APPENDIX H

```
leader  01217pam a22003854a 4500
001     991013438199704436
005     20240613125457.0
008     030501s2004    nyua     b    001 0 eng
010     ##$a  2003050852
020     ##$a0738136689
035     ##$z966700-o1cmu_inst
035     ##$a(OCoLC)52311944
040     ##$aDLC $cDLC $dYDX $dPMC
042     ##$apcc
049     ##$aPMCC
050     00$aTK5105.5668 $b.G55 2004
082     00$a004.6/8 $221
100     1#$aGilb, James P. K., $d1965-
245     10$aWireless multimedia : $ba guide to the IEEE 802.15.3 standard / $cJames P.K. Gilb.
260     ##$aNew York : $bStandards Information Network, IEEE Press, $cc2004.
300     ##$axix, 314 p. : $bill. ; $c23 cm.
440     #0$aIEEE standards wireless networks series
504     ##$aIncludes bibliographical references (p. 295-298) and index.
596     ##$a9
650     #0$aIEEE 802.11 (Standard)
650     #0$aMultimedia systems.
650     #0$aWireless LANs.
901     ##$aocm52311944
916     ##$a20040909
917     ##$a20040805
918     ##$aACQORDER
919     ##$a20150707
920     ##$aCATASSIST
994     ##$aE0 $bPMC
999     ##$aTK5105.5668 .G55 2004 $wLC $c1 $iF091687 $lBY-REQUEST $mOFFSITE $p$41.46 $rY $sY $tBOOK $u9/9/2004 FORM=MARC
```



BOOK

Wireless multimedia : a guide to the IEEE 802.15.3 standard

Gilb, James P. K., 1965-

c2004

Available at Offsite Repository BY-REQUEST (TK5105.5668 .G55 2004)

Available Online

TOP
VIEW ONLINE
FIND IN LIBRARY
LINKS
DETAILS
SEND TO
VIRTUAL BROWSE

Find in Library

Please sign in to check if there are any request options. Sign in

BACK TO LOCATIONS

LOCATION ITEMS

Offsite Repository
Available , BY-REQUEST ; TK5105.5668 .G55 2004
(1 copy, 1 available, 1 request)

Item in place
Sign in for loan information

Links

Report a Problem

Signed in and still can't find what you're looking for? Let us know

Display source record

View Online

Full text availability

Wiley Online Library - AutoHoldings Books

Wiley Online Library UBCM all Online Books

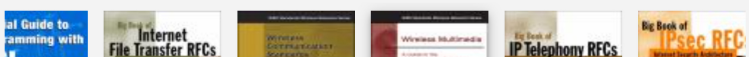
Details

Title	Wireless multimedia : a guide to the IEEE 802.15.3 standard
Creator	Gilb, James P. K., 1965-
Subject	IEEE 802.11 (Standard) Multimedia systems Wireless LANs
Related titles	Available in other form: Wireless multimedia : a guide to the IEEE 802.15.3 standard
Series	IEEE standards wireless networks series
Publisher	New York : Standards Information Network, IEEE Press
Creation Date	c2004
Format	xix, 314 p. : ill. ; 23 cm.
Bibliography Note	Includes bibliographical references (p. 295-298) and index.
Source	Library Catalog

Send to

PRINT | EMAIL | PERMALINK | CITATION | EXPORT RIS | EXPORT BIBTEX | EASYBIB | REFWORKS | ENDNOTE

Virtual Browse




ial guide
ramming
CGI.pm :
standard ...
8
Big book of Internet file transfer RFCs ...
c2000
Wireless communication standards : a study of IEEE 802.11, ...
c2004
Wireless multimedia : a guide to the IEEE 802.15.3 standard ...
c2004
Big book IP telephony RFCs ...
c2001
Big book of IPsec RFCs ...
c2000

APPENDIX I

IEEE Standards Wireless Networks Series

Wireless Multimedia

A Guide to the IEEE 802.15.3™ Standard

Dr. James P. K. Gilb

F0916887

E & S
TK
5105.5668
.G55
2004

IEEE

Published by
Standards Information Network
IEEE Press

SP1132

INTEL-1010
U.S. Patent 8,457,672

EE
Wireless Multimedia
A Guide to the IEEE 802.15.3™ Standard

IEEE Standards Wireless Networks Series

Wireless Multimedia

A Guide to the IEEE 802.15.3™ Standard

Wireless Multimedia: A Guide to the IEEE 802.15.3™ Standard is a view into new applications for wireless multimedia. Consider how wireless network connections and telecommunication have created a technological revolution. Now, we expand that revolution to multimedia!

Written in a clear, concise, and easy to read format by the Technical Editor of the IEEE 802.15.3 standard, this guide is the ultimate resource for those seeking to understand and implement the standard.

IEEE 802.15.3 is the first IEEE standard specifically designed to support wireless multimedia applications in low-cost, high-speed portable devices. The standard is poised for wide adoption and implementation in consumer electronic devices that require multimedia connectivity. Applications for the standard include: digital cameras, telephone-quality audio applications, high-quality audio and video, home theater and stereo components, video conferencing, video intercoms, interactive gaming, remote displays and controls, and more!

This guide details key areas of IEEE 802.15.3, including the development history, a high-level overview of the standard, applications, implementation, interfaces, coexistence, and justification. It is a useful resource, not only for those responsible for implementing the standard, but also for those who plan marketing and business strategies around this technology.

Wireless Multimedia is a MUST READ for anyone who wants to fully understand the inner workings and possibilities of the IEEE 802.15.3 standard.

Look for other books in the series at:

http://standards.ieee.org/StandardsPress

ISBN 0-7381-3668-9

IEEE

INTEL-1010
U.S. Patent 8,457,672

University Libraries
Carnegie Mellon University
Pittsburgh PA 15213-3890

Wireless Multimedia

...A Guide to the IEEE 802.15.3™ Standard

Dr. James P. K. Gilb

IEEE

Published by
Standards Information Network
IEEE Press

INTEL-1010
U.S. Patent 8,457,672

004.68
G146W

Trademarks and Disclaimers

IEEE believes the information in this publication is accurate as of its publication date; such information is subject to change without notice. IEEE is not responsible for any inadvertent errors.

Library of Congress Cataloging-in-Publication Data

Gilb, James P. K., 1965–
Wireless multimedia: a guide to the 802.15.3 standard / by James P. K. Gilb.
p. cm.
Includes bibliographical references and index.
ISBN 0-7381-3668-9
1. IEEE 802.11 (Standard) 2. Multimedia systems. 3. Wireless LANs. I. Title

TK5105.5668.G55 2003
004.6'8—dc21

2003050852

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

Copyright © 2004 by The Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published March 2004. Printed in the United States of America.

No part of this publication may be reproduced in any form, in an electronic retrieval system, or otherwise, without the prior written permission of the publisher.

P√

Yvette Ho Sang, Manager, Standards Publishing
Jennifer Longman, Managing Editor
Andrew Ickowicz, Project Editor
Linda Sibilia, Cover Designer

INTEL-1010
U.S. Patent 8,457,672

IEEE Press/Standards Information Network publications are not consensus documents. Information contained in this and other works has been obtained from sources believed to be reliable, and reviewed by credible members of IEEE Technical Societies, Standards Committees, and/or Working Groups, and/or relevant technical organizations. Neither the IEEE nor its authors guarantee the accuracy or completeness of any information published herein, and neither the IEEE nor its authors shall be responsible for any errors, omissions, or damages arising out of the use of this information.

Likewise, while the author and publisher believe that the information and guidance given in this work serve as an enhancement to users, all parties must rely upon their own skill and judgement when making use of it. Neither the author nor the publisher assumes any liability to anyone for any loss or damage caused by any error or omission in the work, whether such error or omission is the result of negligence or any other cause. Any and all such liability is disclaimed.

This work is published with the understanding that the IEEE and its authors are supplying information through this publication, not attempting to render engineering or other professional services. If such services are required, the assistance of an appropriate professional should be sought. The IEEE is not responsible for the statements and opinions advanced in the publication.

Review Policy

The information contained in IEEE Press/Standards Information Network publications is reviewed and evaluated by peer reviewers of relevant IEEE Technical Societies, Standards Committees and/or Working Groups, and/or relevant technical organizations. The authors addressed all of the reviewers' comments to the satisfaction of both the IEEE Standards Information Network and those who served as peer reviewers for this document.

The quality of the presentation of information contained in this publication reflects not only the obvious efforts of the authors, but also the work of these peer reviewers. The IEEE Press acknowledges with appreciation their dedication and contribution of time and effort on behalf of the IEEE.

To order IEEE Press Publications, call 1-800-678-IEEE.

Print: ISBN 0-7381-3668-9 SP1132

See other IEEE standards and standards-related product listings at: http://standards.ieee.org/

INTEL-1010
U.S. Patent 8,457,672

Trademarks

Bluetooth is a registered trademark of Bluetooth SIG, Inc.(www.bluetooth.org/).

Certicom is a trademark of Certicom Corporation (http://www.certicom.com/).

Cisco is a registered trademark of Cisco Systems, Inc. (www.cisco.com/).

Conexant is a registered trademark of Conexant Systems, Inc. (http://www.conexant.com/).

FireWire is a trademark of Apple Computer, Inc. (http://www.apple.com/).

iDEN is a trademark of Motorola, Inc. (http://www.motorola.com/).

IEEE and 802 are registered trademarks of the Institute of Electrical and Electronics Engineers, Incorporated (www.ieee.org/).

IEEE Standards designations are trademarks of the Institute of Electrical and Electronics Engineers, Incorporated (www.ieee.org/).

Intel is a registered trademark of Intel Corporation (http://www.intel.com/).

Jini is a trademark of the Sun Microsystems, Inc. (http://www.sun.com/).

Kodak is a trademark of Eastman Kodak Company (http://www.kodak.com/).

Maxim is a registered trademark of Maxim Integrated Products (http://www.maxim-ic.com/).

Memory Stick is a registered trademark of the Sony Corporation (http://www.sony.com).

Motorola is a registered trademark of Motorola, Inc. (http://www.motorola.com/).

Nextel is a trademark of the Nextel Communications, Inc. (http://www.nextel.com).

Ntru is a trademark of Ntru Cryptosystems, Inc. (http://www.ntru.com/).

Sharp is a registered trademark of Sharp Corporation (http://www.sharpusa.com).

Sony is trademark of Sony Corporation (http://www.sony.com).

Texas Instruments is a trademark of Texas Instruments Incorporated (http://www.ti.com).

Universal Plug and Play or UPnP is a trademark of Implementers Corporation (http://www.upnp-ic.org/).

Wi-Fi is a registered trademark of the Wi-Fi Alliance (http://www.wi-fi.org/).

INTEL-1010
U.S. Patent 8,457,672

Dedication

I would like to dedicate this work to my wife and family, who are the most important people in my life, for their love, support, encouragement and sacrifice during the completion of the standard and this book.

James P. K. Gilb

University Libraries
Carnegie Mellon University

INTEL-1010
U.S. Patent 8,457,672

Acknowledgement

A standard like IEEE 802.15.3 is not the work of one person, but rather depends critically on the contributions of a diverse group of individuals who dedicate their time to reviewing the drafts and solving problems. I would like to thank all of the members of the 802.15.3 Task Group, the letter ballot voters of the 802.15 Working Group and the sponsor ballot voters for the hard work that went into developing and reviewing this standard.

I would like to especially thank Jim Allen (802.15 Vice Chair and 802.15.3 Vice Chair) for encouraging me to write this book and for his diligent review of the draft.

I would like to thank my family, especially my wife, for all the support in writing this book. I would not have finished this book without their love and support.

As a member of the Task Group that developed this standard, I worked with some great people. Reviewing and analyzing the technical input from these individuals was essential in helping me to write this book. I would like to thank all of the contributors to the 802.15.3 standard, particularly: Robert Heile (802.15 Chair), John R. Barr (802.15.3 Chair), Allen Heberling (MAC Committee Chair and MAC Assistant Editor), Richard Roberts (Systems Committee Chair and Layer Management Assistant Editor), Jay Bain (MAC Assistant Editor), Jeyhan Karaoguz (PHY Assistant Editor), John Sarallo (Layer Management Assistant Editor and MAC Contributing Editor), Ari Singer (Security Assistant Editor), Dan Bailey (Security Contributing Editor), Rajugopal Gubbi (MAC Contributing Editor), Knut Odman (MAC Contributing Editor), Mark Schrader (MAC Contributing Editor), and Bill Shvodian (MAC Contributing Editor).

I would also like to thank the anonymous peer reviewers of this book for their evaluation and suggestions.

Finally, I would like to thank my editor, Jennifer Longman, and the rest of the IEEE Standards Association team for all of their hard work and dedication.

INTEL-1010
U.S. Patent 8,457,672

Author



James P. K. Gilb received the Bachelor of Science degree in Electrical Engineering in 1987 from the Arizona State University, graduating magna cum laude. In 1989, he received the Master of Science degree in Electrical Engineering from the same institution and was named the Outstanding Graduate of the Graduate College. He received the Ph.D. degree in Electrical Engineering in 1999, also from Arizona State University. From 1993 to 1995, he worked as an Electrical Engineer at the Hexcel Corporation's Advanced Products Division, which was subsequently bought by the Northrop Grumman Corporation, developing advanced artificial electromagnetic materials, radar absorbing materials, and radar absorbing structures. He joined the Motorola Corporation in 1995, working initially for the Government Systems Technology Group as an RFIC designer and radio system designer. In 1999, he moved to the Semiconductor Products Sector as a Technical Staff Engineer (Member of Technical Staff) where he worked on a variety of radio systems. He developed radio architectures and specifications for new products and provided input for new process development. He joined the Mobilian Corporation in 2000, as a Senior Staff Engineer, where he developed the radio architecture and wrote the specification for the RF/analog chip that supported simultaneous operation of IEEE Std 802.11 and Bluetooth. He was also responsible for the detailed design and layout for the front-end RF circuits of the chip. He is currently the Director of Radio Engineering at Appairent Technologies where he is responsible for overseeing the implementation of the complete physical layer for IEEE Std 802.15.3. He has been the Technical Editor of the IEEE 802.15.3 Task Group since 2000 and was responsible for issuing all revisions of the draft standard. He has five patents issued and many papers published in refereed journals.

INTEL-1010
U.S. Patent 8,457,672

Table of Contents

Introduction xv

Acronyms and Abbreviations xvii

Chapter 1 Background and History 1
- What is an IEEE standard? 1
- The 802.15 family 2
- Why 802.15.3? 4
- History of 802.15.3 6

Chapter 2 802.15.3 applications 13
- The high-rate WPAN theme 13
- Still image applications 14
- Telephone quality audio applications 16
- High quality audio applications 17
- Gaming applications 18
- Video and multimedia applications 19

Chapter 3 Overview of the IEEE 802.15.3 standard 23
- Elements of the 802.15.3 piconet 25
- PHY overview 28
- Starting a piconet 31
- The superframe 32
- Joining and leaving a piconet 34
- Connecting with other devices 35
- Dependent piconets 36
- Obtaining information 39
- Power management 40
- System changes 43
- Implementation cost and complexity 44

[illegible] Library

INTEL-1010
U.S. Patent 8,457,672

Chapter 4 MAC functionality 47
MAC terminology in IEEE Std 802.15.3 47
Frame formats 49
Piconet timing and superframe structure 51
Interframe spacings 53
Contention access period (CAP) 55
Channel time allocation period (CTAP) 56
Comparing the contention access methods 60
Guard time 63
The role of the PNC 66
Starting a piconet 66
Handing over control 66
Ending a piconet 72
Joining and leaving the piconet 73
Association 74
Broadcasting piconet information 77
Disassociation 78
Assigning DEVIDs 80
Managing bandwidth 81
Acknowledgements 81
Asynchronous data 87
Stream connections 92
Fragmentation/defragmentation 96
Retransmissions and duplicate detection 99
Power management 100
Common characteristics of the SPS modes 104
Analyzing power save efficiencies 107
Switching PM modes 110
Managing SPS sets 114
DSPS mode 118
Allocating channel time for DSPS DEVs 119
PSPS mode 124
APS mode 126
Changing piconet parameters 128
Beacon announcements 129
Dynamic channel selection 132

INTEL-1010
U.S. Patent 8,457,672

Changing the PNID or BSID 134
Moving the beacon or changing the superframe duration 136
Finding information 138
Probe 139
Announce 143
PNC Information 145
Channel status 148
PNC channel scanning 150
Remote scan 152
Piconet services 154
Other capabilities 157
Transmit power control 157
Multirate capabilities 159
Extensibility of the standard 160
Example of the life cycle of a DEV 162

Chapter 5 Dependent piconets 165
Introduction 165
Starting a dependent piconet 168
Parent PNC ceasing operations with dependent piconets 174
Parent PNC stopping a dependent piconet 176
Handing over PNC responsibilities in a dependent piconet 177

Chapter 6 Security 187
Introduction and history 187
Security modes and policies 190
Security services provided in mode 1 191
Security policies 193
Symmetric key security suite 195
Overview of AES CCM 195
Key distribution 197
Security information 199

Chapter 7 2.4 GHz PHY 203
Overview 203
General PHY requirements 205

INTEL-1010
U.S. Patent 8,457,672

Channel plan 205
Timing issues 206
Miscellaneous PHY requirements 213
PHY frame format 213
Stuff bits and tail symbols 214
Frame format 215
PHY preamble 217
Data size restrictions 219
Modulation 220
Receiver performance 224
Transmitter performance 228
Regulatory and requirements 233
Delay spread performance 234
Mitigating the effects of delay spread 236
Fading channel model used for 802.15.3 237
Defining delay spread performance 239
Delay spread measurements 240
Radio architectures 244
Superheterodyne 245
Direct conversion 248
Walking IF 250
Low IF 253
Summary of radio architectures 256

Chapter 8 2.4 Interfacing to 802.15.3 257
The PIBs and their interface 261
MLME SAP 262
PLME SAP 265
MAC SAP 265
PHY SAP 266
The FCSL 268

Chapter 9 2.4 Coexistence mechanisms 271
Introduction 271
Coexistence techniques in 802.15.3 271
Passive scanning 273

INTEL-1010
U.S. Patent 8,457,672

The ability to request channel quality information 273
Dynamic channel selection 273
Link quality and RSSI 274
Channel plan that minimizes channel overlap 274
Transmit power control 275
Lower impact transmit spectral mask 275
Neighbor piconet capability 276
Coexistence results : 278
Assumptions for coexistence simulations 278
BER calculations 280
802.11b and 802.15.3 282
802.15.1 and 802.11 FHSS overlapping with 802.15.3 288

Summary 291

References 295

Glossary 299

Index 305

INTEL-1010
U.S. Patent 8,457,672

Introduction

This book is intended for people who want to learn more about wireless multimedia solutions based on IEEE Std 802.15.3™. This is a new IEEE standard designed to specifically address the needs of high-speed, low-cost portable multimedia devices. Because of this focus, the standard is very different from other IEEE 802® wireless standards. This book addresses four main areas of IEEE Std 802.15.3: history, applications, implementation, and justification.

The history of the 802.15.3 standard is described in Chapter 1. This chapter also includes some of the key points of the IEEE standards process, especially as it relates to the development of 802.15.3. The 802 Task Groups use a cryptic document naming system, which is explained in the chapter as well. This will make it easier for people who are new to IEEE standards to find information from the Task Group's web site.

The applications of IEEE Std 802.15.3 are discussed in Chapter 2. While a wide variety of applications are enabled with 802.15.3, the design of the protocol focused on two areas: high-throughput and low-latency. For some applications, a steady, high-throughput connection is all that is required. Other applications require a predictable, low latency, but do not require quite as much bandwidth. While the focus of 802.15.3 is on quality of service (QoS) support, it also handles asynchronous applications efficiently.

Chapter 3 provides a high-level overview of the standard. Following the overview, the medium access control (MAC) is described in detail in Chapters 4, 5, and 6. Chapter 4 covers the operation of the MAC and includes message sequence charts that illustrate the various operations of the protocol. Chapter 5 describes the operation of dependent piconets, which can be used to improve the coexistence of 802.15.3 piconets as well as coexistence with other wireless networks. Chapter 6 completes the discussion of the MAC functionality by providing a description of the optional security methods supported by the standard.

INTEL-1010
U.S. Patent 8,457,672

The physical layer (PHY) is discussed in detail in Chapter 7. In addition to a discussion of the characteristics of the PHY, this chapter provides additional information regarding the anticipated performance of the PHY in typical applications. At the higher data rates supported by 802.15.3, delay spread is an important factor in the operation of the PHY. Chapter 7 provides references to justify the delay-spread numbers used in the selection criteria for 802.15.3. The chapter finishes with a survey of some radio architectures that may be useful for the implementation of highly-integrated, low-cost radio frequency integrated circuits (RFICs) for 802.15.3.

Chapter 8 discusses the interfaces that were defined in IEEE Std 802.15.3. These interfaces support connections with higher layers in the protocol stack, e.g., IEEE Std 802.2™ and IEEE Std 1394™. Although the interfaces are optional to implement, they do provide insight into the operation of the protocol.

The final chapter in the book, Chapter 9, deals with the issue of coexistence. Because unlicensed allocations do not guarantee clear radio spectrum, wireless devices need to be able to handle the presence of interferers from a wide variety of sources. In the 2.4 GHz band, these include microwave ovens, IEEE Std 802.11b™ and IEEE Std 802.11g™ wireless local area networks (WLANs), IEEE Std 802.15.1™ personal area networks (PANs), etc. Simulation results for 802.15.3 coexisting with these types of interferers are presented in the chapter. In addition, the techniques provided by 802.15.3 to enhance the coexistence with these networks is also discussed in the chapter.

Throughout the book, additional information is given that provides justification for the choices made in the standard. The development process was iterative and so some ideas were introduced, only to be discarded later as further analysis showed problems with the proposed solutions. In some cases, functionality was cut from the standard, not because it was broken, but rather that its marginal utility was too low to justify the added complexity.

INTEL-1010
U.S. Patent 8,457,672

Acronyms and Abbreviations

The following acronyms and abbreviations are used in this book:

ADC	analog-to-digital converter
AES	advanced encryption standard
APS	asynchronous power save
ATP	association timeout period
BcstID	broadcast identifier
BER	bit error ratio
BPSK	biphase-shift keying
BSID	beacon source identifier
CAP	contention access period
CCA	clear channel assessment
CFA	call for applications
CSMA/CA	carrier sense multiple access/collision avoidance
CTA	channel time allocation
CTAP	channel time allocation period
CTRq	channel time request
CTRqB	channel time request block
CWB	continued wake beacon
DAC	digital-to-analog converter.
DBPSK	differential biphase shift keying
DES	data encryption standard
DEV	device
DEVID	device identifier
DestID	destination identifier
DSSS	direct-sequence spread spectrum
DSPS	device synchronized power save
DQPSK	differential quadrature phase-shift keying
D8PSK	differential eight phase-shift keying
EVM	error vector magnitude

INTEL-1010
U.S. Patent 8,457,672

FCS	frame check sequence
FER	frame error ratio
FSK	frequency shift keying
HCS	header check sequence
IE	information element
IF	intermediate frequency
ISI	intersymbol interference
LO	local oscillator
LQI	link quality indication
MAC	medium access control
MCTA	management channel time allocation
MSC	message sequence chart
MSDU	MAC service data unit
OFDM	orthogonal frequency division multiplexing
OID	object identifier
PAN	personal area network
PCTM	pending channel time map
PHY	physical layer
PIB	PAN information base
PNC	piconet controller
PNCID	piconet controller identifier
PNID	piconet identifier
ppm	parts per million
PS	power save
PSD	power spectral density
PSK	phase-shift keying
PSPS	piconet synchronized power save
PVR	personal video recorder
QAM	quadrature amplitude modulation
QPSK	quadrature phase-shift keying
RFIC	radio frequency integrated circuit
RSSI	receiver signal strength indication
SDL	specification description language

INTEL-1010
U.S. Patent 8,457,672

SNR	signal-to-noise ratio
SPS	synchronized power save
SrcID	source identifier
TCM	trellis coded modulation
TDMA	time division multiple access
UnassocID	unassociated ID
UWB	ultra-wideband
VGA	variable gain amplifier
VLIF	very low intermediate frequency
ZIF	zero intermediate frequency

INTEL-1010
U.S. Patent 8,457,672

Chapter 1 Background and History

This book provides an introduction and explanation of the IEEE 802.15.3™ wireless personal area networks (WPANs) standard. The goal of the book is to address three areas:

a) Clarify IEEE Std 802.15.3 for individuals who are implementing compliant devices;

b) Show how IEEE Std 802.15.3 can be used to develop wireless multimedia applications;

c) Provide insight into the reasons for the choices made in developing IEEE Std 802.15.3.

This chapter provides an introduction to IEEE standards, the IEEE 802.15™ family of standards, and the reasons why IEEE Std 802.15.3 was developed.

WHAT IS AN IEEE STANDARD?

IEEE Standard 802.15.3 is part of the family of standards developed by the Institute of Electrical and Electronic Engineers, Inc. (IEEE). In particular, it is a part of the Local Area Network/Metropolitan Area Network (LAN/MAN) standards activity of the IEEE Computer Society. These standards allow different manufacturers to create interoperable networking equipment, such as Ethernet (IEEE Std 802.3™), Token Ring (IEEE Std 802.5™), and Wi-Fi® (IEEE Std 802.11b™). Interoperability is a key requirement for networking equipment because its value comes from its ability to communicate with other equipment. Network equipment, by definition, is not useful by itself. At a minimum, two compatible devices are required to enable communications.



The IEEE develops a variety of standards in many different areas other than the LAN/MAN standards, which are sponsored by the IEEE's Computer Society.

Visit http://standards.ieee.org and http://grouper.ieee.org/groups/index.html for more information.

INTEL-1010
U.S. Patent 8,457,672

A standard helps the user by providing choices for purchasing hardware and keeps prices down by encouraging competition. Networking equipment that conforms to a standard also helps to ensure that the consumer will not be left with an orphaned solution. With proprietary solutions, a decision by a single manufacturer can kill a product and force consumers to change all of their equipment at a potentially large cost or incur costly service fees to maintain obsolete equipment. On the other hand, if there is a market for the equipment compliant to a standard, then it is possible for any manufacturer to provide the solution.

Manufacturers also benefit from the technical efforts that go into developing standards. The development of an IEEE standard involves thousands of hours of top-quality engineering time from many different sources in the proposal, development, and review of the standard. Although the competition inherent in the production of equipment compliant with a standard will keep prices low, this can actually help the manufacturer. While the per-unit pricing will generally be lower for a standardized product, the overall market will be correspondingly larger, providing greater profit potential for the manufacturer. Because a standard prevents proprietary lock-in, consumers tend to adopt these products more rapidly and so the total market often grows faster for a standards-compliant solution than for a proprietary solution.

The product design team benefits as well by developing a product that implements a standard. The standard development process includes an extensive peer review that tends to find almost all of the errors in the standard, so that the design engineer can focus on implementation rather than debugging the design of the protocol. A standard also allows suppliers to develop solutions that will meet the design engineer's needs.

THE 802.15 FAMILY

The IEEE 802 standards activities are divided into many different Working Groups (802.3, 802.11, 802.15, 802.16, etc.) that deal with a single type of networking standard. The 802.15 Working Group is tasked with developing standards and recommended practices for personal area networks (PANs). As of November 2003, within the 802.15 Working Group, there were five Task Groups that were either currently working on standards or had finished their work.

INTEL-1010
U.S. Patent 8,457,672



The SDL model in IEEE Std 802.15.1 does not correspond to the text in the standard or the Bluetooth 1.1 specification. The model was created for an older version of the specification to prove to the SIG that there were significant problems with the Bluetooth 1.0x specifications.

The first 802.15 project, IEEE Std 802.15.1™, was designed to provide a vehicle for creating a standard out of the Bluetooth® specification. The IEEE standards development process is an open one that allows input from any interested party. The Bluetooth specification, however, was developed by a small group of companies called the Bluetooth Special Interest Group (SIG), who retained control of the final specification. While much of the 802.15.1 standard is a word-for-word version of the Bluetooth specification, the 802.15.1 Task Group did provide additional material for the standard, including a specification definition language (SDL) model, an interface to the 802.2™ logical link control (LLC), and other text to help explain the specification. IEEE Std 802.15.1 was approved in April 2002 by the IEEE Standards Association (IEEE-SA), and the Task Group is no longer active.

Soon after the work on IEEE Std 802.15.1 began, the 802.15 and 802.11 Working Groups became concerned about the ability of 802.11b and 802.15.1 networks to coexist when they were operating in close proximity. The 802.15 Working Group then formed a new Task Group, 802.15.2, to develop a Recommended Practice that would improve the ability of the two wireless networks to coexist. A Recommended Practice is like a standard, except that it contains suggestions rather than requirements for interoperability. The suggestions represent "best practices" that can be used by implementers to improve their products. The IEEE 802.15.2 Recommended Practice was approved by both the 802.11 and 802.15 Working Groups and in June of 2003, the IEEE Std 802.15.2 was approved IEEE-SA.

The 802.15.3 Task Group was formed to create a standard that would enable wireless connectivity of high-speed, multimedia-capable portable consumer devices. The goals were to get greater than 20 Mb/s data rate while maintaining quality of service (QoS) for the data streams, low-power, and low-cost. IEEE Std 802.15.3, which is the subject of this book, was approved by the IEEE-SA in June 2003.

The 802.15.4 Task Group also took on the goal of very low-cost and low-power wireless networks. This group was tasked with developing a standard for low data

INTEL-1010
U.S. Patent 8,457,672

rate devices (<250 kb/s) at a cost that was less than either IEEE Std 802.15.3 or IEEE Std 802.15.1. The 802.15.4 standard was approved by the IEEE-SA in May 2003.



New Study Groups and Task Groups are formed in the 802.15 Working Group from time to time. Refer to the 802.15 web page (http://grouper.ieee.org/802/15/index.html) for the latest information.

The final group within 802.15 is the 802.15.3a™ Task Group. Their goal is to develop a new physcial layer (PHY) that will go with the 802.15.3 medium access control (MAC) to enable data rates of greater than 110 Mb/s to more than 400 Mb/s. The Task Group has completed the call for applications (CFA) and the call for proposals (CFP) and has begun the downselection process. Once a single proposal has been selected and confirmed by a 75% vote of the Task Group and Working Group, the process of writing the draft standard will begin.

WHY 802.15.3?

IEEE Std 802.15.3 was specifically written to address a class of applications that did not have a wireless standard. Some of the wireless applications that were submitted in response to the call for applications for IEEE Std 802.15.3 were as follows:

- Connecting digital still cameras to printers or kiosks;
- Laptop to projector connection;
- Connecting a personal digital assistant (PDA) to a camera or a PDA to a printer;
- Speakers in a 5.1 surround-sound system connecting to the receiver;
- Video distribution from a set-top box or cable modem;
- Sending music from a CD or MP3 player to headphones or speakers;
- Video camera display on a television;
- Remote viewfinders for video or digital still cameras.

INTEL-1010
U.S. Patent 8,457,672

These applications are mainly in the consumer electronics arena and have the following characteristics:

- *High throughput requirement:* greater than 20 Mb/s to support video and/or multi-channel audio;
- *Low power usage:* to be useful in battery-powered portable devices;
- *Low cost:* to add value to consumer electronic devices;
- *Guaranteed bandwidth:* to provide QoS for applications sensitive to latency;
- *Ad hoc connectivity:* to provide the benefits of networking without unduly restricting portability or mobility;
- *Simple connectivity:* to make networking easy and eliminate the need for a technically sophisticated user;
- *Independent connectivity:* because there is value in networking devices together even without a connection to the Internet;
- *An assurance of privacy:* to assure the user that only the intended recipients can understand what is being transmitted.

Because the existing wireless standard, 802.11™, was created to address the delivery of the bursty, asynchronous traffic that is common in a LAN, it did not address the special requirements of these applications. Although 802.11 provides 54 Mb/s of instantaneous data rate with either 802.11a™ and 802.11g™, under ideal conditions the actual throughput is less than 31 Mb/s for 802.11a and is less than 30 Mb/s for 802.11g [B32]. The throughput of 802.11g drops to less than 18 Mb/s if there are any 802.11b stations in the wireless local area network (WLAN) [B32]. In addition, the 802.11 MAC does not allow for the consistent reservation of channel time that is a requirement for applications that need QoS. Networks that use 802.11 are mainly set up by professionals in businesses, hotels, airports, coffee shops, and so on, where a networking professional is able to handle the complexities of the network. Although IEEE Std 802.11 supports ad-hoc networking, this mode is difficult to set up and it does not support power-save modes for the devices in the network. Users generally rely on the infrastructure mode, which does not allow direct peer-to-peer connections or ad-hoc networks and requires a dedicated access point (AP).

INTEL-1010
U.S. Patent 8,457,672

This is not to say that IEEE Std 802.11 is not a good protocol. It is an excellent solution for data traffic that is typical on WLANs. It allows stations to quickly get access to the medium to send data without going through a channel time allocation process. It has routing and roaming capabilities that enable users to be mobile while maintaining a connection to the infrastructure. The 802.11 security model for centralized authentication is consistent with its use in extending the wired infrastructure to wireless stations. Network adapters for 802.11b have fallen dramatically in price as well and they are now cheaper than Bluetooth network adapters while providing 11 times the raw data rate[1].

Although IEEE Std 802.11 is able to solve some of the consumer electronics applications under certain configurations and traffic models, it does not solve the broad range of connectivity and QoS requirements for a growing class of existing and emerging portable, multimedia consumer electronic devices. Creating a solution for these applications required more than just a new PHY, it also required a completely new MAC to create a protocol that would be optimized for the consumer electronics space.

HISTORY OF 802.15.3

Multiple groups have worked on the concept of high-speed, wireless connectivity for portable, multimedia devices. The HomeRF[2] Working Group began work on a high-speed multimedia-capable specification in 1999, called HomeRF Multi-Media (or HRF-MM), that was aimed at >20 Mb/s data rates. The HomeRF multimedia specification was never released, however, the group did use a new Federal Communications Commission (FCC) report and order that allowed wideband frequency hopping in the 2.4 GHz band to extend the data rates of HomeRF 1.1 from 2 Mb/s to 10 Mb/s.

In early 2000, the Bluetooth SIG also began work on a higher speed extension for Bluetooth 1.0, which was called Radio 2. The initial goal was to achieve a data rate of least 10 Mb/s while maintaining backward compatibility with Bluetooth 1.x implementations. Because of the closed development process used by the Bluetooth SIG, little information is publicly available about the status of this

[1] Products that use 802.11g are also now available, which offer 28 times the throughput of Bluetooth and are as cheap or cheaper than Bluetooth network adapters.

[2] The HomeRF Working Group disbanded in January of 2003.

INTEL-1010
U.S. Patent 8,457,672

effort. By the middle of 2003, the Bluetooth SIG had announced that a smaller increase in the data rate, from 1 Mb/s to 2 or 3 Mb/s, would be available first with a 12 Mb/s enhancement that would be available later. It is not clear, however, how Bluetooth Radio 2 devices will be able to compete with existing low-cost WLANs[3]. In addition, it has been reported that the high-rate Bluetooth Specification will require an entirely different MAC and PHY from what is currently used in Bluetooth 1.x implementations [B31]. If this difference remains in the final revision of the Radio 2 specification, then it is unlikely that a combination Bluetooth 1.x and Radio 2 implementation would be less expensive than a combination Bluetooth1.x and 802.15.3 solution or even a combination Bluetooth 1.x and 802.11g solution.

In late 1999, Kodak™, Motorola®, and Cisco® approached the IEEE Std 802.15 Working Group to request a new study group to look at wireless MAC and PHY that would support portable, multimedia-capable devices. Unlike the HomeRF and Bluetooth efforts, the IEEE uses an open process that allows anyone to present a proposal for the development of a standard. The Study Group became a Task Group at the March 2000 meeting in Albuquerque, NM, USA. At that meeting, there were already responses to the call for applications. These applications were used to develop a criteria document that set out the requirements for the new standard in three main areas: system, MAC, and PHY. Once the criteria document was finished, there was a call for proposals that resulted in four MAC and nine PHY proposals. The topics of each of the four MAC proposals are listed in Table 1–1.

Table 1–1: IEEE Std 802.15.3 MAC proposals

Description	Document number
Modified HiperLAN/2	00/196r6
802.11 "lite" with time slot additions	02/205r2 and 00/218r1
Bluetooth-like with extensions	00/212r5 and 00/213r0
802.11 "lite"	00/197r2

[3] EE Times reported that "Still, skeptics wonder why anybody needs a high-rate Bluetooth when consumers can safely turn to Wi-Fi for faster wireless file transfer or data streaming." [B31]

INTEL-1010
U.S. Patent 8,457,672

A note about the document numbering used for 802.15.3 is in order. The 802.15 Working Group has had two different naming schemes for documents. The changeover in the document naming scheme occured during the September 2003 meeting. Documents prior to 2003 and those in 2003 with sequence numbers less than 300 use the older numbering scheme. Documents that have been submitted to the 802.15 Working Group are all publicly available on the 802.15 website, http://grouper.ieee.org/groups/802/15/index.html, in the "Doc archives" link.

A typical document name in the older document naming method is "00196r6P802-15_TG3-HiPeRPAN-merged-proposal.pdf." The first two digits are for the year followed by a three-digit number assigned sequentially by the Chair of the Working Group followed by the letter "r" and the revision number. All of the 802.15.3 documents contain "P802-15_TG3-" followed by the title. The "TG3" indicates Task Group 3, which is 802.15.3. Thus, the example document name indicates that it is the sixth revision of 196th document number assigned by the Chair of 802.15 in the year 2000 for Task Group 3 entitled "HiPeRPAN merged proposal" in pdf format. The parts of an 802.15 document name are illustrated in Figure 1–1.

The document numbering convention for each of the 802 Working Groups is different. The best place to find this information is in the Operating Rules of the Working Group. The Working Group rules, as well as the document archive for the Working Group can be found on their web pages. Links to all 802 working groups can be found on http://ieee802.org/dots.html

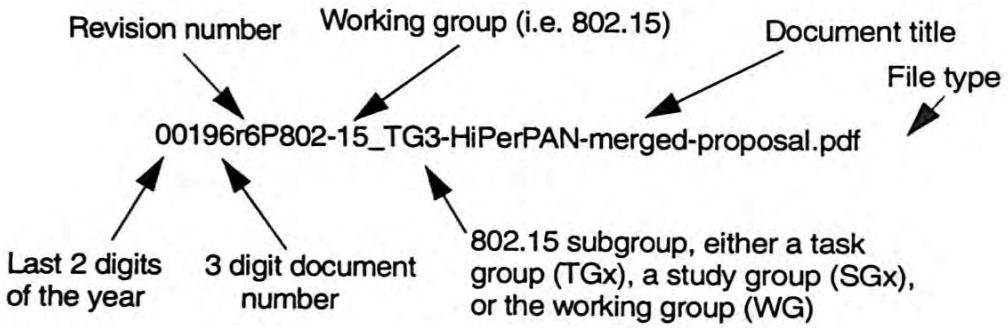


Figure 1–1: Old (prior to September 2003) IEEE 802.15 document naming conventions

INTEL-1010
U.S. Patent 8,457,672

A typical document using the new document naming method is "15-03-0516-01-0030-Draft-editorial-and-enhancements.doc." The string of 14 digits is called the Document Control Number (DCN). The first two digits indicate the Working Group (WG 15) while the next two digits indicate the year that the document was submitted, i.e., 2003. The next four digits are a number assigned sequentially by the document management software to uniquely identify the document. This is followed by a two digit revision number and a four character Task Group identifier followed by the title.[4] Thus the example document name indicates that it is the first revision of the 516th document submitted in 2003 for Task Group 3 entitled, "Draft editorial and enhancements" in doc format. The parts of a new 802.15 document name are illustrated in Figure 1–2.

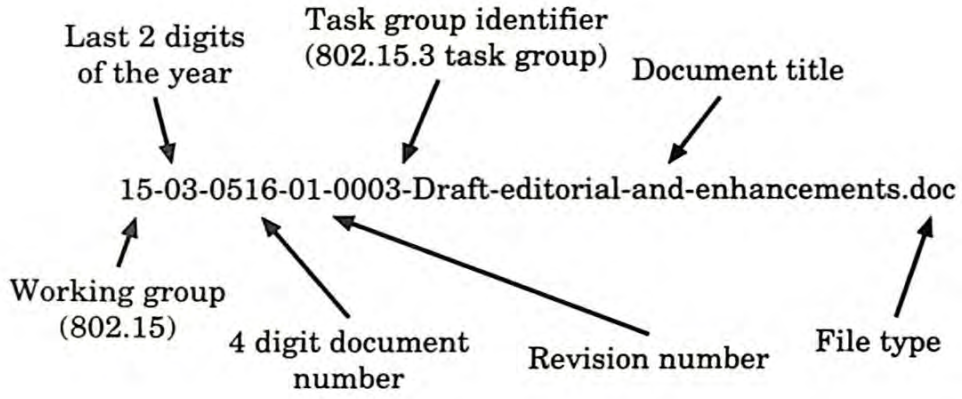


Figure 1–2: Current IEEE 802.15 document naming conventions

In the down-selection process, one of the proposals (00/212r5, 00/213r0) was withdrawn and the other three agreed to a compromise outline of the protocol that used a beacon followed by a contention access period (CAP) followed by assigned time slots. This merged proposal received 100% approval from the Task Group. The PHY selection, however, was much more contentious. The nine PHY proposals are summarized in Table 1–2. The acronyms used in the table are defined on page xvii.

[4] The Task Group identifier can be both digits and characters, e.g., "003a" is Task Group 3a while "mmwi" is the Millimeter Wave interest group.

INTEL-1010
U.S. Patent 8,457,672

Table 1–2: IEEE Std 802.15.3 PHY Proposals

Frequency	Modulation	Coding	Document numbers
5 GHz	4 level FSK	None	00/206r1
5 GHz	OFDM with BPSK, QPSK, 16-QAM, DQPSK and D8PSK	Turbo	00/196r6
2.5 GHz	2 level FSK, QPSK,	None	00/214r7, 00/215r0
2.5 GHz	M-ary Bi-Code Keying with QPSK and 16-QAM	Reed-Solomon	00/210r10
2.5 GHz	QPSK, 16-QAM	Turbo	00/199r2, 00/200r9
UWB	Bi-phase with DSSS	None	00/195r8
2.5 GHz	8PSK, QPSK, 16, 32, 64-QAM	TCM	00/211r5
2.5 GHz	DBPSK, DQPSK, 8-QAM, 16-QAM	Reed-Solomon	00/197r2, 00/198r3

The down-selection vote was held at the IEEE 802 plenary meeting in Tampa, FL, USA, in November 2000. At that point, two of the proposals were withdrawn (00/206r1 and 00/195r8) and the final proposal selected was a compromise of two proposals, (00/214r7, 00/215r0, 00/199r2, and 00/200r9). This final version specified the modulation but left the coding to be determined at the January 2001 meeting in Monterrey, CA, USA. At that meeting, trellis coded modulation (TCM) was selected as the coding with the base rate left as uncoded to allow simple, low-power, low-cost devices. This completed the PHY selection process and enabled the group to continue with the task of drafting the standard.

The initial drafting process proceeded informally with different authors given clauses to write and the intermediate results were posted for comment. In the November 2001 meeting in Austin, TX, USA, the group closed the last outstanding issue on the draft and was able to begin the draft approval process.

The draft approval process involves two stages of balloting: Working Group letter ballot and Sponsor letter ballot. The first balloting process seeks the approval of the entire Working Group: the 802.15 Working Group in this case. If the Working Group approves of the draft, it is forwarded to Sponsor ballot (conducted by the IEEE-SA), where interested individuals review and vote on the contents of the

INTEL-1010
U.S. Patent 8,457,672

draft standard. The IEEE 802.15.3 draft standard began the Working Group letter ballot process in early December 2001 with a 40-day letter ballot. In the letter ballot, the voters are asked to vote yes, no, or abstain and to provide comments on things that they wanted changed in the draft. In the case of a no vote, the voter is required to provide comments that indicate the changes that would be required in order to change the person's no vote to a yes vote. The minimum approval for the ballot is 75%, but traditionally, the 802 Working Groups seek to get greater than 90% approval.

The first letter ballot failed to get 75% approval[5], but with changes to add security suites, coexistence analysis, and improved power management text, a sufficient number of no voters changed their votes to yes to enable a recirculation ballot. The draft went through three recirculation ballots, each time achieving greater than 75% approval before being approved to be sent to Sponsor ballot following the November 2002 802 plenary meeting in Kauai, HI, USA. A summary of the voting and comments for the Working Group letter ballots is given in Table 1–3.

The draft began the Sponsor ballot process in December of 2002. The first Sponsor ballot received 76% approval[6], just making the 75% requirement for passing. In response to the comments received, the Task Group made some changes to the draft that satisfied the majority of the no voters. The mandatory status of the device synchronized power save (DSPS) power-save mode had been a contentious issue since the first Working Group letter ballot. Based on a compromise that was agreed to in Kauai, the group was finally able to close out this issue.



The IEEE SA has two numerical requirements for a ballot to be valid. First, greater than 75% of those eligible to vote must return ballots. Second, the number of abstain votes must be less than 30% of the total votes received. If the ballot is valid, it passes if 75% of the people vote yes of the total number of people voting yes or no (abstains do not count in this tally).

In addition, the Task Group was informed that the authentication protocols that were in the draft were likely outside of the scope of a MAC and PHY standard. This could have caused problems with the approval of the standard with either the

[5] 75 of the 89 eligible voters in the 802.15.3 Working Group voted in the first letter ballot.

[6] There were 101 people in the balloting pool and in the first ballot, 79 people returned ballots in time (> 75% return ratio). There were 56 affirmative votes, 17 negative votes, and 6 abstentions (abstains < 30%). There were 447 technical comments and 379 editorial comments.

INTEL-1010
U.S. Patent 8,457,672

802 Executive Committee (802 ExCom) or the IEEE Standards Review Committee (RevCom). In order to avoid any problems and to ensure that the standard could be quickly approved, the Task Group removed the three authentication methods from the draft and attempted to purge the document of any references to authentication.

The revised (and much shorter) draft was recirculated to the Sponsor balloting group. This ballot came in with better than 96% approval[7] and the comments, much fewer this time, dealt mostly with finishing the removal of the security suites. After implementing changes to address these issues, the draft was recirculated a final time. In this final round of Sponsor balloting, the draft received 97% approval[8] and no new comments were received. The Working Group then forwarded the draft to the 802 ExCom where it was approved and then sent on to the IEEE RevCom for consideration. RevCom approved the draft in their June 2003 meeting and the IEEE Standards Board affirmed their decision the next day, June 12, 2003, making 802.15.3 an official standard.

Table 1–3: 802.15.3 Working Group letter ballot results

Ballot #	Vote results			Comments		
	Yes	No	Abstain	Technical Required	Technical	Editorial
LB12	49 (69%)	22 (31%)	4	554	612	685
LB17	64 (88%)	9 (12%)	2	444	131	622
LB19	66 (89%)	8 (11%)	1	326	72	153
LB22	70 (95%)	4 (5%)	1	62	24	159

[7] The first sponsor ballot recirculation result was 73 approve, 3 disapprove, and 6 abstain with 57 technical comments and 117 editorial comments.

[8] The final sponsor ballot recirculation result was 76 approve, 2 disapprove, and 6 abstain with no comments received.

INTEL-1010
U.S. Patent 8,457,672

Chapter 2 802.15.3 applications

IEEE Std 802.15.3, like all IEEE standards, addresses a set of applications. In the process of developing the standard, a call for applications (CFA) is sent out to solicit responses from manufacturers and users to indicate the types of applications for which they need a solution. Although not every application can be addressed with a single standard, the task group tried to enable as many applications as possible identified in the CFA while aiming for low cost, complexity, and power.

THE HIGH-RATE WPAN THEME

There are several wireless standards that can be applied to some of the following applications, however, only 802.15.3 is able to address all of the requirements for portable, wireless multimedia devices. In addition, it is important to understand the underlying philosophy of 802.15.3 in order to understand what the excitement is about. The general applications of local wireless solutions for consumer electronic devices can be divided into data networks, such as those used for enterprise WLANs or desktop internet applications, and multimedia networks, like those used for entertainment and portable devices. These networks have different purposes and requirements. After the CFA, it became clear that multimedia networks need to include a large class of devices in order to be a complete system. For example, a television set is not just a wireless video distribution opportunity. It had its own class of peripherals such as remote controls, 6.1 speakers, keyboards for on-screen Internet access, input from personal video cameras, and still cameras. In order to be useful in the home or on vacation, the needs of all these devices had to be addressed in a single standardized solution. There had to be a full systems solution, or the customer would not be satisfied.

IEEE Std 802.15.3 was a "ground up" opportunity to develop a new standard that served a large, untapped market, not accessible to Bluetooth, 802.11, and other approaches, and to serve as a platform for future enhancements.

INTEL-1010
U.S. Patent 8,457,672

STILL IMAGE APPLICATIONS

One of the original applications that drove the development of IEEE Std 802.15.3 was still image transfer from a digital camera to a photo kiosk. The increasing size of images in digital cameras combined with the impatience of users created a need for low-cost, high data-rate, ad-hoc wireless connection. Based on a 15-second transaction time limit for the user's attention span, the data rate needed to be at least 20 Mb/s. The wireless photo kiosk is intended to allow owners of digital cameras to create prints, photo CDs, store the picture at a central site, or even email the picture to selected individuals with a simple button push. Digital cameras are now getting the capability of adding information to the picture, when it is taken, that indicates what the user wants to do with the picture, e.g., print to paper, size of the prints, number of copies, where to store the data file, and so on.

One of the advantages of wireless image transfer is that it allows people to "unload" their digital cameras into permanent storage, prints, photo CDs, or a central Internet location, e.g., www.ofoto.com. This is especially important for people when they are are traveling on vacation. The typical digital camera user saves their pictures on their home computer so that they can erase the files from the camera or memory card. However, most people do not have laptops and even fewer take their laptops or computers on vacation, which is where most of the pictures will be taken. Thus, people are not able to use their digital cameras as much as they would use a standard film camera.

The photo kiosk application offers value to everyone involved[9]. The store or facility that houses the kiosks gets rental money from the kiosk owner as well as additional business and traffic from people that use the machine. The kiosk owner is able to collect money for printing pictures or making CDs. If the pictures are sent to an online site for storage, the kiosk owner may collect a transmittal fee and the website owner may also get money for storing the picture. The person using the camera gets to empty their flash card and receive a copy of the pictures (electronic, paper, or both). Because the camera also acts as a display device, it can be used for uploading content, e.g., music, short videos, ads, or even other pictures.

[9] Currently there are approximately 50,000 photo kiosks and they are expanding rapidly.

INTEL-1010
U.S. Patent 8,457,672

The still image transfer capabilities are also a compelling application for the home. Less than 50% of U.S. households have a computer, and yet virtually every household has a film camera[10]. In a typical house with a computer, it is located in a back room in an out of the way place and the computer is not normally turned on. When the user wants to transfer images or print them, they must go to the computer, turn it on, wait for it to boot, and then they can begin to transfer images. In order to expand the market for digital cameras and related services, there needs to be a way to use the camera without requiring the intervention of the PC.

The simple way to do this is to wirelessly enable both the printer and the camera. However, the images that the camera prints can be large and the user is not going to want to wait for a slow data link to transfer the pictures. With an IEEE Std 802.15.3 wireless link, the user can easily send the pictures to the printer. The market has already provided ink jet printers that have slots for the various types of digital camera storage, e.g., compact flash (CF), secure digital (SD) cards, multimedia cards (MMC), and Memory Stick®. As an extra cost option, they also can have a small display for viewing the pictures on the card. However, with a wireless link, the camera can be used as a display and control device for the printer. In addition, manufacturers can offer a less expensive printer because they do not have to add the expense of supporting multiple slots and formats.

Another application of the still image transfer would be the remote display of images on various devices. For example, the user could create a slide show of images and display them on an 802.15.3-enabled television. Because televisions and even high-definition televisions (HDTVs) are relatively low resolution compared with digital pictures, even a 1 megapixel camera would create an image that would display well on an HDTV. This would require application level support on the television for JPEG pictures. Users could also upload pictures to digital picture frames without having to take the frame off the wall to plug in the cable.

In summary, some of the applications that would use still image or bulk file transfer are:

- Printing pictures to a photo kiosk
- Uploading content from a kiosk (movies, ads, still photos, music, etc.)

[10] In 2003 there were approximately 800 million film cameras and 37 million digital cameras.

INTEL-1010
U.S. Patent 8,457,672

- Remote display of still images on televisions or picture frames
- Direct camera to ink jet printing in the home
- Serve as display and control device for the printer

TELEPHONE QUALITY AUDIO APPLICATIONS

In this context, audio applications are those that transmit voice between two parties in a manner similar to the telephone system. Although the data rates for this type of communications are not particularly high, about 9.6 to 64 kb/s depending on the quality required, these applications have the most stringent latency requirements. The reason for this is that although people can deal with delayed video, as long as the associated audio remains in sync, they are not used to long pauses in conversation. In most cultures, a long (>0.5 s) latency in receiving audio traffic from the remote person will cause the other individual to begin talking, often on top of the remote person's response.

The quality required for audio transmission, the latency, dynamic range, bit rates required, and so on, have been extensively studied by the telephone and cellular companies. Because of this research and the experience that people have with telephones today, the requirements for audio applications are well-documented. In many places, people are used to very high-quality voice connections, which sets a high expectation for any protocol/radio combination that seeks to replace a wired solution. Part of the design of IEEE Std 802.15.3 was aimed at solving the audio problem. The 802.2 priority mapping (see Annex A in IEEE Std 802.15.3) specifies less than 10 ms of delay and jitter for voice applications. This number is relatively low because it is expected that there will be other delays and jitter from the wireline portion of the connection. If the connection is only between wireless devices (DEVs) in the piconet, e.g., an intercom system, then the latency could be expanded to greater than 80 ms.

The latency for voice and any retransmissions, if needed, is controlled by the interval requested in the channel time request. For example, if the superframe duration is 10 ms, then the DEV could request allocations every fourth superframe and suffer no more than 40 ms latency. The maximum allowed superframe duration is 65 ms, which puts an upper limit on the latency for a single allocation in a superframe. If the DEV wanted to have a lower latency, it could request that

INTEL-1010
U.S. Patent 8,457,672

its channel time be split into two allocations spread out in the same superframe, i.e., a super-rate allocation.

In summary, some of the applications that would use telephone quality audio are:

- Cordless telephones
- Intercoms

HIGH QUALITY AUDIO APPLICATIONS

The high-quality audio market is distinct from telephone-quality audio. In the case of telephony, the end-to-end latency is very important. A typical high-quality audio application, on the other hand, is relatively insensitive to latency over the air as long as there is sufficient buffering and bandwidth to maintain the data flow. For example, many modern mobile CD players read ahead on the CD and buffer the information to prevent skips in the playback when movement of the player causes the read process to skip. Although some buffering is allowed in this application, users still expect the sound to start soon after the "start" button is pressed and for it to continue uninterrupted for as long as they want to listen to it. In addition, the end points of an audio system, i.e., the speakers and microphones, are relatively low-cost items and so the buffer space will be constrained.

High quality audio also requires a much greater data rate than simple telephone quality audio. While wireline quality voice connections run at 64 kb/s[11], CD quality audio requires (2*44.1 kHz* 8 bits/sample) = 705.6 kb/s if it is sent uncompressed. A 5.1 surround sound system (e.g., 6 channels) would require three times the data rate, a little more than 2.1 Mb/s, to send the data uncompressed. While lossy compression methods, e.g., MP3 or Ogg Vorbis, are acceptable for portable digital audio applications, home and professional sound systems require uncompressed data to maintain greater than 90 dB dynamic range.

One of the challenges in wireless distribution of sound, especially in a 5.1 surround sound system, is to keep the playback synchronized. While the 802.15.3 standard does not specify a method for synchronizing different applications, the beacon provides a time frame of reference for all of the DEVs in the piconet.

[11] Digital cellular telephones use compression for voice transmission and generally use less than 19 kb/s, some as low as 9.6 kb/s.

INTEL-1010
U.S. Patent 8,457,672

This allows all of the DEVs to use a common timing frame of reference for playback. In addition, the standard provides sufficient flexibility for an application to create a method for synchronizing playback among a variety of DEVs.

In summary, some of the high-quality audio applications that are supported by 802.15.3 include:



The data rates for typical high-quality audio applications require from 1.6 to 2.6 Mb/s per audio channel. The data delivery needs to be consistent, so a "best effort" delivery system will fail to provide users with the level of quality to which they are accustomed. These data rates need to be supported over relatively long distances, typically over an entire home. These requirements, high data rate, QoS and long range, are only met by the 802.15.3 standard.

- Digital audio players (e.g., Ogg Vorbis, MP3, etc.)
- Headphones
- Speakers, especially 5.1 surround sound systems
- Wireless microphones

GAMING APPLICATIONS

A low-cost, high-speed wireless network with guaranteed time allocations opens up new types of applications for gaming. In terms of the number of devices, the home console and portable video game market is larger than the home computer market. The most obvious application of wireless connectivity in video games is to remove the wires that connect the controllers to the game console. Although the amount of information that is transferred to and from the controller is small, the users are very sensitive to latency. A controller that fails to get its command through in a timely manner will cause the user to lose the game and will be quickly returned to the store. IEEE Std 802.15.3 allows the application to control the latency by selecting the allocation interval for the data. The data transfers are bidirectional as well because game controllers often implement physical feedback mechanisms to enhance the gaming experience.

The controllers for the game console are only a small part of the total gaming market. Once the game console has become wirelessly enabled with 802.15.3, other applications are available. For example, with multiple game controllers in a relatively small area, the users could engage in a multiplayer game. Because multiple consoles are used, the players could each have a different viewpoint of

INTEL-1010
U.S. Patent 8,457,672

the action in their display. Game consoles now have the ability to access the Internet. In this case, the console could act as the wireless router for the house, distributing information to wireless portable devices such as web pads. The game console could also be used to play music that is sent wirelessly to speakers or headphones. At some point, the game console may take on the function of the home computer for many households.

Another application would be to enable handheld games to interact with each other wirelessly. Multiplayer games that could be formed whenever a group of people with handheld gaming devices was in range. The gaming devices are taking on new roles as well, such as providing playback of digital music. Having a wireless connection in the handheld game would allow the user to upload music, videos or images from kiosks, computers, or the game console. Because the handheld game has a screen as well as buttons, it can serve as the interface for this transaction. Not only that, but new applications will be found for the device using the display and controls to provide input to and feedback from another 802.15.3 device that, for cost reasons, does not have a user interface.

In summary, some of the possible gaming applications are:

- Wireless controllers
- Multiplayer gaming with home console
- Game console as Internet portal
- Handheld multiplayer gaming
- Digital music, video, or image upload to handheld games

VIDEO AND MULTIMEDIA APPLICATIONS

Multimedia applications have the highest requirements for throughput, but they often have more relaxed latency requirements. Streaming multimedia delivery over the Internet is accomplished today with buffering. This model works well when the endpoint is a computer that has a large hard drive and RAM. However, many consumer devices do not have large amounts of RAM or non-volatile storage available. Thus, while the latency requirement is relaxed for streaming multimedia applications, there is still an upper bound on the latency that is permissible for these applications.

INTEL-1010
U.S. Patent 8,457,672

The most commonly cited streaming multimedia application is the set-top box application. Although satellite and cable television are normally delivered to a single box in a house, the typical home has more than one television. The carriers (e.g., the cable or satellite service) would like to enable people to buy more than one pay-per-view service at a time, but this requires distributing the signal to more than one television at the same time. Because most homes are not wired for cable in all of the rooms where the televisions are located, the providers would like to enable the user to self-configure the network without incurring the expense of running cables. In addition, the optimal place for bringing in the signal, either from the curb for cable or from a wall or roof-mount antenna, is often not where the users want to watch television. The content providers incur high costs in running cables for the customers; yet, they have to do this or the customer will not buy the service. The concept of a "self-install" that does not require running cables has made the carriers very interested in wireless video distribution systems.

Even in the case of the home theater system, connecting the cables correctly to all of the components of the system can be quite daunting. Wireless connectivity with service discovery would allow people to bring in a new component and have it automatically find all of the other components and create the required connections. The user would only have to plug in power for the new component, and all of the components could self-configure the connections. For the typical consumer, this is a great advantage from the point of view of ease of use. Not only that, but it also will allow the user to position the components wherever they want.

Streaming multimedia is not the only type of application that 802.15.3 enables. Two-way multimedia applications, e.g., where the person interacts with the application, are also a good application for 802.15.3's QoS capability. If the application includes two-way voice interaction, then the application needs both the low latency for voice as well as the high bandwidth for video. Some examples of this kind of application are video conferencing, video intercoms, and gaming. However, it may be possible to separate the delivery of voice from video and to prioritize the voice delivery. People tend to be more accepting of glitches in the video presentation than in the audio portion. In this case, each DEV would allocate two streams, one for video and one for audio, to the other DEV. If the application supported scheduling audio and video together in a single stream, then the DEVs would require only one time allocation in the superframe.

INTEL-1010
U.S. Patent 8,457,672

Another application is a detachable display for a video camera. Sharp® demonstrated this application using a proprietary radio from their research departments that carried data at 10 Mb/s. An 802.15.3 radio would give greater data rate and lower cost because it was based on a standard. This application allows the user to put a camera on a tripod and take the display to a remote location. Because 802.15.3 enables two-way communications with low latency, the user would be able to control the camera from remote as well as be able to see the images that the camera was recording. Digital still cameras would also benefit by having a remote display with shutter release and focus control.

Sony™ has pioneered the concept of robots as entertainment devices, and this idea is beginning to take hold at other companies as well. One of the enhancements that has been suggested is remote control of the robot with a remote display. The user would be able to give voice commands to the robot and see what the robot is doing. Another application for remotely viewing with a camera would be for remote control vehicles like model cars and possibly model airplanes (although the range would have to be extended). This would allow the person controlling the vehicle to change their viewpoint to be from the inside of the vehicle looking out. It would also allow the vehicle to go behind visual obstructions while remaining under the control of the user or even to integrate the vehicle in a computer game.

Digital and analog camcorders are an important application for wireless connectivity. Although it is possible to view the video on the camera, the displays are usually small (less than 4 inches) because the camera needs to be small as well. Currently, users need to find the video cable for the camera and attach it to a television to show the video. Although some televisions now have connectors in front, not all do, and the cables can get lost. With a wireless link, users would be able to easily display the contents of their camcorder on any display device, e.g., television or computer monitor. Not only that, they could make copies of the video directly to tape or DVD without having to connect the cables to the back of the VCR or DVD recorder. If users had a personal video recorder (PVR), they would be able to use the camera for control, the television for display, and the PVR as an editing device. Once the edited video was complete, the PVR could record it to tape or DVD.

INTEL-1010
U.S. Patent 8,457,672

In summary, some of the possible multimedia applications are:

- Video distribution from set top boxes to televisions
- Wireless connection of home theater and stereo components
- Wireless video conferencing, video intercoms, and interactive gaming
- Wireless displays for entertainment applications like robots and remote-controlled vehicles
- Remote displays and controls for video and digital still cameras
- Displaying video from camcorders to televisions
- Storing video from camcorders to VCRs and DVD recorders
- Editing videos using a camcorder and PVR

INTEL-1010
U.S. Patent 8,457,672

Chapter 3 Overview of the IEEE 802.15.3 standard

Networks are an essential part of computers today and have become a key component to many of the applications that run on them. These networks may serve different geographic areas and subsequently have different implementations. The areas served by these networks can be divided into four types:

a) Wide area networks (WANs): A country or even the entire world

b) Metropolitan area networks (MANs): A city or similar area

c) Local area networks (LANs): A single building or campus

d) Personal area networks (PANs): An area around a person.

The last type of network, the PAN, is a relatively new area of development. In a PAN, the group of connected devices is not referred to as a network, but rather as a piconet, to imply that it is a small group of networked devices. PANs are designed to connect the many electronic devices that a single person might own to enable data sharing, connectivity, and new applications that are still developing. Because of this, PAN devices are used in a way that is very different from other networks.

Early definitions of a PAN, including the one used in the IEEE Std 802.15.3, referred to it as being limited in distance[12]. However, the results of the call for applications (CFA) included multiple applications that required relatively long distances, e.g., coverage of an entire house. Instead of range, one of the key characteristics of a PAN that differentiates it from a LAN is that a PAN normally consists of DEVs that are under the control, in some fashion, of a single user. A LAN, on the other hand, normally consists of devices that belong to multiple users. Thus the connectivity focus of a PAN tends to be inward, whereas a LAN's connectivity focus is outside of the LAN. The typical characteristics of a PAN are as follows:

[12] The definition used in the standard was "... a space about a person or object that typically extends up to 10 m in all directions..."

INTEL-1010
U.S. Patent 8,457,672

a) A single person owns all of the devices.

b) A PAN should form automatically and should exist only for as long as it is needed.

c) The connectivity within a PAN is peer-to-peer among the members of the PAN, although external connectivity is allowed.

In a typical PAN, a single user[13] owns or controls all of the devices. On the other hand, in a LAN, a typical user may not have even seen all of the devices, let alone own or control all of them. In the PAN environment, this allows the user to move the devices around for better reception and to be able to use different authentication processes to enable security. Another difference between PANs and LANs is that although LANs cover relatively small areas, the main use of LANs is to enable connectivity with WANs, in particular, with the Internet. Although a LAN can be valuable in providing connectivity in a building or campus, most people also use the LAN to bridge to the Internet. It is much more likely that a LAN device will be trying to reach other devices on the Internet than in a PAN, where it will be possible that the devices will not have a connection to a WAN.

Another use for a PAN, especially a WPAN, is for disposable networks. In this case, two devices that come in range are able to quickly establish a network to communicate data and just as quickly are able to take down the network when they are done. Fast connect times are important as well as the ability to spontaneously form a network without the intervention of a networking expert. A typical application involves a user walking up to another wireless-enabled device that is able to advertise its services and quickly fill the need. This might involve a digital still camera and a photo-processing kiosk. The user is able to select the number and type of prints on the camera and quickly download the pictures and the printing instructions to the kiosk for printing.

A PAN also has the characteristic that the devices will generally want to connect directly with the other devices in the PAN, i.e., in peer-to-peer mode. Traffic in the LAN, on the other hand, is often directed to devices that are not in that LAN, but rather is routed to other devices that are often quite distant. In the case of IEEE Std 802.11, stations in infrastructure mode are not allowed to communicate

[13] Or a small, closely-related group of people, e.g., a family.

INTEL-1010
U.S. Patent 8,457,672

directly with each other, but need to send all data to and receive data from the access point.

ELEMENTS OF THE 802.15.3 PICONET

There are two key elements to the 802.15.3 piconet: the piconet coordinator (PNC) and the device (DEV), as shown in Figure 3–1. The PNC is the center of the piconet, and the extent of the piconet is determined by the ability of DEVs to hear the beacon transmitted by the PNC. The PNC also acts as a DEV in the piconet and is able to communicate with any other DEV in the piconet. Every 802.15.3-compliant implementation is a DEV, but not all DEVs are capable of becoming a PNC. However, without a PNC, it is not possible to form an 802.15.3 piconet. This leaves open the possibility that two 802.15.3-compliant DEVs would be unable to communicate if neither one of them implemented PNC capabilities. Although this is a potential problem, the task group felt that for some very cost sensitive applications, like speakers, the PNC capability would be an unnecessary expense. In the case of a speaker, it does not have a function in the absence of a sound source, e.g., microphone, DVD player, MP3 player, component receiver, and so on; therefore, the assumption was that two speakers would not need to form a piconet but rather could rely on the sound source to take on the PNC role.

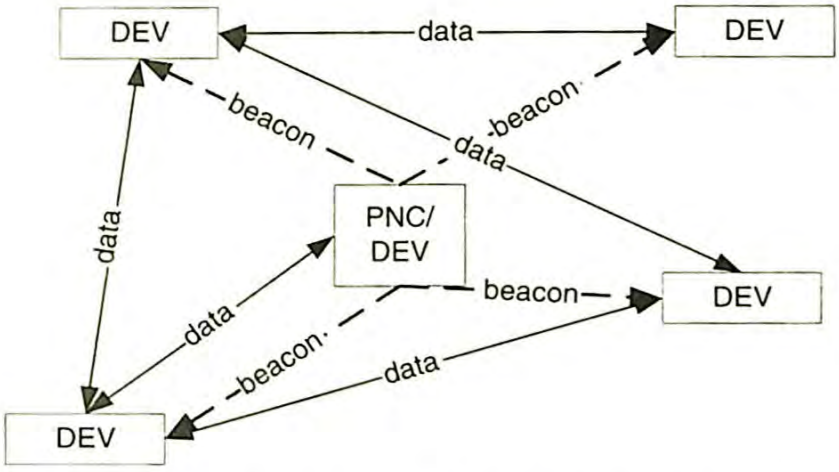


Figure 3–1: Elements of an 802.15.3 piconet

INTEL-1010
U.S. Patent 8,457,672

The PNC is required in every 802.15.3 network because it operates as the arbitrator in assigning channel time. The PNC also provides the timing reference for the piconet via the beacon. All DEVs need the approval of the PNC to join the piconet via association, and if security is enabled in the piconet, they also need to authenticate to the PNC to gain membership in the piconet. The PNC also keeps track of sleeping DEVs to enable them to save power while making sure that their channel time is allocated when they are scheduled to be awake.

Although the PNC has a variety of responsibilities, the standard has made an effort to keep most of the complexity, whenever possible, in the DEV. The reason behind this is that it allows DEVs to be as complex as they need to be to support their application without burdening the DEV that happens to be the PNC. An example of this is channel time allocation. Rather than sending the required data rates, buffer sizes, latencies, and priority to the PNC so that it can determine the appropriate amount of channel time, the DEVs are required to calculate the amount of channel time that they require for their application and the frequency with which it needs to be allocated. The PNC simply decides if there is enough time in the superframe to allocate the channel time. In the case of an application specific DEV, e.g., a microphone or game controller, the DEV already knows the required channel time and allocation frequency and always makes the same channel time request.

The DEVs in the piconet can also take on special roles to fulfill other requirements. For example, a DEV in the piconet can request channel time from the PNC to become the PNC of another piconet, called a dependent piconet. A DEV that is the PNC of a dependent piconet is referred to as a dependent PNC. A dependent piconet shares channel time with the original piconet, called the parent piconet, as shown in Figure 3–2. If a DEV is a dependent PNC, it is acting in two separate roles: the first as a DEV in the parent piconet and the second as the PNC of another piconet that is synchronized to the parent piconet.

An example of a parent and dependent piconet is illustrated in Figure 3–3. In the figure, the dependent PNC is a member of both the parent and dependent piconets. Also note that one of the DEVs is a member of both the parent and the dependent piconet. The standard does not forbid a DEV from joining both piconets. In this case, the DEV would need to keep track of two separate personalities, one as a DEV in the parent piconet and the second as a DEV in the dependent piconet.

INTEL-1010
U.S. Patent 8,457,672

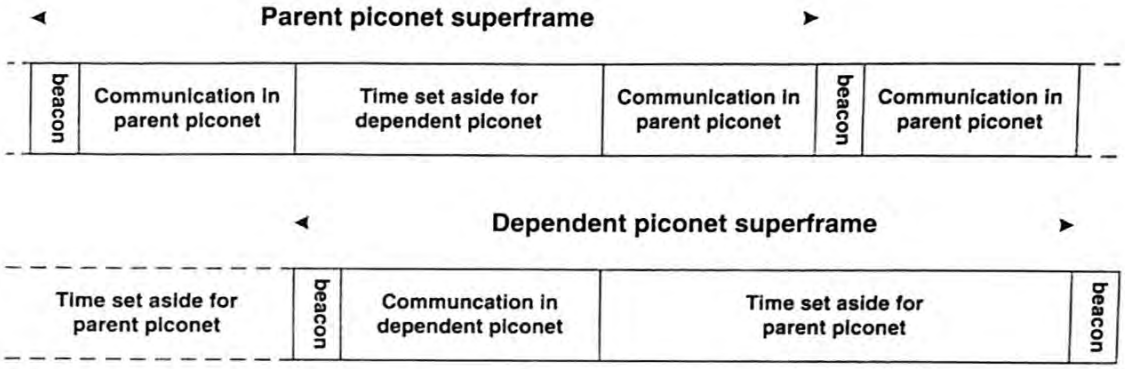


Figure 3–2: Timing relationship for parent and dependent piconets

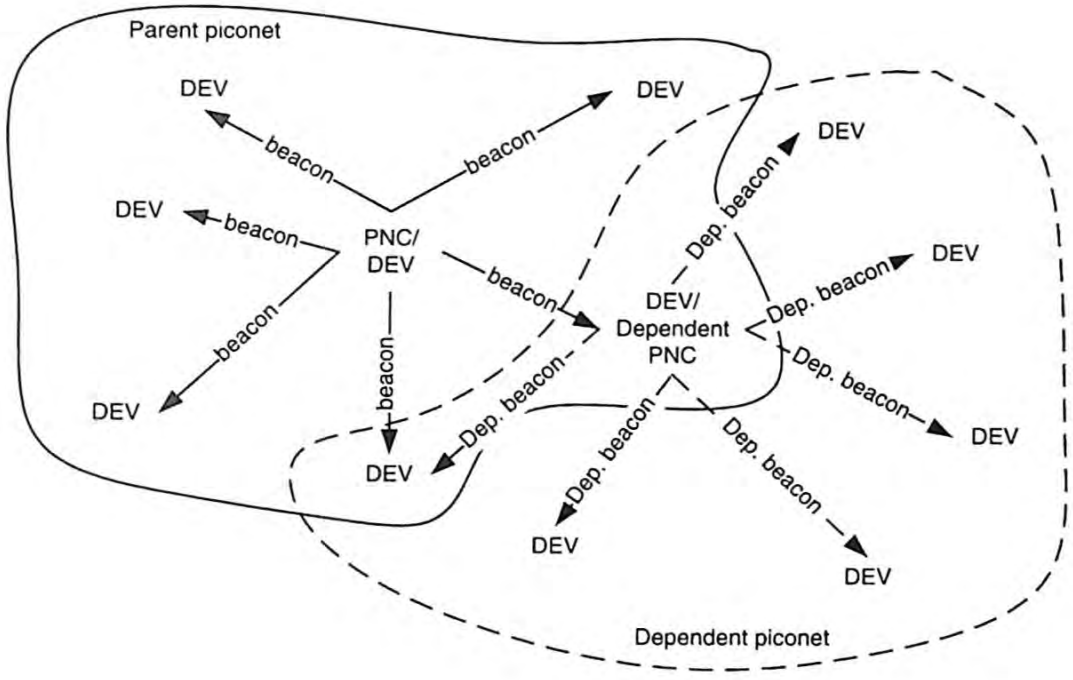


Figure 3–3: Possible connections with parent and dependent piconets

INTEL-1010
U.S. Patent 8,457,672

There are two types of dependent piconets—the child and the neighbor piconet. The difference between the two piconets is determined by the type of membership the dependent PNC has in the parent piconet. If the DEV is a full member of the parent piconet, then it can request channel time to form a child piconet and become a child PNC. A child piconet can be used to extend the range of the original piconet by using the child PNC as a bridge to other DEVs. A child piconet can also be used to provide different level or type security to a subset of the DEVs in the parent piconet.

On the other hand, if the DEV does not want to or is unable to become a full member of the parent piconet, it joins the network as a neighbor PNC. A neighbor PNC is not able to communicate directly with members of the parent piconet (other than with the parent PNC) and it is not given the parent piconet's group data encryption key. A neighbor piconet can be used to support a different level of data security or to share channel time between two piconets controlled by different users.

A DEV may also act as the key originator in the piconet. The key originator is a DEV that generates symmetric encryption keys to protect the confidentiality and integrity of data for subsets of the DEVs in the piconet. In a secure piconet, the PNC is the key originator for all of the DEVs that are members of the piconet. A DEV might take on the role of a key originator if it does not trust the PNC or if it wants to secure an application for a limited set of DEVs with a different security method than the one used by the piconet as a whole. In this role, the DEV authenticates peer DEVs in the piconet and maintains the management and symmetric keys for the DEVs. A key originator performs all of the security-related actions of the PNC with the exception of sending a secure beacon.

PHY OVERVIEW

The 802.15.3 PHY operates in the 2.4–2.4835 GHz frequency band that is available as an unlicensed band in most of the countries in the world. This allows a person to use their 802.15.3-compatible device in Europe, Japan, Canada, and many other countries. The symbol rate is 11 Mbaud with either 1, 2, 3, 4, or 5 bits per symbol to give 11, 22, 33, 44, and 55 Mb/s as a raw data rate.

INTEL-1010
U.S. Patent 8,457,672

The base rate for the 2.4 GHz PHY is 22 Mb/s. The base rate is used for the beacon, association, disassociation, broadcast, multicast, and whenever the DEV is unsure what rate is supported by another DEV or the PNC. All DEVs, therefore, are required to support the base rate, whereas the other data rates and modulations are optional. The 11 Mb/s data rate was added to the standard to enable communications between DEVs that are distant from each other in the piconet. For example, consider the piconet shown in Figure 3–4 with three DEVs and the PNC. As Figure 3–4 illustrates, communication in the piconet can take place directly between members of the piconet, e.g., between DEV-1 and DEV-3, and does not have to first go through the PNC. This is an important difference between 802.15.3 and 802.11's infrastructure mode, which requires that all traffic goes through the access point.

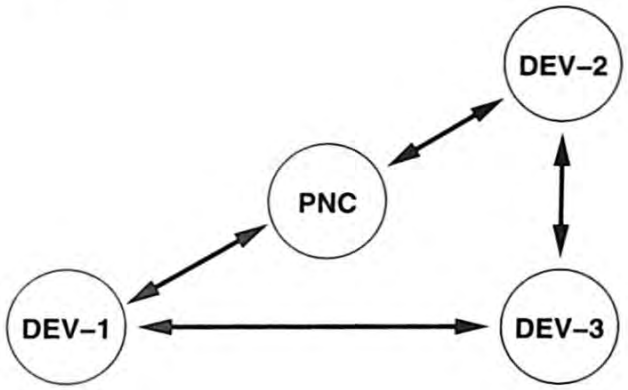


Figure 3–4: DEVs using 11 Mb/s mode to enable data transfer

If the distance between the PNC and DEVs 1–3 is near the limit of what can be reached with 22 Mb/s, the DEVs will be able to join and participate in the piconet. However, although the distance between DEV-2 and DEV-3 is short enough to enable peer-to-peer communication, the distance between DEV-1 and DEV-3 or DEV-1 and DEV-2 will be too long for reliable 22 Mb/s communications. In this case, the DEVs are still able to use the 11 Mb/s transfer mode to communicate between themselves.

The 802.15.3 PHY intentionally reused many of the characteristics of the 802.11 direct-sequence spread spectrum (DSSS) and 802.11b PHYs while seeking to relax the requirements wherever possible. For example, 802.15.3 uses the same frequency accuracy and roughly the same channel plan as 802.11b, but it has a

INTEL-1010
U.S. Patent 8,457,672

relaxed adjacent and alternate channel requirements for the receiver to enable low-cost implementations.

The IEEE Std 802.15.3 Working Group selected a coding technique, trellis-coded modulation (TCM), that has been proven reliable and efficient in wireline standards for decades. This coding method requires a smaller signal-to-noise ratio for the demodulation of 11 Mb/s QPSK-TCM than what is required for the 11 Mb/s CCK mode of 802.11b. Thus, for the same transmit power and receiver noise figure, an 802.15.3 PHY will have a greater range at 11 Mb/s than will an 802.11b radio at 11 Mb/s.

The initial work for the 802.15.3 PHY was aimed at compliance with the FCC's "low power rules" (i.e., 47 CFR 15.249 [B29]) rather than the spread spectrum rules that were used for 802.11b (i.e., 47 CFR 15.247 [B29]). The low power rules would have limited 802.15.3 implementations to about +8 dBm of transmitter power. However, a recent FCC report and order changed the requirements for spread spectrum systems from a jamming resistance test to a power spectral density test. Although this change was intended to allow the proposed 802.11g PHYs to operate in the U.S., it also allowed 802.15.3 radios to use higher transmitter power as well. In the U.S., an 802.15.3 PHY can transmit at up to 1 W with up to 6 dB of antenna gain. In the European Community (EC), both 802.11g and 802.15.3 are limited to 100 mW, including any antenna gain. A typical implementation of 802.15.3, however, would use a transmitter power similar to the current range for 802.11b stations, i.e., about +12 to +18 dBm. However, an extremely low power implementation of 802.15.3 might use a lower transmit power, trading range for battery life. The standard does not have a minimum transmit power requirement.[14]

One of the key differences between 802.11b (or 802.11g) and 802.15.3, other than modulation techniques, is the channel spacing and transmit power spectral density. IEEE Std 802.11b defined a transmit power mask that, by today's standards, is very lax. With digital modulation and pulse shaping that is typically used in current radio implementations, it is relatively easy to control the transmitter spectrum to be less than the 22 MHz that is allowed for 802.11b implementations. Compliant 802.15.3 implementations restrict their transmit

[14] This is also true for all but two of the nine 802.11 PHYs. 802.11 FHSS and 802.11 DSSS do specify a minimum required transmit power.

INTEL-1010
U.S. Patent 8,457,672

bandwidth to less than 15 MHz. This allows up to four non-overlapping channels in the 2.4 GHz band instead of the three channels specified for 802.11b. The narrower transmit bandwidth reduces the interference to other networks, like 802.15.1 and 802.11b, whereas the narrower receiver bandwidth reduces the susceptibility of 802.15.3 to interference from other wireless networks or interference sources. The reduced susceptibility of 802.15.3 to other interferers is very important because the number of wireless devices in the market is increasing at an incredible rate. As the unlicensed frequency bands becomes more congested, the ability of 802.15.3 radios to reject interference and maintain a high data rate will become increasingly important.

There are four primary sources of interference in the 2.4 GHz band: 802.15.1 WPANs, 802.11b WLANs, microwave ovens, and cordless phones. The coexistence performance of 802.15.3 with 802.11b and 802.15.1 was analyzed and placed in Annex C (informative) of IEEE Std 802.15.3, and it is also discussed in Chapter 9 of this book. In general, 802.15.3 piconets can achieve an acceptable level of coexistence with other devices in the 2.4 GHz band. In addition, IEEE Std 802.15.3 and IEEE Std 802.15.2 provide methods that can enhance the coexistence of these networks. These methods are also discussed in Chapter 9. Cordless telephones in the 2.4 GHz band usually are narrowband frequency hopping devices, and so they would tend to affect 802.15.3 WPANs in a manner similar to 802.15.1 WPANs. Microwave ovens, on the other hand, are wideband interferers that operate periodically in the band. There are techniques, out of the scope of IEEE Std 802.15.3, that can be used to mitigate the effects of the microwave oven noise.

STARTING A PICONET

The first activity for an 802.15.3 piconet is its creation. A PNC-capable DEV, instructed by its higher layers, begins to search the channels to find other 802.15.3 piconets and to determine which channel has the least amount of interference. If the DEV is unable to find another 802.15.3 piconet, it would then start its own piconet by broadcasting a beacon. The initial draft for IEEE Std 802.15.3 included a selection process for the PNC wherein PNC-capable DEVs announced their capabilities so that the "most capable" DEV would be selected to be the PNC. However, after some consideration, this process was removed because it is

INTEL-1010
U.S. Patent 8,457,672

unlikely to happen unless all of the DEVs begin scanning simultaneously. Even in the case where all of the DEVs are turned on at the same time, e.g., if there was a power outage, the DEVs will take different amounts of time to "boot up" before turning on the wireless interface.

Instead of a selection process, the first PNC simply looks at the capabilities of DEVs that associate with the piconet to see if a more capable DEV has joined. If the new DEV is better suited to become the PNC, the old PNC will normally hand over responsibility to the new PNC. Thus, in the typical case where the piconet grows slowly, the "best" DEV will eventually become the PNC without the complexity of the selection process. The handover process also allows a PNC that is turning off to pass over coordination responsibilities to another DEV that is remaining active in the piconet. The handover process passes the association information, DEV capabilities, channel time requests, security information and power management states of the DEVs to the new PNC. The intention is that the handover process will be seamless and will only minimally disrupt traffic in the piconet.

Although a network normally needs at least two devices to be considered an actual network, an 802.15.3 piconet is said to be in operation as long as the PNC is sending beacons, even if there are no other DEVs in the piconet. If the beacon from the PNC disappears, due to power outage, interference, etc., the piconet will cease to exist after the timeout specified in the standard. The remaining DEVs in the old piconet that are PNC capable have the option to start a new piconet using the regular piconet start procedure.

THE SUPERFRAME

The basic timing for the piconet is the superframe, as shown in Figure 3–5. The superframe is made up of three parts:

a) The beacon

b) The contention access period (CAP)

c) The channel time allocation period (CTAP)

The beacon is sent by the PNC at a regular interval such that the superframe duration is normally constant. The PNC can change the superframe duration, but it announces the change to the DEVs in the piconet via the beacon before it makes

INTEL-1010
U.S. Patent 8,457,672

the change. The superframe duration is sent in each beacon so that all of the DEVs will know when the next beacon will occur. The beacon also indicates when the CAP ends as well as the channel time allocations (CTAs) in that superframe. The beacon is used to communicate management information as well as to indicate which DEVs are sleeping and when they will be awake.

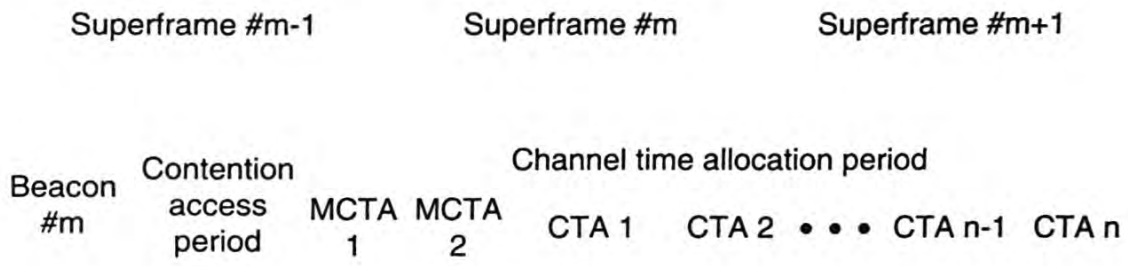


Figure 3–5: The 802.15.3 superframe

The CAP is used to transfer commands and data either between the PNC and a DEV or between DEVs in the piconet. The access method for the CAP is carrier sense multiple access with collision avoidance (CSMA/CA) in a manner similar to IEEE Std 802.11. This access method allows a very efficient use of the time in the CAP while providing a low latency for DEVs that wish to send commands or data. The CAP is well-suited to sending small amounts of asynchronous data between DEVs in the piconet because it does not require a formal channel time reservation process. The CAP is also used by DEVs that are associating with the piconet to send the Association Request commands to the PNC. The length of the CAP may vary between superframes, and the PNC may even set it to be a zero length for some of the superframes. The PNC is allowed by the MAC standard to restrict the type of commands or data that may be sent in the CAP, but for the 2.4 GHz PHY, the PNC is required to allow all DEVs to use the CAP for commands, association, or data.

The CTAP is composed of one or more CTAs that are allocated by the PNC to a pair of DEV identifiers (DEVIDs). Only the DEV that is listed as the source is allowed to transmit in a CTA, and so the access method for CTAs is time division, multiple access (TDMA). In the case where the CTA has the PNC identifier (PNCID) as either the source ID (SrcID) or destination ID (DestID), the CTA is

INTEL-1010
U.S. Patent 8,457,672

called a management CTA (MCTA). In most cases, there is no difference between an MCTA and a CTA. The only exceptions to this are when the source address is either the unassociated ID (UnassocID) or the broadcast ID (BcstID). In this case, there are multiple DEVs that are allowed to send information in the MCTA. To handle potential collisions, a third type of access method, called slotted aloha, is used. Although this method is generally not as efficient as using the CAP for association, commands, and data, it was put in to allow for future PHYs that may not be able to perform clear channel assessment (CCA) efficiently.

Using MCTAs instead of the CAP also adds complexity to the PNC because it now needs to keep track of the potential needs of all the DEVs in the piconet and assign MCTAs periodically to all of them. The DEVs need periodic MCTAs so that they can send commands to the PNC and maintain their association in the piconet. When the PNC allocates the CAP, it does not have go through the effort to separately allocate time for association, commands, and periodically directed links to the DEVs in the piconet. The use of open (BcstID as the SrcID) or association (UnassocID as the SrcID) MCTAs is optional for the protocol and for the 2.4 GHz PHY. Because there is no difference between a regular MCTA and a CTA, all DEVs are required to support the TDMA access method.



The unassocID was originally called the AssocID because it was used by the DEV to associate with the piconet. The name was changed to be the UnassocID in the letter ballot because the only DEVs that use this DEVID are unassociated ones. On the other hand, MCTAs that are reserved for the association process are referred to as association MCTAs.

JOINING AND LEAVING A PICONET

Once the piconet is established by the PNC, DEVs begin to join the piconet using the association process. The association process is used to limit the size of the piconet so that the PNC is not overloaded and to assign an ID to the incoming DEV. Rather than sending the complete MAC address, which is 8 octets, for the source and destination in the MAC header, IEEE Std 802.15.3 requires that new DEVs joining the piconet get an 8-bit ID, the DEVID, from the PNC to identify the DEVs messages in the piconet. This choice cuts 14 bytes of overhead from the MAC header, which improves the efficiency of the 802.15.3 protocol. The PNC is

INTEL-1010
U.S. Patent 8,457,672

able to refuse an association request for many reasons, including the reason that the PNC does not have the resources to handle another DEV in the piconet.

Once the DEV has joined the piconet, it ceases to be a member of the piconet either because it requests to leave, the PNC tells it to leave, or through inactivity in the piconet. When the DEV wants to leave the piconet, it sends a Disassociation command to the PNC. Likewise, if the PNC wants the DEV to leave the piconet, it sends a Disassociation command to the DEV to tell the DEV that it is no longer a part of the piconet. When the DEV joins the piconet, the PNC assigns the DEV an association timeout period (ATP). If the DEV does not successfully send a frame to the PNC at least once during an ATP, the PNC will consider the DEV to be disassociated and will eventually reuse the DEVID it assigned to that DEV for another DEV. In a similar manner, if the DEV does not hear any communication from the PNC for an ATP, it considers itself disassociated and needs to search for a piconet to join.

CONNECTING WITH OTHER DEVICES

Once a DEV has joined the piconet, it finds out about other DEVs in the piconet when the PNC broadcasts the PNC Information command. This command contains a list of all the DEVs in the piconet, their MAC addresses, and their capabilities. With this information, the DEV is able to send small amounts of data to the other DEVs during the CAP or to request channel time from the PNC to send frames to other DEVs.

There are two types of channel time requests (CTRqs) that the DEV may make to the PNC. The first type of CTRq, called an isochronous CTRq, is a request for channel time on a periodic basis that does not expire. Until the request is terminated by either the source, destination, or PNC, the CTRq remains valid. If the PNC agrees to the request, it issues a stream index that is unique in the current piconet and that corresponds to a specific source and destination pair of DEVs. A DEV is able to request more than one allocation per superframe (a super-rate allocation), one allocation per superframe, or one allocation every n superframes (a sub-rate allocation).

The different types of isochronous allocations are used so that the DEV can request the latency and throughput that is consistent with its application.

INTEL-1010
U.S. Patent 8,457,672

Insufficient bandwidth or too much latency can make or break a product. Thus, the ability to have a DEV request and receive a specific amount of latency and throughput from the network is extremely important. This ability alone will open several new market segments for 802.15.3 solutions.

The second type of CTRq is called an asynchronous CTRq. This CTRq is a request for a total amount of channel time that will be allocated whenever the PNC is able to fit it into the superframe. Because it is a request for a total amount of time, once the PNC has allocated all of the requested time, it drops the asynchronous CTRq and no longer attempts to allocate the CTAs. This method is slightly more efficient than using an isochronous CTRq because the DEV does not need to explicitly terminate the CTA when it is done sending data. An asynchronous allocation is not assigned a stream index by the PNC. Instead, all DEVs use the same stream index for asynchronous data.

For both asynchronous and isochronous CTAs, the source may request a modification of the current allocation using the CTRq command. As with the initial allocation, the PNC is able to grant or refuse the request depending on the capabilities of the PNC and the current traffic conditions in the channel. A request to modify an asynchronous allocation results in an update to the amount of time that the DEV is requesting the PNC to allocate.

The source, destination, or PNC may terminate the existing isochronous allocation. The PNC always grants the termination request if all of the following are true:

a) the allocation exists,

b) the DEV that sent the request is a valid member of the piconet, and

c) the requesting DEV is either the source or destination of the allocation.

The source DEV or the PNC may terminate an asynchronous allocation using the CTRq command. Because the asynchronous allocation is somewhat transient, the destination DEV is not allowed to request a termination of an asynchronous CTA.

DEPENDENT PICONETS

Dependent piconets are WPANs that rely on the parent piconet for their channel time and synchronization. The basic idea is that the two piconets are synchronized

INTEL-1010
U.S. Patent 8,457,672

in time and set aside time in their allocations to allow quiet time for the other piconet to operate.

One type of a dependent piconet is the child piconet. The PNC of the child piconet is a full member of the parent piconet and so it is able to communicate with other DEVs in the parent piconet as well as with DEVs in the child piconet. Thus, the child PNC is able to act as a bridge between the two piconets. Because the child PNC is a full member of the parent piconet, it must be a fully compliant 802.15.3 DEV, as well as being PNC capable. Some of the applications for child piconets are:



The child piconet was originally called the daughter network, but it was renamed to be somewhat more politically correct. Task group discussion of creating dependents for a parent, the responsibilities of the parent for the child, or what happens with an ill-behaved child always seemed to evoke side jokes about how this might apply to real life. Because of the wide variety of anthropomorphizing available for parents, children, dependents, and neighbors, the discussion of dependent piconets was often one of the more humorous ones in the standards development process.

- Extending the range of the parent piconet by using the child PNC as a relay to DEVs that the child PNC is able to reach;
- Setting up a separate group for security purposes if the child PNC does not trust the parent PNC;
- Allowing different piconets to share the same channel without interference when there are no other channels available;
- Allowing the formation of dedicated sub-networks that need to control their time allocations separately (e.g., a set of speakers and a component amplifier that are pre-configured as a network at the time of manufacture).

The first step in becoming a child PNC is for the DEV to join the parent piconet using the association process. Once the DEV has joined the parent piconet, it requests channel time using the CTRq command with its own DEVID as both the source and destination. If it is granted time by the PNC, the child PNC can then begin transmitting its beacon in the allocated time. The child PNC sets aside time in its superframe that corresponds to the time that the parent piconet is operating.

INTEL-1010
U.S. Patent 8,457,672

The other type of dependent piconet is the neighbor piconet. The key difference between the neighbor PNC and the child PNC is that the neighbor PNC is not a full member of the parent piconet. In fact, the neighbor PNC is not even required to be a fully compliant 802.15.3 DEV. The neighbor PNC is only required to support a subset of the 802.15.3 commands, and it only needs to support the base rate. One of the unique applications of the neighbor piconet is that it allows controllers from other standards to implement a method of coexistence in which the two networks can negotiate to share time in the same channel without causing collisions. As an example, if an 802.11 AP implemented a few 802.15.3 commands and modified the modem slightly (the PHYs are essentially the same for the base rate and the DSSS mode), it would be able to request time from an 802.15.3 PNC so that it could schedule the traffic for the WLAN when the WPAN was quiet. Similarly, the 802.15.3 PNC would use the quiet time in the WLAN to operate the WPAN.



The name "neighbor" was meant to indicate that the parent PNC would be willing to give the DEV some time, but that it was not really part of the parent's piconet. This is like loaning lawn tools to the neighbors, but not making them a part of the family like your children. See, dependent piconets are one of the funniest parts of the standard as well as being quite useful.

The neighbor piconet can also be used if the PNC-capable DEV is unable to become a secure member of the parent piconet. It would then be able to set up a separate piconet with different security while sharing the same channel. Note that the neighbor PNC is not able to communicate with the DEVs in the parent piconet (unless they also join the neighbor piconet). The key difference in forming a neighbor piconet is that the DEV associates with the parent piconet not as a regular DEV, but as a neighbor PNC. If the parent PNC allows the formation of a neighbor piconets, it will assign the neighbor PNC a special DEVID, the neighbor ID (NbrID), that indicates that it is a neighbor PNC and not a member of the piconet. Once the neighbor PNC has joined the piconet, it can then request channel time for its piconet in the same way that the child PNC requests channel time.

The parent PNC is allowed to force the dependent piconet to stop operations whenever it wants for any reason. For a child piconet, it only terminates the CTA allocated for the child piconet because the child PNC may want to continue as a DEV in the parent piconet. In the case of the neighbor piconet, the PNC uses the

INTEL-1010
U.S. Patent 8,457,672

Disassociate command to cause the neighbor PNC to cease operations. In this case, there is no reason for the neighbor PNC to remain in the parent piconet if it is not going to be given channel time for the neighbor piconet. In all cases, the PNC will listen for the child or neighbor PNC's beacons to allow it time to shutdown the dependent piconet. However, the PNC will eventually remove the allocation even if the child or neighbor has not been able to complete its shutdown process.

OBTAINING INFORMATION

IEEE Std 802.15.3 provides a variety of commands that are used to get information about other DEVs in the piconet or about the current channel conditions. The Probe Request and Probe Response commands are used to exchange information elements (IEs) with other DEVs. On the other hand, the Announce command is used to send, unrequested, an IE to another DEV in the piconet.

The PNC Information Request and PNC Information commands are used to request and provide information about the DEVs that are currently members in the piconet. The PNC Information command contains information about the DEV's capabilities, its MAC address, its DEVID, and other characteristics that are unique to the DEV. For example, other DEVs in the piconet can use the capability information to determine the data rates that are supported by the DEV for communication in the piconet.

The Remote Scan and Channel Status commands are used to determine information about the channel quality in one of the supported PHY channels. Because 802.15.3 operates in the unlicensed band, the DEVs are required to accept interference from primary and secondary users in the band. In addition, the DEVs could encounter interference from other 802.15.1 or 802.15.3 WPANs, 802.11 WLANs, microwave ovens, cordless phones, and so on. A DEV can get information about the quality of a link to a specific DEV with which it is communicating using the Channel Status Request command. The response to this command, the Channel Status Response command, contains information that can be used to determine the frame error ratio (FER) for that link. The PNC is also able to send a request for the collective channel status information to all of the

INTEL-1010
U.S. Patent 8,457,672

DEVs in the piconet by sending the Channel Status Request command as a broadcast frame.

Furthermore, the PNC can request that another DEV in the piconet scan a set of channels to determine if other 802.15.3 piconets are present as well as the channel that appears to be the best from an interference point of view. This information, together with its own assessment of the available PHY channels, can be used by the PNC to move the piconet to another channel.

POWER MANAGEMENT

One of the key advantages of IEEE Std 802.15.3 is its support of power management modes. The goal in any power management strategy is for the DEV to turn off as many of its operations as possible for the longest period of time. When the DEV is turned off or "sleeping," it will not be listening to beacons or other traffic. A DEV is "woken up" when it begins listening to beacons or time allocations on a regular basis. However, it is not enough for the DEV to "disappear" for some period of time. To maintain communications with a sleeping DEV, the other DEVs in the piconet will need to know when it will be awake again. These DEVs also need a method to wake up the sleeping DEV if traffic needs to be sent to it. In addition, if more than one DEV is sleeping, then the DEVs that are running the same application need to wake up at the same time to communicate.

To do this, the standard provides three power-save modes, asynchronous power save (APS), device synchronized power save (DSPS), and piconet synchronized power save (PSPS). A DEV that is not in a power-save mode and is therefore listening to every beacon is in ACTIVE mode. In any of the four power management modes, ACTIVE or the three power-save modes, the DEV may be in SLEEP state or AWAKE state, depending on the traffic in the piconet. If the DEV sees that it is not the source or destination of a particular CTA, it is not required to have its receiver active for that duration. This allows DEVs, even those in ACTIVE mode, to save power. The various groupings of power-related modes in the standard are listed in Table 3–1.

INTEL-1010
U.S. Patent 8,457,672

Table 3–1: Power-related modes in 802.15.3

Name of mode grouping	Valid modes
Power management modes	ACTIVE, APS, DSPS, PSPS
Power save modes	APS, DSPS, PSPS
Synchronized power-save modes	DSPS, PSPS

For greater power savings, the DEV may choose to switch to one of the three power-save modes. The APS mode offers the best power savings by giving up synchronization with the rest of the piconet. In APS mode, the DEV simply stops listening to beacons. All of the streams with the DEV as the source or destination are terminated by the PNC. The only responsibility that a DEV in APS mode has is to send a frame to the PNC at least once every association timeout period (ATP) in order to maintain its status as being associated with the piconet. When another DEV wants to request channel time with a DEV in APS mode, it sends a CTRq command the PNC. The PNC then sets the DEV's bit in the Pending Channel Time Map (PCTM) IE to tell the DEV that there is a CTA pending for the DEV. The sleeping DEV has the option to terminate the newly allocated stream if it does not want to wake up.



The ATP is used by the PNC as a heartbeat for each of the associated DEVs. If the PNC does not hear from a DEV at least once every ATP, it assumes that the DEV has disappeared and the PNC disassociates the DEV. Likewise, if a DEV doesn't hear from the PNC within an ATP, it assumes that it is no longer a part of the piconet and that it has been disassociated.

The next-best power savings method is the DSPS mode. In this mode, the DEVs wake up periodically to listen to beacons. A beacon during which the DEV is in AWAKE state is referred to as a wake beacon. The time period between wake beacons is called the Wake Beacon Interval. The beacon number when the DEV will be awake is called the Next Wake Beacon. A group of DEVs that share the same Wake Beacon Interval and Next Wake Beacon is called a DSPS set. Multiple DEVs can join the same DSPS set, which allows them to be awake at the same time. Because an 802.15.3 communications link is peer-to-peer, it is important that both the source and destination are awake at the same time so that they can exchange frames. DSPS mode allows DEVs with similar application requirements

INTEL-1010
U.S. Patent 8,457,672

to be asleep for long periods of time while maintaining synchronization of their wake beacons. Furthermore, the DEVs can allocate sub-rate channel time, i.e., channel time every N beacons, aligned with their wake beacons. The only requirement is that the sub-rate interval is greater than the Wake Beacon Interval so that the sleeping DEVs are not required to wake up more often than their normal wake cycle. Figure 3–6 shows the wake beacons and sub-rate allocation for a DSPS DEV that has a Wake Beacon Interval of 4 (i.e., it is awake every 4th beacon) and has sub-rate allocation every 8th beacon (i.e., its CTA rate factor is 8).

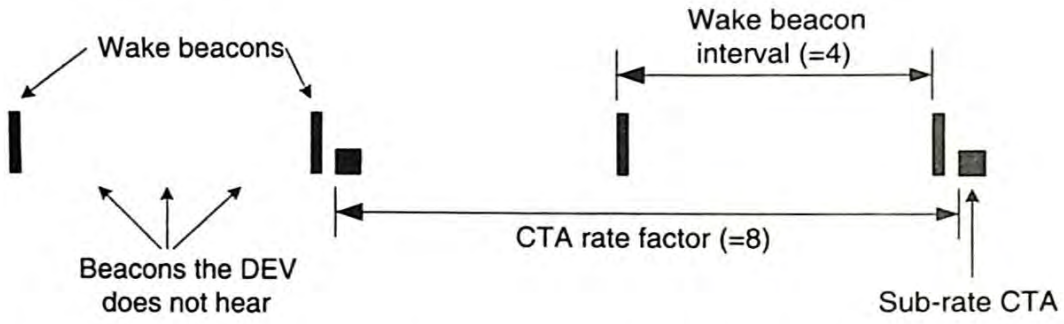


Figure 3–6: Wake beacons and Wake Beacon Interval for a DSPS DEV



All of the DEVs that belong to a particular DSPS set do not have to be in the same power management mode. Some of the DEVs in the set may be in ACTIVE mode while other DEVs in the set may be DSPS, PSPS or even APS modes.

The DSPS process begins when a DEV uses the SPS Configuration Request command to request that the PNC create a DSPS set with a specified interval. The PNC cannot change the interval; it must either reject or accept the request. The PNC is able to set the Next Awake Beacon in order to spread out the traffic in the piconet. The PNC is also able to change the Next Awake Beacon occasionally as the channel time requirements in the piconet change. Once the DSPS set has been created, it is assigned a DSPS set index to identify it. Other DEVs are then able to join this set by sending an SPS Configuration Request command to the PNC. A DEV may be a member of a DSPS set while it is in ACTIVE mode. The DEV then switches between ACTIVE and DSPS modes by using the PM Mode Change command. When the DEV is in DSPS mode, the PNC schedules asynchronous allocations for the DSPS DEV during its wake beacons. However, broadcast and multicast traffic is not

INTEL-1010
U.S. Patent 8,457,672

necessarily aligned to the DSPS DEV's awake beacon because there can be more than one DSPS set, each with different intervals and Next Wake Beacons. If a DEV wants to send broadcast traffic to DSPS DEVs, it needs to send the frames during that DEV's wake beacons. The PNC must support a minimum of one DSPS sets, but it may support more depending on the implementation.

The last power-save mode is piconet synchronized power save (PSPS) mode. The primary goal of PSPS mode is not power savings, but rather ensuring that DEVs are awake for system changes and announcements. The DEVs in PSPS mode wake up at a time specified in the PS Status IE for the PSPS set (i.e., set 1). However, unlike DSPS modes, the DEVs in PSPS mode are not allowed to specify the interval between wake beacons. Instead the PNC chooses this interval based on requests from the DEVs in PSPS mode. The beacon when DEVs in PSPS mode are required to be in AWAKE state is called the system wake beacon. Because there is only one PSPS set for the entire piconet, the sleep interval, and hence the power savings, is a compromise between the requests of the DEV and the PNC's desire to change piconet parameters quickly. The PNC is allowed to change the interval between every system awake beacon. The PNC is prohibited from making major piconet changes until at least one system wake beacon has passed. This is intended to improve the chances that a DEV will hear the system changes before they happen. Of course, if any DEV misses the system change announcements, it will be able to recover from the changes by scanning for its piconet.

DEVs in PSPS mode keep their allocations when they switch into PSPS mode. However, because the system wake beacon may not be aligned to the PSPS DEV's sub-rate allocations, the PSPS DEVs need to wake up for not only the system wake beacons, but also superframes that contain its sub-rate allocations. Note that a DEV in PSPS mode with a CTA in every beacon is essentially in ACTIVE mode, because ACTIVE mode is defined as a DEV that listens to every beacon.

SYSTEM CHANGES

Because the PNC coordinates the activities of the piconet, it is also responsible for changing the parameters of the piconet whenever it is required. In particular, the PNC has the responsibility of choosing the PHY channel for the piconet and changing it whenever the channel conditions deteriorate. The PNC also sets and

INTEL-1010
U.S. Patent 8,457,672

changes the superframe duration and is able to move the location of the start of the superframe in time. The PNC also selects the piconet identifier (PNID) and the beacon source identifier (BSID), which are used by the DEVs to identify the piconet.

Whenever the PNC wants to change one of these key parameters of the piconet, it puts the Piconet Parameter Change IE in the beacon to warn the DEVs of the impending change and to indicate the beacon number when the change will occur. The PNC will keep this IE in the beacon until the change occurs. In addition, if there is a DEV in PSPS mode, the PNC needs to ensure that this IE is placed in at least one system wake beacon before the change takes place.



The BSID acronym was based on the initials of one of the major contributors to the 802.15.3 standard. The acronym was selected first and then words were found that fit the initials. While not quite as funny as anthropomorphizing dependent piconets, it did seem humorous at the time.

IMPLEMENTATION COST AND COMPLEXITY

One of the most important goals in the development of IEEE Std 802.15.3 was to create a standard for which low-cost implementations could be built. For the MAC, the cost and complexity of the standard was reduced in the following ways:

- Not all DEVs are required to be PNC capable. This allows very low cost implementations (e.g., a wireless speaker or remote control).
- The number of distinct frames, commands, and information elements was kept to a minimum.
- A DEV only needs to support the power-save modes that it needs. If the implementation is not battery-powered, the DEV would not need to support any power-save modes.
- The stream index for isochronous connections allows the DEV to easily separate data based on priority, both for transmission and reception.
- The fragmentation numbering was selected to allow the DEV to easily defragment the received frame with a minimum amount of processing.
- A simple PNC can be built that only uses a very rudimentary channel allocation algorithm. This is useful in applications such as personal digital

INTEL-1010
U.S. Patent 8,457,672

assistants (PDAs) which are cost sensitive, but require the ability to run a simple piconet for data transfer (e.g., PDA to PDA transfer).

- The PNC is not required to forward data between DEVs. Unlike the 802.11 AP, which is required to store and forward data for power save STAs, the 802.15.3 PNC only keeps track of when a DEV will next be awake. The sending DEV can use this information to determine when it will be able to successfully send data to the power save DEV.

The 802.15.3 PHY was also selected to allow for very low cost implementations. Some of the design decisions that allow for cost reductions in some implementations are:

- A DEV is only required to implement the DQPSK mode, which does not require coherent reception. In addition, it can be received even when there is clipping in the receiver chain.
- The transmit power can be selected for the application. For some applications, a power amplifier (PA) will not be necessary and so the cost can be reduced.
- No special coding is required for the base mode (22 Mb/s DQPSK)
- The frequency and timing accuracies (+/- 25 ppm) as well as the symbol rate are the same as in 802.11b. This allows the implementer to re-use crystals that are now in relatively high volume production for 802.11b and 802.11g.
- Standard 802.11b RFICs can be used for the 22 Mb/s mode while 802.11g RFICs can be used for the higher order modulations. The design work has already been completed for these ICs and they are in relatively large volume production, reducing risk and price.
- The required sensitivity for the standard was relaxed to allow lower-cost radios.
- The frequency range (2.4 to 2.5 GHz) allows RFIC designers to use low-cost SiGe or RF-CMOS processes for the radio design.
- The PHY was designed to allow direct-conversion receivers, which can greatly reduce the cost and area of an implementation.

INTEL-1010
U.S. Patent 8,457,672

Chapter 4 MAC functionality

MAC TERMINOLOGY IN IEEE STD 802.15.3

One of the tasks in writing a good standard is to choose a set of terminology and to consistently use it. For example, in IEEE Std 802.15.3, the basic data structure is called a frame instead of a packet. Although the term "packet" would have been acceptable and is used in other documents to indicate essentially the same thing, the technical editor, with input from the Task Group, decided instead to use the term "frame." There are five types of frames in the standard: beacon, immediate acknowledgment (Imm-ACK), delayed acknowledgement (Dly-ACK), command, and data. "Command frames," or simply "commands," are used to communicate control information between the DEVs and the PNC or among DEVs. The standard also uses information elements (IEs) that are a type, length, and value encoding of information. IEs are used in cases when there is a variable number of elements that need to be transferred among the PNC and DEVs. The standard also adopted the use of the word "octet" instead of "byte" when referring to 8 bits of data.

Probably the most obvious difference in terminology between IEEE Std 802.15.3 and IEEE Std 802.11 is that members of the 802.11 WLAN are called stations (STAs), whereas in 802.15.3, they are called devices (DEVs). In 802.11, the STAs are coordinated, in infrastructure mode, by the access point (AP), whereas in 802.15.3, the DEVs rely on the piconet coordinator (PNC) to maintain order in the piconet. The decision to use different terminology was to draw a distinction between the architecture of the two standards. In 802.11, the key purpose of an AP is to provide a link to another network, either wired or wireless, that expands the reach of the WLAN. Often, this is a link to a wired network that then has access to the WAN, particularly to the Internet. On the other hand, in 802.15.3, the PNC is not required to have access to any other network. The PNC is used, not for access to other networks, but rather to control access to the time resources of the piconet. The piconet may have a DEV that acts as an access point, but that DEV is not required to be the PNC of the piconet. In addition, the PNC is also a DEV in the

INTEL-1010
U.S. Patent 8,457,672

piconet and so it is just as likely to be a source or sink for a data stream as any other DEV in the piconet. An 802.11 AP, on the other hand, usually functions only as the access point, it does not act as an endpoint for the data traffic.

The 802.15.3 standard differentiates among four different types of piconets that can be formed: independent, parent, child, and neighbor. In addition, child and neighbor piconets are also referred to collectively as dependent piconets when the text applies equally to both types of a piconet. An independent piconet is an 802.15.3-compliant piconet that is not a dependent of a parent piconet and does not have any dependent piconets as a part of its piconet. The PNC of the independent piconet does not have to synchronize with any other PNC, and it is not required to provide consideration for any other dependent PNCs. A parent piconet is one in which a dependent piconet has been given a channel time allocation (CTA). The PNC of the parent piconet, referred to as the parent PNC, does not have to synchronize with any other PNC. However, it is required to take additional action when it stops a dependent piconet (see "Parent PNC stopping a dependent piconet" on page 176).

The child piconet relies on the parent PNC for its channel time and it is synchronized to the beacon of the parent PNC. The PNC of the child piconet, called the child PNC, is a full member of the parent piconet and is allowed to communicate with any of the other DEVs in the parent piconet.

The neighbor piconet is like the child piconet in that it relies on the parent PNC for its channel time and it is synchronized to the beacon of the parent PNC. However, the PNC of the neighbor piconet, referred to as the neighbor PNC, is not a full member of the parent piconet. The neighbor PNC is not allowed to communicate directly with other DEVs in the parent piconet. The neighbor PNC is not even required to be a PNC-capable, fully-compliant 802.15.3 DEV; it could be an entity that is compliant to another standard and that implements the minimal amount of functionality required to create a neighbor piconet within an existing 802.15.3 piconet. Dependent piconets are described in more detail in Chapter 5.

A frame exchange sequence involves at least two DEVs. A DEV that is sending a frame is the source of the frame, whereas the DEV (or DEVs) that are the intended recipient of the frame is the destination. In a sequence of frames, e.g., data frame followed by Imm-ACK, the DEV that begins the sequence is referred to as the originator of the exchange.

INTEL-1010
U.S. Patent 8,457,672

The other DEV (or DEVs) involved in the exchange are referred to as the target of the exchange. As an example, consider a data frame/Imm-ACK exchange with two DEVs, DEV-1 and DEV-2. DEV-1 sends the data frame to the DEV-2 requesting an Imm-ACK. Thus, DEV-1 is the originator of the sequence and DEV-2 is the target for the duration of the exchange. The source of the data frame is DEV-1, and its destination is DEV-2. When DEV-2 sends the Imm-ACK frame, it is the source of the Imm-ACK frame and DEV-2 is the destination. However, DEV-1, the destination of the Imm-ACK frame, is still the originator and DEV-2, the source of the Imm-ACK frame, is the target.



The key concept in understanding originator, target, source and destination is that the originator and target DEVs are unchanged during a frame exchange sequence, whereas the source and destination can change with every frame that is sent.

FRAME FORMATS

The goal of a standard is to enable interoperability between devices built by different manufacturers. One of the key clauses (IEEE Standards terminology for a chapter in a standard) in the standard for interoperability is Clause 7, which defines the frame formats. Although it is extremely important that implementers agree in every detail of the frame formats, the actual formats are somewhat arbitrary. Thus, although Clause 7 of the standard is key in the actual implementation, the details of it are not very interesting from the point of view of understanding how the standard works. There are a few areas in Clause 7, however, that do require some special attention.

The first area of interest is the bit ordering for the frames. There are two types of ordering defined in the beginning of the clause; the order in which the bits, octets, and fields are transmitted over the medium and the order that they are represented in the standard. In both cases, there is no correct way to order the bits, octets, and fields; rather, there are some minor trade-offs in choosing one method over the other. The frame formats clause in the initial draft was based very closely on IEEE Std 802.11-1999 and directly copied some of the wording. In particular, most of the text on the bit ordering over the air is exactly the same as the text in IEEE Std 802.11. Thus, in IEEE Std 802.15.3, the least significant bits of the least significant octet, e.g., bit 0 of octet 0, is sent first, whereas bit 7 of octet N (of an N octet frame) is sent last over the air, as shown in Figure 4–1.

INTEL-1010
U.S. Patent 8,457,672

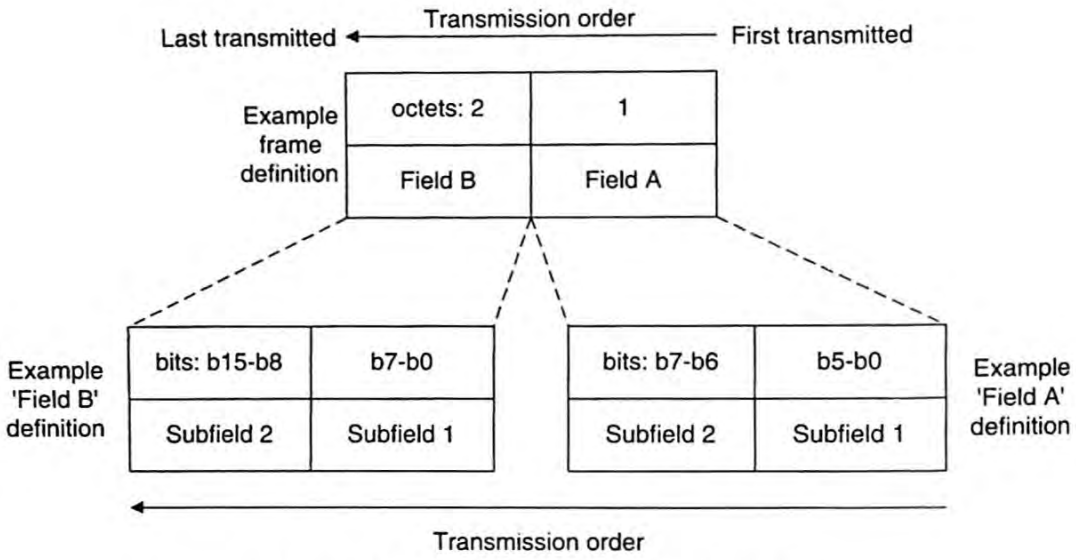


Figure 4–1: Bit and octet ordering

Despite the fact that the wording was taken directly from 802.11, which had multiple vendors providing interoperable solutions, some members felt that the text was a little confusing. In response to this, additional text and figures were added to clarify the bit and octet ordering.

One of the questions regarding bit ordering had to do with strings, like the one in the BSID IE. Suppose that users wanted to call their network "Wireless." Would the BSID IE be sent with "Wireless" or as "sseleriW;" i.e., would "W" or "s" be sent first? The draft indicates that the lowest order octet is always transmitted first. In the C language (and in many other programming languages), the first character in a string is indexed by the number zero. Thus, the lowest-order octet in "Wireless" is the ASCII encoding of "W," and so the string is sent as "Wireless," with the encoding of "W" sent first over the air. An example of the order that the data payload or a text field is transmitted is shown in Figure 4–2.

The other issue with bit ordering had to do with the representation of the frame formats in the draft. This mainly affected Clause 7, but it also affected a few figures and tables in the 2.4 GHz PHY clause, Clause 11. In this case, there is no technical difference if the figures have the first bits and octets on either the left or

INTEL-1010
U.S. Patent 8,457,672

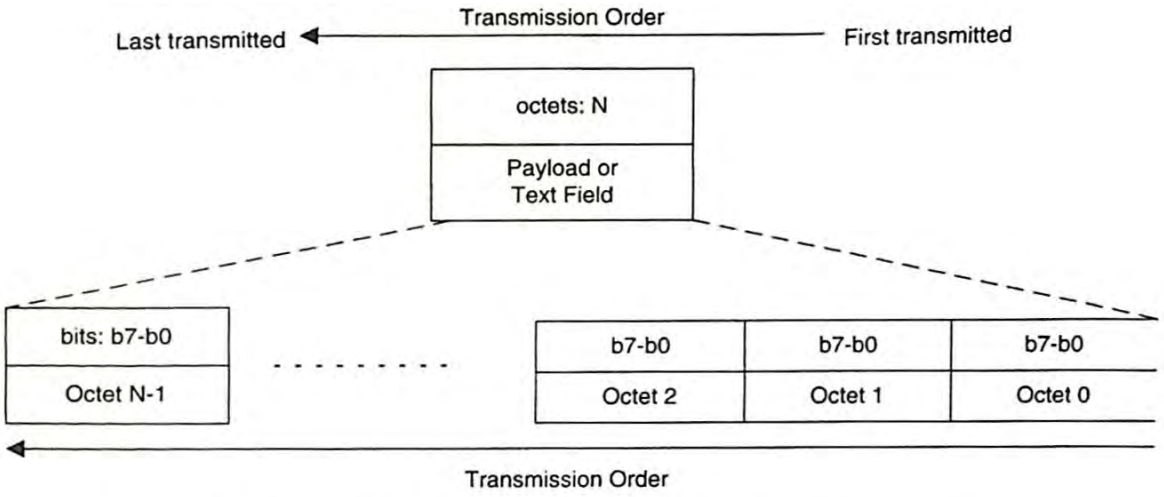


Figure 4–2: Example of payload or text field transmission order

the right side of the page. The key was to be consistent throughout the draft to minimize confusion. The issue is that numbers are usually represented with the most significant digits, octets, or bits on the left. However, many people are used to seeing the first bit sent on the left. Because the first bit over the air is the least significant bit, it was not possible to simultaneously satisfy both concerns. The decision was made to put the first bits, octets, and fields transmitted over the air on the right-hand side of the page so that the frame is transmitted from right to left.

The task group also attempted to minimize the number of frames, commands, and information elements. At the same time, the group also tried to avoid overloading any single command with too many functions. As with many things in the standard, however, it was not possible to simultaneously achieve both goals. In general, the group would err on the side of overloading the use of an existing command or IE rather than creating a new command or IE.

PICONET TIMING AND SUPERFRAME STRUCTURE

Timing in the piconet is based on the superframe. The superframe begins with the first symbol in the PHY preamble of the beacon and ends a superframe duration later. The DEVs are required to set their timers to zero at the start of the PHY

INTEL-1010
U.S. Patent 8,457,672

preamble of the beacon. Because the DEV will not know that the beacon has arrived until the MAC header is processed, the DEV never really sets its internal timer to zero; rather, it sets it to a time that corresponds to the time of the event when it knows the beacon has arrived. For example, if the PHY reports when the HCS has completed, the DEV would set its timer to the sum of the duration of the PHY preamble, PHY header, MAC header and HCS. The beginning of the PHY preamble is used because it represents the first transmission in the medium during the superframe. Thus, the last transmission during the superframe needs to end prior to the superframe duration. The timing for the piconet is illustrated in Figure 4–3.

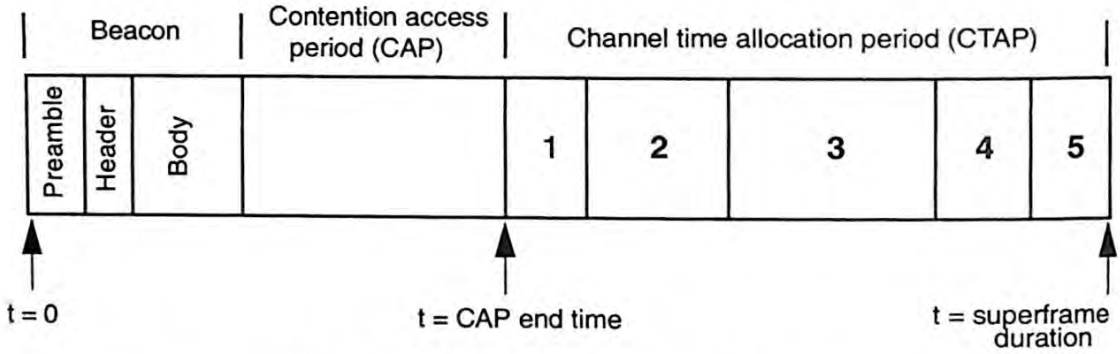


Figure 4–3: Piconet timing relative to the beacon

The value of the superframe duration is contained in the beacon and is generally constant during piconet operations. This allows DEVs that are sleeping to predict the timing of their wake beacon will occur so that they can wake up on time. The PNC is able to change the superframe duration, but it is required to warn the DEVs in the piconet via the Piconet Parameter Change IE in the beacon before it changes the value. This process is described in more depth in "Moving the beacon or changing the superframe duration" on page 136.

The beacon determines the start of the superframe and carries critical information regarding operation in the superframe. If a DEV misses the beacon, it will not be able to transmit unless it has been assigned a pseudo-static CTA (see "Channel time allocation period (CTAP)" on page 56). The beacon is used to communicate quite a bit of information to the DEVs in the piconet, and so it may grow to be a fairly large frame. As the number of octets in the beacon increases, the probability

INTEL-1010
U.S. Patent 8,457,672

that it will be correctly received decreases. In addition, it may not be possible to fit all of the required information elements in the beacon due to PHY limitations on maximum frame size.

IEEE Std 802.15.3 overcomes this limitation by allowing the PNC to split the beacon into the beacon frame and one or more broadcast Announce commands that follow the beacon. The broadcast Announce commands that follow the beacon frame are referred to as the beacon extension and the combination of the beacon frame and the beacon extension is simply referred to as the beacon. The PNC will always send the CTA IEs, the BSID IE, and the Parent Piconet IE, if present, in the beacon frame. These IEs are the most critical to the operation of the piconet, and so they must occur in the beacon frame. In addition, the CTAs are all referenced to the start of the beacon frame, so if a DEV did not receive the beacon frame, it would not be able to use the allocations because it would not know when the superframe started. Other IEs that the PNC needs to send can be placed in the broadcast Announce commands that follow the beacon. The PNC indicates that the Announce commands will be following the beacon by setting the More Data bit to 1 in the beacon and in all but the last broadcast Announce command. Because the Announce commands in the beacon extension are sent to the broadcast address, the ACK Policy field is set to no-ACK.

The rest of the superframe is made up of the contention access period (CAP) and the channel time allocation period (CTAP), as shown in Figure 4–3.

Interframe spacings

IEEE Std 802.15.3 defines four different timings that govern the interframe spacing (IFS) in the superframe. All of these timings are PHY dependent because they are determined by the specific characteristics of the PHY. The shortest of these IFSs is called the minimum interframe space (MIFS). This IFS is the amount of time that the PHY needs either between successive transmissions or successive receptions. This timing is only used in directed CTAs when the DEV that is sending the frame does not expect a response from the receiving DEV, i.e., when the source DEV either requests no ACK or is in the middle of a Dly-ACK



In one of the proposals for the TG3a PHY [B5], the throughput improvement from using MIFS with Dly-ACK (5 frame bursts) was 11% at 110 Mb/s, 18% at 200 Mb/s, and 33% at 480 Mb/s for 1024 octet frames.

INTEL-1010
U.S. Patent 8,457,672

burst. The MIFS was introduced to reduce the overhead and improve the efficiency for very fast PHYs.

The next shortest time is called the short interframe space (SIFS). The SIFS is the length of time required for the PHY to switch between transmit and receive. The SIFS is used when the DEV that is transmitting expects a response from another DEV, e.g., an ACK. It is also used between CTAs because a DEV may be receiving or transmitting at the end of one allocation and be required to be in the opposite state at the beginning of the next CTA.

The backoff interframe space (BIFS) is only used in the CAP. It is based on the transmit/receive turnaround time and the length of time required to determine that the medium is clear. Thus, the BIFS is the sum of the SIFS and the time required to perform a clear channel assessment (CCA).



The BIFS acronym was not named for anyone involved in the 802.15.3 standards development. Rather, the acronym is intended to imply the name Biff, which, for some reason, the technical editor thought was funny.

The final IFS is called the retransmission interframe space (RIFS). A DEV that is expecting an ACK from another DEV will wait at least one REFS duration before it either resends the frame or gives up on it and sends a different frame. There were two viewpoints in the Task Group regarding the value of the RIFS. The first view was that the RIFS should be similar to the definition in 802.11; in which it is defined to be the amount of time to change from transmit to receive plus the time to listen for the ACK to start plus the time to change back to transmit from receive. Based on this viewpoint, the RIFS would be equal to SIFS plus CCA detect time plus SIFS, or equivalently, it would be equal to a BIFS plus a SIFS.

The other viewpoint was based on two assumptions. The first is that the there may be PHYs in the future that are unable to perform CCA or do it so slowly that it is unuseable. The other assumption is that it might be likely that the source DEV would miss the ACK starting and begin transmitting the next frame on top of the ACK being sent by the receiver. If this was true, then not only would the first frame transmission be lost, but also the second transmission would be lost as well. Using these assumptions, the RIFS would be equal to two SIFS plus the length time required for the ACK. If the DEV was requesting an Imm-ACK, then this time is easily calculated. On the other hand, if the DEV was requesting a Dly-ACK, then the source DEV cannot be sure of the exact duration of the

INTEL-1010
U.S. Patent 8,457,672

Dly-ACK frame. However, the source DEV can put an upper limit on this time and it would have to use the worst-case time for this method.

In the case of the 2.4 GHz PHY, the time required for CCA (7.3 µs) is much shorter than the time required for an Imm-ACK (about 23 µs), so the penalty for waiting for the entire ACK is quite large. In addition, the PHY preamble for the 2.4 GHz PHY has very good auto- and cross-correlation properties. These properties give the PHY preamble greater than 10 dB improvement of the signal-to-noise performance as compared with the data modulation. Thus, for the 2.4 GHz PHY, it is highly unlikely that a DEV would have reliable communications with another DEV in the piconet and not be able to detect the PHY preamble. Because there is a low probability of missing the preamble of the ACK and a large time penalty for waiting, the 2.4 GHz PHY uses the CCA method for determining RIFS. Because this IFS, like all of the others, is PHY dependent, the definition of it will be revisited when the next PHY, 802.15.3a, is selected.

Contention access period (CAP)

The CAP is used to provide efficient access for short bursts of data or commands. The PNC, however, is not required to allocate a usable CAP every superframe. If the PNC feels that the CAP traffic in the piconet is low enough, it could skip allocating the CAP to allow more CTAs in the superframe or to handle instances of superframe overloading (see "Managing bandwidth" on page 81). Unlike the channel time allocations (CTAs) in the CTAP, a DEV is not required to request time in the CAP before it is given a chance to use it. Because the DEV is not required to reserve time before using the CAP, it makes the CAP a very efficient method for transmitting small amounts of asynchronous data.



All transmissions in the CAP either have to be ACKed in the CAP or sent as a no-ACK frame. The sending DEV needs to make sure that there is enough time remaining in the CAP for its frame and the anticipated ACK, if required. Dly-ACK is not allowed in the CAP, it can only be used during a CTA.

INTEL-1010
U.S. Patent 8,457,672



The PHY parameter, pBackoffSlot is the same value as the BIFS. In fact, BIFS could have been used instead of pBackoffSlot throughout the standard. This is a good candidate for things to fix in a Corrigendum.

When a DEV is going to transmit a frame in the CAP, it picks a random number between 0 and 7 to use as its backoff counter, n. It then waits until the channel has been clear for n*pBackoffSlot period of time and then gets to try and transmit its frame. Unlike 802.11, the counter is not reset if the medium is determined to be busy. Likewise, the counter is not reset for each superframe. One of the reasons for this choice is that the CAP is only a portion of the total time in the superframe. If the DEV picked a long backoff counter with a short CAP, it might never get a chance to transmit a long frame.

If the DEV transmits the frame but does not receive an acknowledgment, it may attempt to retransmit the frame. In this case, the backoff window increases from 0 to 7 to be 0 to 15. The window increases each time the frame is transmitted but not ACKed, next to 0 to 31 and finally to a window of 0 to 63. The DEV continues to use this backoff window until it gives up on sending the frame. Regardless if the previous frame was sent successfully or unsuccessfully, the next frame uses a backoff window of 0 to 7 for its first attempt to be transmitted during the CAP.

Channel time allocation period (CTAP)

The CTAP is defined as the time in the superframe that follows the end of the CAP up until one SIFS before the start of the next beacon (which is one superframe duration after the start of the previous beacon). The PNC is not required to fill up the CTAP with CTAs or to expand the CAP to take up any unused time, so it is possible that there would not be any CTAs in the CTAP of a particular superframe. There might also be large gaps in time in between CTAs in the CTAP where the PNC has not allocated any channel time. The CTAs are communicated to the DEVs in the piconet via the CTA IE in the beacon. Each CTA IE contains one or more CTA blocks, each of



The CTAP was originally called the contention free period (CFP) throughout most of the life of the draft. However, the standard does allow the PNC to create open and association management CTAs (MCTAs) in the CTAP. Thus, the CTAP contains time when DEVs are contending for the medium using slotted aloha, and then so it could no longer be called a contention free period. This change was applied only in the latest drafts, so earlier documents in the task group will refer to this portion of the superframe as the CFP rather than the CTAP.

INTEL-1010
U.S. Patent 8,457,672

which has the source, destination, stream index, start time, and duration of the CTA. The CTA blocks are ordered by increasing start time of the allocation, with the lowest start time sent first over the air. Because IEs are limited to 255 octets in length, the PNC may be required to put multiple CTA IEs in the beacon to handle all of the allocations for the superframe. Only the source DEV is allowed to initiate a frame exchange sequence during a CTA. The destination is only allowed to transmit the requested ACK frames, either Imm-ACK or Dly-ACK, during the CTA.



A CTA is the "property" of the source DEV. The source can use the channel time for almost any purpose, including making connections with the destination DEV using a non-802.15.3 protocol. This feature is used to support neighbor piconets.

Although every CTA has a stream index, the source DEV may send data with any stream index during the CTA. If the DEV has higher priority traffic it needs to send to the destination, it can use any or all of the CTAs that it has allocated to the destination to send the data. This includes sending asynchronous data or commands to the destination DEV. In general, once the DEV has been given the CTA, it is allowed to use the time for any purpose.

IEEE Std 802.15.3 always allowed, and in some cases required, the PNC to allocate channel time with another DEV in the piconet. In this case, the allocation is not for the PNC's DEV personality, i.e., using its DEVID, but rather for its PNC personality, i.e., using the PNCID. These CTAs are exactly the same as regular CTAs, except that the PNCID is either the SrcID or the DestID.[15] The standard refers to these types of CTAs as management CTAs (MCTAs). When the SrcID is either the PNCID or one of the DEVIDs of a DEV in the piconet, then the allocation is the same as other CTAs. However if the SrcID is either the unassociated ID (UnassocID) or the broadcast ID (BcstID), then the CTA is no longer contention free, but rather uses the slotted-aloha contention method for access. An MCTA with the UnassocID as the SrcID is called an association MCTA, whereas an MCTA with the BcstID as the SrcID is called an open MCTA. Open and association MCTAs were introduced to allow the possibility of a future PHY that was not able to do CCA in an efficient manner.

[15] These CTAs also use the MCTA Stream Index, but that does not affect the traffic in the CTA because frames with any valid stream index can be sent in the CTA as long as the SrcID matches the SrcID for the CTA.

INTEL-1010
U.S. Patent 8,457,672

The slotted-aloha mechanism is similar to the CSMA/CA method used in the CAP. Instead of using a backoff interval based on listening to the channel, the slotted-aloha method uses the entire MCTA as a backoff slot and counts down until it can access the medium. However, unlike the CSMA/CA method, the DEVs using slotted aloha do not have to listen first to see if the medium is clear, they just transmit when they have counted down to their slot. Consider two DEVs trying at the same time to access a single association MCTA that appears in every superframe. If it is the first time that the DEVs have attempted to send the Association Request command, they each choose an integer between one and two, inclusive (i.e., they pick either one or two). If the DEV has picked one, then it can attempt to transmit in this superframe. If the DEV has picked two, it decrements it by one for each association MCTA that is in the superframe and transmits when it reaches one.



If there is more than one MCTA in a superframe and the DEV has multiple frames it needs to send to the PNC, the DEV could end up using more than one MCTA in the superframe. Once the DEV has sent a frame during an MCTA, it may begin the slotted-aloha backoff process as early as the next MCTA in that superframe. The backoff for the next frame starts with either 1 or 2 irrespective of the status of the backoff of the previous frame.

In the case of the two DEVs trying to associate, there is 50% chance that they will both choose the same number; thus, there is a 50% chance that they will collide in either the first or second superframe trying to send the Association Request command. If they do not collide, then both will be successful in sending the Association Request command within two superframes. If they collide in either the first or second superframes, then both DEVs will choose a new random number, this time between 1 and 4, inclusive, for their backoff. The range for the random number doubles with each failure until it reaches 256. The same process is used for open MCTAs where the DEVs are trying to send command frames, other than Association Request commands, to the PNC.

Another special type of CTA is the private CTA. This CTA is one where the source and destination DEVIDs are the same. This type of allocation is used to set aside time in the piconet for other types of communications, e.g., to allow time for dependent piconets or for the parent piconet. The dependent piconet is given a private CTA with its DEVID as the source and destination in the parent piconet's superframe. In the dependent piconet's superframe, the PNC provides a private

INTEL-1010
U.S. Patent 8,457,672

CTA with the PNCID as the source and destination in its own CTAP to set aside time for the parent piconet's operations.

Because the PNC is allowed to change the length of the CAP and location of the CTAs in every superframe, every DEV is required to hear the beacon before it transmits in the superframe. Thus, errors in the beacon reception will result in reduced throughput during the allocated CTA. To prevent the beacon FER from limiting the throughput of a stream, the standard allows a DEV to request a CTA, called a pseudo-static CTA, where the location of the allocation in the superframe is relatively constant between superframes. Because the PNC will not be moving the CTA, the DEV may continue to use this allocation for up to mMaxLostBeacons superframes, even if the DEV did not receive the beacon. The constant mMaxLostBeacons is set to a value of four in the standard. Thus, if the source DEV of a pseudo-static allocation only hears every other beacon from the PNC, it will not experience any reduction in throughput due to the lost beacons. If the source DEV had a regular CTA, it would have at least a 50% reduction in the throughput for the CTA.

The PNC is allowed to change the location of a pseudo-static CTA, but it needs t. take care to ensure that enough time is allowed for the source and destinations of the pseudo-static CTAs to see the change in the allocation before the time is given to another DEV. Early versions of the draft used directed frames, first to the destination and then to the source, to communicate the change in the allocation. The PNC was not allowed to change the allocation until it had received ACKs to the directed frames from both DEVs participating in the CTA. However, because this method could potentially take a long time, especially if the PNC was having trouble communicating with either DEV, the task group decided to just use the beacon to communicate the change. The current process is very simple: when the PNC wants to move the pseudo-static CTA, it simply updates the allocation in the beacon to the new time. The PNC will not allocate the old time for the CTA to any other DEV until at least mMaxLostBeacons has passed. When the source DEV sees the new allocation in the beacon, it begins using the new allocation for its transmissions. When the destination DEV sees the new allocation, it begins listening during the new time and should also listen during the old time as well, in case the source DEV has not yet seen the new allocation. The destination DEV will stop listening during the old allocation and will begin receiving only during the new allocation after one of the following occurs:

INTEL-1010
U.S. Patent 8,457,672

a) the destination DEV receives a frame from the source DEV in the new allocation;

b) mMaxLostBeacons superframes have passed; or

c) the PNC has allocated the channel time to another set of DEVs.

If the destination sees the change in the allocation either before or at the same time that the source sees the change, no frames will be lost due to the change in the allocation. However, if the source sees the change before the destination sees the change, the source will be sending frames during a time when the destination is not listening, and the frames will be lost, requiring retransmission if the source is requiring ACKs. The destination DEV will either hear the new allocation within mMaxLostBeacons or it will have to begin scanning the channel to find PNC again. If the implementer wants the destination DEV to have the maximum probability of correctly receiving frames intended for it, then any time the DEV fails to hear the beacon, it should listen to the entire superframe for frames addressed to it. In this case, the DEV is allowed to ACK the frames that it receives, even if it has not heard the beacon because it the responsibility of the sending DEV to ensure that there is enough time left for the destination to send the ACK.

Comparing the contention access methods

IEEE Std 802.15.3 has defined two contention access methods. The original reasoning in adding slotted aloha as an access method was that UWB PHYs would be unable to perform clear channel assessment (CCA) in an efficient manner. However, the work of the TG3a alternative PHY task group has indicated that CCA for UWB PHYs could be done in approximately the same time as for the 2.4 GHz PHY. The requirements for this are the same as for the 2.4 GHz PHY, i.e., the CCA is based on either detecting the beginning of the PHY preamble at sensitivity or detecting any large energy in the receiver bandwidth. Under these assumptions, CCA for a UWB PHY could be done fairly efficiently.

Another argument for keeping open association MCTAs in the standard was the claim that slotted aloha could be as efficient or more efficient than a short CAP.[16] Depending on the traffic assumptions and the size of the CAP, it can be shown that associations MCTAs in some cases can offer less collisions and shorter average

INTEL-1010
U.S. Patent 8,457,672

time to gain successful access to the channel. However, this analysis is based on the PNC allocating the minimum time that could be used to allow DEVs to associate in the piconet. The minimum usable CAP for association is one that allows time for just one Association Request command, the Imm-ACK, and two SIFSs. For the 2.4 GHz PHY at the 22 Mb/s base rate, the time required for the Association Request command, the Imm-ACK, and two SIFS is approximately 75 μs. If the PNC uses a CAP of this size, then the latency for an association can be longer than using a single association MCTA (which would be the same size) every superframe. However, if the PNC expands the CAP by even four backoff slots (e.g., 4 * 7.3 μs = 29.2 μs) to a total of 104 μs, then the CAP is more efficient on average than even two MCTAs, which would take up 150 μs.

For both contention methods, the PNC needs to determine the amount of time to be set aside for the DEVs to contend for the medium. In the case of the CAP, the PNC only needs to determine one number, the length of time to allocate for the CAP. One method to do this would be for the PNC to keep track of the ratio of the number of corrupted frames to the number of correctly received frames, regardless of the destination. Corrupted frames could be indicated by failed header check sequences (HCSs), frame check sequences (FCSs), or high receiver signal strength indication (RSSI) without a corresponding frame. Another method the PNC can use is to monitor the CAP to determine the length of time during which it was clear of transmissions. Regardless of the method used, if the PNC determines that the traffic during the CAP is high, it should increase the duration of the CAP. Conversely, if there is little traffic in the CAP and almost all of the frames are getting through, the PNC could reduce the duration of the CAP.

The task of determining the correct mixture of MCTAs is more complex for the PNC than is the task of determining the duration of the CAP. Not only is the PNC required to determine the number and frequency of MCTAs to allow for association, but it also has to determine the number and frequency of MCTAs to allow for other commands. The PNC must keep track of separate allocations for association and open MCTAs because Association Request commands may only be sent in an association MCTA and not in an open MCTA. Without the CAP, the PNC either needs to create uplink allocations to each of the DEVs in the piconet

[16] The slotted-aloha protocol has been shown to have a maximum theoretical throughput of only 36.8%. On the other hand, CSMA/CA protocols, such as the one used in IEEE 802.11 or in the CAP in IEEE 802.15.3, have been shown to have throughput in excess of 70%.

INTEL-1010
U.S. Patent 8,457,672

periodically to allow them to send commands to the PNC or it needs to allocate open MCTAs often for the same purpose. To do this, the PNC is required to keep track of the needs of the devices individually. If only a few devices are sending commands on a regular basis, then it would create uplink allocations to those DEVs while using an occasional open MCTA to allow the other DEVs a chance to communicate with the PNC.



A sub-rate allocation is a channel time allocation that occurs less often than every superframe. For example, the allocation may occur every other superframe (CTA Rate Factor = 2) or every 8th superframe (CTA Rate Factor = 8). A super-rate allocation, on the other hand, occurs one or more times in every superframe. For example, a super-rate allocation that occurs twice in every superframe has CTA Rate Factor of 2. The Channel Time Request command differentiates between a sub-rate and a super-rate request with the CTA Rate Type field.

Another issue with the use MCTAs to replace the CAP is that the allocation of MCTAs is a problem similar in complexity to managing sub-rate allocations. In the case of a sub-rate allocation, the requirements for the length and frequency of the allocation is known by the DEV requesting the channel time whereas for MCTAs, the PNC does not know the requirements of the DEV when it makes the allocation. Sub-rate allocations require the PNC to keep track of the frequency of the allocation and to make sure that the allocations are spread out among the superframes so that one superframe does not end up with more allocations than will fit in the allowed time. Once the PNC has determined the number of MCTAs to create and the timing for their allocation, it needs to find space in the superframe for the time allocation. In the absence of the CAP, all of the DEVs must explicitly request time to send asynchronous frames over the air. This increases the load on the PNC because it will have more CTRqs to handle as opposed to the CAP where the DEVs are directly responsible for getting time to send the data.

On the other hand, the slotted-aloha mechanism is easier, in some ways, to implement than the random access method in the CAP. Slotted aloha does not require that the DEV sense the medium before it transmits, which simplifies the transmit state machines. Accessing the medium during an open MCTA is similar to using an assigned CTA, when the DEV determines that it has access, it simply begins transmitting. Using the CAP requires that the MAC and PHY implementation are capable of efficiently implementing CCA, which may not

INTEL-1010
U.S. Patent 8,457,672

always be the case for future PHYs. In addition, the slotted-aloha method also enforces a slot time that is understood by all of the DEVs in the piconet. When the CAP is used, however, it is not unusual for the DEVs in the piconet to calculate different slot boundaries because they do not all have the same probability of hearing a frame. When the timing for the slot boundaries among the DEVs is different, it may lead to more collisions and retransmissions in the CAP.

Guard time

In between every CTA is at least one SIFS duration. However, because the clocks of the individual DEVs are only synchronized once per superframe and they may have differing accuracies, the PNC is also required to set aside guard time to ensure that transmissions from one CTA do not overlap in time with transmissions in other CTAs. The worst-case scenario is when the clock of the source DEV in CTA n is slow and the clock of the source DEV in CTA $n + 1$ is fast. This is illustrated in Figure 4–4 where the ideal CTA positions are based on the DEVs having the same clock accuracy.

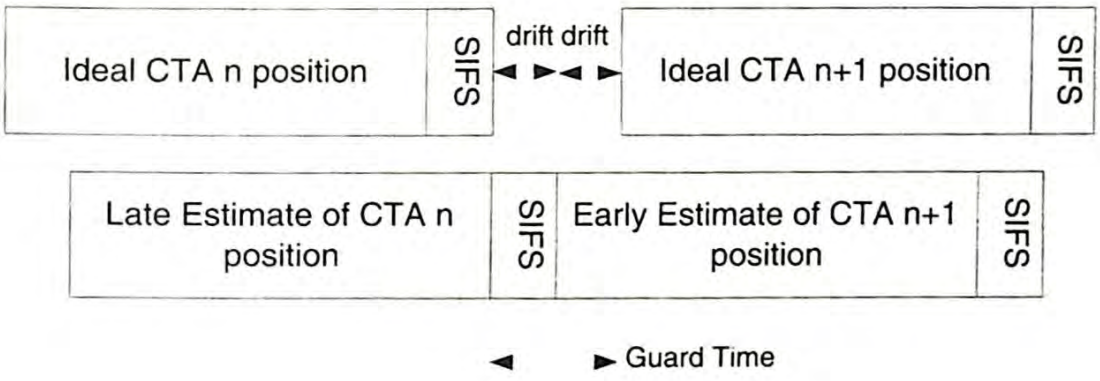


Figure 4–4: Illustration of worst-case CTA timing drift

INTEL-1010
U.S. Patent 8,457,672



A pseudo-static allocation can be moved by the PNC, but it needs to follow a special procedure (see "Channel time allocation period (CTAP)" on page 56). This procedure ensures that the DEV will still be able to use the old allocation for the maximum number of lost beacons that are allowed for this allocation.

The maximum drift that can occur is used to determine the guard time between successive CTAs in the CTAP and the time between the end of the last CTA and the start of beacon transmission. The drift is a function of the clock accuracy of the DEV and amount of time that has elapsed since the clocks were last synchronized. In 802.15.3, the synchronizing event is the beacon. For regular CTAs, the longest period of time between synchronizing events is the superframe duration. If the DEV does not hear the beacon, it may not transmit during the CTA and so it will not collide with other CTAs in the CTAP. In the case of pseudo-static CTAs, the DEVs may miss up to mMaxLostBeacons while still being allowed to transmit in the CTA. In this case, the maximum time between synchronizing events is equal to the product of mMaxLostBeacons and the superframe duration. The maximum drift for a single clock can be calculated with:

$$\text{MaxDrift(interval)} = \text{clock accuracy (ppm)} * 1{,}000{,}000 * \text{interval}$$

The clock accuracy for a DEV is PHY dependent and is ±25 ppm for the 2.4 GHz PHY. The interval is defined to be the time elapsed from the last timing event, i.e., the beacon.

With a maximum superframe duration of approximately 65 ms, the maximum drift between clocks for two DEVs in the piconet using regular CTAs is $(65\text{ ms})*2*(25*10^{-6}) = 3.3\ \mu s$. A summary of the maximum drift for regular and pseudo-static CTAs for various superframe durations is given in Table 4–1.

For typical superframe durations of less than 40 ms, the total drift for regular CTAs is pretty small, less than 2 µs, and so the PNC could add 2 µs to the 10 µs SIFS in between every CTA. However, if there are pseudo-static slots in the superframe, then choosing only a single guard time would almost double the amount of dead time on the air from 10 µs to 18 µs. In this case, it would be beneficial for the PNC to calculate the guard for each of the CTAs. Alternately, the PNC could calculate a finite number of guard times and then assign the appropriate guard time during the correct portion of the superframe. For example,

INTEL-1010
U.S. Patent 8,457,672

Table 4–1: Maximum clock drift for guard time calculation

Superframe duration	Max drift for regular CTA	Max drift for pseudo-static CTA
10 ms	0.5 µs	2.5 µs
20 ms	1 µs	5 µs
30 ms	1.5 µs	7.5 µs
40 ms	2 µs	10 µs
50 ms	2.5 µs	12.5 µs
65.535 ms	3.3 µs	16.4 µs

if the PNC was only going to use two guard times, it could assign CTAs in the first half of the super frame with one-half of the maximum drift added as guard time while CTAs in the second half of the superframe would have the maximum drift added as guard time. The GuardTime may be calculated as a function of the CTA intervals using:

$$\text{GuardTime(interval)} = [\text{MaxLostBeacons}(\text{CTA}_n) + \text{MaxLostBeacons}(\text{CTA}_{n+1}) + 2] * \text{MaxDrift(interval)}$$

The value of MaxLostBeacons for each CTA depends on whether the CTA is pseudo-static or dynamic. MaxLostBeacons is zero for a regular CTA and is equal to mMaxLostBeacons if the CTA is pseudo-static. The PNC can implement any method for allocating the guard time in the CTAP as long as it ensures that, even with the worst-case timing drifts, adjacent CTAs will always be separated by a SIFS.



The CTA blocks in the beacon do not indicate that a CTA is either dynamic or pseudo-static. The DEV that requested the CTA and the PNC both know that it is a pseudo-static allocation because this is contained in the Channel Time Request command. The destination of the allocation will be informed of the type of allocation via the CTA Status IE that is placed in the beacon when the allocation is first granted. If the DEV misses this announcement, it can request the CTA Status IE from the PNC using the Probe Request command.

INTEL-1010
U.S. Patent 8,457,672

THE ROLE OF THE PNC

The PNC is key to the operation of the piconet. The DEV that takes on the role of the PNC is responsible for starting and stopping the piconet, admitting DEVs into the piconet, allocating channel time, changing the piconet channel if conditions deteriorate, and keeping the piconet parameters current. If the DEV supports security (Chapter 6), then it is also takes on the responsibility as the key originator for the piconet.

Starting a piconet

The first step in starting a piconet is to scan the available PHY channels to determine if there are other piconets in range. The DEV is also required to rate the channels based on interference. If there are no other piconets or if the DEV does not want to join a piconet, it may decide to start its own piconet. If the DEV is PNC capable, it will choose the channel with the least interference to start its piconet. The DEV will then listen to the selected channel to ensure that it is still quiet, i.e., that another piconet has not started on the channel or switched into the channel. When that scan is complete, the DEV will begin sending its beacon. At this point, the PNC will assign itself two DEVIDs: the PNCID for PNC related traffic and one of the unreserved DEVIDs for traffic that is intended for its application layers.

Once the PNC has started sending a beacon, the piconet has been formed, even though there is only one member. At this point, other DEVs may find the beacon and decide to join the piconet. The PNC will continue sending the beacon until it either hands over responsibility for the piconet to another DEV or it shuts down the piconet.

Handing over control

As DEVs join the piconet using the association process (see "Association" on page 74), they pass their capabilities field to the PNC. One of the purposes of this field is to tell the current PNC about the PNC capabilities of the new DEV. For example, the capabilities field indicates if the DEV is capable of becoming the PNC, if it is battery-powered, the number of associated DEVs it would support as PNC, and so on. The PNC looks at these fields and then determines if it is either

INTEL-1010
U.S. Patent 8,457,672

required to or should hand over the control of the piconet to the new DEV. The order that these fields are compared in is given in Table 4–2.

As shown in Table 4–2, the PNC Des-Mode is the highest priority. In fact, it is the only entry in the table that requires the PNC to hand over its responsibility. The idea behind the PNC Des-Mode field is that there may be one DEV that the owner wants to be the PNC. In the case of the home, the person may want the DEV that is in the center of the house to be the PNC because that would maximize the range for the piconet. Users may select a DEV that they view as more capable, even if it is not judged to be so from the table above. The manufacturers may also set the PNC Des-Mode bit at the time of manufacture, e.g., in the receiver of a home theater system, due to the anticipated usage of the product. If the current PNC has the PNC Des-Mode field set to 1, then it does not have to hand over to a new DEV that does have the field set to 1. Otherwise, the PNC is forced to hand over responsibilities.

Table 4–2: Comparison order of fields for PNC handover

Order	Information	Note
1	PNC Des-Mode	If set to 1, the DEV prefers to be the PNC.
2	SEC	The DEV supports security as PNC is preferred.
3	PSRC	AC power as the DEV's power source is preferred.
4	Max associated DEVs	Higher number supported as PNC is preferred.
5	Max number CTRqBs	Higher number supported as PNC is preferred.
6	Transmitter power level	Higher transmitter power is preferred.
7	MAX PHY rate	Higher supported data rates is preferred.
8	DEV address	Tie breaker, no two DEVs have the same address.

If security is enabled in the piconet, then the PNC may not want to hand over to an new DEV if doing so would violate its security policies. In addition, if the PNC Des-Mode bit is not set, the PNC is not required to hand over to another DEV that rates higher based on Table 4–2. The standard only states that the DEV may hand over.

INTEL-1010
U.S. Patent 8,457,672



A DEV is allowed to be the PNC of two piconets simultaneosly. However, this is significantly more demanding than running a single piconet, and so the DEV may wish to refuse a hand-over if it is already the PNC of a dependent piconet. If the DEV refuses the hand-over, it is possible that the current PNC will shut down the piconet because it is turning off and cannot find another DEV to take over as PNC. In this case, the DEV that is a dependent PNC may want to take on the added complexity of dual-PNC implementation to keep the dependent piconet operational.

In addition to handing over to a new, more capable DEV that has joined the piconet, the PNC may wish to hand over control of the piconet if it is turning off. In either case, the PNC informs the DEV that it has been selected to be the new PNC by sending the PNC Handover Request command to that DEV. The complete frame exchange is illustrated in Figure 4–5. The PNC Handover Request command includes the number of DEVs currently in the piconet, the number of channel time request blocks (CTRqBs)[17] that it will send to the new PNC, and the number of PS Sets. The selected DEV is only allowed to refuse the hand-over request in two cases:

a) The DEV selected to be the new PNC is currently the PNC of a dependent piconet and it is unable or unwilling to simultaneously run both piconets.

b) The DEV and PNC are in a dependent piconet and the selected DEV is unable to join the parent piconet either as a DEV or as a neighbor PNC.

Otherwise, the DEV is required to accept the nomination to become the new PNC and so it will then prepare to become the new PNC of the piconet.

The current PNC needs to give the new PNC all of the information about the associated DEVs in the piconet, the channel time requests, the power mode information, and any relevant security information. To accomplish this, the first command sent by the PNC is the PNC Information command. This command is a list of all the DEVs that are associated in the piconet and includes information about each of the DEVs and an indication if the DEV is a member of the piconet or if has only associated. A DEV is a member of the piconet if it is associated with the piconet and, if required, has established a secure relationship with the PNC.

[17] CTRqBs are the individual channel time requests that are sent by DEVs to PNC to secure channel time (see "Asynchronous data" on page 87 and "Stream connections" on page 92).

INTEL-1010
U.S. Patent 8,457,672

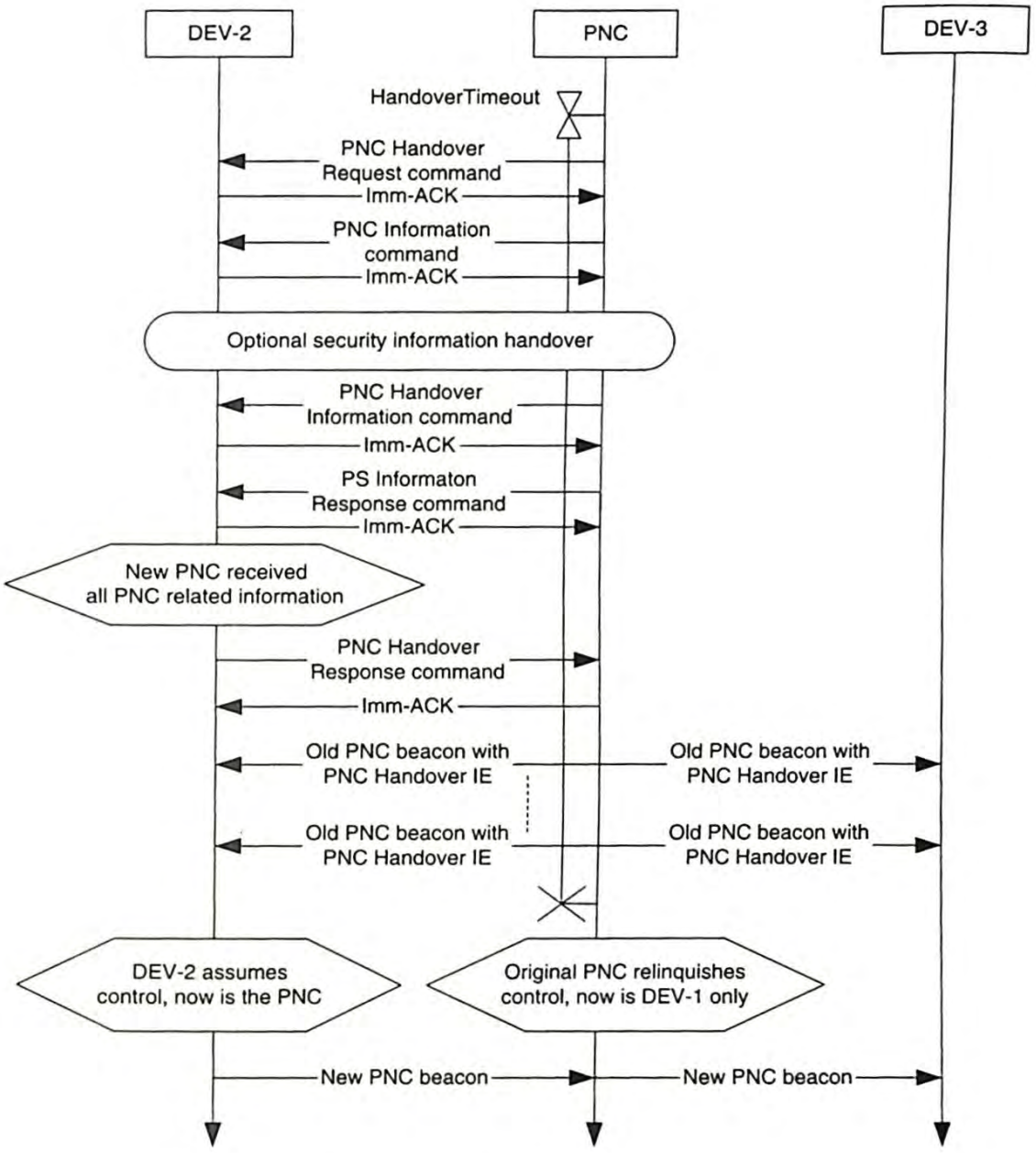


Figure 4–5: Frame exchange for PNC hand over

INTEL-1010
U.S. Patent 8,457,672

At this point, the new PNC may request the security information from the current PNC using the Security Information Request command. The Security Information commands that are sent in response to a Security Information Request command are intended to be used to handover authentication-related information about the DEVs that may or may not be currently members of the piconet. In particular, the Security Information command might contain a list of MAC addresses and associated information, e.g., cryptographic certificates, hashed keys, etc., of DEVs that should be allowed to join the piconet.

As an example, consider a DEV that has authenticated with the PNC but has left the piconet. The owner of the piconet and the DEV will have had to indicate to the PNC in some manner that the DEV should be allowed access to the piconet. The owner should not be required to re-authorize the DEV in the case of a PNC hand over. Instead this authorization should be passed in a trusted manner to the new PNC so that the DEV will automatically be re-authorized to become a member of the piconet as soon as it rejoins the piconet. The new PNC also needs this information because it will be required to re-authenticate with all of the DEVs in the piconet so that it can establish new management keys. Again, because some of this information normally requires user intervention in the authentication process, securely passing this information during the hand over process allows the hand-over to happen seamlessly. The standard does not define the content or format of this information because it depends on the authentication method that was used to allow DEVs to become members of the piconet. Authentication methods are outside of the scope of a MAC and PHY standard and so they are not specified in the 802.15.3 standard. A more complete description of the use of the Security Information command is given in "Security information" on page 199.

The next command that is sent is the PNC Handover Information command that contains all of the isochronous CTRqBs that the current PNC has received. Asynchronous requests are not passed, because some of them may expire before the hand-over is complete. The new PNC will use these requests to allocate channel time once it has taken over as PNC. The new PNC is also allowed to use the actual CTAs from the previous beacons to determine the CTAs for the beacons it will send once it has taken over. The last command that the old PNC sends to the new PNC is the PS Set Information Response command. This command contains the status of all of the DEVs in PS mode as well as all of the information about the PS sets (see "DSPS mode" on page 118). If there are no DEVs in PS mode, then

INTEL-1010
U.S. Patent 8,457,672