identified by indices zero through $K_z - 1$ in the direction of increasing polar angle and shall correspond one to one with the beam vectors zero to $K_z - 1$ from the selected z-beams codebook. This is further illustrated in Figure 215 for a 2-D antenna array with 8 beams in each direction.

13.2.4 Clusters

A cluster is a group of beams around a center beam. The clustering concept is introduced to facilitate tracking. The number of clusters per sector(s) is left to the implementor. Figure 215 gives examples of clusters of different sizes. The circles in Figure 215 represent hypothetical beams.

Cluster encoding shall only be used for DEVs that support the pattern estimation and tracking (PET) option. For DEVs implementing the beam switching and steering option, cluster encoding support is not required.

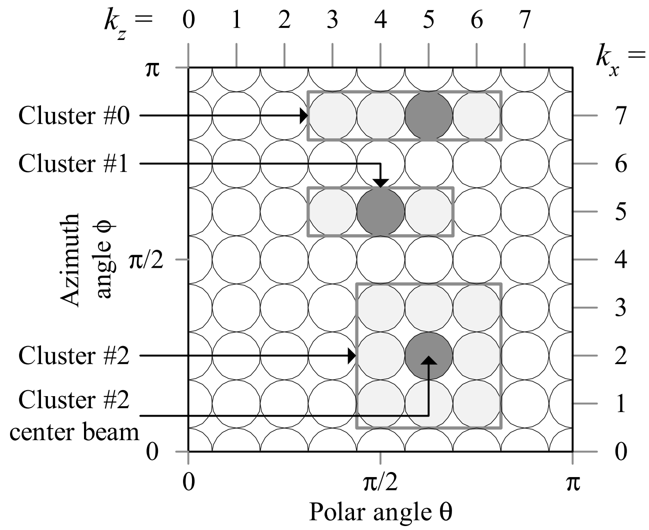

Figure 215—Beam numbering and clusters

A cluster shall be encoded by an 8-bit field: $c_7c_6c_5c_4c_3c_2c_1c_0$. The first three lsb bits, i.e., $c_2c_1c_0$, encode the beams in the polar angle direction in reference to Figure 215, while the second set of three bits, i.e., $c_5c_4c_3$, encodes the beams in the azimuth angle direction. The last set of two bits, c_7c_6, specifies three different 2-D puncturing patterns, i.e., different cluster geometries.

Bits c_1c_0 shall encode the total number of beams in the polar direction (excluding the center beam). The maximum number of beams (excluding the center beam) shall be 3, which corresponds to the setting c_1c_0=11. Bit c_2 shall be set to zero if the number of beams to the left of the center beam is smaller than or equal to the number of beams to the right of the center beam; otherwise, bit c_2 shall be set to one.

Bits c_4c_3 shall encode the total number of beams in the azimuth direction (excluding the center beam). Bit c_5 shall be set to zero if the number of beams below the center beam is smaller than or equal to the number of beams above the center beam; otherwise, bit c_5 shall be set to one.

Bits c_7c_6 shall encode the 2-D puncturing patterns as follows. When bit c_6 is set to one, this shall indicate that the cluster is punctured; otherwise, all 2-D beams within a cluster shall be used. When c_7c_6 = 11, the cluster is fully punctured; i.e., only the beams along the polar angle direction and azimuth angle direction around the center beam are used. When c_7c_6 = 01, the beams along the polar angle direction and azimuth angle direction around the center beam as well as the adjacent beams to the center beam are used. Figure 216 shows some examples of cluster encoding.

Finally, beams in a cluster are ordered in increasing index k_z and decreasing index k_x in reference to Figure 215. When a cluster is transmitted, the first beam, i.e., the beam with lowest k_z index and highest k_x index, shall be transmitted first, and the last beam, i.e., the beam with highest k_z index and lowest k_x index, shall be transmitted last.

13.3 Beam forming codebooks

This subclause specifies codebooks for sectored and switched antennas, 1-D and 2-D arrays with uniform spacing of λ/2. For all other configurations, the BST criterion shall be used, in which case, the knowledge of codebooks at the receiving side is no longer required.

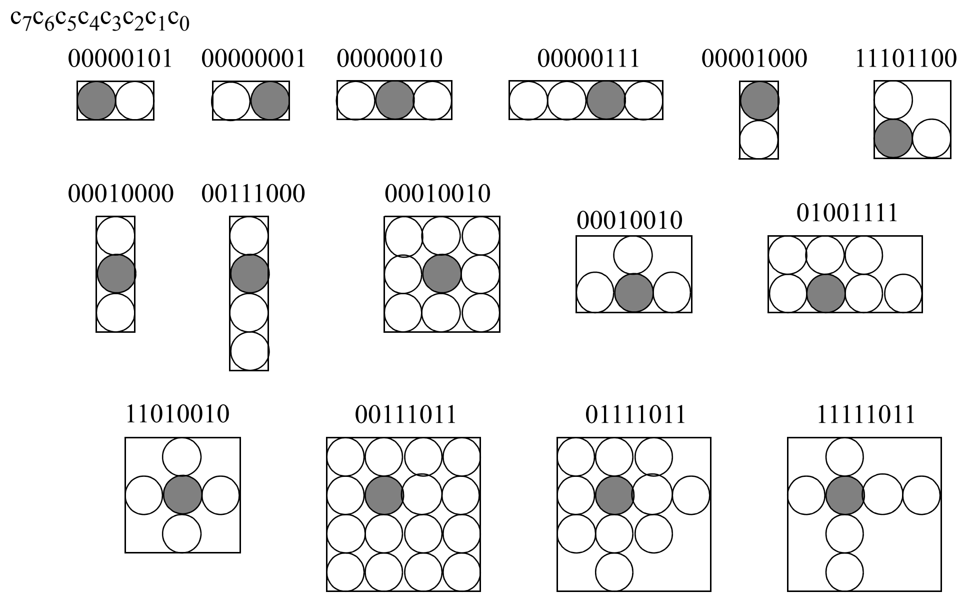

Figure 216—Examples of cluster encoding

A codebook is a matrix where each column specifies the beam former vector or combiner vector to be used. Throughout this subclause, the term codeword is used as a generic term for beam former vector and combiner vector. Each column specifies a specific pattern or direction. The set of columns span the entire space, which is 360 degrees. Columns shall be numbered in increasing order starting with zero.

The codebook for a sectored antenna array and switched antenna array of M elements is a special case of beam forming antenna given by the $M \times M$ identity matrix in Equation (70):

$$\mathbf{W}_{M \times M} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \tag{70}$$

For the purpose of training, it is sufficient to provide codebooks in one dimension. The 2-dimensional antenna arrays can be trained by separable codebooks along the polar angle direction and azimuth angle direction. This is illustrated in Figure 217 where the 2-D antennas weights w_{m_x, m_z}, $m_x = 0{:}M_x - 1$ and $m_z = 0{:}M_z - 1$ can be computed from the antenna weights along the x-axis w_{m_x}, $m_x = 0{:}M_x - 1$ and the antenna weights along the z-axis w_{m_z}, $m_z = 0{:}M_z - 1$ as follows in Equation (71):

$$w_{m_x, m_z} = w_{m_x} w_{m_z} \text{ for } m_x = 0{:}M_x - 1 \text{ and } m_z = 0{:}M_z - 1 \tag{71}$$

Consequently, if the polar codebook has M_x codewords and the azimuth codebook has M_z codewords, the 2-D codebook will have $M_x \times M_z$ codewords.

For a 1-D antenna array with uniform spacing of λ/2, the beam forming codebook for various number of antenna elements is given in 13.3.1.

The following convention shall be used for antenna numbering in reference to Figure 217: the antenna element on the m_x^{th} row ($m_x = 0{:}M_x - 1$) and m_z^{th} column ($m_z = 0{:}M_z - 1$) shall be numbered $m_x M_x + m_z M_z$.

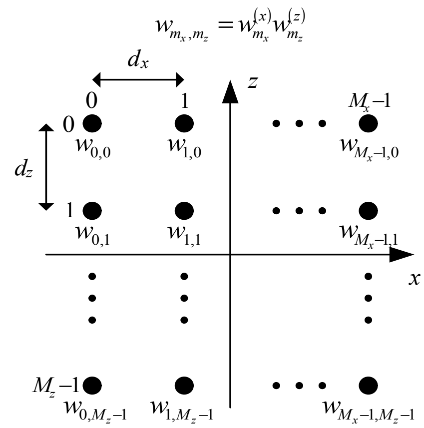

Figure 217—Separable 2-dimensional antenna array

13.3.1 Beam forming codebooks

This subclause provides the beam forming codebooks that should be used for 1-D and 2-D arrays with uniform spacing of λ/2. The codebooks specified here are one-dimensional. The 2-D codebooks can be obtained from the corresponding 1-D polar codebook and azimuth codebook as detailed in 13.3.

Each beam forming codebook is identified by the number of antenna elements, M, and the desired number of beam patterns, K. For the case where $K \geq M$, the codebook beam vectors are given by the column vectors of the matrix shown in Equation (72).

$$W(m, k) = j^{\mathrm{fix}\left\{\frac{m \times mod[k + (K/2), K]}{K/4}\right\}} \text{ for } m = 0{:}M-1 \text{ and } k = 0{:}K-1 \tag{72}$$

The function fix() returns the biggest integer smaller than or equal to its argument. It is also possible to substitute the function round() for the function fix(), where the function round() returns the closest integer to the input argument.

For the special case where $K = M/2$, the codebook beam vectors are given by the column vectors of the matrix shown in Equation (73).

$$W(m, k) = \begin{cases} (-j)^{mod(m, k)} & m = 0{:}N-1 \text{ and } k = 0 \\ (-1)^{\mathrm{fix}\left\{\frac{m \times mod[k + (K/2), K]}{K/4}\right\}} & m = 0{:}N-1 \text{ and } k = 1{:}K-1 \end{cases} \tag{73}$$

The function round() can be substituted for the function fix() as before.

The Codebook ID field is an 8 bit number where the 7 lsb bits indicate the number of desired beam patterns and the msb bit indicates whether the round (msb bit = 0) or fix (msb bit = 1) function is used. The Codebook ID field is used only if both DEVs support PET.

13.4 Beam forming reference model

The beam forming reference model is illustrated in Figure 218. In this figure, DEV1 has $M^{(1,t)}$ transmit antennas and $M^{(1,r)}$ receive antennas while DEV2 has $M^{(2,t)}$ transmit antennas and $M^{(2,r)}$ receive antennas. Depending on the implementation, the transmit and receive antenna weight vectors **w** and **c** belong to specific alphabets. For example, for switched/sectored antennas where one antenna is active at a time, the weights belong to the alphabet {0, 1}. For a phased antenna array implementing specific phase shifts, the weights are restricted to those specific phase shifts. For a complex beam forming antenna array, the weights can be adjusted in both phase and amplitude.

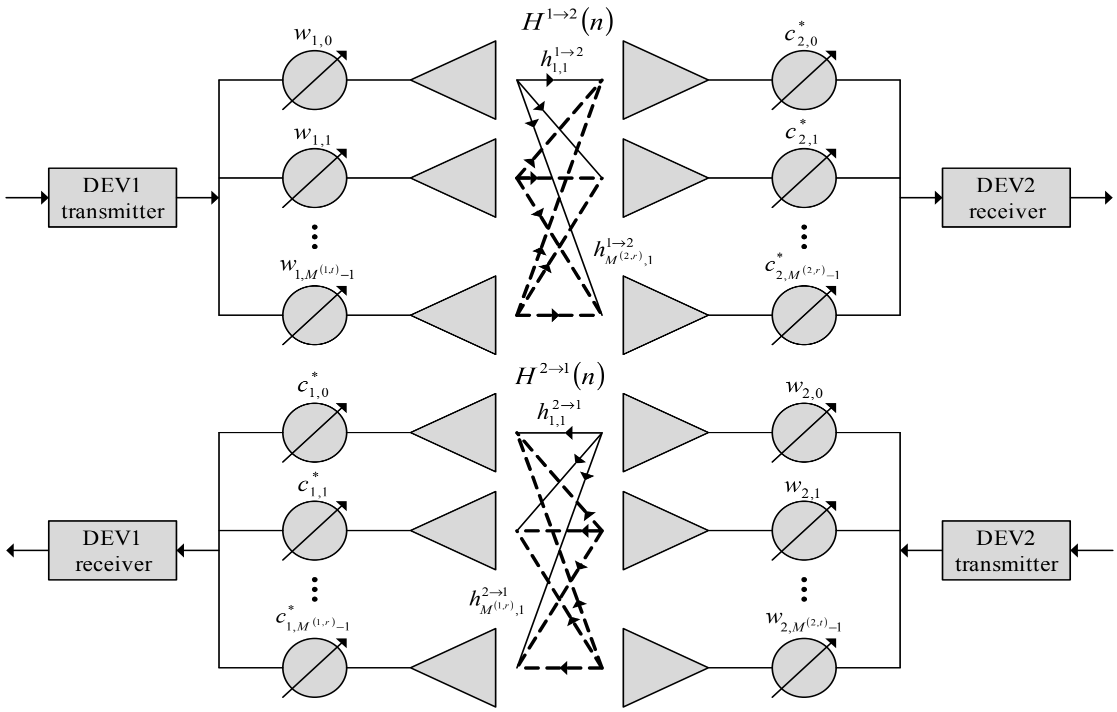

Figure 218—Beam forming reference model

The system model is developed in reference to Figure 218. At DEV1's transmitter, the SC PHY, AV PHY, or HSI PHY bit stream is multiplied by the beam former vector shown in Equation (74).

$$\mathbf{w} = \begin{bmatrix} w_{1,0} & w_{1,1} & \ldots & w_{1,M^{(1,t)}-1} \end{bmatrix}^T \tag{74}$$

and then transmitted through a multipath MIMO channel with frequency domain channel state information (CSI) matrix shown in Equation (75).

$$\mathbf{H}^{1\to2}(n) \in C^{M^{(1,t)} \times M^{(2,r)}} \tag{75}$$

at frequency bin number n, $n = 0, 1, 2, \ldots, N_f$:

$$\mathbf{H}^{1\to2}(n) = \begin{bmatrix} h_{1,1}^{1\to2}(n) & h_{1,2}^{1\to2}(n) & \ldots & h_{1,M^{(2,r)}}^{1\to2}(n) \\ h_{2,1}^{1\to2}(n) & h_{2,2}^{1\to2}(n) & \ldots & h_{2,M^{(2,r)}}^{1\to2}(n) \\ \ldots & \ldots & \ldots & \ldots \\ h_{M^{(1,t)},1}^{1\to2}(n) & h_{M^{(1,t)},2}^{1\to2}(n) & \ldots & h_{M^{(1,t)},M^{(2,r)}}^{1\to2}(n) \end{bmatrix} \tag{76}$$

where $h_{i,j}^{1 \to 2}(n)$ represents the channel response between DEV1 j^{th} transmit antenna and DEV2's i^{th} receive antenna, where j and i range from 1 to the number of transmit and receive antennas, respectively. The number of frequency bins, N_f, corresponds to the subblock length in SC PHY(i.e., $N_f = 256$) or to the number of used carriers in HSI PHY or AV PHY (i.e., $N_f = 352$).

At DEV2's receiver, the received signals a processed through the combiner vector shown in Equation (77).

$$\mathbf{c}_2^T = \begin{bmatrix} c_{2,0} & c_{2,1} & \dots & c_{2,M^{(2,r)}-1} \end{bmatrix} \tag{77}$$

The equivalent channel between DEV1's transmitter and DEV2's receiver is a single input single output (SISO) channel with frequency response at bin n given by Equation (78).

$$G^{1 \to 2}(n) = \mathbf{c}_2^H \mathbf{H}^{1 \to 2}(n)\mathbf{w}_1 \text{ for } n = 0, 1, \dots, N_f - 1 \tag{78}$$

In a similar way, the equivalent channel between DEV2's transmitter and DEV1's receiver is a single input single output (SISO) channel with frequency response at bin n given by Equation (79).

$$G^{2 \to 1}(n) = \mathbf{c}_1^H \mathbf{H}^{2 \to 1}(n)\mathbf{w}_2 \text{ for } n = 0, 1, \dots, N_f - 1 \tag{79}$$

The objective of the pattern estimation beam forming is to select the beam former vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ and the combiner vectors $\mathbf{c}_1$ and $\mathbf{c}_2$ that optimize a cost function that measures the link quality according to a selected criterion. If for example an effective SNR criterion is selected, then DEV2 has to be able to acquire and track the CSI matrices $\mathbf{H}^{1 \to 2}(n)$ for $n = 0{:}N_f - 1$ in the region of space of interest. Furthermore, if the channel is asymmetric, then DEV1 has to be able to acquire and track the CSI matrices $\mathbf{H}^{2 \to 1}(n)$ for $n = 0{:}N_f - 1$. If a beam-switching option is selected, then a DEV needs only to measure the link quality per beam pair. The exact optimization criterion is left to the implementor.

For the special case of a symmetric antenna system (SAS) the same antenna array is used for transmission and reception and for a symmetric channel and so the optimal beam former and combiner vectors are related as follows in Equation (80).

$$\mathbf{c}_1 = \mathbf{w}_1{}^* \text{ and } \mathbf{c}_2 = \mathbf{w}_2{}^* \tag{80}$$

where * indicates the complex conjugate.

For this special case, it is sufficient to determine one of the two vectors for each DEV.

The general case where at least one of the DEVs uses a different antenna system for transmission and reception shall be referred to as asymmetric antenna system (AAS).

Measuring the link quality between all beam pairs, or acquisition of the entire set of CSI matrices, is time costly and incurs high overhead. In order to reduce the amount of time and overhead required for training, a two-level beam forming mechanism shall be used as detailed in 13.5.1.

13.5 Beam forming protocol

The beam forming protocol consists of a two-level training mechanism and an optional tracking phase. The two-level training mechanism consists of a sector level and beam level training and is used to find the best pair of beam patterns between two DEVs with a given beam resolution. Tracking is used to achieve higher resolution and to track the best set of beam patterns between the two DEVs.

13.5.1 Two-level training mechanism

The sector level is used to limit the region of space that is of interest and to find the best pair of sectors. These sectors are then sliced into beams in preparation for beam level training. Beam level training is used to select the optimal transmit beam vector and receiver beam vector as outlined in 13.4. In the simple case of beam switching, this reduces to selecting the best transmit and receive pair of beams. Beam level training is achieved using a set of beam former and combiner vectors (from selected beam forming codebooks) covering the sector(s) selected during sector level training. The beam former (combiner) codebooks are specified in 13.3.1.

13.5.1.1 Sector level training

If the two DEVs are both SAS, then the sector level training shall be performed as described in 13.5.1.1.2. If either or both of the DEVs are AAS, then sector level training shall be performed as described in 13.5.1.1.1.

13.5.1.1.1 AAS sector level training

The sector level consists of following stages:

— Sector training
— Sector feedback
— Sector to beam mapping
— Acknowledgement

The first stage, sector training, shall be divided into two parts: sector training from DEV1 to DEV2 and sector training from DEV2 to DEV1 and shall be formatted as illustrated in Figure 219.

SIFS	Sector training DEV2→DEV2 $J^{(2,t)}$ cycles	SIFS	Sector training DEV1→DEV2 $J^{(1,t)}$ cycles

Figure 219—AAS Sector training

The sector training from DEV1 to DEV2 is illustrated in Figure 220.

Sector training cycle #$J^{(1,t)}-1$	...	Sector training cycle #1	Sector training cycle #0

Figure 220—AAS Sector training DEV1→DEV2

The sector training cycle is illustrated in Figure 221.

BSIFS	sector training #$J^{(2,r)}-1$ $S^{(1,t)}_{j^{(1,t)}} \rightarrow S^{(2,r)}_{J^{(2,r)}-1}$	...	sector training #1 $S^{(1,t)}_{j^{(1,t)}} \rightarrow S^{(2,r)}_{1}$	sector training #0 $S^{(1,t)}_{j^{(1,t)}} \rightarrow S^{(2,r)}_{0}$

Figure 221—AAS Sector training cycle #$J^{(1,t)}$ for sector training DEV1→DEV2

For the SC PHY and HSI PHY, the sector training sequence shall be identical to the CMS preamble, which consists of a CMS SYNC followed by a CMS CES. For the AV PHY, the sector training sequence shall be identical to the HRP preamble.

The sector training from DEV1 to DEV2 consists of $J^{(1,t)}$ cycles. During each cycle, DEV1 shall send $J^{(2,r)}$ repetitions of a sector training sequence in the same direction, i.e., the direction specified by the corresponding sector codeword. Each cycle except the last one shall end with a BSIFS. The $J^{(1,t)}$ cycles shall be sent in $J^{(1,t)}$ different directions, $[S_0^{(1,t)}, S_1^{(1,t)}, \ldots, S^{(1,t)}_{J^{(1,t)}-1}]$ corresponding to the chosen $J^{(1,t)}$ transmit sector codewords.

During a cycle, DEV2 shall attempt to receive each of the $J^{(2,r)}$ sector training sequences using a different listening (receive) direction. The $J^{(2,r)}$ different listening directions, $[S_0^{(2,r)}, S_1^{(2,r)}, \ldots, S_{J^{(2,r)}-1}^{(2,r)}]$, during a cycle shall correspond to DEV2's $J^{(2,r)}$ chosen sector codewords.

At the completion of the full $J^{(1,t)}$ cycles, DEV2 will have had an opportunity to receive a sector training sequence using each combination of DEV1 transmit sector direction (0 to $J^{(1,t)} - 1$) and DEV2 receive sector direction (0 to $J^{(2,r)} - 1$). Based on this information, DEV2 selects the best sector pair, i.e., DEV1's optimal transmit sector, $S_{j^{(1,t)}}^{(1,t)}$, and DEV2's optimal receive sector, $S_{j^{(2,r)}}^{(2,r)}$.

Following the sector training from DEV1 to DEV2, a similar sector training from DEV2 to DEV1 takes place where DEV2 transmits sector training sequences over $J^{(2,t)}$ cycles as shown in Figure 222.

Sector training cycle #$J^{(2,t)} - 1$	...	Sector training cycle #1	Sector training cycle #0

Figure 222—AAS Sector training DEV2→DEV1

The sector training cycles shown in Figure 222 is illustrated in Figure 223.

BSIFS	sector training #$J^{(1,r)} - 1$ $S_{j^{(2,t)}}^{(2,t)} \rightarrow S_{J^{(1,r)}-1}^{(1,r)}$	...	sector training #1 $S_{j^{(2,t)}}^{(2,t)} \rightarrow S_1^{(1,r)}$	sector training #0 $S_{j^{(2,t)}}^{(2,t)} \rightarrow S_0^{(1,r)}$

Figure 223—AAS Sector training cycle #$J^{(1,t)}$ for sector training DEV2→DEV1

At the completion of the cycles, DEV1 selects the best sector pair, i.e., DEV2's optimal transmit sector, $S_{j^{(2,t)}}^{(2,t)}$, and DEV1's optimal receive sector, $S_{j^{(1,r)}}^{(1,r)}$.

The second stage of sector level training is sector feedback and shall be formatted as in Figure 224.

SIFS	DEV2→DEV1 Feedback $S_{j^{(2,t)}}^{(2,t)} \rightarrow S_{j^{(1,r)}}^{(1,r)}$	SIFS	DEV1→DEV2 Feedback $J^{(1,t)}$ repetitions

Figure 224—AAS Sector Feedback

For the SC PHY and HSI PHY, the DEV1→DEV2 feedback is defined as in Figure 225. For the AV PHY with LRP omni feedback, only one repetition is required for the DEV1 to DEV2 feedback repetitions block.

Announce command #$J^{(1,t)} - 1$ $S_{J^{(1,t)}-1}^{(1,t)} \rightarrow S_{j^{(2,r)}}^{(2,r)}$	…	BSIFS	Announce command #$j^{(1,t)}$ $S_{j^{(1,t)}}^{(1,t)} \rightarrow S_{j^{(2,r)}}^{(2,r)}$	BSIFS	Announce command #0 $S_0^{(1,t)} \rightarrow S_{j^{(2,r)}}^{(2,r)}$

Figure 225—AAS DEV1→DEV2 feedback

The Announce command shall be sent $J^{(1,t)}$ times in the $J^{(1,t)}$ different transmit directions, $[S_0^{(1,t)}, S_1^{(1,t)}, \ldots, S_{J^{(1,t)}-1}^{(1,t)}]$. For the AV PHY with LRP omni feedback, only one repetition is required for the DEV1 to DEV2 feedback repetition block. When more than one Announce command is sent as feedback, all frames shall be sent with ACK policy set to no-ACK, with the exception of the last one, which shall be sent with ACK policy set to Imp-ACK.

This is required since DEV1 does not yet know its optimal transmit sector. DEV2 switches to its optimal receive sector, $J_{j^{(2,r)}}^{(2,r)}$, and attempts to receive at least the transmission sent on DEV1's optimal transmit sector, $J_{j^{(1,t)}}^{(1,t)}$. The Feedback IE informs DEV2 of its optimal transmit sector, $S_{j^{(2,t)}}^{(2,t)}$, second best transmit sector, and the corresponding LQIs.

In return, DEV2 shall transmit its sector feedback in a Feedback IE in an Announce command with Imp-ACK requested. The Announce command shall be sent on DEV2's optimal transmit sector, $S_{j^{(2,t)}}^{(2,t)}$,and DEV1 shall listen on its optimal receive sector, $S_{j_{(1,r)}}^{(1,r)}$. The Feedback IE informs DEV1 of its optimal transmit sector, $S_{j_{(1,t)}}^{(1,t)}$, second best transmit sector, and the corresponding LQIs. For the AV PHY, the feedback IE sent from both DEV1 and DEV2 shall be transmitted using LRP mode. For SC PHY and HSI PHY, the feedback IE sent from both DEV1 and DEV2 shall be transmitted using CMS.

Upon completion of the feedback stage, both DEV1 and DEV2 know their optimal transmit and receive sectors. These shall be used for any further frame exchanges in this level.

The mapping stage follows the feedback stage. The mapping stage format is as illustrated in Figure 226.

SIFS	DEV2→DEV1 mapping $S_{j_{(2,t)}}^{(2,t)} \rightarrow S_{j_{(1,r)}}^{(1,r)}$	SIFS	DEV1→DEV2 mapping $S_{j_{(1,t)}}^{(1,t)} \rightarrow S_{j_{(2,r)}}^{(2,r)}$

Figure 226—AAS sector to beam mapping

DEV1 shall transmit its sector to beam mapping in an Announce command with the ACK Policy set to Imp-ACK with the following IEs:

— The Mapping IE field, as defined in 7.4.28, which for the SC PHY and HSI PHY, this IE specifies the number of DEV1 transmit and receive beams and the preamble type (short or long) to be used in the beam level training. For the AV PHY, this IE specifies are the number of DEV1 transmit and receive beams and the HRP preamble. This IE is always present.
— The BST Clustering IE, as defined in 7.4.29, or PET Clustering IE, as defined in 7.4.30, that contains number of DEV1 transmit and receive clusters and the number of beams in each cluster, and the cluster encoding (when both DEVs use PET.) The BST clustering IE shall be exchanged only when both DEVs support tracking.
— The Beam PET IE, as defined in 7.4.31 that contains the number of transmit and receive antennas on the *z*-axis and *x*-axis, the corresponding codebook IDs, and the amplitude and phase resolution capabilities. The Beam PET IE may be exchanged only when both DEVs use PET.

DEV2 shall reply by sending back its own sector to beam mapping in an Announce command with the ACK Policy set to Imm-ACK with the following IEs:

— The Mapping IE field, as defined in 7.4.28, which for the SC PHY and HSI PHY, this IE specifies the number of DEV2 transmit and receive beams and the preamble type (short or long) to be used in the beam level training. For the AV PHY, these are the number of DEV1 transmit and receive beams and the HRP preamble. This IE is always present.
— The BST Clustering IE, as defined in 7.4.29, or PET Clustering IE, as defined in 7.4.30, that specifies the number of DEV2 transmit and receive clusters and the number of beams in each cluster, and the cluster encoding (when both DEVs use PET). The BST clustering IE shall be exchanged only when both DEVs support tracking.
— The Beam PET IE, as defined in 7.4.31, that specifies the number of transmit and receive antennas on the *z*-axis and *x*-axis, the corresponding codebook IDs, and the amplitude and phase resolution capabilities. The Beam PET IE may be exchanged only when both DEVs use PET.

DEV1 shall reply with an Imm-ACK that completes the sector level training.

Figure 227 illustrates the message flow for a successful AAS sector level training process.

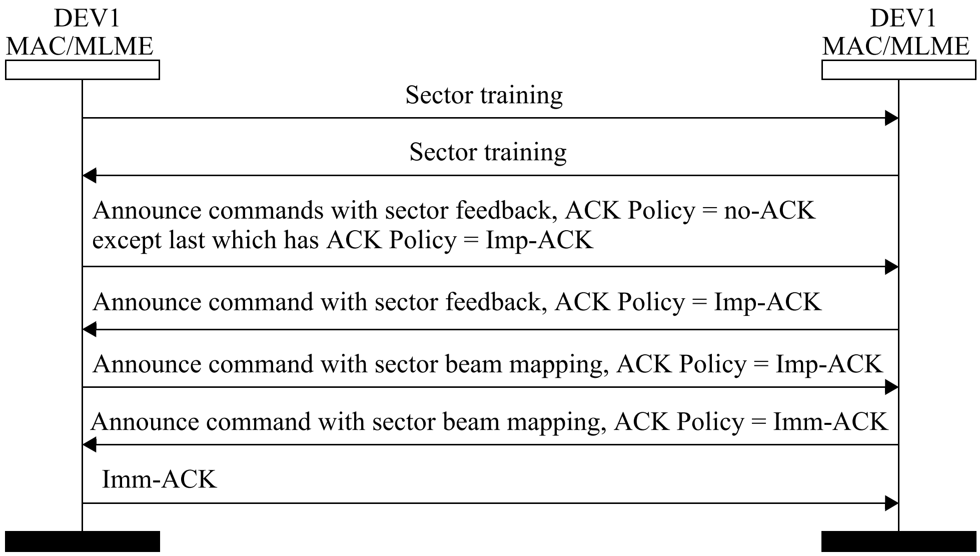

Figure 227—AAS sector level training process

13.5.1.1.2 SAS sector level training

Sector level training for a SAS consists of the following stages:

- — Sector training
- — Sector feedback
- — Sector beam-mapping
- — Acknowledgement

The sector training stage shall be formatted as in Figure 228.

SIFS	Sector training DEV1→DEV2 $J^{(1)}$ cycles

Figure 228—SAS Sector training

The sector training from DEV1 to DEV2 consists of $J^{(1)}$ sector training cycles and is illustrated in Figure 229.

Sector training cycle # $J^{(1)}-1$	...	Sector training cycle #1	Sector training cycle #0

Figure 229—SAS Sector training DEV1→DEV2

The sector training cycle is illustrated in Figure 230.

BSIFS	sector training # $J^{(2)}-1$ $S^{(1)}_{j^{(1)}} \to S^{(2)}_{J^{(2)}-1}$	...	sector training #1 $S^{(1)}_{j^{(1)}} \to S^{(2)}_{1}$	sector training #0 $S^{(1)}_{j^{(1)}} \to S^{(2)}_{0}$

Figure 230—SAS Sector training cycle # $j^{(1)}$ for sector training DEV1→DEV2

The sector training sequence shall be identical to the CMS preamble for the SC PHY and HSI PHY. The sector training sequence shall be identical to the HRP preamble for the AV PHY.

The sector training consists of $J^{(1)}$ cycles. During each cycle, DEV1 shall send $J^{(2)}$ repetitions of a sector training sequence in the same direction. Each cycle except the last one shall end with a BSIFS. The $J^{(1)}$ cycles shall be sent in $J^{(1)}$ different directions, $[S_0^{(1)}, S_1^{(1)}, \ldots, S_{J^{(1)}-1}^{(1)}]$ corresponding to the chosen $J^{(1)}$ sector codewords.

During a cycle, DEV2 shall attempt to receive each of the $J^{(2)}$ sector training sequences using a different direction. The $J^{(2)}$ different directions, $[S_0^{(2)}, S_1^{(2)}, \ldots, S_{J^{(2)}-1}^{(2)}]$ during a cycle shall correspond to DEV2's $J^{(2)}$ chosen sector codewords.

At the completion of the full $J^{(1)}$ cycles, DEV2 will have had an opportunity to receive an sector training sequence using each combination of DEV1 transmit sector (0 to $J^{(1)} - 1$) and DEV2 receive sector (0 to $J^{(2)} - 1$). Based on this information, DEV2 selects the best sector pair, i.e., DEV1's optimal transmit and receive sector, $J_{i^{(1)}}^{(1)}$, and DEV2's optimal transmit and receive sector, $J_{i^{(2)}}^{(2)}$.

The second stage of sector level training is sector feedback and shall be formatted as in Figure 231.

SIFS	DEV2→DEV1 Feedback $J^{(1)}$ repetitions

Figure 231—SAS Sector Feedback

The DEV2→DEV1 feedback frame is defined as in Figure 232.

Announce Command # $J^{(2)} - 1$ $S_{j^{(2)}}^{(2)} \to S_{J^{(1)}-1}^{(1)}$	…	BSIFS	Announce Command # $j^{(2)}$ $S_{j^{(2)}}^{(2)} \to S_{j^{(1)}}^{(1)}$	BSIFS	Announce Command #0 $S_{j^{(2)}}^{(2)} \to S_0^{(1)}$

Figure 232—DEV2→DEV1 feedback

For the SC PHY and HSI PHY, DEV2 shall transmit its sector feedback in a Feedback IE, as defined in 7.4.27, by sending an Announce command with ACK policy set to no-ACK with the exception of the last Announce command, which shall have the ACK policy set to Imp-ACK. The Announce command shall be sent in the optimal transmit sector, $S_{j^{(2)}}^{(2)}$, and shall be repeated $J^{(1)}$ times as shown in Figure 232. This is required since DEV1 does not yet know which sector to use to receive frames from DEV2, and therefore shall listen on each of the $J^{(1)}$ sectors until it hears the Announce command from DEV2. For the AV PHY, the feedback IE sent from DEV2 shall be transmitted using LRP mode. For the SC PHY and HSI PHY, the feedback IE sent from both DEV1 and DEV2 shall be transmitted using CMS. For the AV PHY with LRP omni feedback, only one repetition is required for the DEV2 to DEV1 feedback repetitions block. The feedback IE informs DEV1 of its optimal sector, $S_{j^{(1)}}^{(1)}$, second best sector, and the corresponding LQIs.

Upon completion of the feedback stage, both DEV1 and DEV2 know their optimal transmit and receive sectors. These shall be used for any further frame exchanges in this level.

The mapping stage follows the feedback stage. The mapping stage format is as illustrated in Figure 233.

SIFS	DEV2→DEV1 mapping $S_{j^{(2)}}^{(2)} \to S_{j^{(1)}}^{(1)}$	SIFS	DEV1→DEV2 mapping $S_{j^{(1)}}^{(1)} \to S_{j^{(2)}}^{(2)}$

Figure 233—SAS Sector Beam mapping

DEV1 shall transmit its sector to beam mapping in an Announce command with the ACK Policy set to Imp-ACK with the following IEs:

— The Mapping IE field, as defined in 7.4.28, which for the SC PHY and HSI PHY, this IE contains the number of DEV1 transmit and receive beams and the preamble type (short or long) to be used in the beam level training. For the AV PHY, these are the number of DEV1 beams and HRP preamble. This IE is always present.

- The BST Clustering IE, as defined in 7.4.29, or PET Clustering IE, as defined in 7.4.30, that specifies the number of DEV1 transmit and receive clusters and the number of beams in each cluster, and the cluster encoding (when both DEVs use PET.) The BST clustering IE may be exchanged only when both DEVs support tracking.
- The Beam PET IE, as defined in 7.4.31, that specifies the number of transmit and receive antennas on the *z*-axis and *x*-axis, the corresponding codebook IDs, and the amplitude and phase resolution capabilities. The Beam PET IE shall be exchanged only when both DEVs use PET.

DEV2 shall reply by sending back its own sector to beam mapping in an Announce command with the ACK Policy set to Imm-ACK with the following IEs:

- The Mapping IE field is defined in 7.4.28: For the SC PHY and HSI PHY, these are the number of DEV1 transmit and receive beams and the preamble type (short or long) to be used in the beam level training. For the AV PHY, these are the number of DEV1 beams and HRP preamble. This IE is always present.
- The BST Clustering IE defined in 7.4.29, or PET Clustering IE, as defined in 7.4.30: number of DEV2 transmit and receive clusters and the number of beams in each cluster, and the cluster encoding (when both DEVs use PET.) The BST clustering IE shall be exchanged only when both DEVs support tracking.
- The Beam PET IE is defined in 7.4.31: number of transmit and receive antennas on the *z*-axis and *x*-axis, the corresponding codebook IDs, and the amplitude and phase resolution capabilities. The Beam PET IE may be exchanged only when both DEVs use PET.

DEV1 shall reply with an Imm-ACK that completes the sector level training.

Figure 234 illustrates the message flow for a successful SAS sector level training process.

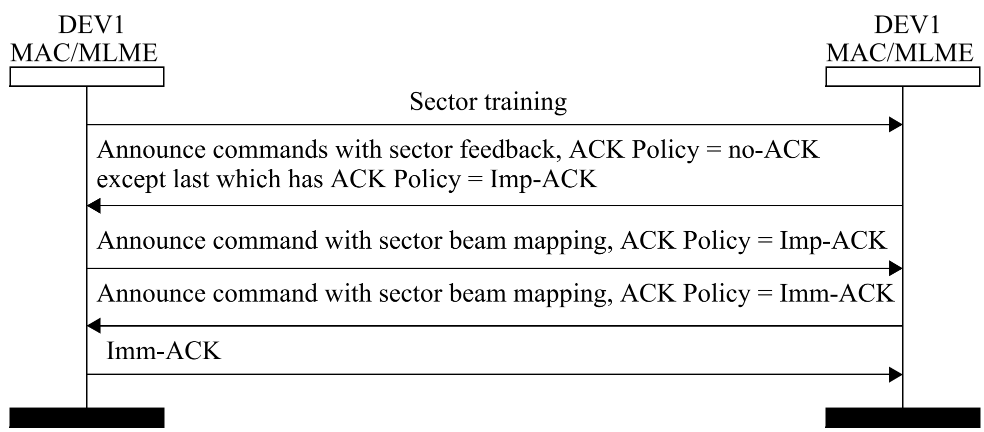

Figure 234—SAS Sector training process

13.5.1.1.3 Sector level training failure remedy

Announce commands may be retransmitted according to 8.8. Should the Announce command in either direction fail to be acknowledged after a number of retransmissions that is implementation dependent, or if there is not enough time remaining in the CTA for the entire frame exchange, the DEV that is attempting the retransmissions shall terminate the stream. DEV1 shall then request a CTA again from the PNC and restart the beam forming process from the beginning.

13.5.1.2 Beam level training

Once sector level training is completed, DEV1 and DEV2 shall start beam level training. The beam level training explores beams within the best sectors to find the best beam pair (best transmit and receive patterns) for DEV1 and DEV2. The AAS case is described first, followed by the SAS case.

If the two DEVs use PET or if tracking is enabled, then when referring to beams transmission, the cluster mapping shall be transmitted in increasing index number, i.e., cluster number 0 shall be transmitted first. Furthermore, beams within a cluster shall be transmitted first in the order specified in 13.2.4.

The AAS beam level training is described first, followed by the SAS case.

13.5.1.2.1 AAS beam level training

The beam level training consists of the following stages:

— Beam training
— Beam feedback
— Beam to HRS beam mapping
— Acknowledgement

The first stage, beam training, shall be divided into two parts: beam training from DEV1 to DEV2 and beam training from DEV2 to DEV1 and shall be formatted as illustrated in Figure 235.

SIFS	Beam training DEV2→DEV2 $K^{(2,t)}$ cycles	SIFS	Beam training DEV1→DEV2 $K^{(1,t)}$ cycles

Figure 235—AAS Beam training

The beam training from DEV1 to DEV2 is illustrated in Figure 236.

Beam training cycle #$K^{(1,t)}-1$	...	Beam training cycle #1	Beam training cycle #0

Figure 236—AAS Beam training DEV1→DEV2

The beam training cycle is shown in Figure 237.

BBIFS	beam training #$K^{(2,r)}-1$ $B^{(1,t)}_{k^{(1,t)}} \to B^{(2,r)}_{K^{(2,r)}-1}$	...	beam training #1 $B^{(1,t)}_{k^{(1,t)}} \to B^{(2,r)}_{1}$	beam training #0 $B^{(1,t)}_{k^{(1,t)}} \to B^{(2,r)}_{0}$

Figure 237—AAS Beam training cycle #$k^{(1,t)}$ for beam training DEV1→DEV2

For the SC PHY and HSI PHY, the beam training sequence shall be transmitted in the mode (preamble type) agreed upon during the sector mapping stage of the sector level training. For the AV PHY, the beam training sequence shall be the HRP preamble.

The beam training from DEV1 to DEV2 consists of $K^{(1,t)}$ cycles. During each cycle, DEV1 shall send $K^{(2,r)}$ repetitions of a beam training sequence in the same direction, i.e., the direction specified by the corresponding beam codeword. Each cycle except the last one shall end with a BBIFS. The $K^{(1,t)}$ cycles shall be sent in the $K^{(1,t)}$ different directions, $[B_0^{(1,t)}, B_1^{(1,t)}, \ldots, B_{K^{(1,t)}-1}^{(1,t)}]$ corresponding to the chosen $K^{(1,t)}$ transmit beam codewords.

During a cycle, DEV2 shall attempt to receive each of the $K^{(2,r)}$ beam training sequence repetitions using a different listening (receive) direction. The $K^{(2,r)}$ different listening directions, $[B_0^{(2,r)}, B_1^{(2,r)}, \ldots, B_{K^{(2,r)}-1}^{(2,r)}]$, during a cycle shall correspond to DEV2's $K^{(2,r)}$ chosen beam codewords.

At the completion of the full $K^{(1,t)}$ cycles, DEV2 will have had an opportunity to receive a beam training sequence using each combination of DEV1 transmit beam (0 to $K^{(1,t)} - 1$) and DEV2 receive beam (0 to $K^{(2,r)} - 1$). Based on this information, DEV2 selects the best beam pair, i.e., DEV1's optimal transmit beam, $B_{k^{(1,t)}}^{(1,t)}$, and DEV2's optimal receive beam, $B_{k^{(2,r)}}^{(2,r)}$.

Following the beam training from DEV1 to DEV2, a similar beam training from DEV2 to DEV1 takes place where DEV2 transmits beam training sequences over $K^{(2,t)}$ cycles as shown in Figure 238.

Beam training cycle #$K^{(2,t)} - 1$	...	Beam training cycle #1	Beam training cycle #0

Figure 238—AAS Beam training DEV2→DEV1

The beam training cycle is shown in Figure 239.

BBIFS	beam training #$K^{(2,r)} - 1$ $B_{k^{(2,t)}}^{(2,t)} \rightarrow B_{K^{(1,r)}-1}^{(1,r)}$	...	beam training #1 $B_{k^{(2,t)}}^{(2,t)} \rightarrow B_1^{(1,r)}$	beam training #0 $B_{k^{(2,t)}}^{(2,t)} \rightarrow B_0^{(1,r)}$

Figure 239—AAS Beam training cycle #$K^{(2,t)}$ for beam training DEV2→DEV1

At the completion of the $K^{(2,t)}$ cycles, DEV1 selects the best beam pair, i.e., DEV2's optimal transmit beam, $B_{k^{(2,t)}}^{(2,t)}$, and DEV1's optimal receive beam, $B_{k^{(1,r)}}^{(1,r)}$.

The second stage of beam level training is beam feedback and shall be formatted as in Figure 240. For the AV PHY with LRP omni feedback, only one repetition is required for the DEV1 to DEV2 feedback repetitions block. For SC PHY and HSI PHY, the feedback IE sent from both DEV1 and DEV2 shall be transmitted using CMS.

SIFS	DEV2→DEV1 feedback $B_{j^{(2,t)}}^{(2,t)} \rightarrow B_{j^{(1,r)}}^{(1,r)}$	SIFS	DEV1→DEV2 feedback $B_{j^{(1,t)}}^{(1,t)} \rightarrow B_{j^{(2,r)}}^{(2,r)}$

Figure 240—AAS Beam feedback

The DEV1→DEV2 feedback shall be sent using an Announce command on DEV1's optimal transmit sector, $S_{j^{(1,t)}}^{(1,t)}$ and DEV2 shall listen on its optimal receive sector, $S_{j^{(2,t)}}^{(2,t)}$. The Announce command shall contain the following IEs:

— The Feedback IE, as defined in 7.4.27, that contains DEV2's optimal transmit beam, $B_{k^{(2,t)}}^{(2,t)}$, the second best transmit beam, and the corresponding LQIs.
— The PET Phase IE, as defined in 7.4.34, that contains DEV2's phase vector, i.e., the phase for each of the $M^{(2,t)}$ transmit antenna elements. This IE may be exchanged only when both DEVs use PET and DEV2's transmit phase resolution is greater than zero.
— The PET Amplitude IE, as defined in 7.4.33, that contains DEV2's amplitude vector, i.e., the amplitude for each of the $M^{(2,t)}$ transmit antenna elements. This IE may be exchanged only when both DEVs use PET and DEV2's transmit amplitude resolution is greater than zero.

In return, DEV2 shall transmit its beam feedback by sending an Announce command with Imp-ACK requested containing the same IEs. The Announce command shall be sent on DEV2's optimal transmit sector, $S_{j^{(2,t)}}^{(2,t)}$ and DEV1 shall listen on its optimal receive sector, $S_{j^{(1,r)}}^{(1,r)}$. For the AV PHY, the beam feedback sent from both DEV1 and DEV2 shall be transmitted using LRP mode. For SC PHY and HSI PHY, the feedback IE sent from both DEV1 and DEV2 shall be transmitted using CMS. The Announce command shall contain the following IEs:

— Feedback IE
— PET phase IE
— PET amplitude IE

Upon completion of the feedback stage, both DEV1 and DEV2 know their optimal transmit and receive beams (patterns). These shall be used for any further frame exchanges in this level.

If either or both of the DEVs do not support tracking, the beam to HRS beam mapping exchange shall be skipped and the last Announce command from DEV2 to DEV1 in the feedback stage shall be sent with Imm-ACK instead.

If tracking is supported, then following the feedback stage, the beam to HRS beam mapping is performed. The Beam to HRS beam mapping shall be formatted as illustrated in Figure 241.

SIFS	DEV2→DEV1 mapping $B^{(2,t)}_{k^{(2,t)}} \to B^{(1,r)}_{k^{(1,r)}}$	SIFS	DEV1→DEV2 mapping $B^{(1,t)}_{k^{(1,t)}} \to B^{(2,r)}_{k^{(2,r)}}$

Figure 241—AAS beam to HRS beam mapping

DEV1 shall transmit its beam to HRS beam mapping in an Announce command with the ACK Policy set to Imp-ACK with the following IEs:

— The Mapping IE, as defined in 7.4.28, that for the SC PHY and HSI PHY, these are the number of DEV1 transmit and receive HRS beams and the SYNC mode to be used in the HRS beam tracking phase. For the AV PHY, these are the number of DEV1 transmit and receive HRS beams and the HRP preamble. This IE is present if both DEVs support tracking.
— The BST Clustering IE is defined in 7.4.29, or PET Clustering IE, as defined in 7.4.30: number of DEV1 transmit and receive clusters and the number of HRS beams in each cluster, and the cluster encoding (when both DEVs use PET). The BST clustering IE shall be exchanged only when both DEVs support tracking.
— The HRS Beam PET IE is defined in 7.4.32: the *z*-axis and *x*-axis HRS codebook IDs. The HRS Beam PET IE shall be exchanged only when both DEVs use PET.

DEV2 shall reply by sending back its own beam to HRS beam mapping in an Announce command with the ACK Policy set to Imm-ACK containing the following IEs:

— Mapping IE
— PET Clustering IE
— HRS beam PET IE

DEV1 shall reply with an Imm-ACK that completes the beam level training.

Figure 242 illustrates the message flow for a successful AAS beam level training process.

13.5.1.2.2 SAS beam level training

The beam level consists of the following stages:

— Beam training
— Beam feedback
— Beam to HRS beam mapping
— Acknowledgement

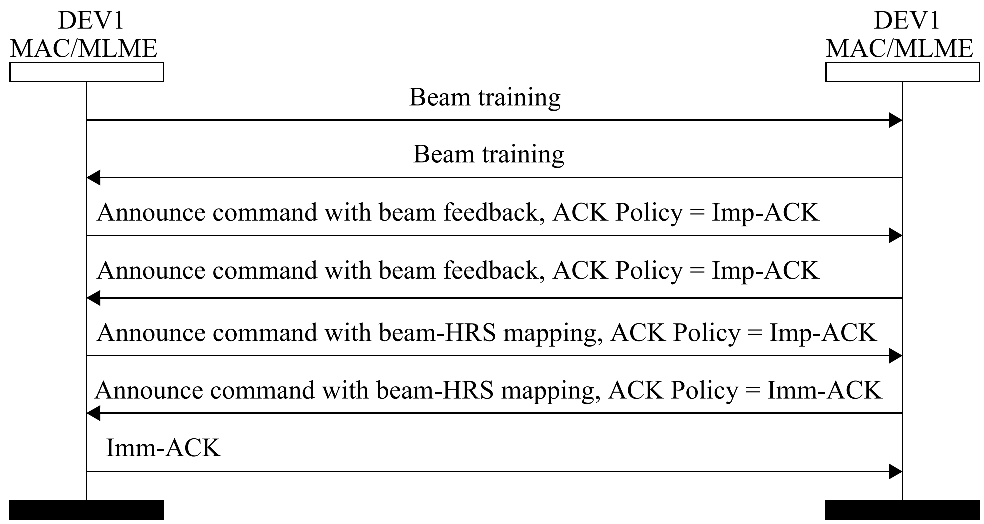

Figure 242—AAS beam level training process

The first stage, beam training, shall be formatted as illustrated in Figure 243.

SIFS	Beam training DEV1→DEV2 $K^{(1)}$ cycles

Figure 243—SAS Beam training

The beam training from DEV1 to DEV2is illustrated in Figure 244.

Beam training cycle #$K^{(1)} - 1$	...	Beam training cycle #1	Beam training cycle #0

Figure 244—SAS Beam training DEV1→DEV2

The beam training cycle is illustrate in Figure 245.

BBIFS	beam training #$K^{(2)} - 1$ $B^{(1)}_{k^{(1)}} \rightarrow B^{(2)}_{K^{(2)}-1}$	...	beam training #1 $B^{(1)}_{k^{(1)}} \rightarrow B^{(2)}_{1}$	beam training #0 $B^{(1)}_{k^{(1)}} \rightarrow B^{(2)}_{0}$

Figure 245—SAS Beam training cycle #$K^{(1)}$ for beam training DEV1→DEV2

For the SC PHY and HSI PHY, the beam training sequence shall be transmitted in the mode agreed upon during the sector mapping stage of the sector level training. For the AV PHY, the beam training sequence shall be the HRP preamble.

The beam training from DEV1 to DEV2 consists of $K^{(1)}$ cycles. During each cycle, DEV1 shall send $K^{(2)}$ repetitions of a beam training sequence in the same direction, i.e., the direction specified by the corresponding beam codeword. Each cycle except the last one shall end with a BBIFS. The $K^{(1)}$ cycles shall be sent in the $K^{(1)}$ different directions, $[B^{(1)}_0, B^{(1)}_1, \ldots, B^{(1)}_{K^{(1)}-1}]$ corresponding to the chosen $K^{(1)}$ transmit beam codewords.

During a cycle, DEV2 shall attempt to receive each of the $K^{(2)}$ beam training sequence repetitions using a different listening (receive) direction. The $K^{(2)}$ different listening directions, $[B^{(2)}_0, B^{(2)}_1, \ldots, B^{(2)}_{K^{(2)}-1}]$, during a cycle shall correspond to DEV2's $K^{(2)}$ chosen beam codewords.

At the completion of the full $K^{(1)}$ cycles, DEV2 will have had an opportunity to receive a beam training sequence using each combination of DEV1 transmit beam (0 to $K^{(1)} - 1$) and DEV2 receive beam (0 to $K^{(2)} - 1$). Based on this information, DEV2 selects the best beam pair, i.e., DEV1's optimal transmit and receive beam, $B^{(1)}_{k^{(1)}}$, and DEV2's optimal receive beam, $B^{(2)}_{k^{(2)}}$.

Following the beam training, beam feedback begins. The beam feedback is formatted as illustrated in Figure 246.

SIFS	DEV2→DEV1 Feedback $S^{(2)}_{j^{(2)}} \rightarrow S^{(1)}_{j^{(1)}}$

Figure 246—SAS Beam feedback

DEV2 shall transmit its beam feedback in an Announce command with Imp-ACK requested. The Announce command shall be sent on DEV2's optimal transmit sector, $S^{(2)}_{j^{(2)}}$ and DEV1 shall listen on its optimal receive sector, $S^{(1)}_{j^{(1)}}$. The Announce command shall contain the following IEs:

— The Feedback IE is defined in 7.4.27: DEV1's optimal transmit and receive beam, $B^{(1)}_{k^{(1)}}$, second best transmit and receive beam, and the corresponding LQIs.
— The PET Phase IE is defined in 7.4.34: DEV1's phase vector, i.e., the phase for each of the $M^{(1)}$ antenna elements. This IE shall be exchanged only when both DEVs use PET and DEV1's phase resolution is greater than zero.
— The PET Amplitude IE is defined in 7.4.33: DEV1's amplitude vector, i.e., the amplitude for each of the $M^{(1)}$ antenna elements. This IE shall be exchanged only when both DEVs use PET and DEV1's amplitude resolution is greater than zero.

Upon completion of the feedback stage, both DEV1 and DEV2 know their optimal transmit and receive beams (patterns). These shall be used for any further frame exchanges in this level.

If either or both of the DEVs do not support tracking, the beam to HRS beam mapping exchange shall be skipped and the last Announce command from DEV2 to DEV1 in the feedback stage shall be sent with Imm-ACK instead.

If tracking is supported, then following the feedback stage, the beam to HRS beam mapping stage is utilized. The Beam to HRS beam mapping is formatted as illustrated in Figure 247.

SIFS	DEV2→DEV1 Mapping $B^{(2)}_{k^{(2)}} \rightarrow B^{(1)}_{k^{(1)}}$	SIFS	DEV1→DEV2 Mapping $B^{(1)}_{k^{(1)}} \rightarrow B^{(2)}_{k^{(2)}}$

Figure 247—SAS Beam to HRS beam mapping

DEV1 shall transmit its beam to HRS beam mapping in an Announce command with Imp-ACK requested with the following IEs:

— The Mapping IE, as defined in 7.4.28, which for the SC PHY and HSI PHY, these are the number of DEV1 transmit and receive HRS beams and the SYNC mode to be used in the HRS beam tracking phase. For the AV PHY, these are the DEV1 HRS beams and the HRP preamble. This IE is present if both DEVs support tracking.
— The BST Clustering IE, as defined in 7.4.29, or PET Clustering IE, as defined in 7.4.30, that contains the number of DEV1 transmit and receive clusters and the number of HRS beams in each cluster, and the cluster encoding (when both DEVs use PET). The BST clustering IE shall be exchanged only when both DEVs support tracking.
— The HRS Beam PET IE, as defined in 7.4.32, that contains the polar angle and azimuth angle HRS codebook IDs. The HRS Beam PET IE shall be exchanged only when both DEVs use PET.

DEV2 shall reply by sending back its own beam to HRS beam mapping in an Announce command with Imm-ACK requested with the following IEs:

— Mapping IE,
— BST clustering IE
— HRS beam PET IE

DEV1 shall reply with an Imm-ACK that completes the beam level training.

Figure 248 illustrates the message flow for a successful SAS beam level training process.

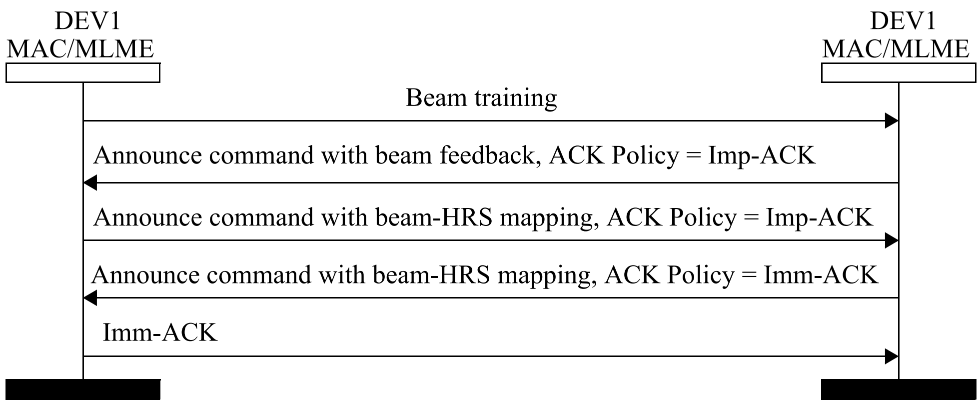

Figure 248—SAS beam level training process

13.5.2 Beam tracking

To improve connectivity, a beam tracking phase is provided. When tracking is enabled, beams within a sector are further grouped into HRS clusters and tracked as described below. Transmission of clusters and beams within a cluster shall follow the order outlined in 13.5.1.2. During tracking, each DEV tracks the best and second best clusters.

The tracking phase is used to achieve higher beam resolution and to track the changes in the beam former and combiner vectors due to channel characteristics variability over time. In the more general case, the beam former and combiner directions (patterns) need to be adjusted dynamically to achieve optimal link quality. Tracking is enabled by clustering, as described in 13.2.4. The best beam and its adjacent beams that are to be tracked shall be grouped as the best cluster. Accordingly the second best beam and its adjacent beams shall be grouped into the second best cluster.

Tracking the best cluster and second best cluster is performed quasi-periodically as defined by the tracking frequency in the mapping IE. The tracking frequency of the best cluster of HRS beams is higher than the tracking frequency of the second best cluster of HRS beams.

Tracking shall take place in the CTA allocated to data transfer from DEV1 to DEV2. Each data frame sent from DEV1 to DEV2 with the Beam Tracking field in the PHY header set to one, shall be followed by a BBIFS followed by one or more high resolution beam training (HT) sequences sent in the HRS beams (directions) identified during the beam level tracking. In the following, such a frame is referred to as a tracking frame. Not every frame sent by DEV1 is a tracking frame and the frequency at which tracking frames are sent is left to the implementor.

The odd tracking frame is used to estimate the best cluster and the even tracking frame is used to estimate the second best cluster. The high resolution beam training sequence frame shall be formatted as illustrated in Figure 249.

SIFS	HRS training cycle # $L^{(1,t)}_{1+mod(m,2)} - 1$	…	HRS training cycle #1	HRS training cycle #0

Figure 249—High resolution beam training (HT)

The HRS training cycles shall be formatted as illustrated in Figure 250 and Figure 251, respectively.

BBIFS	HT #$L_1^{(2,r)} - 1$ $H^{(1,t)}_{1,l^{(1,t)}} \rightarrow H^{(2,r)}_{1,L_2^{(2,r)}-1}$	…	HT #1 $H^{(1,t)}_{1,l^{(1,t)}} \rightarrow H^{(2,r)}_{1,1}$	HT #0 $H^{(1,t)}_{1,l^{(1,t)}} \rightarrow H^{(2,r)}_{1,0}$

Figure 250—HRS training cycle #$l^{(1,t)}$ when mod(m, 2) = 0

BBIFS	HT #$L_1^{(2,r)} - 1$ $H^{(1,t)}_{2,l^{(1,t)}} \rightarrow H^{(2,r)}_{2,L_2^{(2,r)}-1}$	…	HT #1 $H^{(1,t)}_{2,l^{(1,t)}} \rightarrow H^{(2,r)}_{2,1}$	HT #0 $H^{(1,t)}_{2,l^{(1,t)}} \rightarrow H^{(2,r)}_{2,0}$

Figure 251—HRS training cycle #$l^{(1,t)}$ when mod(m, 2) = 1

For DEVs operating in the low-latency mode, as detailed in 8.7a.2, only a single high resolution beam training sequence shall be transmitted in each of the tracking frames as shown in Figure 252. The orders in which the HRS beams are interrogated remain the same as above. It is worth noting that the low-latency tracking from DEV1 to DEV2 is distributed over $L(1,t) \times L(1,r) + L(2,t) \times L(2,r)$ tracking frames.

HRS beam HT	BBIFS	Frame

Figure 252—Frame with beam tracking field set to one for the low-latency case

The tracking frame that interrogates the last beam in either cluster index may be sent with ACK policy set to Imp-ACK and the Last HRS Beam field in the PHY header set to one. The frequency with which this is done is implementation dependent. If DEV2 does not have results indicating that a change in transmit beam is desirable, DEV2 shall respond with an Imm-ACK and tracking continues. If, however, DEV2 has results indicating that DEV1 has a better choice of transmit beam, DEV2 shall respond with a feedback of the results in an empty data frame with ACK policy set to Imm-ACK. Upon receiving any results, DEV1 shall acknowledge the results with an Imm-ACK still using the old transmit beam. It shall then continue the data exchange using the new transmit beam.

If both DEV1 and DEV2 support best and second best beam, switching between best and second best beam will be then triggered by the tracking results at DEV2 if the DEV2 finds the LQI of receive beam is lower than the required threshold and so cannot continue its data exchange anymore. The threshold shall be decided by implementors. If DEV2 does not have results indicating that a change in receive beam is desirable, DEV2 shall respond with an Imm-ACK and tracking continues.

If, however, DEV2 has results indicating that the receive beam cannot be used anymore, DEV2 shall switch to the second best beam immediately and then respond with an empty data frame with ACK policy set to Imm-ACK. Upon receiving any results, DEV1 shall acknowledge the results with an Imm-ACK still using the old transmit beam. It shall then continue the data exchange using the new receive beam.

If both the best and second best beam cannot be used for data exchange, the beam forming procedure will be restarted after a waiting timer is expired (the timer value shall be defined in MLE).

The switching beams sequence is shown in Figure 253.

DEV2 and DEV1 continue data exchange using new beam set	SIFS	DEV1→DEV2 Imm-ACK (old beam set)	SIFS	DEV2→DEV1 Empty frame with Imm-ACK & Feedback results

Figure 253—Switching beam sequence

If the switching beams takes place, than it should be understood to the two DEVs that the old cluster is automatically replaced by the new cluster of the same size and identified by the new beam center.

The tracking and switching process from DEV2 to DEV1 is implemented in a similar way and can be run independently of tracking and switching from DEV1 to DEV2.

13.6 On-demand beam forming

On-demand beam forming takes place in the CTA allocated to the DEV. DEV1 shall reserve a CTA for the special purpose of beam forming acquisition. The sector level training as described in 13.5.1.1 shall occur first followed by the beam level training as described in 13.5.1.2.

13.7 Pro-active beam forming

In the pro-active beam forming, the sector training shall be performed according to sector level training in 13.5.1.1 and shall take place in the sector training section of the beacon part of the superframe. The beacon structure is shown in Figure 254.

Section Level Training Section	PNC Quasi Omni Tracking Section	Quasi Omni Beacon Section

Figure 254—Beacon structure for pro-active beam forming

The message exchange following the sector training as specified in 13.5.1.1 and the beam level tracking as specified in 13.5.1.2 shall take place in the beam forming CTA allocated to the PNC and DEV.

13.8 Transmit switched diversity

Transmit switched diversity (TSD) may be used in case that the PNC is capable of multiple directional antennas that are switched one to another and the DEV is capable of the antenna with omni. The TSD capability is indicated in the TSD supported field in Capability IE, as described in 7.4.11.

If the PNC sends quasi-omni beacons, then the association procedure of TSD shall follow the directional association as described in 8.6.6.3. During data transmission from the PNC to the DEV in a CTA, TSD may be used. When TSD is used, the DEV shall check the LQI from the data frame and shall compare the LQI with TXDiversityTheshold, as described in 6.3.18. If the LQI is lower than TXDiversityTheshold, the DEV shall send the Announce command to the PNC with the Transmit Switched Diversity IE, as defined in 7.4.23, which shall set the Mode field to one and set the Transmit Direction Index field to the next transmit antenna of the PNC. If the PNC successfully receives the Announce command, the PNC shall respond by sending an Announce command to the DEV with TSD IE, which shall set the Mode field to three and the Transmit Direction Index field to the corresponding transmit direction of the PNC. The PNC may transmit data frames after switching the transmit direction to the new transmit direction.

The PNC may announce the TSD switching request time to the DEV by sending an Announce command with TSD IE with the Mode field to zero and the TSD Switching Request Time field set to the new value.

The DEV shall feedback to the PNC periodically by using the Announce command according to TSD switching request time whether the current transmit direction of the PNC should be changed or not, where the Announce command includes TSD IE with the Mode field to one and the Transmit Direction Index field set to the corresponding transmit direction of the PNC.

If the DEV does not know the PNC transmit direction index, the DEV may send the Announce command with TSD IE to the PNC with the mode field set to two. The PNC respond with an Announce command with TSD IE with the mode field set to three and set the Transmit Direction Index field set to the current antenna index of the PNC.

Annex A

(normative)

Frame convergence sublayer

Insert the following subclause after A.2:

A.3 Stream SAP

The Stream SAP is used to transfer time sensitive data streams. Examples of the data intended for this interface include audio and video, particularly uncompressed audio and video. The primitives defined for this SAP are listed in Table A.2.

Table A.2—Stream SAP primitives

Name	Request	Confirm	Indication	Response
STREAM_INITIATE	A.3.1.1	A.3.1.2	A.3.1.3	—
STREAM_MODIFY	A.3.1.4	A.3.1.5	—	—
STREAM_END	A.3.1.6	A.3.1.7	A.3.1.8	—
STREAM_DATA	A.3.2.1	A.3.2.2	A.3.2.3	—

A.3.1 Stream creation, modification and deletion

These primitives are used to start, change or stop a stream connection. The parameters for the primitives are defined in Table A.3.

Table A.3—STREAM_INITIATE, STREAM_MODIFY and STREAM_END primitive parameters

Name	Type	Range	Description
RequestID	Integer	0–255	A unique value created by the originating DEV to match the request primitive to the response primitive
TargetAddress	DEVAddress	Any valid MAC address as defined in 7.1	The address of the target of the primitive.
SourceAddress	DEVAddress	Any valid MAC address as defined in 7.1	The address of the Source of the stream.
StreamIndex	Integer	Any valid stream index, as defined in 7.2.3.	The index of the stream that was created or the index of the stream to be modified or ended.
MinThroughput	Integer	$(1 - 2^{64}) - 1$	The minimum required throughput in bits per second at the MAC SAP.

Table A.3—STREAM_INITIATE, STREAM_MODIFY and STREAM_END primitive parameters

Name	Type	Range	Description
DesiredThroughput	Integer	$(1 – 2^{64}) – 1$	The desired throughput in bits per second at the MAC SAP.
MaxTransmitDelay	Duration	$(1 – 2^{64}) – 1$	The maximum delay in microseconds.
TypicalFrameSize	Integer	0-pMaxFrameBodySize	The typical size, in octets, of an MSDU that would be presented to the MAC SAP.
SECMode	Boolean	TRUE, FALSE	Indicates if security is applied to the stream.
ReliabilityExponent	Integer	0–31	The negative power of 10 that is the that is the maximum FER including retries and frames lost to MaxTransmitDelay. For example, a value of 4 corresponds to an FER < $10^{–4}$. If the value of the parameter is zero, then the parameter is ignored.
TimeOut	Integer	0–65535	The time in milliseconds for the primitive to complete.
AvailableThroughput	Integer	$(1 – 2^{64}) –1$	The estimate of the throughput in bits per second available in the allocated stream.
ResultCode	Enumeration	SUCCESS, FAILURE	Indicates the result of the request.
ReasonCode	Enumeration	REQUEST_TIMEOUT, NOT_ASSOCIATED, TARGET_UNAVAILABLE, TERMINATED_BY_PNC, TERMINATED_BY_DEST, INVALID_STREAM_INDEX, TRANSMIT_DELAY_ UNSUPPORTED RESOURCES_ UNAVAILABLE, OTHER	The reason for a ResultCode of FAILURE.

A.3.1.1 STREAM_INITIATE.request

This primitive is used to setup a data stream. The semantics of this primitive are as follows:

```
STREAM_INITIATE.request      (
                             RequestID,
                             TargetAddress,
                             MinThroughput,
                             DesiredThroughput,
                             MaxTransmitDelay,
                             TypicalFrameSize,
                             SECMode,
                             ReliabilityExponent,
                             TimeOut
                             )
```

The primitive parameters are defined in Table A.3.

A.3.1.2 STREAM_INITIATE.confirm

This primitive is used to report the result of a request to set up a data stream. The semantics of this primitive are as follows:

```
STREAM_INITIATE.confirm      (
                             RequestID,
                             StreamIndex
                             AvailableDataRate,
                             ResultCode,
                             ReasonCode
                             )
```

The primitive parameters are defined in Table A.3.

A.3.1.3 STREAM_INITIATE.indication

This primitive is used to report that the creation of a stream by another DEV. The semantics of this primitive are as follows:

```
STREAM_INITIATE.indication   (
                             SourceAddress,
                             StreamIndex
                             )
```

The primitive parameters are defined in Table A.3.

A.3.1.4 STREAM_MODIFY.request

This primitive is used to modify the parameters of an existing data stream. The semantics of this primitive are as follows:

```
STREAM_MODIFY.request          (
                               RequestID,
                               StreamIndex,
                               MinThroughput,
                               DesiredThroughput,
                               MaxTransmitDelay,
                               TypicalFrameSize,
                               SECMode,
                               ReliabilityExponent,
                               TimeOut
                               )
```

The primitive parameters are defined in Table A.3.

A.3.1.5 STREAM_MODIFY.confirm

This primitive is used to report the result of a request to modify the parameters of an existing data stream. The semantics of this primitive are as follows:

```
STREAM_MODIFY.confirm          (
                               RequestID,
                               StreamIndex
                               AvailableDataRate,
                               ResultCode,
                               ReasonCode
                               )
```

The primitive parameters are defined in Table A.3.

A.3.1.6 STREAM_END.request

This primitive is used to setup a data stream. The semantics of this primitive are as follows:

```
STREAM_END.request             (
                               StreamIndex,
                               TimeOut
                               )
```

The primitive parameters are defined in Table A.3.

A.3.1.7 STREAM_END.confirm

This primitive is used to report the result of a request to set up a data stream. The semantics of this primitive are as follows:

```
STREAM_END.confirm          (
                            StreamIndex
                            ResultCode,
                            ReasonCode
                            )
```

The primitive parameters are defined in Table A.3.

A.3.1.8 STREAM_END.indication

This primitive is used to report that the creation of a stream by another DEV. The semantics of this primitive are as follows:

```
STREAM_END.indication       (
                            StreamIndex,
                            ReasonCode
                            )
```

The primitive parameters are defined in Table A.3.

A.3.2 Stream data interface

These primitives are used to send stream data and report the reception of stream data. The parameters for the primitives are defined in Table A.4.

Table A.4—STREAM_DATA primitive parameters

Name	Type	Range	Description
RequestID	Integer	0–255	A unique value created by the originating DEV to match the request primitive to the response primitive.
SourceAddress	DEVAddress	Any valid MAC address as defined in 7.1	The address of the source of the data.
DestinationAddress	DEVAddress	Any valid MAC address as defined in 7.1	The address of the destination of the data.
StreamIndex	Integer	Any valid stream index, as defined in 7.2.3.	The index of the stream that was created or the index of the stream to be modified or ended.
TransmitTimeout	Integer	$(1 – 2^{32}) – 1$	The maximum allowed delay in microseconds from when the data is presented to the SAP until the frame has finished transmission and the acknowledgement, if required, is received.

Table A.4—STREAM_DATA primitive parameters *(continued)*

Name	Type	Range	Description
SNAPHeaderPresent	Boolean	TRUE, FALSE	TRUE indicates that an LLC/SNAP header is present in the data frame.
ConfirmRequested	Boolean	TRUE, FALSE	Indicates when a confirm primitive is required for the request.
Length	Integer	$0 – (1 – 2^{32}) – 1$	The length of the Data in octets.
Data	Octet string		The information to be sent in the stream connection.
DataType	Enumeration	VIDEO, AUDIO, DATA, UNCOMPRESSED_VIDEO, UNCOMPRESSED_AUDIO	Indicates the type of data that is sent in the stream.
UEPAllowed	Boolean	TRUE, FALSE	Indicates if UEP is allowed for the transmission of the data.
ErrorsFreeData	Boolean	TRUE, FALSE	Indicates if the data that was received contains errors or is free from errors.
InterlacedFieldIndication	Enumeration	TOP_FIELD, BOTTOM_FIELD, NOT_INTERLACED	Indicates if the pixels belong to the top or bottom field of an interlaced image or that the image is not interlaced.
VideoFrameNumber	Integer	As defined in 7.2.9.1.4	A sequential numbering of the video frames that are being transferred.
HPosition	Integer	As defined in 7.2.9.1.4	The horizontal position of the first pixel in the MSDU.
VPosition	Integer	As defined in 7.2.9.1.4	The vertical position of the first pixel in the MSDU.
ResultCode	Enumeration	SUCCESS, FAILURE	Indicates the result of the request.
ReasonCode	Enumeration	TRANSMIT_TIMEOUT, NOT_ASSOCIATED, TARGET_UNAVAILABLE, INVALID_STREAM_INDEX, OTHER	The reason for a ResultCode of FAILURE.

A.3.2.1 STREAM_DATA.request

This primitive is used to send stream data to another DEV. The semantics of this primitive are as follows:

```
STREAM_DATA.request            (
                               RequestID,
                               StreamIndex,
                               TransmitTimeout,
                               SNAPHeaderPresent,
                               ConfirmRequested
                               Length,
                               Data,
                               DataType,
                               UEPAllowed,
                               InterlacedFieldIndication,
                               VideoFrameNumber,
                               HPosition,
                               VPosition
                               )
```

The primitive parameters are defined in Table A.4.

A.3.2.2 STREAM_DATA.confirm

This primitive is used to send stream data to another DEV. The semantics of this primitive are as follows:

```
STREAM_DATA.request            (
                               RequestID,
                               StreamIndex,
                               TransmitDelay,
                               ResultCode,
                               ReasonCode
                               )
```

The primitive parameters are defined in Table A.4.

A.3.2.3 STREAM_DATA.indication

This primitive is used to send stream data to another DEV. The semantics of this primitive are as follows:

```
STREAM_DATA.request          (
                             SourceAddress,
                             DestinationAddress,
                             StreamIndex,
                             SNAPHeaderPresent,
                             Length,
                             Data,
                             DataType,
                             ErrorFreeData,
                             InterlacedFieldIndication,
                             VideoFrameNumber,
                             HPosition,
                             VPosition
                             )
```

The primitive parameters are defined in Table A.4.

Annex D

(normative)

Protocol implementation conformance statement (PICS) proforma

D.7 PICS proforma—IEEE Std 802.15.3-2003[2]

D.7.3 Major capabilities for the MAC sublayer

D.7.3.1 MAC frames

Insert and change the following rows to Table D.3. The other rows are not changed and are not shown.

Table D.3—MAC frames

Item Number	Item Description	Reference	Transmitter		Receiver	
			Status	Support N/A Yes No	Status	Support N/A Yes No
MF3.22	Synchronization	7.4.22	O		O	
MF3.23	Transmit switched diversity	7.4.23	O		O	
MF3.24	UEP specific	7.4.24	O		O	
MF3.25	IFS	7.4.25	O		O	
MF3.26	CTA relinquish duration	7.4.26	O		O	
MF3.27	Feedback	7.4.27	O		O	
MF3.28	Mapping	7.4.28	O		O	
MF3.29	BST clustering	7.4.29	O		O	
MF3.30	Pattern estimation and tracking (PET) cluster-ing	7.4.30	O		O	
MF3.31	Beam PET	7.4.31	O		O	
MF3.32	HRS beam PET	7.4.32	O		O	
MF3.33	PET amplitude	7.4.33	O		O	
MF3.34	PET phase	7.4.34	O		O	

[2] *Copyright release for PICS proforma:* Users of this standard may freely reproduce the PICS proforma in this annex to use it for its intended purpose and may further publish the completed PICS.

Table D.3—MAC frames *(continued)*

Item Number	Item Description	Reference	Transmitter		Receiver	
			Status	Support N/A Yes No	Status	Support N/A Yes No
MF3.35	Sync frame frequency	7.4.35	O		O	
MF4.8	PNC handover request	7.5.3.1	~~M~~O		~~M~~O	
MF4.9	PNC handover response	7.5.3.2	~~M~~O		~~M~~O	
MF4.10	PNC handover information	7.5.3.3	~~M~~O		~~M~~O	

D.7.3.2 MAC sublayer functions

Insert and change the following rows to Table D.4. The other rows are not changed and are not shown.

Table D.4—MAC sublayer functions

Item number	Item Description	Reference	Status	Support		
				N/A	Yes	No
MLF3	PNC handover capable	8.2.3	~~FD2: M~~O			
MLF23	Aggregation	8.7a	O			
MLF24	Block ACK	8.8.3b	O			
MLF25	Beam forming	Clause 13	O			
MLF26	Channel probing	8.9.7	O			
MLF27	UEP	8.16	O			
MLF28	Transmit switched diversity	13.8	O			

Annex D1

(informative)

Implementation considerations

D1.1 Channel time requests

D1.1.1 Types of CTAs

Change Table D1.1 as follows:

Table D1.1—Types of CTAs in the standard

CTA type	SrcID	DestID	Stream Index	Access method(s)
CAP	N/A	N/A	N/A	Uses CSMA/CA, not a real CTA, but it is assigned time in the superframe.
Regular S-CAP	N/A	N/A	N/A	As defined in 8.6.6.
Association S-CAP	N/A	N/A	N/A	As defined in 8.6.6.
Regular CTA	Any valid single DEVID	Any valid single DEVID	A regular stream index	TDMA with transmit control transfer.
Regular MCTA	Any valid single DEVID	Any valid DEVID	MCTA stream index	TDMA with transmit control transfer. This is the same functionality as a regular CTA.
Association CTA	UnassocID	PNCID	Asynchronous stream index	CSMA/CA
Association MCTA	UnassocID	PNCID	MCTA stream index	Slotted aloha
Private CTA	Any valid single DEVID	Any valid single DEVID that is the same as the SrcID	A regular stream index	Not defined by PNC, handled by DEV that has control of the CTA.
Open CTA	BcstID	Any valid DEVID	Asynchronous stream index	CSMA/CA
Open MCTA	BcstID	Any valid DEVID	MCTA stream index	Slotted aloha

Insert the following subclause to Annex D1 as D1.4:

D1.4 Use of multiple mmWave PHYs in a single piconet

The MAC protocol allows the use in a CTA of any PHY mode or rate that is supported by both the source and the destination. Because of this, a DEV that joins a piconet that implements one PHY mode is able to use a different PHY mode when communicating with a DEV that also supports this mode.

As an example, a DEV that supports CMS and an HSI MCS, as defined in 12.3.2.1, is able to join an SC piconet and request a CTA to another DEV that also supports CMS and HSI MCS. The two DEVs are allowed to carry on communications in that CTA using HSI MCSs, even though the piconet is operating in SC mode.

As another example, if a DEV supports HSI, but not CMS, it would need to find an HSI piconet, one of independent, parent or child, to join before it can transfer data. A DEV that supports both CMS and the HSI base rate could act as a bridge to an HSI-only DEV by creating an HSI child piconet in the SC parent piconet. The HSI only DEV could then join this child piconet and transfer data with the dual mode DEV. The dual mode DEV would then be able to either bridge the communications using the IEEE 802.1 sublayer or route the data using layers above layer 2.

As a further example, an HSI PNC-capable DEV may implement the CMS mode instead of HSI mode 0. In that case, the HSI PNC-capable DEV would operate the piconet in SC PHY, sending the beacon using CMS. DEVs in the piconet that support HSI PHY are then able to use the HSI MCSs for communication in CTAs.

Insert the following new annex to fall after Annex D1:

Annex D2

(normative)

Optional OOK/DAMI Modes

D2.1 Introduction

PNC-capable OOK/DAMI DEVs shall use child piconet, as described in D2.2, for non-PNC-capable OOK/DAMI DEV communication.

D2.2 Child piconet operation

When a PNC-capable OOK/DAMI DEV detects an SC piconet, it may join the SC piconet and request to create child piconet using the procedure described in 8.2.5, for non-PNC-capable OOK/DAMI DEV communication.

Alternatively, if a PNC-capable OOK/DAMI DEV does not detect any piconet, it may first start an SC piconet by using CMS. To support communication with other OOK/DAMI DEVs, the PNC-capable OOK/DAMI DEV shall reserves private CTA(s) (as described in 8.4.3.1) and operate a piconet within this reserved private CTA(s) by transmitting OOK/DAMI beacons.

Within the OOK/DAMI piconets in both scenarios described above, OOK/DAMI DEVs shall follow the optional OOK/DAMI mode descriptions in 12.2.8 and the following PHY requirements.

D2.3 DAMI

In DAMI modulation, the coded binary serial input data, $b[k]$, where $k = 0, 1, 2,\ldots$, shall be first precoded to form an intermediate data, $\bar{b}[k]$, defined as follows in Equation (D2.1):

$$\bar{b}[k] = \bar{b}[k-2] \oplus b[k] \tag{D2.1}$$

where the two initial values $\bar{b}[-2]=\bar{b}[-1]=0$ shall be used for precoding. The output, $d[k]$ are formed by Equation (D2.2):

$$\bar{d}[k] = K_{mod}(I[k] + jQ[k]) \tag{D2.2}$$

where $I[k]$ and $Q[k]$ are given by Table D2.1. The resulting constellation is illustrated in Figure 179(b). The normalization factor is sqrt(2).

The transmitted RF signal for a DAMI system is a single-sideband (SSB) modulated signal accompanied by two low-power pilot tones. The SSB signal can be written as shown in Equation (D2.3):

$$s_{SSB}(t) = s(t)\cos(2\pi f_c t) + \bar{s}(t)\sin(2\pi f_c t) \tag{D2.3}$$

where f_c is the center frequency, $s(t)$ is the baseband signal, and $\bar{s}(t)$ is the Hilbert transform of $s(t)$. The baseband signal $s(t)$ can be represented by Equation (D2.4):

$$s(t) = \sum_{k=0}^{N_p - 1} d[k]g(t - kT_{sym}) \tag{D2.4}$$

where N_p is the number of symbols in the frame, T_{sym} is the symbol length, and $g(t)$ is the baseband pulse shape. It is noted that one symbol corresponds to one bit for a DAMI system, meaning that the symbol length is the same as the bit length. The two pilot tones shall have frequencies f_c and f_c–1/(2T_{sym}), respectively. Both of them shall be in phase with the SSB signal. Their amplitudes shall be chosen such that the integrated power of each pilot is 25 dB (with ±1 dB tolerance) below the integrated power of the SSB signal.

Table D2.1—DAMI encoding table

Precoded input bits $\bar{b}[k–2]\bar{b}[k]$	$I[k\}$	$Q[k]$
00	0	0
01	1	
10	–1	
11	0	

D2.4 PHY preamble

In the OOK piconet, the preambles shall be modulated in OOK waveform with spreading factor of 2 in bit repetition. A single mandatory preamble is defined based on the Golay sequence of length 128, denoted $\mathbf{a}_{128}$, as shown in Table 99. Figure D2.1 shows the structure of the PHY preamble.

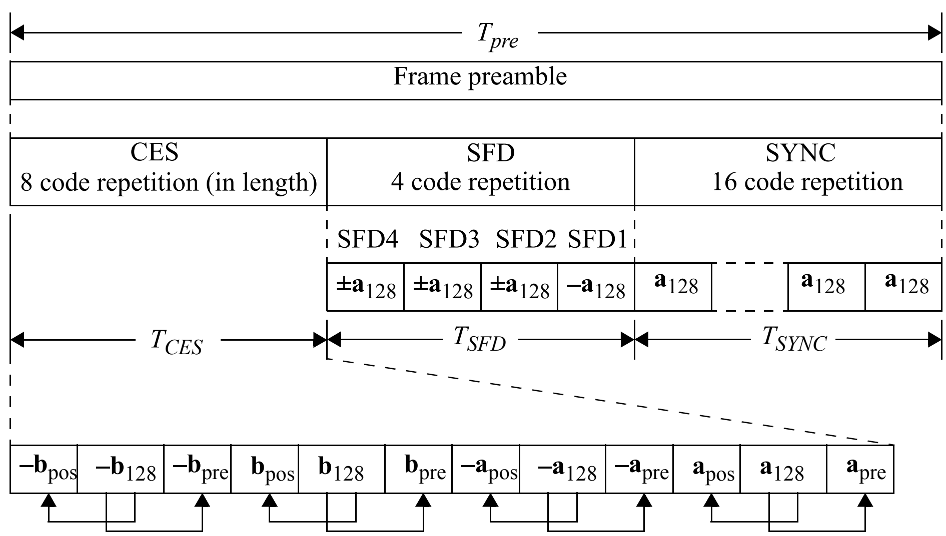

Figure D2.1—OOK preamble structure

The frame synchronization (SNYC) field consists of 16 code repetitions of Golay sequence $\mathbf{a}_{128}$ as given in Table 99. The SFD consists of 4 code repetitions of Golay sequence $\mathbf{a}_{128}$ as given in Table 99. The usage of the four SFD codes shall be as follows: 1 for delimiter, 1 for CES selection and 2 for OOK MCS selections. SFD1 is defined as the delimiter. SFD2, the code for CES selection shall specify whether CES is adopted after the SFD4. If the value of SFD2 is positive, then CES shall be adopted, and if SFD2 is negative, then CES shall not be adopted. SFD3 and SFD4 are OOK MCS selection codes as shown in Table D2.2.

Table D2.2—SFD for OOK MCS selection

SFD pattern, SFD3 SFD4	OOK MCS	Spreading factor
+ +	OOK1	2
+ –	OOK2	1
– +	Reserved	Reserved
– –		

In OOK waveform, a negative sequence shall be derived by bit inverting as follows: –x = Bit_Inverting(x), where x is a sequence in the form of binary bit 0 and 1. Bit_Inverting(x) is an operation to invert all the binary bits 0 of a sequence x to 1 and invert all the binary bits 1 of a sequence x to 0.

If SFD2 is positive and CES is adopted, the CES field shall be constructed from four Golay complementary sequences $\mathbf{a}_{128}$, $-\mathbf{a}_{128}$, $\mathbf{b}_{128}$ and $-\mathbf{b}_{128}$ as shown in Figure D2.1. Each sequence shall be preceded by a cyclic prefix (i.e., a copy of the last 64 bits of the sequence) and followed by a cyclic postfix (i.e., a copy of the first 64 bits of the sequence). The pair of Golay complementary sequences $\mathbf{a}_{128}$ and $\mathbf{b}_{128}$ is given in Table 99, where both sequences in the form of binary bit 0 and 1. Another pair of Golay complementary sequences $-\mathbf{a}_{128}$ and $-\mathbf{b}_{128}$ shall be derived from the previous pair of $\mathbf{a}_{128}$ and $\mathbf{b}_{128}$ by bit inverting.

D2.5 PHY frame format

In the OOK piconet, the frame header and the frame payload shall be modulated in OOK waveform with spreading factor of 2 in bit repetition.

The frame header that is defined in 12.2.3.2 shall be used in the OOK piconet. Then the frame header shall be modulated as shown in Figure D2.2.

— Spread with a PRBS generated using an LFSR as defined in 12.2.2.8.2, and the spreading factor is 16. The 15 bit seed value of the LFSR shall be equal to the value defined in 12.2.2.8.2,
— Spread with bit repetition and the spreading factor is 2,
— Map the frame header onto OOK waveform.

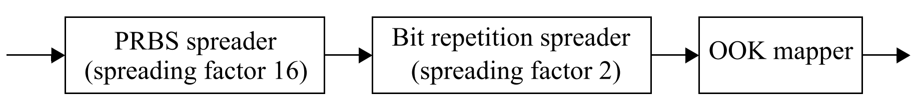

Figure D2.2—OOK frame header spreading process

In OOK data payload, the transmit symbols shall be divided into block of length $N = 508 \times \text{SF}$, where SF is the spreading factor in Table 104. This transmit symbol block shall be appended with pilot symbol as described in Figure D2.3. The pilot symbols consist of a sequence of length $N_p = 4 \times \mathrm{SF}$. The pilot symbols for OOK1 and OOK2 modes shall be chosen according to Table D2.3.

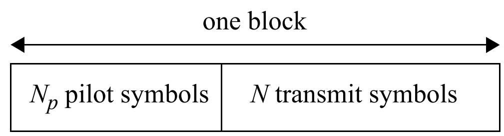

Figure D2.3—OOK frame format with pilot symbols

Table D2.3—OOK pilot symbols

Mode	Pilot symbols
OOK1	11001100
OOK2	1010

D2.6 Interframe space

In the OOK/DAMI piconet, compliant implementation shall use the PHY layer timing parameter values as defined in Table D2.4.

Table D2.4—PHY layer timing parameters

PHY parameter	Value	Subclause
pPHYMIFSTime	0.6 µs, 1.0 µs, 2.0 µs (default)	12.2.6.4
pPHYSIFSTime	1.0 µs, 2.0 µs, 2.5 µs, 6.0 µs (default)	12.2.6.3
pCCADetectTime	5 µs	12.2.5.4
pPHYChannelSwitchTime	100 µs	12.2.6.5

D2.7 Eye opening for OOK

OOK shall have eye amplitude opening of 70% or more. The eye amplitude opening is defined as 2B/(A + B) × 100%, where A is the maximum amplitude and B is the minimum amplitude in eye diagram, as illustrated in Figure D2.4. The test equipment shall have an amplitude measurement accuracy of 1/100 of the average amplitude of the recovered baseband signals. Timing measurement accuracy to detect the signal shall be 1/10 of the clock rate recovered from data.

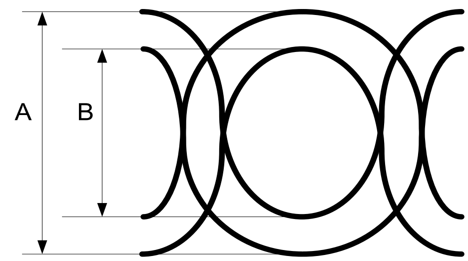

Figure D2.4—OOK eye opening measurement

D2.8 EVM for DAMI

The EVM for DAMI shall be measured following the EVM calculation given in 12.1.7.1 for other SC MCS averaging over 1000 bits. The EVM value for DAMI shall be –14 dB or less.