Tampereen teknillinen korkeakoulu
Julkaisuja 386

Tampere University of Technology
Publications 386

Marko Hännikäinen

Design of Quality of Service Support for Wireless Local Area Networks

Tampere 2002

Tampereen teknillinen korkeakoulu
Julkaisuja 386

Tampere University of Technology
Publications 386

Marko Hännikäinen

Design of Quality of Service Support for Wireless Local Area Networks

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB103, at Tampere University of Technology, on the 1st of November 2002, at 12 o'clock noon.

Tampere 2002

ISBN 952-15-0896-5 (printed)
ISBN 952-15-1566-X (PDF)
ISSN 0356-4940

TTKK- PAINO, Tampere 2002

Abstract

Wireless Local Area Networks (WLAN) operating on unlicensed radio frequency bands are emerging in a number of application areas. They both extend and replace traditional wired LANs. Wireless Personal Area Network (WPAN) is a type of WLAN that targets low cost of technology, small size, and low power consumption.

As the multitude of application types utilising data networks is increasing, new requirements are placed on both wired LAN and WLAN services. Many projected applications require time bounded data transfer. Consequently, Quality of Service (QoS) support is required in order to enable different applications to operate.

This thesis presents a design of Quality of Service (QoS) support for WLANs. The work concentrates on a WLAN called Tampere University of Technology Wireless Local Area Network (TUTWLAN). TUTWLAN is being developed for research purposes, not destined to meet current or emerging WLAN standards. This enables to more freely experience with different functional alternatives.

The QoS support requirements and justification for the TUTWLAN design are drawn from the interoperability requirements with higher layer protocols and peer wired LAN and WLAN technologies. The main entity for QoS support in TUTWLAN is the Medium Access Control (MAC) protocol called TUTMAC. Other central components are a TUTWLAN Access Point (AP) and a transport layer protocol for a wireless video demonstrator application. The work also contains a hardware demonstrator platform and several other software components and applications for the TUTWLAN implementation.

TUTWLAN has a centrally controlled topology, as data transfer services are enabled by changing control and management messages between a TUTWLAN base station and portable terminals. The channel access is based on dynamic reservation Time Division Multiple Access (TDMA). Interconnection to backbone wired LAN is implemented by the TUTWLAN AP module located in a TUTWLAN base station. AP adapts the QoS signalling between different connected LANs and supports QoS in data forwarding.

The TUTMAC protocol is implemented for the MAC processor of the platform using the Specification and Description Language (SDL). SDL has been found to be a suitable specification, simulation, and implementation tool for the TUTMAC protocol. In addition, programmable logic is needed for accelerating the most time critical TUTMAC functions.

Preface

The research work for this Thesis has been carried out during the years 1997 and 2002 at Tampere University of Technology. The work begun at the Signal Processing Laboratory and was completed in the Institute of Digital and Computers Systems that was separated from the Signal Processing Laboratory at the beginning of the year 2000.

I wish to express my gratitude to my supervisor Prof. Timo D. Hämäläinen for his persistent motivation, support, and farsighted guidance from the beginning of my research work.

The research project was started under the supervision of Prof. Jukka Saarinen, who I wish to thank for advice and guidance. Also, thanks to Prof. Jarno Knuutila and Markku Niemi M.Sc who have shared their expertise during my research work. I would like to thank the reviewers of my Thesis, Prof. Petri Mähönen and Dr. Jouni Mikkonen, for their constructive comments on the manuscript.

I am also grateful to all who have been a part of the TUTWLAN research team. I have enjoyed working with skilful colleagues, and friends, in an inspiring atmosphere since the days in FC117.

This Thesis was financially supported by the Tampere Graduate School in Information Science and Engineering (TISE), the National Technology Agency of Finland (TEKES), Foundation of Technology (TES), and the Research and Training Foundation of Sonera Corporation. Their support is appreciatively acknowledged.

Finally, thanks to Jaana for her love and understanding.

Tampere, September 30, 2002

Marko Hännikäinen

Contents

List of Publications

This Thesis consists of an introduction section and twelve publications [P1]-[P12]. Supplementary publications [P13]-[P22] are not included into this Thesis but they are closely related to its contents and therefore separated from the list of references.

[P1] Hännikäinen M., Hämäläinen T., Niemi M., Saarinen J., "Trends in Personal Wireless Data Communications", Computer Communications, Volume 25, Issue 1, pp. 84-99, January 2002.

[P2] Hännikäinen M., Rantanen T., Ruotsalainen J., Niemi M., Hämäläinen T., Saarinen J., "Coexistence of Bluetooth and Wireless LANs", IEEE International Conference on Telecommunications (ICT 2001), Volume 1, pp. 117-124, Bucharest, Romania, June 4-7, 2001.

[P3] Hännikäinen M., Niemi M., Hämäläinen T., "Performance of the Ad-hoc IEEE 802.11b Wireless LAN", International Conference on Telecommunications (ICT 2002), Volume 1, pp. 938-945, Beijing, China, June 23-26, 2002.

[P4] Hännikäinen M., Knuutila J., Letonsaari A., Hämäläinen T., Jokela J., Ala-Laurila J., Saarinen J., "TUTMAC: A Medium Access Control Protocol for a New Multimedia Wireless Local Area Network", IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC 1998), Volume 2, pp. 592-596, Boston, USA, September 8-11, 1998.

[P5] Kari M., Hännikäinen M., Hämäläinen T., Knuutila J., Saarinen J., "Configurable Platform for a Wireless Multimedia Local Area Network", International Workshop on Mobile Multimedia Communications (MoMuC 1998), pp. 301-306, Berlin, Germany, October 12-14, 1998.

[P6] Tikkanen K., Hännikäinen M., Hämäläinen T., Saarinen J., "Advanced Prototype Platform for a Wireless Multimedia Local Area Network", The European Signal Processing Conference (EUSIPCO 2000), Volume 4, pp. 2309-2312, Tampere, Finland, September 5-8, 2000.

[P7] Saari T., Hännikäinen M., Hämäläinen T., "Hardware Acceleration of Wireless LAN MAC Functions", IEEE International Workshop on Design and Diagnostics of Electronic Circuits and Systems (DDECS 2002), pp. 398-401, Brno, Czech Republic, April 17-19, 2002.

[P8] Hännikäinen M., Vanhatupa T., Lemiläinen J., Hämäläinen T., Saarinen J., "Windows NT Software Design and Implementation for a Wireless LAN Base Station", ACM International Workshop on Wireless Mobile Multimedia (WoWMoM 1999), pp. 2-9, Seattle, USA, August 20, 1999.

[P9] Hännikäinen M., Knuutila J., Hämäläinen T., Saarinen J., “Using SDL for Implementing a Wireless Medium Access Control Protocol”, IEEE International Symposium on Multimedia Software Engineering (MSE 2000), pp. 229-236, Taipei, Taiwan, December 11-13, 2000.

[P10] Kuorilehto M., Hännikäinen M., Niemi M., Hämäläinen T., Saarinen J., “Design for a Wireless LAN Access Point Driver”, IEEE International Conference on Telecommunications (ICT 2001), Volume 3, pp. 167-173, Bucharest, Romania, June 4-7, 2001.

[P11] Hännikäinen M., Lehtoranta O., Kuorilehto M., Suhonen J., Niemi M., Hämäläinen T., “Architecture of a Wireless Video Transfer Demonstrator”, International Zurich Seminar on Broadband Communications (IZS 2002), pp. 53-1 - 53-6, Zurich, Switzerland, February 19-21, 2002.

[P12] Suhonen J., Hännikäinen M., Lehtoranta O., Kuorilehto M., Niemi M., Hämäläinen T., “Video Transfer Control Protocol for a Wireless Video Demonstrator", IEEE International Conference on Information Technology: Coding and Computing (ITCC 2002), pp. 462-467, Las Vegas, USA, April 8-10, 2002.

Supplementary Publications

[P13] Hännikäinen M., Vanhatupa T., Lemiläinen J., Hämäläinen T., Saarinen J., “Design and Implementation of a Wireless LAN Interface Card Driver in Windows NT”, International Conference on Telecommunications (ICT 1999), Volume 2, pp. 347-351, Cheju, Korea, June 15-18, 1999.

[P14] Hännikäinen M., Vanhatupa T., Lemiläinen J., Hämäläinen T., Saarinen J., “Architecture for a Windows NT Wireless LAN Multimedia Terminal”, IEEE International Workshop on Multimedia Signal Processing (MMSP 1999), pp. 535-540, Copenhagen, Denmark, September 13-15, 1999.

[P15] Hännikäinen M., Knuutila J., Takko A., Hämäläinen T., Saarinen J., “Automatic C-Code Generation from SDL for a Wireless MAC Protocol", IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2000), Volume 1, pp. 533-538, Honolulu, Hawaii, USA, November 5-8, 2000.

[P16] Hännikäinen M., Takko A., Knuutila J., Hämäläinen T., Saarinen J., “SDL-to-C Conversion for Implementing Embedded WLAN Protocols”, IEEE International Conference on Industrial Electronics, Control, and Instrumentation (IECON 2000), Volume 4, pp. 2455-2460, Nagoya, Japan, October 22-28, 2000.

[P17] Takko A., Hännikäinen M., Knuutila J., Hämäläinen T., Saarinen J., "Embedding SDL Implemented Protocols into DSP", International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES 2000), pp. 48-56, San Jose, USA, November 17-18, 2000.

[P18] Rantanen T., Hännikäinen M., Niemi M., Hämäläinen T., Saarinen J., "Design of a Quality of Service Management System for Wireless Local Access Networks", IEEE International Conference on Telecommunications (ICT 2001), Volume 3, pp. 107-114, Bucharest, Romania, June 4-7, 2001.

[P19] Laitinen A., Hännikäinen M., Hämäläinen T., "Using SDL as a Tool for System Simulations", IEEE International Symposium on Circuits and Systems (ISCAS 2002), Volume 5, pp. 17-20, Phoenix, USA, May 26-29, 2002.

[P20] Hännikäinen M., Laitinen A, Rekonius J., Hämäläinen T., "Implementing Demonstrators for SDL Systems", International Conference on Telecommunications (ICT 2002), Volume 3, pp. 169-174, Beijing, China, June 23-26, 2002.

[P21] Ruotsalainen J., Hännikäinen M., Suhonen J., Lehtoranta O., Hämäläinen T., "Video over Bluetooth", International Conference on Telecommunications (ICT 2002), Volume 2, pp. 709-714, Beijing, China, June 23-26, 2002.

[P22] Kuorilehto M., Hännikäinen M., Niemi M., Hämäläinen T., "Implementation of Wireless LAN Access Point with Quality of Service Support", IEEE International Conference on Industrial Electronics, Control, and Instrumentation (IECON 2002), Sevilla, Spain, November 5-8, 2002, accepted, 6 pages.

List of Abbreviations

ACK	Acknowledgement
ACL	Asynchronous Connection Less
AIFS	Arbitration Inter Frame Space
AP	Access Point
ARQ	Automatic Repeat reQuest
ATM	Asynchronous Transfer Mode
b	bucket depth
B	Byte (8 bits)
BRAN	Broadband Radio Access Networks
CAC	Channel Access Control
CC	Controlled Contention
CCA	Clear Channel Assessment
CCI	Controlled Contention Interval
CCK	Complementary Code Keying
CCOP	Controlled Contention OPportunity
CEPT	Conference of Postal and Telecommunications Administrations
CFP	Contention Free Period
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CTS	Clear To Send
CW	Contention Window
DCF	Distributed Coordination Function
DiffServ	Differentiated Services
DIFS	Distributed Coordination Function Inter Frame Space
DLC	Data Link Control
DSSS	Direct Sequence Spread Spectrum
EC	Error Control
EIFS	Extended Inter Frame Space
EIRP	Effective Isotropic Radiated Power
ETSI	European Telecommunications Standards Institute
EY-NPMA	Elimination Yield - Non-Pre-emptive priority Multiple Access
FCC	Federal Communication Commission
FCS	Frame Check Sequence
FDD	Frequency Division Duplex

FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FHSS	Frequency Hopping Spread Spectrum
FPGA	Field Programmable Gate Array
GSM	Global System for Mobile telecommunications
HCF	Hybrid Coordination Function
HCI	Host Controller Interface
HEC	Header Error Check
HIPERLAN	HIgh PErformance Radio Local Area Network
ID	IDentification
IDEA	International Data Encryption Algorithm
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IFS	Inter Frame Space
IntServ	Integrated Services
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISM	Industrial, Scientific, and Medical
IWEP	Improved Wireless Equivalent Privacy
L2CAP	Logical Link Control and Adaptation Protocol
LAN	Local Area Network
LLC	Logical Link Control
LMP	Link Manager Protocol
MAC	Medium Access Control
MIB	Management Information Base
MPDU	MAC Protocol Data Unit
MSDU	MAC Service Data Unit
NRL	Normalised Residual Lifetime
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open Systems Interconnection
PC	Personal Computer
PCF	Point Coordination Function
PCI	Peripheral Component Interconnect
PDA	Personal Digital Assistant
PDU	Protocol Data Unit
PEP	Performance Enhancing Proxy
PF	Persistence Factor
PHB	Per-Hop Behaviour

PIFS	Point Coordination Function Inter Frame Space
QoS	Quality of Service
r	number of backoff slots (in 802.3 MAC)
R	token rate
RED	Random Early Detection
RF	Radio Frequency
RFID	Radio Frequency IDentification
RLC	Radio Link Control
RR	Reservation Request
RSpec	Service Request Specification
RSVP	Resource Reservation Protocol
RTP	Real Time Transport Protocol
RTS	Request To Send
SAP	Service Access Point
SBM	Subnet Bandwidth Manager
SCO	Synchronous Connection Oriented
SDL	Specification and Description Language
SDP	Service Discovery Protocol
SIFS	Short Inter Frame Space
SIG	Special Interest Group
SRD	Short Range Device
TCP	Transmission Control Protocol
TCS-BIN	Telephony Control Specification Binary
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
ToS	Type of Service
TSpec	Traffic Specification
TUTMAC	Tampere University of Technology Medium Access Control
TUTWLAN	Tampere University of Technology Wireless Local Area Network
TXOP	Transmission Opportunity
UDP	User Datagram Protocol
UI	User Interface
UMTS	Universal Mobile Telecommunications Systems
U-NII	Unlicensed National Information Infrastructure
VCP	Video Control Protocol
VHDL	Very high-speed integrated circuit Hardware Description Language
VLAN	Virtual Local Area Network
WEP	Wired Equivalent Privacy
WLAN	Wireless Local Area Network

WMAN	Wireless Metropolitan Area Network
WPAN	Wireless Personal Area Network
WWAN	Wireless Wide Area Network

1. Introduction

Mobile voice communication has been the driving force behind the rapid growth of wireless networking during recent years. Cellular telephone networks have advanced from analogue to digital, which has increased both voice quality and network capacity, and are now moving towards higher data rates for enabling new applications. Following this general acceptance of the wireless voice, a variety of other wireless data communications technologies targeted at different application areas is emerging [P1].

A practical approach to classify technologies is to compare network coverage and nominal data rates. The classification used throughout this Thesis is depicted in Figure 1 that illustrates the relative placement of different technological categories without destining to exact performance figures.

Wireless Wide Area Network (WWAN) corresponds to current digital cellular telephone networks and their future extensions, such as Global System for Mobile telecommunications (GSM) and Universal Mobile Telecommunications Systems (UMTS). Their strength is the wide geographical coverage. Wireless Metropolitan Area Networks (WMAN) are emerging for data transfer services in urban areas. They are generally limited to fixed point-to-point or point to multipoint connections with limited mobility. Therefore, WMAN can be seen as a wireless alternative for a digital subscriber line [51][47].

Wireless Local Area Networks (WLAN) have originally been designed to replace and extend legacy computer LANs. WLAN enables a quick network installation and easy topology changes. Therefore, WLAN can be established on purely temporary basis, for

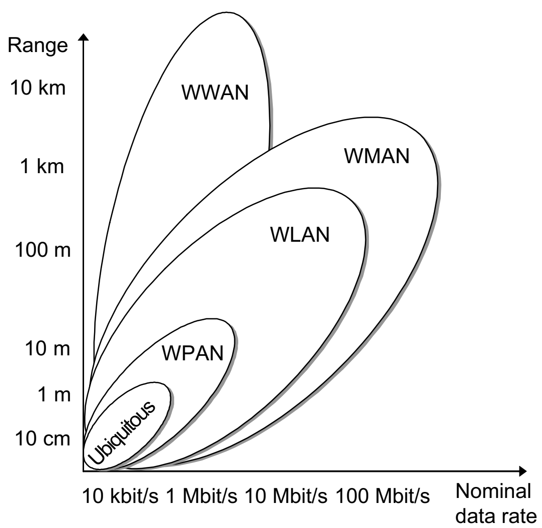


Figure 1. Classification of wireless communication technologies.

example during a meeting to exchange documents or for synchronising the calendar applications of two Personal Digital Assistants (PDA). WLAN also facilitates data networking in places without an existing wired LAN infrastructure, such as in old buildings and homes. As WLAN enables continuous access to the Internet or company database while moving, warehousing, education, and health care have been the first areas to adopt the new technology [117][49].

WLANs, as LANs in general, are moving towards supporting a multitude of services, for example to cover broadband wireless Internet access in hot spot areas [5], as well as short range serial cable replacement [46]. Consequently, the separation between WLAN and a Wireless Personal Area Network (WPAN) is not distinct. WPAN technology is targeted at connecting different personal devices, such as a mobile phone, laptop computer, and PDA. The WPAN technology currently differs from WLAN mainly by its non-functional requirements, such as cost and power consumption. Generally, WPAN has a smaller operational area, lower data rate, and fewer terminals per network compared to WLAN.

The wireless technology class in Figure 1 having the smallest coverage is named ubiquitous [116]. Ubiquitous technologies are projected for various control and automation applications, but generally not for personal communications. Therefore, very small size, minimal power consumption, and low cost are required, while also data rates can be significantly lower compared to WPAN technologies. On the contrary, the number of nodes per network can be extensively higher. Radio Frequency Identification (RFID) technologies at least partly meets these requirements [32]. The emerging low complexity WPAN technologies are also extending to meet the requirements for very short range ubiquitous networking [50].

The number of available wireless technologies allows the choosing of a suitable technology according to specific requirements. At the same time, interoperability requirements are emphasised, as same applications may need to operate over heterogeneous wireless technologies. Thus, an important element for a wireless technology is to provide interoperable data transfer services.

The data transfer services can be characterised by the term Quality of Service (QoS). As traditional LANs have been targeted at file transfer, applications have expected that the service provided by a network technology is reliable and offers a sufficient throughput. More demanding real-time applications have extended the QoS requirements to include new parameters, such as delay, delay variance, and error rate. Wireless data transfer is further adding new requirements such as security and mobility [107][31][61].

The QoS support means that a network has mechanisms to fulfil the placed data transfer requirements. Since the requirements may vary, a network technology must differentiate the provided transfer service in each case. To realise this, the technologies must be able to signal the QoS requirements and have QoS support functions. QoS support itself does not create new bandwidth, but enables the managed use of existing bandwidth and thus a wider range of different applications to operate.

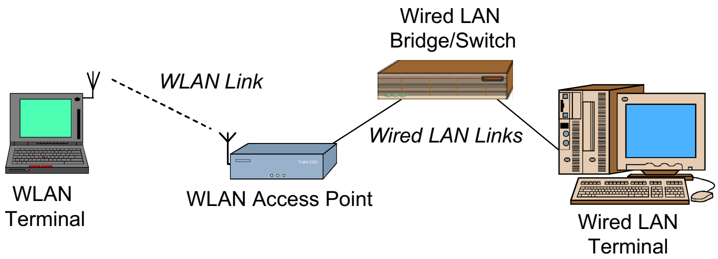


Figure 2. Reference LAN topology consisting of WLAN and wired LAN.

1.1. Scope and Objectives of the Thesis

Figure 2 presents the main network nodes addressed in this Thesis. WLAN is the central technology, but because of the targeted interoperability, wired LAN is also discussed. The wired LAN side of the network topology is constructed with a wired LAN terminal and a bridge (i.e. a switch), which interconnects different LAN segments.

The WLAN nodes of Figure 2 are a WLAN terminal and a WLAN Access Point (AP). WLAN AP also contains the necessary LAN bridge functionality for forwarding traffic between the WLAN and wired LAN segments. In a centrally controlled WLAN topology, AP usually contains the control and management functions for the WLAN segment. In an ad-hoc network topology, a central controller is not available and data transfers occur directly between WLAN terminals. Both the centrally controlled and ad-hoc WLAN topologies are discussed in this Thesis.

The corresponding protocol architecture with relation to the Open Systems Interconnection (OSI) reference model is shown in Figure 3 [70][109]. The presented protocol architecture of a WLAN terminal follows the Transmission Control Protocol/Internet Protocol (TCP/IP) suite from the network layer upwards. In this Thesis, the User Datagram Protocol (UDP) is also a central transport protocol of the TCP/IP suite.

WLAN related layers in Figure 3 are the physical layer and the data link layer of the OSI model. The data link layer has been divided by the Institute of Electrical and Electronics Engineers (IEEE) in its 802 LAN/MAN standards committee into Medium Access Control (MAC) and Logical Link Control (LLC) sub layers [58]. In this Thesis, a separate LLC layer is not addressed. The MAC layer, on the other hand, is the key protocol layer for managing and controlling WLAN and therefore the data transfer service. WLAN physical layer technologies are expected to operate on Radio Frequencies (RF) in this Thesis.

The number of standard WLAN technologies is rapidly increasing targeting at support for different application types. Research work for developing new standards and for improving existing standard technologies is needed. Research work based firmly on standards also narrows the gap between research and its commercial utilisation.

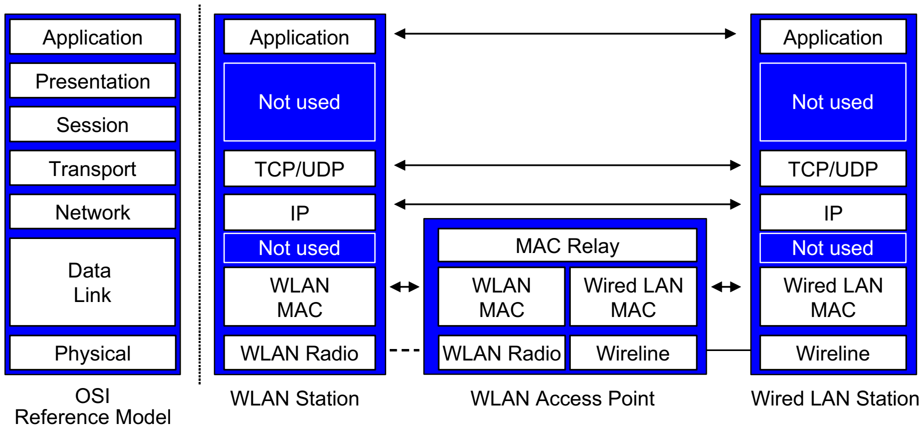


Figure 3. WLAN/wired LAN protocol architecture of this Thesis with the OSI reference model.

However, research based on available standards may also be too restricted. While some technology improvements such as higher data rate radio layers are promptly integrated to WLAN standards, many of the basic design choices remain the same. This can lead to non-optimal solutions. Consequently, standard-free research is important for developing and experimenting with new solutions that are better suited for specific requirements. In turn, these solutions can then be utilised in the standardisation work.

The main objective of this thesis is to present a design of Quality of Service (QoS) support for WLANs. This is carried out with the help of a WLAN called Tampere University of Technology Wireless Local Area Network (TUTWLAN). TUTWLAN is being developed for research purposes, not destined to meet current WLAN standards. This enables to more freely experience with different functional alternatives.

The work on TUTWLAN was started in 1997. At the beginning of the work, the main reference was a draft version of the IEEE 802.11 standard. As will be discussed in Chapter 4, the support functionality for QoS was generally missing from the standard, which motivated the emphasis of the QoS support in TUTWLAN design. As standard technologies have more recently proceeded towards QoS support, the standard-free TUTWLAN research has produced comparable results. The QoS is not separately specified for TUTWLAN, but the QoS support requirements and justification for the design are drawn from the interoperability requirements with higher layer protocols and peer wired LAN and WLAN technologies.

The basic metrics defining the QoS support are MAC layer performance parameters, such as throughput for an application data stream, transfer delay, and delay variance [84]. These parameters are important for time-critical applications needing a dedicated throughput, while controllable error protection and security against eavesdropping are examples of other QoS parameters addressed in TUTWLAN.

The central entities of TUTWLAN are a MAC protocol called TUTMAC, a TUTWLAN access point, and a transport layer protocol for a video demonstrator application. The main functions for realising the QoS support are the signalling of QoS requirements, centrally controlled network topology of TUTWLAN, dynamic reservation based MAC protocol, and functions for protecting and adapting data for wireless transfer. The work also contains a hardware demonstrator platform and several other software components and applications for the TUTWLAN implementation. Video is seen as a demanding test application for validating the QoS support of TUTWLAN.

For the TUTWLAN QoS support presented in this Thesis, the emphasis is on the design. The work is also a base for further research work, for which TUTWLAN presents a platform. Especially, further development of queue management and base station scheduling algorithms is needed. These are outside the scope of this Thesis.

TUTWLAN experiments and results can be generalised and applied to other WLAN designs as well. Therefore, the statement of this thesis is that the *results presented in this Thesis can be used for realising the QoS support for WLANs.*

1.2. Outline of the Thesis

This Thesis consists of an introduction part and twelve publications. The publications embody the main results of the Thesis.

The introduction part provides the technological background of the Thesis. It presents the most significant existing and emerging standard WLAN and WPAN technologies concentrating on their QoS support. The introduction part also gives an introduction to TUTWLAN. The part is organised in the following way.

Chapter 2 starts with an introduction to WLANs. The chapter introduces protocol architectures and the main characteristics of significant existing and emerging standard WLAN and WPAN technologies. The technologies are further discussed in Chapter 4.

Chapter 3 discusses the QoS concept used in this Thesis and introduces the basic tools for supporting QoS. The chapter targets placing a WLAN as a part of the existing fixed network infrastructure. The QoS support in IP layer and in parallel wired LAN technologies are discussed. The purpose of the chapter is also to present the interoperability requirements placed on the TUTWLAN QoS support by wired networks.

In Chapter 4, WLAN service requirements resulting from the characteristics of the wireless medium are discussed. The QoS support of the introduced WLAN and WPAN technologies are presented in detail. These technologies also originate the QoS support requirements placed on TUTWLAN and provide reference functional designs for realising the support.

Chapter 5 concentrates on TUTWLAN. The chapter starts by presenting the design requirements for TUTWLAN and proceeds by the functional design for the TUTMAC protocol. The TUTMAC QoS support architecture is next discussed. The chapter introduces the design and QoS support functionality of TUTWLAN AP and the video transfer application. The different components of TUTWLAN and their implementation status are clarified.

In Chapter 6, the summary of the publications is given and the contribution of the author clarified. Chapter 7 gives the conclusions of the Thesis.

2. WLAN Standards

Standardisation enables compatible products for users. As such, the development of standards and standard-like specifications has been a driving force behind the WLAN markets. WLAN standardisation is a recent activity of the major standardisation and specification bodies, such as IEEE and the European Telecommunications Standards Institute (ETSI). The WLAN standardisation work in the IEEE 802.11 working group began in 1990 while the first version of the standard emerged in 1997. WLAN products based on draft versions of the standard appeared already before that.

In WLAN standardisation, the development in general is to update the existing standards with new functionality and capacity improvements. For another, standardisation bodies are extending the selection of standard technologies. New standards are developed for meeting the specific requirements of different application areas [43][46][47][48][52][75]. For example, in IEEE the WLAN standardisation work has enlarged to cover WPAN and ubiquitous network technologies.

WLAN standardisation and standard technologies are a central topic also for university research work. As WLAN technologies are being actively developed by the standardisation bodies, much of the current university research work is also concentrating on improving the existing WLAN standards.

2.1. WLAN Topologies

LANs can be generally distinguished by their restricted size, transmission technology, and network topology. LAN typically covers a geographically limited area from a few meters to a few kilometres, such as home, office, and university campus. LAN technologies usually rely on wired connections between network devices, which provides a high capacity (range of Gbit/s) and low delay (range of few microseconds) transmission medium [112][109].

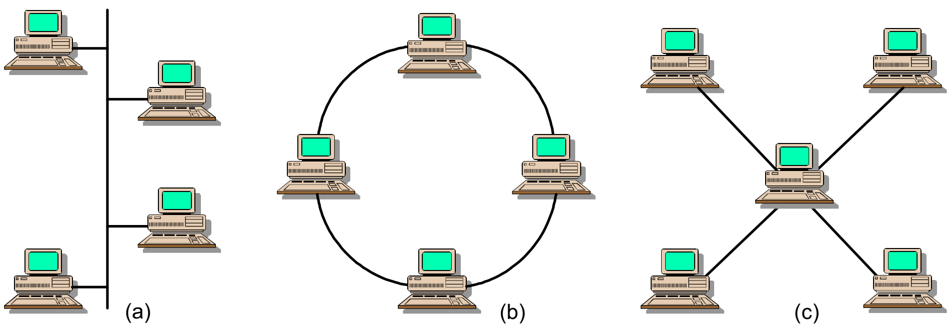


Figure 4. Basic wired LAN topologies: shared bus (a), ring (b), and star (c).

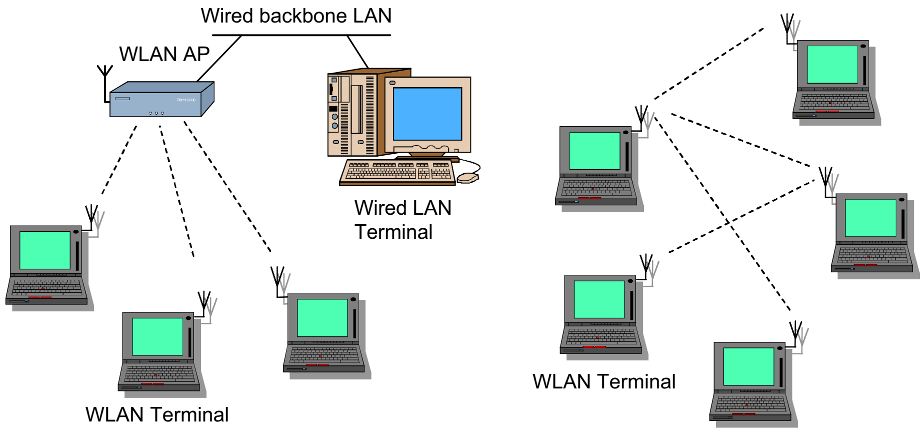


Figure 5. Basic WLAN topologies: infrastructure (left) and independent (right).

The physical topology of a LAN describes how the LAN terminals are connected with each other. Three common wired LAN topologies, a shared bus, a ring, and a star, are depicted in Figure 4. The functional topology created by the access control protocol of the LAN technology can differ from the physical topology. Due to wiring, the wired LAN capacity, coverage, security, and topology can be managed more accurately compared to WLANs.

The transmission medium of a WLAN is physically a shared broadcast radio channel. However, according to the utilised control scheme, two basic WLAN topologies can be distinguished: infrastructure based topology and independent (ad-hoc) topology. These are presented in Figure 5. In the infrastructure topology, a central WLAN AP provides the bridging service between wireless and wired LANs. In addition, the management and control of WLAN are commonly integrated into AP. AP can thus centrally control the transmissions of WLAN terminals and forward frames between them. The MAC protocol of a centrally controlled WLAN is thus an asymmetric shared function between AP and a terminal [117][80][92][93].

In the ad-hoc topology all terminals independently, and equally in most cases, access the wireless medium. Thus, control and management of an independent WLAN topology are distributed and the utilised MAC protocol operates symmetrically in all terminals. Data transfer takes place directly between terminals on a peer-to-peer basis. Also in this topology, an access to outside network resources can be available with a bridge device.

2.2. WLAN Radio Bands

Potential operational frequencies for privately owned WLANs are radio bands that do not require a specific licence, special permissions, or carry licence fees. The existing WLAN and WPAN technologies mostly utilise the Industrial, Scientific, and Medical (ISM) bands. In Europe, ISM bands are part of the frequencies allocated for Short Range Devices (SRD) by the European Conference of Postal and Telecommunications Administrations (CEPT) [30]. Some of the ISM bands are also available globally, but

Table 1. Potential WLAN frequency bands in Europe.

Band	Frequencies	Maximum power (EIRP)	Existing/expected technologies and applications
433 MHz ISM	433.05 - 434.79 MHz	1 mW	RFID technologies, baby monitors, cordless headphones, walkie-talkie phones
862 MHz SRD	862 - 870 MHz (with several sub band divisions)	5 mW - 500 mW	Cordless audio devices, radio microphones, general purpose telemetry, general purpose alarms
2.4 GHz ISM	2.4000 - 2.4835 GHz	100 mW	WLANs, WPANs
5 GHz HIPERLAN	5.150-5.350 GHz	200 mW	WLAN (indoor only)
	5.470-5.725 GHz	1W	WLAN, WMAN
5 GHz ISM	5.725 5.875 MHz	25 mW	WLAN, WPAN
17 GHz HIPERLAN	17.1-17.3 GHz	100 mW	WMAN, WPAN
60 GHz and higher ISM bands	61 - 61.5 GHz 122 - 123 GHz 244 - 246 GHz	N/A	Future development

national and regional exceptions to the use of them may apply [32]. The potential WPAN, WLAN, and WMAN frequency bands in Europe with Effective Isotropic Radiated Power (EIRP) limits are summarised in Table 1. For WLANs and WPANs, the bands around the 2.4 GHz and 5 GHz area are currently the most important.

Due to license free utilisation, radio communication technologies operating within ISM bands must tolerate potential interference [29]. Especially, the 2.4 GHz band already contains different WLAN and WPAN technologies. In addition, microwave ovens, military radar, amateur radio, garage door remote control applications, and cordless phones (in USA) utilise this band [53]. Thus, the coexistence with other systems operating on the same frequency band is an increasingly important design requirement for WLANs [P2].

The 5 GHz band encounters interference from radar systems. Other communication technologies on this band are mobile satellite systems and radiolocation systems [30]. The 5 GHz band is currently available for unlicensed devices in the United States, where the Federal Communication Commission (FCC) has allocated the Unlicensed National Information Infrastructure (U-NII) band. In Europe, the 5 GHz band is mostly allocated for HIgh PErformance Radio Local Area Network (HIPERLAN) type of devices and the transmission power levels are limited [P1].

The potential frequency bands over 5 GHz are currently around 17 GHz and 60 GHz. Communication on these frequencies generally requires a line of sight between sender and receiver, which makes the frequencies more suitable for WMAN and WPAN technologies [108].

2.3. WLAN Technologies

This section presents briefly the protocol architectures and utilised network topologies of the most significant existing and emerging WLAN and WPAN standards, as well as

Table 2. Summary of standard technologies of the WLAN/WPAN field.

Technology	RF band	Link data rate (nominal)	Existing/expected services and applications
IEEE 802.11	2.4 GHz	2 Mbit/s	Legacy LAN traffic
IEEE 802.11b	2.4 GHz	11 Mbit/s	Legacy LAN traffic
IEEE 802.11a	5 GHz	54 Mbit/s	Broadband data transfer enabling multiple services over WLAN
IEEE 802.11g	2.4 GHz	54 Mbit/s	Broadband data rates enabling multiple services over WLAN (compatibility with previous radios)
HIPERLAN/1	5 GHz	23.5 Mbit/s	Legacy LAN traffic, support for real-time services
HIPERLAN/2	5 GHz	54 Mbit/s	Access to broadband core networks, such as ATM, IP, Ethernet, UMTS
Bluetooth	2.4 GHz	1 Mbit/s	Serial cable replacement
IEEE 802.15.1	2.4 GHz	1 Mbit/s	Serial cable replacement (same as Bluetooth)
IEEE 802.15.3	2.4 GHz	55 Mbit/s	Personal multimedia applications
IEEE 802.15.4	868MHz (EU) 915MHz (USA) 2.4 GHz	20 kbit/s 250 kbit/s	Low power and low cost applications, such as remote control, wireless sensors, interactive toys

their targeted applications and data transfer capacities. The introduced technologies are summarised in Table 2.

2.3.1 IEEE 802.11 WLAN

The IEEE standard 802.11 is currently the most widely used WLAN [49]. The original 802.11 standard specifies a single common MAC protocol with three different physical layer alternatives. The MAC protocol supports both ad-hoc networking under *Distributed Coordination Function* (DCF) and infrastructure topology under *Point Coordination Function* (PCF). DCF is the basic functional module of the MAC protocol, while PCF operates on top of it to provide a centrally controlled medium access. The basic IEEE 802.11 protocol architecture is presented in Figure 6 [57].

One of the original 802.11 physical layers uses infrared technology while two of them are 2.4 GHz spread spectrum radios. The spreading of the radio transmission spectrum is done using Frequency Hopping Spread Spectrum (FHSS) or Direct Sequence Spread Spectrum (DSSS). All physical layers provide up to 2 Mbit/s link rate.

The further development of 802.11 has proceeded with new physical layer technologies targeted at adapting the WLAN data rates to better meet the wired LAN capacity. Similarly, higher performance ad-hoc networking has been targeted [49].

The 802.11b physical layer standard updates the link rate to 11 Mbit/s. The used modulation is Complementary Code Keying (CCK) that is based on the DSSS technology. The high rate radio is also backwards compatible with the DSSS radio layer [60]. The 802.11b is currently the most utilised physical layer of the existing 802.11 technologies [43][67].

The IEEE 802.11a standard specifies a physical layer for the 5 GHz band [59]. The standard utilises Orthogonal Frequency Division Multiplexing (OFDM), which divides

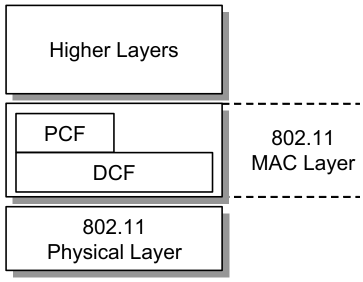


Figure 6. IEEE 802.11 protocol layer architecture.

the transmitted information onto 52 sub-carriers. Different sub-carrier modulations, depending on the link quality and targeted data rate are available. The maximum link rate achieved is 54 Mbit/s. Furthermore, the 802.11g draft standard is specifying a 2.4 GHz band radio that has an equal capacity of 54 Mbit/s [62].

The support of WLAN QoS is being well approached in terms of throughput with the development of higher rate radio layers. The solution of increasing network bandwidth for achieving strict delay requirements is commonly not a viable approach because of the excess capacity [63]. Thus, the QoS support also requires active functionality from the MAC layer of a WLAN.

The QoS support for IEEE 802.11 WLANs is being developed in the task group 802.11e. The purpose is to define MAC procedures to support LAN applications with specific QoS requirements. Transport services for audio, voice, and video applications have been appointed in the design requirements [61]. The proposed IEEE 802.11 QoS support based on the work of the IEEE 802.11e task group is discussed in Chapter 4.

2.3.2 ETSI HIPERLANs

ETSI is developing WLAN technology specifications in the Broadband Radio Access Networks (BRAN) project that was established in 1997. Corresponding to the IEEE 802 standards, the BRAN specifications cover physical and Data Link Control (DLC) layers. DLC contains a MAC protocol and a LLC layer when appropriate [47]. There are also specifications for interfacing existing wired networks.

ETSI BRAN is the successor of the sub-technical committee RES10 that developed the HIPERLAN type-1 (HIPERLAN/1) specification [23]. HIPERLAN/1 specifies a 5 GHz band radio, with the maximum signalling rate of 23.5 Mbit/s for data transmission [23]. ETSI HIPERLAN/1 was the first of the modern WLAN standards, as it was published in 1996.

The HIPERLAN/1 data link layer consists of Channel Access Control (CAC) and MAC sub-layers, as presented in Figure 7. HIPERLAN/1 has a fully distributed network topology. In addition, for extending the network coverage, the HIPERLAN/1 MAC protocol utilises multi-hop relaying, in which intermediate terminals can forward received frames towards their final destination. Interconnection with a peer LAN is enabled with a bridging terminal.

HIPERLAN type 2 (HIPERLAN/2) is a mobile short-range access technology for broadband networks, such as IP (over wired LANs) and ATM. HIPERLAN/2 has a centralised network topology and mobile terminals communicate through AP in a connection oriented manner. A capability for ad-hoc type direct communication between terminals is provided, but a central controller entity is still required for controlling the data transfer [25].

The protocol layer architecture of HIPERLAN/2 contains a physical layer, a DLC layer, and convergence layers, as presented in Figure 7. The DLC contains MAC and Error Control (EC) protocol sub layers for data transfer. The Radio Link Control Protocol (RLC) is used for transporting control messages between a mobile terminal and AP. Both the AP and mobile terminal posses similar protocol layer architectures for data transfer [71].

The HIPERLAN/2 physical layer technology is similar to the IEEE 802.11a. The OFDM modulation is used with 52 sub-carriers, and the utilised frequency band is the 5 GHz HIPERLAN band. The achieved link rate is similarly 54 Mbit/s. Convergence layers adapt the above network (core network) technologies to the HIPERLAN/2 DLC layer. For each of the supported core network, such as IEEE 802.3 (Ethernet), Asynchronous Transfer Mode (ATM), UMTS, and IEEE 1394, a separate convergence layer is specified [25].

Other ETSI BRAN technologies are HIPERACCESS and HIPERMAN, which are broadband fixed wireless access technologies. The projected applications contain internet service providing, LAN bridging, video-telephony, and video conferencing [26].

2.3.3 Bluetooth

Bluetooth is a WPAN technology specified by an industry driven organisation called the Bluetooth Special Interest Group (SIG) [46]. Bluetooth targets voice and data transfer services over short range radio links. Consequently, the main target of the technology is to replace a common serial cable with a wireless alternative. The non-functional requirements have been emphasised. Thus, the technology targets low cost, low power consumption, and small size [86].

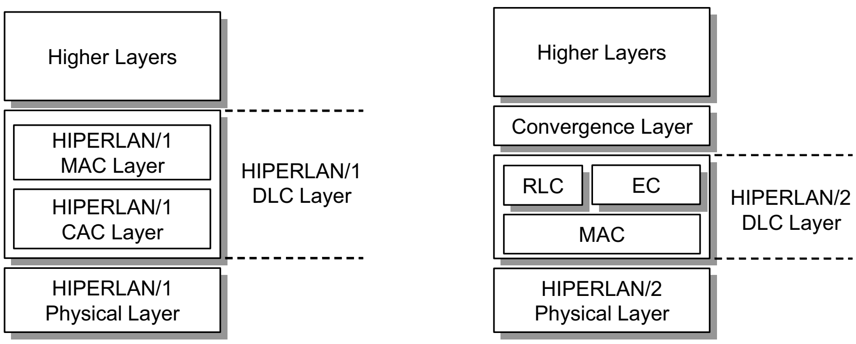


Figure 7. HIPERLAN/1 (left) and HIPERLAN/2 (right) protocol architectures.

The Bluetooth protocol stack is presented in Figure 8. Unlike the IEEE 802 or ETSI WLAN standards, Bluetooth also defines higher layer protocols in addition to the physical layer radio and the link layer that consists of the baseband, Link Manager Protocol (LMP), and Logical Link Control and Adaptation Protocol (L2CAP) [12].

The higher layer protocols over L2CAP are the Service Discovery Protocol (SDP), Telephony Control Specification Binary (TCS-BIN), and RFCOMM that emulates a serial port over Bluetooth. Other higher layer protocols have been adopted from other technologies, such as the Point-to-Point Protocol (PPP), Object Exchange Protocol (OBEX), and AT-Commands. A central interface in Bluetooth is the Host Controller Interface (HCI) that generally separates a physical Bluetooth module and a host computer [83].

The Bluetooth radio operates on the 2.4 GHZ ISM band utilising FHSS over 1 MHz sub-channels. Compared to the 802.11 FHSS physical layer, the hopping rate during data transfer is higher (up to 1600 hops/s). The achieved link rate is 1 Mbit/s. The achieved data rate is dependent of the utilised link and packet types. Two link types categorise Bluetooth services. The Asynchronous Connectionless (ACL) data link provides up to 721 kbit/s asymmetric data rate. The Synchronous Connection Oriented (SCO) voice link has a 64 kbit/s rate, and up to 3 voice connections can exist at the same time.

Bluetooth MAC (baseband and LMP) has a centralised access control, but the network (piconet) is constructed automatically in ad-hoc fashion between a master node and up to 7 slave nodes. In addition, a number of other slaves can be associated to the same piconet while being in a power save mode. Several piconets can form a scatternet, as a Bluetooth node can participate in several piconets at the same time. The Bluetooth scatternet topology is presented in Figure 9.

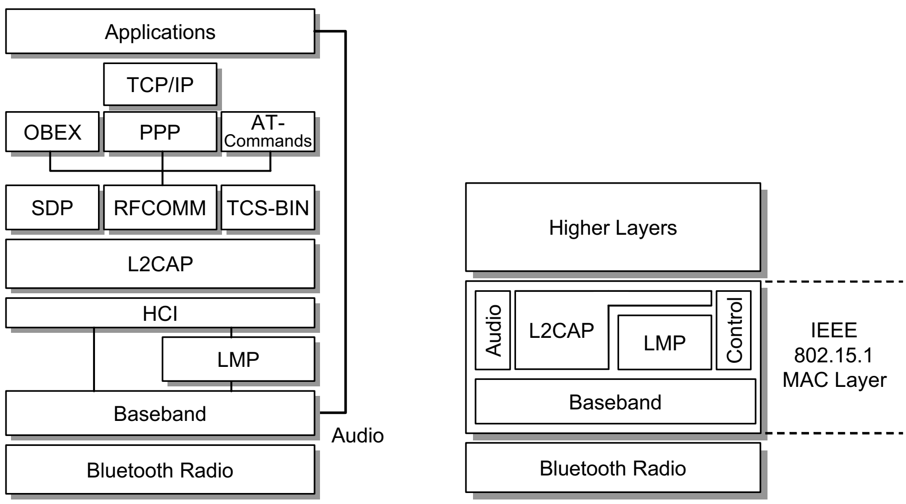


Figure 8. Bluetooth (left) and 80.15.1 (right) protocol architectures.

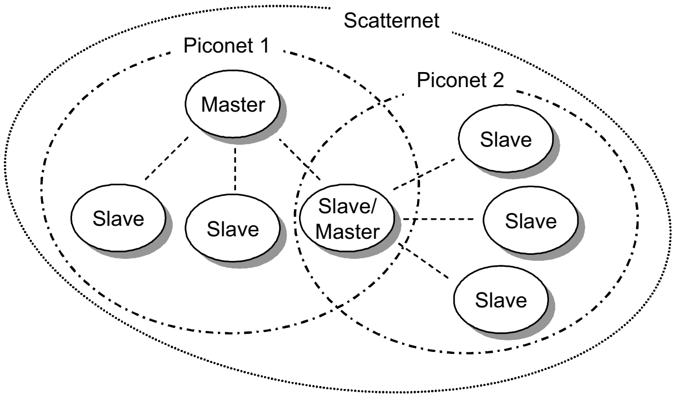


Figure 9. Bluetooth scatternet topology with two piconets.

2.3.4 IEEE 802.15 WPANs

The IEEE 802.15 working group was formed in 1999, focusing on WPAN and short distance wireless data communications in general. There are four task groups [50], of which the task group 2 is developing practices and mechanism to facilitate the coexistence of 802.11 WLAN and 802.15 WPAN [81][75].

The task group 1 (802.15.1) has adopted a WPAN standard from the Bluetooth specification [54]. As other 802 LAN standards, the standard 802.15.1 defines only the physical and MAC layers. Corresponding to the Bluetooth stack, the included layers are the Bluetooth radio for the physical layer, and baseband, LMP, and L2CAP for the data link layer. The 802.15.1 protocol architecture is presented in Figure 8.

The 802.15 task group 3 develops a new standard for a high rate WPAN technology. The target rate has been over 20 Mbit/s, which enables a wider range of personal area applications, such as image and multimedia transfer for consumer electronics appliances [107]. The draft standard defines a physical layer with up to 55 Mbit/s link speed for the 2.4 GHz radio band and a MAC protocol destining to support multimedia applications [55]. Furthermore, the study group 802.15.3a has been recently established for specifying an alternative higher rate physical layer targeting over 100 Mbit/s link rate. Both radio layers will utilise the common MAC protocol.

The 802.15 working group 4 is developing specifications for low data rate WPANs. Target applications are low complexity embedded systems that require a long battery life, such as sensors, interactive toys, remote controls, and home automation devices. The draft 802.15.4 standard proposes a single physical layer, but the layer can operate on two different frequency bands. Furthermore, the lower band is either the 868 MHz SRD band in Europe or the 902 MHz ISM band in USA (see Table 1). The higher band is the 2.4 GHz ISM. The MAC protocol specifies both centralised control and distributed peer-to-peer operation. The targeted link rate is 20 kbit/s for the lower two bands, and 250 kbit/s for the 2.4 GHz band [56].

3. Quality of Service

Applications are the source of service requirements by expecting sufficient data transfer capacity from the underlying network technologies [44]. When an application provides a User Interface (UI), the origin of service requirements is generally extended to be a human user. For example, in audio and video applications, the human expectations for adequate video and audio quality must be met.

The term QoS in relation to data communications can thus be defined as the general capability to meet the expectations of the human user of a network application [31]. This qualitative service definition covers the operations of the application providing UI, the application environment such as the host operating system, and the network data transfer [18]. The qualitative expectations of users, such as the ease of use, short waiting times, reliability, and good sound reproduction, cannot be directly mapped into quantitative requirements that are placed by applications on data transfer. This is because qualitative and quantitative QoS are not measured using the same parameters [18][31].

Applications themselves may try to adapt to the underlying network service for meeting user expectations [15]. However, this is generally not enough for reaching the transfer requirements, e.g., for interactive video communications. In addition, more demanding requirements on network transfer service can be placed with control and automation applications.

The construction of an end-to-end data transfer service between two or more communicating applications is a product of the operation of intermediate network nodes and protocol layers [P1][98][101]. For supporting data transfer QoS, two basic tools are required. First, *signalling* for defining the required QoS over heterogeneous network technologies is needed. QoS signalling is generally layer specific and thus varies in different link technologies. The mapping of the QoS signalling between different layers and technologies is performed by network terminals and intermediate nodes. Second, the data transfer service is constructed by the *QoS support functionality* of intermediate protocols according to the signalled requirements [20].

This chapter concentrates on the QoS signalling and support functionality on the network and especially on the link layer. The discussion on network QoS contains the basic network and transport layer protocols developed mostly within the Internet Engineering Task Force (IETF) for Internet and especially for IP [69]. Following the approach of the previous chapter on WLAN standards, the most common standard wired LAN technology, the IEEE 802.3, is introduced. The functionality of the link layer as a part of the end-to-end QoS is clarified. The quantitative service requirements placed on link technologies are discussed at the end of the chapter.

For summarising the contents of the chapter, the protocol layer architecture of a QoS supporting IP terminal with the related signalling interfaces is depicted in Figure 10.

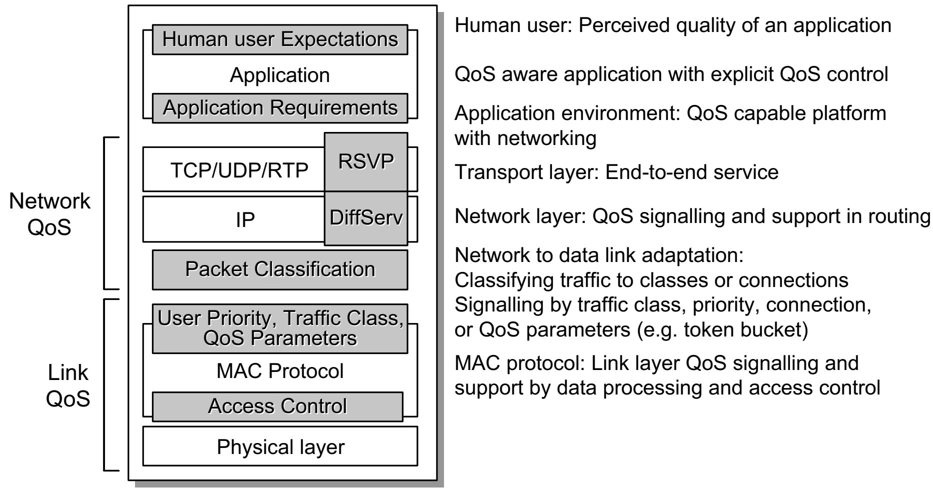


Figure 10. QoS architecture of an IP network terminal.

3.1. Tools for QoS

The basic approaches to provide QoS for network data transfer are reservation and prioritisation. Both can exist together in a same network. In reservation, network resources are allocated based on signalled requests originating from applications. In prioritisation, exchanged packets or frames are usually associated with a priority value that defines the handling in relation to other priorities. The priority value can be used to label the packet to belong in a certain traffic class. Each traffic class can contain a predefined QoS support [31][98][76].

Traffic flow identification is a tool for assigning the configured priority or reservation to appropriate packets or frames. A flow can be identified explicitly by a separate flow label set by the end nodes of the flow, or implicitly by examining the header information of a network packet or LAN frame. The 5-tuple header field information, consisting of source and destination network addresses, source and destination port numbers of a transport protocol, and the used transport protocol type, is used to identify a flow in IP networks. A flow aggregate contains two or more flows that are handled in a same way. An aggregate can be identified e.g. by a shared flow label or by a priority value [98][20][101][76].

A tool used in both the prioritisation and reservation technologies is policing. Policing refers to the administrative decisions and the monitoring of the offered traffic in order to ensure that it does not violate the agreed traffic characteristics. Traffic can be limited in order to protect a network from malicious behaviour. Shaping refers to technologies that meter or regulate a traffic flow to meet the specified reservation. Admission is the initial control for accepting QoS requests. Admission control verifies that available network resources are available to accept a new reservation [97][20].

Queuing with different queue management algorithms [105][110] is a commonly used function for differentiating the waiting delays of packets and frames in network nodes.

Queuing can take place in several protocol layers. Flow control is needed for avoiding lost packets due to insufficient network throughput or the processing capacity of the receiver. If packets are lost due to network congestions or unreliable transmission medium, functionality for recovering from transfer error situations is needed.

3.2. Network QoS

The only QoS class generally provided by the current Internet is best-effort, which means that no delay or throughput bounds are defined. Therefore, also the current applications have been forced to accommodate the changing and unpredictable transfer service. The two basic transport layer protocols of the IP suite are TCP and UDP. Next, their QoS is briefly discussed.

3.2.1 TCP Service

TCP is the most important connection-oriented protocol of the IP suite. It provides a reliable information transfer service for higher layer applications. The main QoS tools in TCP are connection establishment, error recovery, and flow control for avoiding errors due to congestions. Error recovery is implemented by retransmissions and packet reordering [14][109].

TCP provides two main functions for the dynamic flow control. First, a retransmission timer is used for determining a lost packet at the sending TCP/IP host. The timer value can be dynamically adapted to the changing network conditions. The second approach is to control the window size for the sent but unacknowledged packets.

In practice, the slow start algorithm of TCP constantly increases the sending rate for achieving the maximum throughput. However, when a packet is lost due to a transmission error or more commonly, a congestion situation in some network router, the sending rate is dropped. For increasing the robustness of the flow control, the Random Early Detection (RED) function has been widely adopted [110]. RED increases the power of TCP congestion avoidance by dropping packets in network routers before congestion. The original proposed RED algorithm uses random discarding, thus targeting an equal QoS for the different TCP flows. However, the dropping of packets can also be based on packet priority or flow label, in order to control QoS [101].

3.2.2 UDP Service

UDP is functionally a very light transport layer protocol. It is connectionless, and does not provide a reliable transport. On the other hand, UDP gives an application a direct access to the datagram service of the IP layer. This is beneficial in situations when the complexities of packet transfer over IP can be more efficiently managed by an application itself. Also, multicast and broadcast services are available by using UDP [14].

Similarly, UDP can be utilised as a base for other transport layer protocols that target the provision of certain types of services. An important protocol for multimedia applications is the Real Time Transport Protocol (RTP). RTP tools for QoS support are the payload data identification, sequence numbering, and time stamping for synchronising the packets from several sources [101].

UDP has also been exploited as a base transport protocol for implementing the Video Control Protocol (VCP) of the wireless video demonstrator over TUTWLAN. The demonstrator is presented in Chapter 5.

3.2.3 RSVP and Integrated Services

The Integrated Services (IntServ) working group of IETF has specified a flow based QoS support [15]. Because the support is reservation based, new traffic control functions are needed in the IntServ routers and hosts [33][104][20][101][105].

IntServ defines three service classes: guaranteed, controlled load, and best-effort. The controlled load service provides only a single type of service, as there are no parameters to further define the service. The controlled load is targeted at applications that are sensitive to network overload situations, but tolerate throughput and delay variations experienced under normal conditions. Thus, the service approaches the best-effort service in a lightly loaded network. The best-effort service is for tolerable applications, and does not provide QoS guarantees [101]. The guaranteed service is targeted at applications that require delay and throughput bounds. The service assures that packets are transferred within the guaranteed time.

The establishment of a connection with guaranteed service requires signalling of the traffic characteristics and reservations. This is usually made using the Resource Reservation Protocol (RSVP). Traffic flow requirements are defined by a sender in PATH messages using the TSpec (Traffic Specification) and reservations are defined by receivers in REVS messages using RSpec (Service Request Specification). The QoS for a flow is defined using the token bucket model that is further explained in a later section of this chapter [98][101][94].

3.2.4 Differentiated Services

IETF Differentiated Services (DiffServ) defines QoS implementation based on independent network packet handling at network nodes [89][20]. This leads into predictable QoS for the given traffic [101]. The data transfer service is divided into different QoS classes inside a DiffServ domain. No end-to-end reservations by signalling for a traffic flow are made, but the handling of a packet depends on the DiffServ code point value carried in packet headers. This simplifies the management, provides better scaling, and reduces the signalling overhead compared to IntServ approach [44][76].

In DiffServ architecture, ingress and egress boundary nodes of the DiffServ domain classify packets based on the header information, and interior nodes of the domain forward packets according to the assigned classification. A boundary node, such as a host, can also contain traffic conditioning tools, for marking, metering, and shaping. A meter measures if the classified traffic meets the traffic profile that is entitled for the agreed QoS. The shaper can modify the traffic profile by using different QoS tools, such as the token bucket filter and dropping of packets.

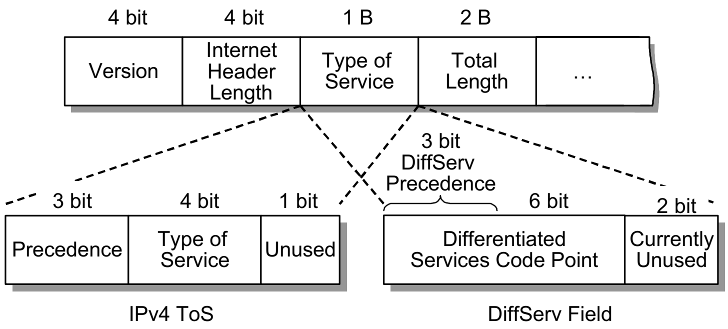


Figure 11. Structure of the IPv4 ToS field.

DiffServ is redefining the meaning of the Type of Service (ToS) field of IP Version 4 (IPv4) and the traffic class octet of IPv6 headers [101][89]. ToS field in the IPv4 has been available as long as the protocol itself, but the use of the field has been mostly neglected. For DiffServ, the ToS byte of the IPv4 header is renamed to DiffServ field that contains a DiffServ code point value, as presented in Figure 11 [89]. The code point for a packet may be chosen from a set of recommended values, or the value may have purely local meaning.

For backward compatibility, the first three bits of the DiffServ field are used to provide near the same forwarding treatment as specified in the IP precedence scheme for the ToS field [69]. The IP and DiffServ precedence values and their mapping to the target traffic types are presented in Table 3. Classes 1 to 4 define assured forwarding, in which the QoS of a class is provided only in relation to other assured forwarding classes [44].

3.3. Link QoS

In the pervious section, the QoS technologies of the network and transport layers were discussed. For extending the QoS support into the LAN domain according to the end-to-end QoS model, the link layer QoS support functionality is required. In this Thesis, the link layer of a LAN corresponds to the utilised MAC protocol. Some link technologies, such as IEEE 802.5 token ring [109] and ATM [31] have had QoS support integrated from the beginning. Especially, an important result from the ATM development work has been the accurately specified QoS.

Table 3. IP and DiffServ Precedence classes [69][44].

Precedence value	IP precedence	DiffServ precedence
111	Network Control	Network Control
110	Internetwork Control	Internetwork Control
101	Critical	Express Forwarding
100	Flash Override	Class 4
011	Flash	Class 3
010	Immediate	Class 2
001	Priority	Class 1
000	Routine	Best-effort

Regardless of ATM and token ring QoS support, in the widely used IEEE 802.3 terminals are equal when competing for the chance to transmit a MAC frame. However, the 802.3 LAN technology has evolved in terms of link rate and QoS support.

3.3.1 IEEE 802.3 LAN (Ethernet)

The legacy 802.3 technology utilises a shared broadcast medium (a coaxial cable). The used MAC protocol is called Carrier Sense Multiple Access with Collision Detection (CSMA/CD). The CSMA operation avoids collisions resulting from the simultaneous transmissions of two or more terminals, while CD enables faster collision recovery. The operation of CSMA/CD is depicted in Figure 12.

When a terminal has a frame to transmit, it first listens to medium activity by utilising the carrier sensing function of the physical layer. If the medium is detected idle, the terminal begins to transmit after a short inter-frame delay. As the physical layer is capable of continuously sensing the medium, it can detect other simultaneous transmissions on the medium and indicate a collision to the MAC protocol. When a collision is detected, the MAC protocol enforces the collision by transmitting a bit sequence called jam. The jam ensures that the other colliding terminals also detect the collision situation and terminate their transmission, thus improving the medium utilisation efficiency.

Next, the terminal schedules a retransmission after a selected *collision backoff* time. The collision backoff time is constructed with an integer number of backoff slots. A slot time is dependent on the type of physical layer, being larger than the sum of the physical layer round-trip time and the maximum jam time. The collision backoff time for the *n*th retransmission attempt is a uniformly selected random integer number of backoff slots *r*. The range of *r* is:

$$0 = r < 2^{k}, k = min(n, 10) \text{ [65] .} \qquad (1)$$

The increasing of *r* with transmission attempts is depicted in Figure 13. The maximum collision backoff is frozen after 10th retransmission, and retransmission attempts are continued until a physical medium dependent attempt limit is reached [65]. The 802.3 does not utilise acknowledgements, but the integrity of the received frame is verified by a Frame Check Sequence (FCS) that is a Cyclic Redundancy Check (CRC) value.

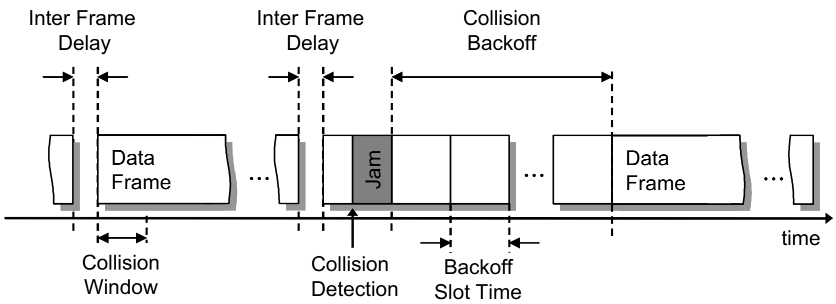


Figure 12. IEEE 802.3 CSMA/CD.

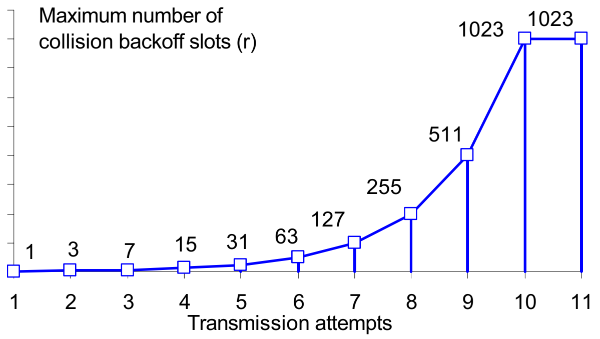


Figure 13. Selection of the number of maximum collision backoff slots in IEEE 802.3.

For the higher layer QoS signalling, the MAC service specification for IEEE 802.3 defines a parameter for a requested service class for each MAC Service Data Unit (MSDU). However, the CSMA/CD MAC protocol supports only a single data transfer QoS, regardless of the service class requested [65].

3.3.2 IEEE 802.1 Bridge

The development of QoS support in IEEE 802.3 takes place through switched network architecture. The IEEE 802.1D standard defines a MAC layer bridge for interconnecting different LANs [63]. In a fully switched topology, only two terminals exist on a LAN segment. Consequently, the physical medium can be used as a full duplex point-to-point link. As there is no longer need to contend for the medium access, the utilisation of the shared CSMA/CD protocol is not required [65][109].

The IEEE 802.1D standard incorporates the 802.1p for Traffic Class Expediting and Dynamic Multicast Filtering [63]. Thus, the 802.1D supports time critical information transfer in a LAN environment. The QoS support is based on different *traffic classes* implemented by preferential queuing and assignment of access priorities by the bridge. The classes are further discussed later in this Chapter.

The IEEE 802.1Q standard for Virtual Local Area Networks (VLAN) extends the priority signalling in IEEE 802.3 networks by introducing a new header field called tag header [64]. It is used to identify the VLAN in which the LAN frame belongs to, but it also carries a 3 bit *user priority* value for the MSDU. A frame can be tagged for priority only, but the overall frame structure remains the same. The updated 802.3 frame structure is presented in Figure 14.

The 802.1D bridge architecture consists of two or more ports, a MAC relay entity, and higher layer entities for bridge management, as depicted in Figure 15. At least the bridge protocol entity is required as a higher layer protocol for managing the bridged LAN topology [63]. The MAC relay entity performs the frame forwarding between source and destination ports. The filtering database contains static or dynamically maintained information of the bridged LAN topology. Therefore, a bridge is able to filter frames according to the MAC addresses of frame header fields, if a destination

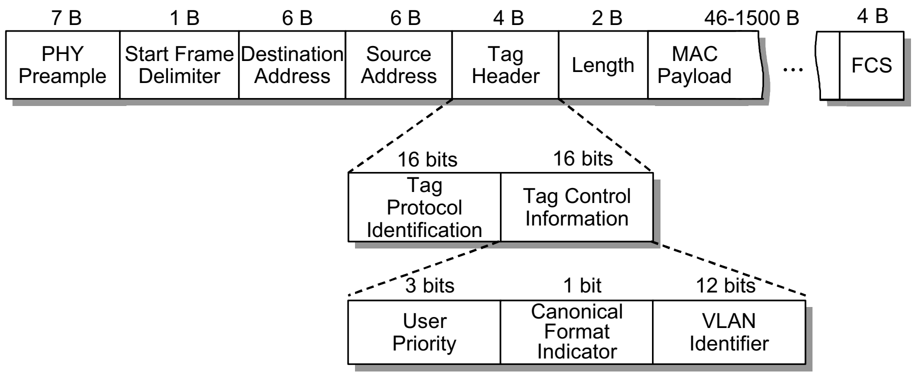


Figure 14. IEEE 802.3 tagged frame format.

terminal cannot be reached through a port. State information for each port indicates whether a port is available for frame forwarding [63].

The signalled priority formation is used as the basis for supporting QoS in bridges. A bridge maps the user priority into one or more traffic classes of the forwarding process, which utilises a separate queue for each supported class [63]. Since the number of queues can be less than the number of user priorities, several priorities can be mapped into one queue. A bridge with only a single supported class provides a best-effort forwarding service. In addition, a bridge can use a different number of classes per each port. For output, the numerically highest traffic class is selected from queues for frame transmission and given to the MAC protocol of the destination port. The 802.1D standard also allows other algorithms to be used for queue scheduling [63].

Another mapping is made from the original user priority to the outbound *access priority*. This access priority is used by the output port MAC protocol for frame transmission [63]. Consequently, the assignment of the access priority depends on the utilised MAC protocol. In the current standard, a bridge does not support different

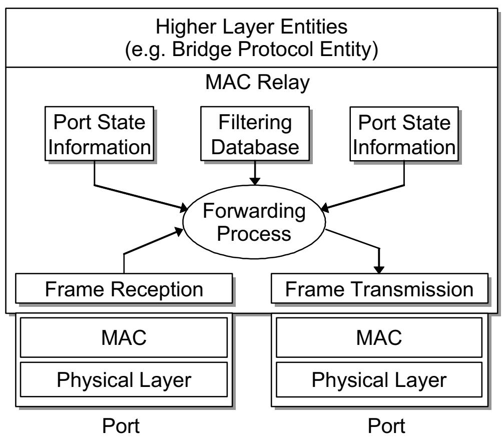


Figure 15. IEEE 802.1D bridge architecture.

access priorities for the 802.3 LAN or 802.11 WLAN by default. In addition to the access priority, the original user priority will also be passed to the underlying MAC protocol to be possibly signalled over the destination LAN [63].

3.3.3 Subnet Bandwidth Manager

IETF has developed mappings between higher layer QoS management protocols and the supported QoS of the link level. One result is the Subnet Bandwidth Manager (SBM) architecture. SBM extends the RSVP based admission control to be used over IEEE 802 type LANs for controlling LAN capacity allocations for different traffic flow reservations. However, SMB is not limited to the use of RSVP, but can utilise other signalling protocols [120][33]. In the SBM architecture, each LAN segment contains a designated SBM that performs admission control to the LAN resources. The designated manager must therefore maintain the state information of the current reservations. SBM clients locating in end terminals reserve bandwidth from the designated manager using RSVP messages.

In a general SBM operation, an SBM client performs the mapping between higher layer QoS management (such as RSVP) to the link layer QoS. In 802 based LANs, the SBM client will select a 802 traffic class by assigning a user priority for an arriving traffic flow. Traffic flows requiring a similar service can be mapped into a single class. Consequently, the admission control and controlling of the traffic class selection ensure that enough capacity for the service requirements exists [104].

The default mappings suggested in [104] between the 802 user priorities and expected service characteristics are presented in Table 4. The mapping to 802 traffic classes can be made according to the IntServ service type (guaranteed, controlled load, best-effort) or from TSpec and RSpec parameters.

3.4. LAN QoS Parameters

The general approach for QoS support is to divide transferred LAN traffic into classes and to control the service of a class with queue and access priorities [63]. The traffic class information is signalled across a LAN using a separate QoS signalling protocol or the signalling is embedded into LAN frame header fields. The classification is generally done at the end terminals, and can utilise the QoS signalling of higher layer protocol entities, such as the application and IP layers [31][66][65][61].

Table 4. Default user priority mappings for services in SBM architecture [104].

User priority	Service characteristics
0	Default (best-effort)
1	Reserved, less than best-effort
2	Reserved
3	Reserved
4	Delay sensitive, no delay bound
5	Delay sensitive, 100 ms delay bound
6	Delay sensitive, 10 ms delay bound
7	Network control

Increasing of the number of supported traffic classes shifts the weight between transfer delay and throughput management. With a small number of classes, the management focuses on delay. Thus, a suitable approach with only two classes is to support a small number of time critical applications with low throughput requirements in addition to the basic transfer service. With increasing the number of supported classes, the management of throughput between traffic flows is improved [63]. The eight possible traffic classes introduced with 802.1D, their target services, and mappings to user priorities are presented in Table 5.

3.4.1 MAC Service Parameters

The IEEE 802.1D standard for MAC bridges parameterises MAC layer QoS requirements. For a bridge, the basic requirement for QoS support is not to affect the service provided by a LAN without a bridge [63]. In addition, while 802.3 provides only a single QoS, traffic classes can be supported by the switched network. Next, the general requirements set on MAC services are discussed.

The service availability is the portion of time when the MAC transfer service is available. This service parameter covers characteristics such as fault tolerance. Some MAC layer network components, such as bridges, can increase the availability with automatic reconfiguration. In traffic class support, the preferential transfer of higher importance classes increases availability for them.

The MAC layer frame loss can occur due to physical layer transfer errors or queue overflows. The frame loss rate can be decreased by implementing error recovery functionality into the MAC protocol. In QoS support, frame loss can be experienced by

Table 5. IEEE 802 user priority to LAN traffic class mappings and targeted services.

User priority	Traffic class	Targeted services
7	Network control	The network control service has critical requirements on reliability. The throughput requirements are generally low, but delay bounds and reliability are significant.
6	Voice	Voice service is delay and delay variance critical. The expected maximum end-to-end delay and delay variance is 10 ms.
5	Video	Video service has relaxed delay requirements compared to voice. The maximum 100 ms end-to-end delay is acceptable.
4	Controlled load	Controlled load service is targeted for important applications using the LAN. These types of applications are expected to have admission control, and the capacity of the LAN is adapted to meet the requirements
3	Excellent effort	Excellent effort service is intended to be better than the best-effort service.
0	Best-effort	Best-effort service is currently the most utilised traffic class. No delay or throughput differentiation or bounding is made.
2	<not specified>	
1	Background traffic	Background service is for applications not requiring or not authorized for any dedicated service, such as bulk transfer of large files. These services can exist in the background of a LAN, and shall not affect to the QoS of any other services.

a traffic class when the buffering capacity or the frame lifetime are exceeded while a medium is used by higher importance traffic.

The MAC service should not misorder frames of a flow sharing the same traffic class. However, due to error recovery the order in which frames are successfully transmitted over a link can change, and the MAC protocol should buffer and reorder the frames for the destination MAC user (application or a higher protocol layer). Another result from multiple transfer attempts, and multiple paths between source and destination terminals, is the possible frame duplication. Duplicates must be identified and discarded by the MAC protocol. Undetected errors are fatal for the operation of the MAC, and their probability should be negligible. However, there can be applications that can use data with known residual errors. This type of transfer can be provided with a MAC protocol if it is implemented by the QoS support [11].

The maximum MSDU size that is accepted to be transferred depends on the MAC and physical layer technologies. A MAC protocol can adapt MSDUs with fragmentation and reassembly functions to suite them better for the transfer service.

Adequate throughput is a central requirement placed on the MAC protocol services, and the basic prerequisite for delay control. The delay is to be supported as a separate QoS parameter for avoiding excessive capacity requirements [63]. The end-to-end MAC frame transfer delay is the time between issuing a transfer request primitive by a source MAC protocol user to the reception of a delivery primitive by the user of the destination MAC protocol. This delay consists of delays through the source and destination MAC protocols, where a frame can encounter processing, queuing, and access delays, and the delays through the physical layers and medium transfer.

The access delay is defined as the waiting time when a MAC protocol has selected a frame to be transmitted, but waits for a reserved time slot or contends with other MAC protocols before actual transmission [66][63]. Because the access delay is a result of several components that depend on the network traffic load and quality of the underlying medium, traffic encounters delay variance.

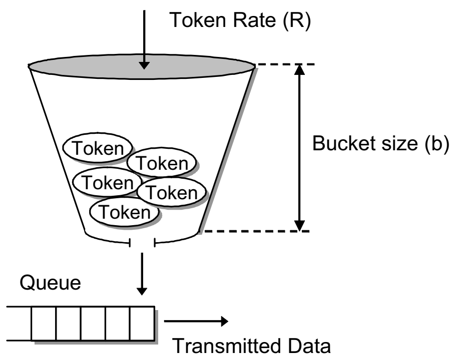


Figure 16. Token bucket algorithm.

For quantitatively specifying the service requirements placed by applications, the token bucket model is widely used [98][109][105]. The model can be seen as a queue control algorithm, as depicted in Figure 16. The basic parameters are the token rate (R) and the bucket size (b). In addition, three additional parameters are commonly defined: peak rate, minimum policed size, and maximum packet size.

The token rate defines the sustainable, average data rate for a traffic flow. The bucket size characterises the maximum burst size. For a period of time, the token rate can be exceeded according to the bucket size. Consequently, the maximum amount of data sent within any time period T is RT+b.

The peak rate gives the maximum allowed data rate that cannot be exceeded. The minimum policed size is the smallest packet size handled by the algorithm. All smaller size packets are handled as being the minimum policed size in resource allocation. Finally, the maximum packet size defines the size of the maximum packet allowed.

4. WLAN QoS

As discussed in Chapter 3, the network and transport layers provide end-to-end transport services for applications over several heterogeneous link technologies. The transport layer manages the reliability of transfer, but the delay and delay variance between network entities is generally shorter at the link layer [109]. QoS support implementation is a central requirement for a wired LAN MAC layer functionality, and the support is further emphasised in WLANs.

Various proposals for improving the performance of standard WLANs and their QoS support are available in research publications [3][90][34][22]. Because of the popularity of the IEEE 802.11, most of the proposals concern the 802.11 MAC protocol [17][96][6][7][106]. Furthermore, many of the proposals target at compatibility with the existing standard. These proposals are commonly associated with a proposed local QoS scheduling algorithm for dividing the available throughput according to the required QoS [111][3].

Another central research approach is to improve the adaptation between WLAN and legacy higher layer protocols [87]. Especially, the TCP flow control assumes that packet loss is due to congestion in some network node, which decreases the performance of wireless links [119]. Consequently, a Performance Enhancing Proxy (PEP) can be used as an intermediate entity on a standard WLAN to improve the performance of TCP/IP and Internet applications [13]. The development of new routing functions to enable seamless ad-hoc networking with TCP/IP has also been a key research topic for WLAN QoS [114].

4.1. Medium Challenges and MAC Services

The wireless link bandwidth is a scarce resource, as there is a common competition for the usable radio bands between different technologies [29]. Consequently, the bandwidth of a wireless link is usually significantly lower compared to wired LAN and cannot be multiplied with switches and new cabling [85].

The basic expectations placed with wired MAC protocols on the underlying medium are stability and reliability, which do not apply with a wireless medium. The wireless medium is time variant, and the link characteristics change with the movement of terminals. The same frequency band can be shared by different systems causing interference [45]. Forward Error Correction (FEC) coding and Automatic Repeat Request (ARQ) algorithms are the two basic recovery functions that are associated with error checking using CRC codes. Error avoidance and recovery can be enhanced by fragmentation and reassembly functions [10][73][4][100].

Two basic challenges for WLAN communications, especially present in ad-hoc networking, are depicted in Figure 17. The hidden node problem occurs when two terminals (A and B), unaware of each other due to a missing signal path, transmit at the

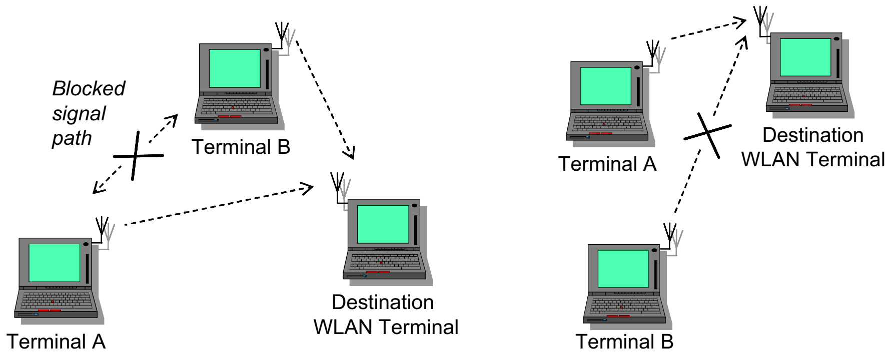


Figure 17. Basic WLAN challenges: hidden node (left) and power capture (right).

same time to a common destination terminal. This leads to the collision of transmissions and wasted medium capacity. The other challenge is called the power capture effect, in which the transmission of a nearer terminal (A) is successful despite the contending transmission from a more distant terminal (B) with a weaker signal level at the destination terminal. The capture disturbs the operation of the MAC protocol, causing unequal medium access. A transmission power control can be used to minimise the effect of power capture [4][112][10][92].

The coverage area of a wireless medium cannot be accurately limited and it cannot be trusted that every terminal operating on the same WLAN is actually receiving all significant transmissions. The topology of a network is dynamic as new terminals arrive to the signal coverage area, join the network, operate, and finally leave the network. A terminal can also disappear from a network due to a temporary signal fading and power save state that is utilised for extending battery lifetime.

Consequently, terminal registration (also known as association) with the network (with AP or other terminals) and the advertising of the network availability are needed. A modified association procedures reusing the existing registration information may be exploited for enabling faster handovers between APs [23][57].

The shared radio medium with undefined coverage area results in security risks, as anyone with a suitable terminal may eavesdrop network traffic. Data security by encryption is generally required because of the insecure medium. Another requirement is the authentication of a peer MAC, as a wireless WLAN terminal cannot be identified by a physical location [65][57][4].

The fundamental MAC protocol service in both wireless and wired LANs is to deliver MSDU between peer terminals. For this, the basic function is to divide transmission medium between MAC entities for reliable exchange of MAC Protocol Data Units (MPDU). In the following discussions, this function in WLANs and WPANs is referred to as channel access.

A wireless MAC protocol divides the radio bandwidth into frequency sub bands and time slots, thus creating transfer channels. Because of the limited radio bandwidth and asymmetric traffic characteristics, Time Division Multiple Access (TDMA) is generally utilised in a WLAN segment, while Frequency Division Multiple Access (FDMA) is used for dividing the available bandwidth between separate WLANs or WLAN segments. Consequently, Time Division Duplex (TDD) transfer is preferred against Frequency Division Duplex (FDD) [19][95].

The MSDU delivery QoS is controlled by the requests of higher protocol layers or local management entity. The requested QoS is supported by the operation of the radio channel access and the data processing functions, such as queue scheduling and error control of the MAC protocol.

4.2. IEEE 802.11

The basic target of the original IEEE 802.11 has been to extend the wired LAN services, namely 802.3 networking over wireless links [80]. Therefore, the projected WLAN applications have also been conventional LAN applications.

In the 802.11 MAC, several data processing functions affecting QoS are available before channel access. The MAC protocol supports fragmentation of long MSDUs into shorter MPDUs for improving transfer reliability. MPDU payloads can be encrypted using the Wired Equivalent Privacy (WEP) algorithm. Immediate positive acknowledgements and retransmissions are utilised for error recovery [57].

The requested QoS in the 802.11 MAC protocol contains two parameters that are passed to the protocol with a MSDU transfer service primitive. These are the *priority* and *service class*. The priority parameter specifies the MSDU priority and can have two values, contention and contention free. The service class also has two possible values: strictly ordered and reorderable multicast, defining whether the MAC protocol is allowed to reorder MSDUs [57]. The support for time-bounded services is implemented only in the centrally controlled network topology under PCF. The basic service class under DCF is contention that provides a best-effort QoS.

The 802.11 MAC protocol used in DCF is Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). The access resembles the CSMA/CD utilised in 802.3 LANs, as carrier sensing is employed for verifying an idle medium before transmission. In wireless medium, though, the carrier sensing is less reliable. Also, collision detection is difficult to implement. It would require the doubling of most transceiver parts of the system and accurate signal processing since the transmission energy of the own transmitter easily covers other signals [9][1].

In 802.11, the carrier sensing mechanism is further refined for avoiding collisions, especially due to hidden terminals [1]. In addition to the Clear Channel Assessment (CCA) function for physical carrier sensing, a virtual carrier sensing is utilised by two methods. Prior to the exchange of a data frame, terminals may exchange short control frames, namely Request to Send (RTS) and Clear to Send (CTS) messages. These frames carry the duration of the frame transfer, thus explicitly reserving the medium for the following transfer, as other terminals receiving the control frames use the duration information for virtual carrier sensing. The duration information is also carried on the header of each directed data frame.

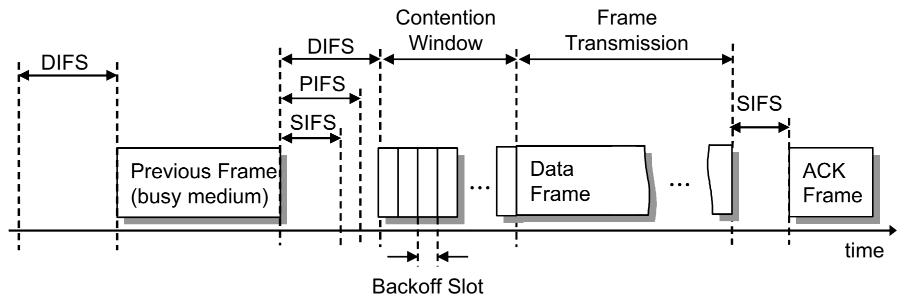


Figure 18. Basic MAC operation of the IEEE 802.11 WLAN.

In the IEEE 802.11 standard, the priority of medium access is selected according to the frame type, not directly according to the traffic type. The priority is based on Inter Frame Spaces (IFS), where a higher priority frame can be sent earlier on an idle sensed medium than a lower priority frame. The 802.11 MAC contains three basic IFSs[1]. The highest priority called Short IFS (SIFS) is reserved for immediate control frames, such as positive acknowledgements (ACK) and CTS, and for the following segments of a fragmented MSDU. The lowest priority is DCF IFS (DIFS), and all competing terminals under DCF utilise this access priority. The basic 802.11 MAC operation under DCF is presented in Figure 18.

DIFS is followed by another Contention Window (CW) for resolution. A random backoff time is selected before the start of transmission while continuously sensing the medium activity. The backoff is used for avoiding collisions, and it provides a fair access among all competing terminals. The backoff time consists of an integer number of short backoff slots, of which duration is a physical layer specific parameter. The selection of CW is dynamic, as the average value changes according to experienced collisions with the increasing network traffic load. The backoff time is selected using the function

*Backoff Time = Random (0, CW)*Slot Time* [57]. (2)

Thus, the number of backoff slots is randomly selected using an equal distribution between 0 and the *CW* integer number of backoff slots. CW takes values between CW_{min} and CW_{max}. CW is increased exponentially from the initial CW_{min} after each unsuccessful transmission attempt, until the maximum CW_{max} value is reached. Retransmissions are limited by two variables: a short retry limit for frames not utilising the RTS/CTS reservation procedure and a long retry limit for frames sent with the RTS/CTS procedure. The CW backoff algorithm of the IEEE 802.11b is depicted in Figure 19 [P3][88][74].

Under PCF, a point coordinator in AP uses the middle channel access priority called PCF IFS (PIFS) to seize control of the channel. It may then generate a Contention Free Period (CFP) by transferring data to terminals and by polling terminals for an exchange

[1] Extended IFS (EIFS) is the longest IFS. Rather than frame priority, it is used for recovering from possible error situations.

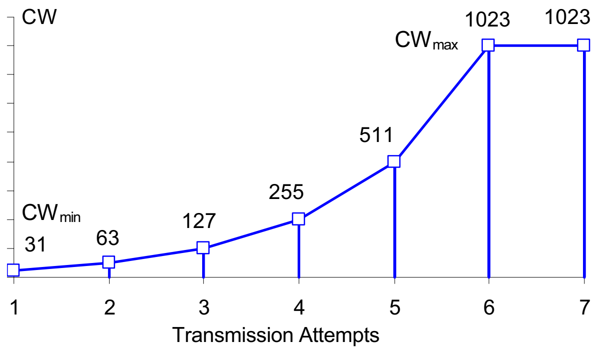


Figure 19. Selection of the maximum number of CW back backoff slots in IEEE 802.11b.

of a single data frame, always acquiring the channel control with the higher access priority compared to DCF. In addition, contention free operation is secured by beginning CFP with a beacon frame that broadcasts the duration of CFP. The duration information is used for the virtual carrier sensing by network terminals. The operation of CPF for data transfer under PCF is presented in Figure 20.

A defined repetition rate for CFPs enables more controlled delay and delay variation compared to the contention based transfer. The assignment of contention free transmission opportunities for a pollable terminal depends on the implementation of the polling algorithm in AP. In the default case of 802.11 PCF, a terminal requests to be placed on the polling list during the association procedure and AP sequentially polls the terminals on the list with a round robin algorithm [3], in the order of their association identifier values. According to the standard, AP may also utilise other maintenance algorithms for the polling list. These algorithms can be chosen by product vendors [49].

4.3. IEEE 802.11e Enhancements

In the recent research proposals for improving the 802.11 QoS two basic approaches are to change the polling algorithm under PCF [3][111] and the contention used for DCF.

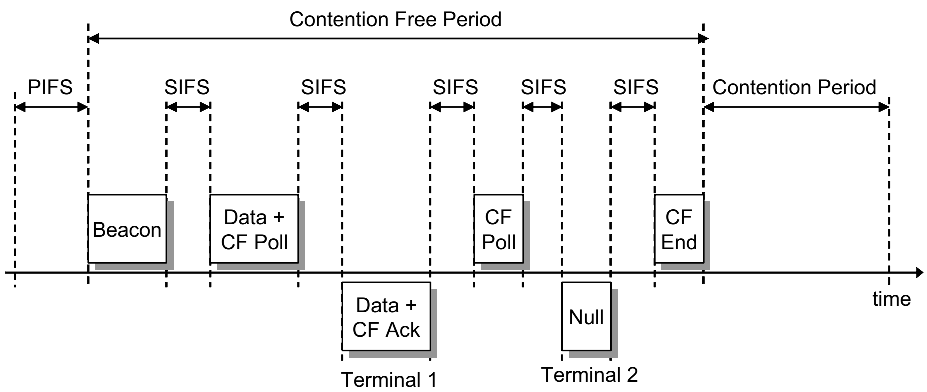


Figure 20. CFP under PCF.

New algorithms for calculating CW according to traffic type [7][17][6] and new traffic scheduling algorithms [90][22] have been the most common alternatives for DCF. Also, the utilisation of PIFS under distributed control for transferring real-time traffic, or the increasing of the number of inter frame spaces according to frame priorities have been proposed [106][1][90]. Link adaptability has been proposed to be improved with new transfer error control algorithms, fragment size management, and physical rate selection in [34][96][1][118].

The IEEE 802.11e draft standard [61] includes a number of new features for the MAC QoS enhancements, which are closely related to the above research proposals. The new MAC is developed to be backwards compatible, which also complicates the design of QoS support. Consequently, the traffic originating from non-QoS capable terminals is categorised as best-effort. The MAC QoS is supported on both independent and infrastructure network topologies [49].

A new QoS related data processing function introduced with the 802.11e draft is MAC layer FEC for reducing the probability of retransmissions and lost MSDUs. The FEC uses a (224,208) Reed-Solomon code. A stronger FEC is performed on the MAC header using a (48,32) Reed-Solomon code [61].

For a MAC user, the draft protocol contains the two QoS parameters that are passed along a MSDU transfer service request primitive: the *priority* and *service class*. As only a single traffic class is defined, the service class parameter is omitted and can be a null value. The original contention and contention free values are also supported as priority parameter values, but the range of the parameter is expanded to cover 16 separate values.

The priority parameter is used in two ways, to directly assign a *user priority* for an MSDU (values 0-7), which is later utilised as delivery (access) priority. The priority value can also assign the MSDU to a TSpec (values 8-15). A TSpec is configured through the MAC management, and the protocol attempts to transfer MSDUs assigned with a TSpec according to QoS definitions contained in the traffic specification. The MAC user priority parameter signalling is depicted in Figure 21.

The assigned priority parameter value is signalled between terminals using a new QoS control field of the MAC frame header. The default user priority ordering for 802.11e is

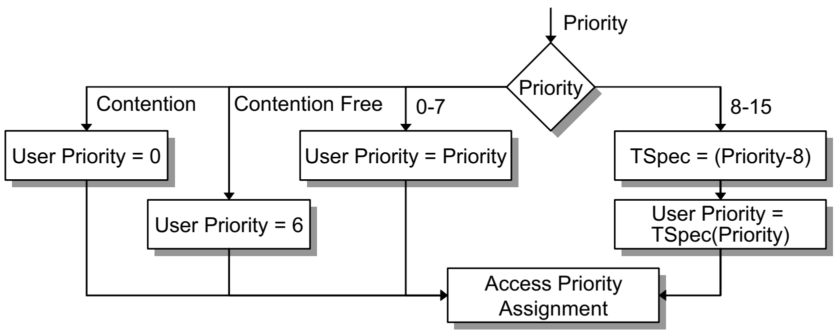


Figure 21. MAC user priority parameter signalling.

{7, 6, 5, 4, 3, 0, 2, 1}, which is the same as that recommended by the IEEE 802.1D for 802 LAN traffic classes presented in Table 5. The contention priority value of the original 802.11 is expected to be mapped to priority 0 and the contention free value to priority value 6.

A single traffic stream is identified with a stream identification and an association identification of the source or destination terminal. These parameters are part of a Tspec in the draft standard. The TSpec also defines the source and destination MAC addresses for a stream. The traffic stream properties and consequently its QoS requirements are specified using a number of parameters introduced in Table 6. All parameters may not be available in every TSpec or in the final standard.

A new coordination function called Hybrid Coordination Functions (HCF) has been developed for providing selective handling of MSDUs according to the QoS support. HCF provides two modes for channel access, *HCF contention* and *HCF polling*. QoS traffic is supported by Transmission Opportunities (TXOP) that can be obtained using both access modes.

4.3.1 HCF Contention

The HCF contention access is presented in Figure 22. Compared to the original standard, a new Arbitration IFS (AIFS) has been introduced. The AIFS is not a single value, but assigned separately for each access priority (marked as AIFS[i]) by a local management entity of a terminal or by AP. Similarly, new dynamics have been

Table 6. IEEE 802.11e TSpec parameters for transfer QoS support.

Tspec parameter	Description
Periodic traffic	Defines whether the traffic stream is continuous or periodic
Bi-directionality	Defines whether the traffic stream is bidirectional or unidirectional
ACK policy	Defines the used acknowledgement policy. Acknowledgements can be prevented when no retransmissions are used, or delayed.
FEC	Enables the use of FEC coding
Delivery priority	Defines the utilised delivery (access) priority value used for the traffic stream MSDUs
Retry interval	Defines the minimum time a transmitting terminal waits for a delayed acknowledgement before initialising retransmissions
Inactivity	Defines the maximum time that can elapse without any MSDU using this stream being transmitted before AP deletes this stream (for freeing capacity)
Inter arrival interval	Specifies the nominal inter arrival time of MSDUs of this traffic stream
Nominal MSDU size	Defines the nominal size of MSDUs belonging to this traffic stream
Minimum data rate	Defines the lowest data rate that is acceptable for this traffic stream (within the delay and jitter limits)
Mean data rate	Defines the nominal sustained data rate for this traffic stream (within the delay and jitter limits)
Maximum burst size	Defines the size of the maximum burst that can occur for this stream (within the delay and jitter limits)
Delay bound	Defines the maximum delay for a MSDU between its reception from the local MAC user to the start of successful the transmission of the MSDU to its destination
Jitter bound	Defines the maximum delay difference for a MSDU between its reception from the local MAC user to the start of successful the transmission of the MSDU

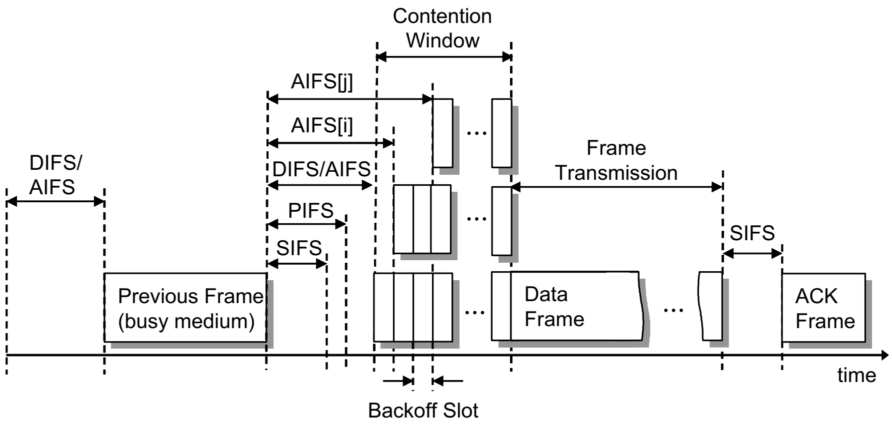


Figure 22. Channel access in 802.11e for an independent WLAN topology.

proposed for the calculation of the backoff time. CW_{min} and CW_{max} are not fixed according to physical layer, but assigned separately for each access priority ($CW_{min}[i]$, $CW_{max}[i]$). The algorithm for selecting backoff time is rephrased as

*Backoff Time[i] = Random (1, (CW[i] +1)) * Slot Time* [61], (3)

where $CW_{min}[i] \leq CW[i] \leq CW_{max}[i]$.

The selection of a new CW[i] after an unsuccessful attempt to transfer a MSDU belonging to the associated access priority is done using the following criteria with a Persistence Factor (PF) parameter:

CW[i] =((CW[i]+1) PF[i]) - 1* [61]. (4)

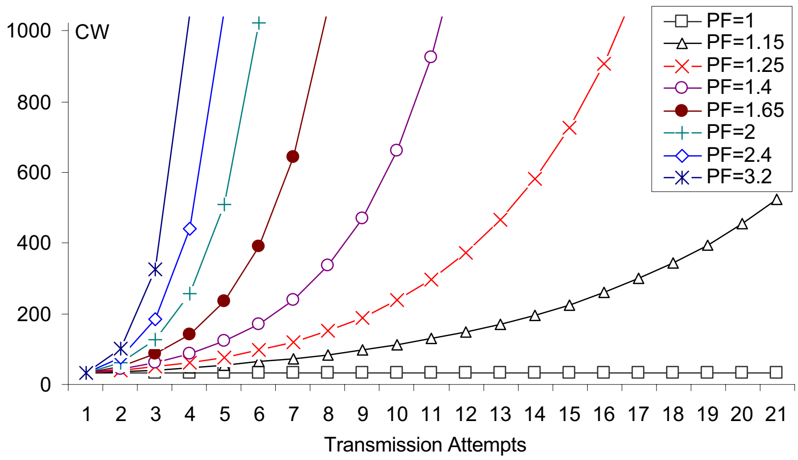


Figure 23. Example selection of CW in IEEE 802.11e with different PFs.

Increasing of CW with eight different PF[i] scaling values for each access priority is depicted in Figure 23. The scaling of PFs has not been taken from the proposed standard, but selected as a possible set of values for differentiating access priorities. Also, the CW_{min} or CW_{max} limits are not considered. By using the PF[i] value 2, the algorithm is the binary exponential backoff presented in Figure 19. Value 1 leads to no change of CW[i].

4.3.2 HCF Polling

Under the centralised HCF, AP is able to assign TXOPs to network terminals. TXOPs may occur during both the contention and contention free periods. TXOP is defined as a maximum duration after the preceding frame that allocated the period. A terminal may transfer a multiple of frames within a single TXOP period.

TXOPs can be acquired several ways. A TXOP request can be send after a won HCF contention, when polled by AP during a CFP, or by utilising a Controlled Contention Interval (CCI). The controlled contention access is presented in Figure 24. CCI is initialised by AP transmitting a Controlled Contention (CC) frame. The CCI consists of CC opportunities (CCOP) that are timeslots where terminals can send Reservation Request (RR) frames to AP. The reservation attempts can be limited by using a priority mask in the CC frame. The priority mask defines the access priorities for which the requests can be sent during CCI. The selection of the used CCOP slot is done randomly with a terminal.

4.4. HIPERLAN/1

HIPERLAN/1 has included QoS support into the standard from the beginning. For a HIPERLAN/1 MAC user, the protocol provides two QoS parameters associated with MSDU transfer requests. The *user priority* states the priority of the MSDU data. The priority parameter has two possible values: high (value 0) and normal (value 1). The other parameter is *MSDU lifetime* that specifies the maximum transfer time for the user data until it becomes obsolete and can be discarded [23][80].

HIPERLAN/1 also utilises a third internal QoS parameter called *residual MSDU lifetime*. This parameter is used to specify the remaining lifetime for a MSDU. The mapping of queued MSDUs to different MPDU access priorities is done according to the user priority and the residual lifetime. The lifetime is normalised, which means that

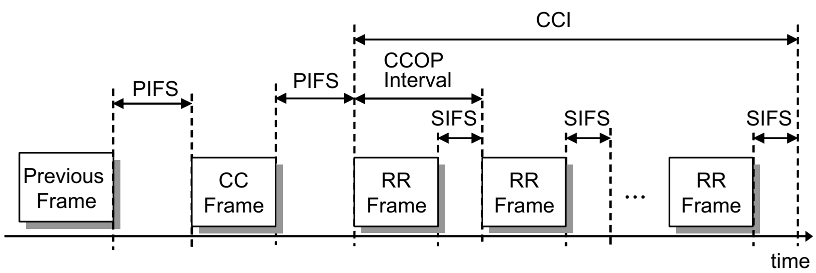


Figure 24. Controlled contention access in IEEE 802.11e.

Table 7. HIPERLAN/1 access priority mapping [23].

Normalised residual lifetime (NRL)	Channel access priority	
	User priority = 0 (high)	User priority = 1 (normal)
NRL < 10 ms	0	1
10 ms = NRL < 20 ms	1	2
20 ms = NRL < 40 ms	2	3
40 ms = NRL < 80 ms	3	4
80 ms = NRL	4	4

the value of residual lifetime is divided by the estimated number of hops to reach its final destination terminal. The mapping of access priorities is presented in Table 7.

HIPERLAN/1 utilises an Elimination-Yield Non-Pre-emptive priority Multiple Access (EY-NPMA) algorithm for channel access. The EY-NPMA operation is presented in Figure 25. The non-pre-emptive access means that once an access cycle has started, no new terminals are able to interfere with the ongoing cycle despite their access priority. EY-NPMA consists of three access phases: prioritisation phase, contention phase, and transmission phase. In addition, another form of the access cycle called free channel access cycle has been defined, in which only the transmission phase takes place [23]. A terminal or terminals with the highest access priorities survive the prioritisation phase. Next, the contention phase provides statistically equal probabilities of winning for the remaining terminals [72].

During the prioritisation phase, a contenting terminal having an access priority *n* shall sense the channel activity during *n* prioritisation slots. If the medium remains idle, the terminal starts to transmit a channel access burst. This informs any lower priority terminal to back off from the cycle. The access burst continues to the elimination phase of the contention phase. The elimination burst lasts for a selected number of elimination slots. The number of slots is a random variable, as the probability of transmitting on a single slot is given. After the elimination burst, a contenting terminal again senses the medium during the elimination survival verification interval. If the channel is concluded idle, the terminal has survived the elimination phase. Next, during the yielding phase the terminal senses the medium during a random number of yield slots. The remaining terminal having selected the smallest number of yield slots wins the contention by

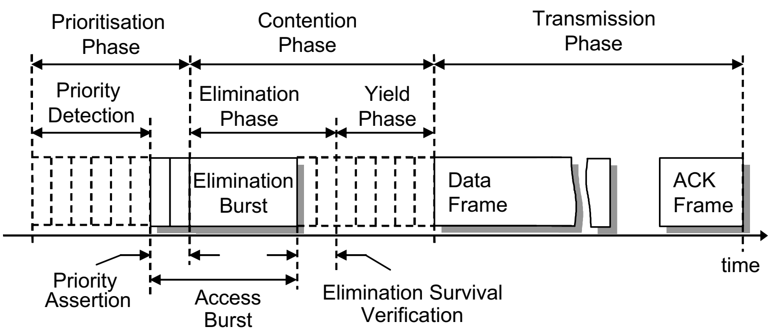


Figure 25. HIPERLAN/1 channel access cycle.

entering the transmission phase. The transmission phase contains the transmissions of an MPDU and the corresponding acknowledgement from the receiving terminal [10].

4.5. HIPERLAN/2

HIPERLAN/2 MAC protocol is based on reservation TDMA. The MAC protocol has a centralised architecture, despite whether the topological architecture of the network is centralised or direct [24][25]. The basic structure of the TDMA frame is shown in Figure 26. The frame consists of broadcast, downlink, direct link, uplink, and random access phases. The duration of the whole access cycle is fixed to 2 ms, but the boundaries between the separate phases are flexible and can be adapted according to the traffic load.

The broadcast phase carries network information, such as power levels and AP identifier, and defines the structure of the TDMA frame and the reservation status of time slots on different link phases. During the downlink phase, MPDUs are transmitted from AP to terminals and the uplink phase contains the reverse direction. The direct link phase is optional, and used for direct terminal to terminal communications.

The random access phase contains uplink slots that are used for terminals to send messages to AP when they do not have reserved uplink slots. The phase is divided into time slots that are utilised using the slotted ALOHA protocol with binary exponential backoff after collisions [115].

The basic unit of transfer is an ATM cell payload of 48 bytes that is transferred in 54 bytes long MPDUs. For increasing efficiency, several MPDUs from a singe terminal can be multiplexed into a single MPDU train [28][103][21]. Selective repeat ARQ with CRC calculation over MPDUs are utilised by the EC. Also, the physical layer performs convolutional FEC coding [115].

The HIPERLAN/2 MAC protocol is connection oriented. The QoS for each connection is a result of resource requests sent by mobile terminals and an AP scheduling function that allocates time slots for the DLC connections of a terminal. For scheduling, the AP concerns the amount of queued traffic at each terminal and the QoS definitions for a connection [84].

As HIPERLAN/2 has been targeted at the support of the QoS classes defined for ATM networks for virtual connections [31][28]. Each HIPERLAN/2 connection can thus be associated with separate parameters for throughput, delay, delay variation, and bit error

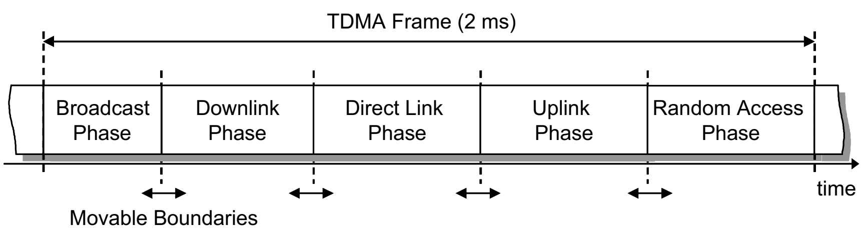


Figure 26. HIPERLAN/2 MAC TDMA frame.

rate [115]. For compatibility with IEEE 802 based LANs, an Ethernet service specific convergence sub layer has been specified for HIPERLAN/2 [27]. The support of the IEEE 802 based QoS is optional in the convergence layer. If supported, the different 802 traffic classes are mapped into separate HIPERLAN/2 DLC connections.

4.6. Bluetooth

In Bluetooth, the basic QoS differentiation is made by selecting the used link type. The SCO link provides a synchronous data link connection for time bounded data, while ACL provides connectionless transfer.

A SCO connection has a fixed throughput, but the QoS can be affected by packet type selection. A packet type defines the applied error recovery processing for the data transfer. The selection is also a parameter for ACL links, for which it results into different data throughputs.

The Bluetooth data processing contains encryption and retransmissions with different error detection and acknowledgement policies. Two levels of FEC coding can be applied. A frame header in Bluetooth is protected by a strong (1/3) repetition FEC that can be used for audio payload as well. FEC (2/3) scheme is a shortened Hamming (15,10) code that can be utilised for both audio and data payloads.

For ACL, the Bluetooth user QoS is defined at the L2CAP layer. The layer must also monitor the utilised resources and ensure that QoS contracts are honoured. Currently, L2CAP is required to provide only a best-effort service without QoS guarantees [11]. The L2CAP QoS negotiation between Bluetooth terminals uses a QoS flow specification, of which definition is similar to [94]. As L2CAP performs multiplexing of several flows, each flow can have a separate QoS configuration. The main parameters of a Bluetooth flow specification for defining transfer QoS are presented in Table 8 [16][11].

The Bluetooth baseband utilises TDMA, in which the medium time is divided into 625 μs duration timeslots. The Bluetooth channel access is depicted in Figure 27. A continuous transmission of a frame can last over 1, 3, or 5 timeslots and some operations during connection establishment use half-size timeslots. In basic operation

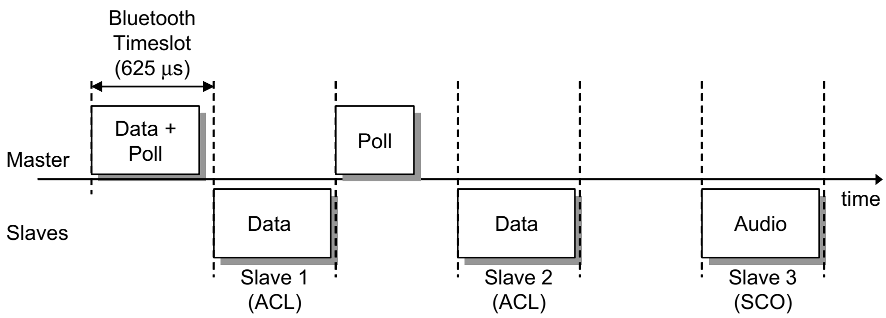


Figure 27. Bluetooth channel access.

Table 8.Bluetooth L2CAP flow specification.

Parameter	Values	Description
Service type	No traffic Best-effort Guaranteed	Defines the level of service requested. For the best-effort, the other parameters can be ignored.
Token rate	Rate in B/s	A Bluetooth user may send data at this rate continuously
Token bucket size	Size in B	Defines the burst size
Peak bandwidth	Bandwidth in B/s	Maximum data rate for a Bluetooth user
Latency	Latency in µs	Maximum delay for a bit through a Bluetooth terminal. The delay ending point is when the bit is first time transmitted by the radio.
Delay variation	Variation in µs	Maximum difference of the minimum and maximum latencies that a packet may experience

with 1 timeslot packets, all Bluetooth terminals on the same piconet hop to a new frequency channel for every slot according to the Bluetooth FHSS scheme. A master of a Bluetooth piconet always starts transmissions at even numbered timeslots. A slave is able to respond to the master using an odd numbered slot if it has been addressed in the preceding even numbered slot.

The transmission opportunities in Bluetooth are scheduled by a piconet master. For SCO links, the timeslots are reserved. In this case, a slave is able to transmit on the reserved slot regardless of whether it has successfully received a master transmission during the previous slot.

For controlling the transfer of different L2CAP flows, only a single ACL link is available at the baseband layer between a master and a slave terminal. The QoS of the ACL link for data transfer is defined by a LMP parameter called *poll interval.* The poll interval defines the maximum duration between subsequent transmissions from the master to a slave on the ACL link. Also, a parameter defining the number of repetitions of broadcast traffic packets is part of the LMP QoS configuration.

4.7. Summary

The properties of a wireless medium and the development of LAN applications towards multimedia results in the need of QoS support in WLANs. This need is widely recognised by the WLAN standardisation and specification bodies, as well as research institutions. The QoS support proposals for WLANs are strongly dependent on topology. The emerging IEEE 802.11e will support QoS in both centralised and distributed topologies, which has resulted in relatively complex functionality.

The QoS parameters, their possible values, and the access control functionality for the introduced technologies are summarised in Table 9. In general when concerning the state of the current products available, the best-effort service is supported by Bluetooth and IEEE 802.11 WLAN, while the rest of the standard technologies are still emerging.

Table 9. QoS parameters and access control mappings in different WLAN/WPAN technologies.

<table>
<tr><th>WLAN</th><th>MSDU QoS parameters</th><th>MSDU QoS parameter values</th><th>QoS in access control</th></tr>
<tr><td rowspan="4">802.11</td><td rowspan="2">Priority</td><td>Contention</td><td rowspan="2">DCF
PCF</td></tr>
<tr><td>Contention free</td></tr>
<tr><td rowspan="2">Service Class</td><td>Strictly ordered</td><td rowspan="2">-</td></tr>
<tr><td>Reorderable multicast</td></tr>
<tr><td rowspan="2">HIPERLAN/1</td><td>User priority</td><td>High/low (0/1)</td><td rowspan="2">Access priority
0-4</td></tr>
<tr><td>MSDU lifetime</td><td>0 - 16 000 ms</td></tr>
<tr><td rowspan="2">802.11e</td><td>Priority
or
Tspec</td><td>Contention free (6)
Contention (0)
Priorities (0-7)
Traffic category (8-15)</td><td rowspan="2">Access priority in contention HCF
TXOP reservation in polled HCF
Controlled contention</td></tr>
<tr><td>Service class</td><td>Not used</td></tr>
<tr><td>Bluetooth</td><td>Flow specification</td><td>Token rate
Token bucket size
Peak bandwidth
Latency
Delay variation</td><td>Poll rate
Packet type
Link type</td></tr>
<tr><td>HIPERLAN/2</td><td>Connection parameters</td><td>Connection parameters</td><td>AP scheduling decisions</td></tr>
</table>

5. Introduction to TUTWLAN

TUTWLAN research has been carried out in the Institute of Digital and Computer Systems at Tampere University of Technology (TUT). The work started in 1997 consisting of different areas of local wireless data communications. During the research work, a new WLAN called TUTWLAN has been developed. TUTWLAN research work has consisted of link level protocols, hardware demonstrators, TUTWLAN management and security architectures, and applications for demonstration and testing purposes. A seamless design flow from protocol specification to the implementation has also been a key research topic.

TUTWLAN research covers the technological areas of WPAN and WLAN. Bluetooth and other WPAN standard technologies were not available at the time when most of the TUTWLAN design was carried out. Still, many of the design objectives for TUTWLAN have been similar to those later assigned to WPANs. The purpose of TUTWLAN has been to develop a scalable technology, enabling a wide range of applications from multimedia laptops to small handheld devices, and further to wireless sensors. This application area introduces different throughput and other QoS support requirements for the implemented data transfer.

A set of design requirements for TUTWLAN were concluded from the demands of the target environment, projected applications, and the from the interoperability requirements with fixed networks. Also, the characteristics of the available WLAN related radio technology was concerned. The concluded design requirements are presented in Table 10. The motivation has been to develop a technology having relatively simple architecture and efficient implementation. Consequently, a central requirement is the capability to support real time services [P4].

Table 10. Design requirements of TUTWLAN.

Objective	Design requirement
Multimedia services	❑ Supporting of MAC user requested QoS for data transfer ❑ Support for real time services
Service security	❑ Utilisation of configurable security ❑ Wired equivalent security level ❑ End-to-end secure service between terminals
Inter-operability	❑ Hiding of the complexities of wireless medium ❑ Interoperability with legacy Internet protocols ❑ Interoperability with existing network infrastructure
Low cost	❑ Modular design ❑ Reconfigurability ❑ Off-the-shelf components ❑ Unlicensed frequency band ❑ Seamless design flow
Power saving	❑ Power saving functionality into the wireless protocols ❑ Low power consumption components
Simple control and management	❑ Centralised control and management ❑ Limited number of simultaneous connections

5.1. TUTWLAN Topology

The basic network architecture of TUTWLAN consists of a pico sized radio cell. The cell radius depends on the radio layer. The maximum transmission power level of the 2.4 GHz ISM bands restricts the transmission power to 100 mW in Europe, thus resulting in a radius of tens of meters in indoor environments.

A TUTWLAN cell is controlled and managed by a TUTWLAN *base station*. A base station also provides a connection to a wired backbone network, generally to a 802.3 LAN (Ethernet). This interconnection is enabled by the *TUTWLAN AP* module of the base station. A number of *portable terminals* can register to a base station and gain access to the wireless cell and to wired LAN resources through the base station. The TUTWLAN topology, with basic connections and operational modes, is depicted in Figure 28.

There are two types of applications present. A network application refers to legacy applications that usually operate on top of the IP protocol suite. The management application is a local control and management entity in a portable terminal or the cell management entity in a base station. In the TUTWLAN demonstrator, the management application also provides a user interface [P14]. Similarly, other applications can operate directly on top of TUTMAC protocol. These applications are referred to as *native* applications in the publications of this Thesis.

A portable terminal refers to any device that contains the necessary physical network adapter and protocol functionality for TUTWLAN access. For example, a wireless microphone application should include only the minimal functionality for connection establishment and data transfer.

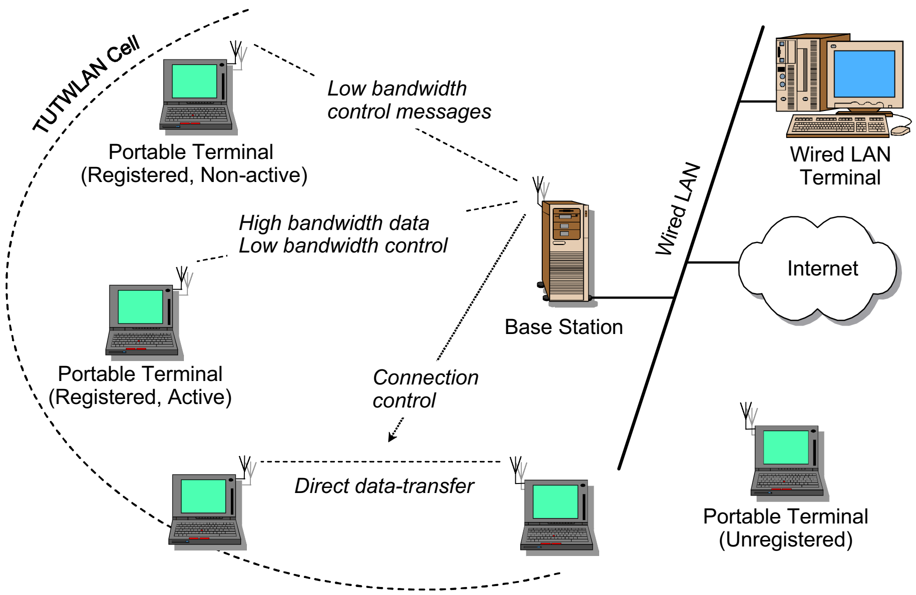


Figure 28. TUTWLAN topology.

A portable terminal can appear in two modes: *registered* and *unregistered*. The registration procedure involves a terminal authentication and the corresponding updating of the TUTWLAN cell topology. The registered mode of a portable terminal can be seen as logically equivalent to a wired network connection.

According to the current data transfer capability, a registered terminal can further reside in active and non-active modes. A non-active terminal is able to exchange low bandwidth control and management messages with a base station and to receive broadcast data. However, a non-active terminal has no dedicated access for data transfer services. A non-active terminal can utilise power saving mode where parts of the system components, such as the radio subsystem, can be temporarily powered down. By acquiring a data transmission opportunity, a terminal proceeds into the active mode. The designed signalling between TUTWLAN base stations do not support seamless handovers for active mode portable terminals. Instead, handovers are expected to occur when a portable terminal resides non-active.

In normal operation, the base station forwards all traffic inside a TUTWLAN cell, also between the active portable terminals. However, it may be possible for a portable terminal to reliably communicate with another terminal directly depending on the operation environment. Thus, for performance improvement a direct data transfer operation between two portable terminals has also been designed as an option. This type of transfer is possible only under the control of a common base station. The direct transmission leads to improved medium utilisation efficiency, but introduces a more complex network topology [4].

A base station is a natural network component for an interconnection to outside network resources. While it is not necessary to provide the access through a controlling station, for traffic balancing it is an efficient solution. In a network with laptop computers, most of the traffic is expected to be transferred downlink from a base station to portable terminals.

Because of the central placement and possibly more efficient transceiver, the base station enables a fairer access between portable terminals. Also, this enables a simpler implementation of management operations, such as security initialisation and power saving.

Due to the centralised control, the data transfer QoS support can be more efficiently implemented [9][8]. However, it is recognised that the QoS negotiation between a portable terminal and TUTWLAN base station is a source of overhead, and the scheduling algorithm of the base station is the central function for QoS support. The design of the scheduling function needs to combine the QoS support with efficient medium utilisation.

5.2. TUTMAC Protocol

TUTMAC is the MAC protocol designed for TUTWLAN. TUTMAC is a dynamic reservation TDMA based protocol [P4]. The protocol consists of two entities, as the medium is accessed by exchanging control messages between portable terminals and a base station. Also, a practical implementation of dynamic reservation TDMA requires a central control point. The complexities of the medium and the QoS support requirements are taken into the MAC design.

The protocol is divided into data and management planes, as depicted in Figure 29 that presents the functional architecture of TUTMAC. The data transfer service is assembled using the required set of functions on the data processing plane, under the control of management plane. The main functional blocks of the protocol are the interfaces, data processing, and management. In addition, the Management Information Base (MIB) is referred to as a separate module. The interface blocks provide a Service Access Points (SAP). A single SAP is provided from the network user interface for general higher layer protocols and TUTMAC users. Another SAP is available for a protocol management user (M-SAP). Native user SAPs (N-SAP) are similar to the basic SAP, but provide a separate identification for each user. The radio subsystem layer is accessed through the physical layer SAP (P-SAP) and the physical management SAP (PM-SAP) enables radio control.

Both a base station and a portable terminal share common functional blocks. In addition, the station specific functions are added to common functional architecture. These are the base station data processing and management blocks. In the functional reference model, these blocks are considered to contain all the base station specific functions that complement, utilise, or replace the TUTMAC functionality of a portable terminal.

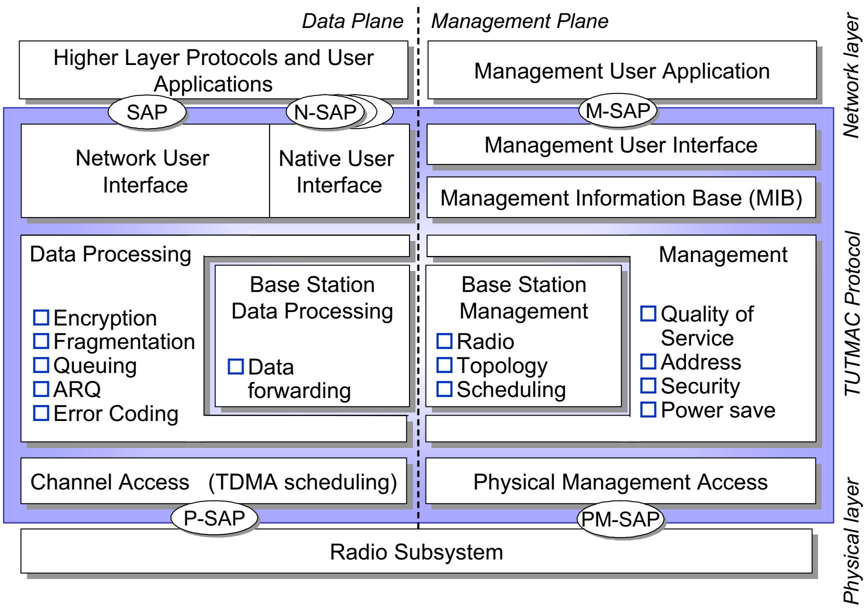


Figure 29. Functional architecture of the TUTMAC protocol.

5.2.1 Data Processing

The functional blocks in the data processing module have a uniform interface, thus they can be separately applied for a MSDU according to the assigned QoS.

Encryption can be applied on MSDUs for enabling data privacy. The function can be configured to utilise different algorithms and encryption keys and the same algorithms are applied for authentication. The two algorithms originally specified for TUTWLAN are the Improved Wireless Equivalent Privacy (IWEP) and International Data Encryption Algorithm (IDEA). IDEA is a well-known strong encryption algorithm, while IWEP has been developed in the project to be used as a default algorithm for TUTWLAN encryption [102][39][37][38].

Fragmentation is applied on the MAC level for adapting MSDUs to the wireless medium. The frames are divided according to the slot structure of the TDMA cycle. In addition, shorter frames are less likely to encounter transfer errors caused by temporary fading or interference peaks on the radio channel and are more suited for faster retransmissions when such an error is detected [96].

MPDUs are buffered on separate queues according to assigned TUTMAC specific priority value before the channel access and transmission. The further development of queuing algorithms for TUTMAC is projected in future. Next, ARQ scheme is used in TUTMAC for recovering from the occurred transfer errors. The utilisation of ARQ depends on the configured QoS support. Thus, it is necessarily not used for all transmissions or the maximum number of retransmissions varies. FEC coding is the second method for recovering from transmission errors. In TUTMAC, FEC code is applied separately for frame headers that contain vital information for addressing and frame structure interpretation [P7]. For the frame payload, no specific algorithm has been yet assigned.

The data forwarding function is needed in a TUWLAN base station for the forwarding of frames between portable terminals inside a single TUTWLAN cell. MSDU fragments are not reassembled during forwarding, but MPDU retransmission requests are utilised for recovering from transmission errors.

5.2.2 Protocol Management

The management block controls the data processing functions and performs various network management procedures, such as authentication and registration, power saving, address translation, network environment scanning, and controls TDMA reservations.

During a system start up, the management module initialises the operational configuration of the system and provides required management parameters for the other modules. The TUTMAC MIB functions as an interface to a higher layer management entity (e.g. TUTMAC management user application). The MIB is also constantly updated according to the self initiated changes in the protocol configuration [P18].

The security function of the management block controls the encryption, decryption, and authentication operations. The initialisation of the algorithms, as well as the exchange and storage of encryption keys are handled by the function. The power management controls the utilisation of the power save modes. There are two power save modes specified. The first mode can be entered without explicit control signalling, as the protocol is aware of the slot reservations of the next TDMA frame. For the second mode of power saving, a portable terminal informs the base station of the duration of the power save period. The base station in a TUTWLAN cell controls the radio environment, such as the used link speed and the utilised radio frequencies.

The topology function of a base station and the address management function of both station types are closely related. During the terminal registration, a portable terminal is added to the wireless cell topology. For compatibility, TUTWLAN utilises the 48-bit IEEE MAC address format for identifying terminals [58]. After the registration procedure, these addresses are translated into 8-bit TUTWLAN terminal identification (TUTWLAN ID) values that are utilised for addressing TUTWLAN terminals. This procedure preserves wireless bandwidth. Also, with the encryption of registration messages, the address translation can improve the location privacy of TUTWLAN terminals, as TUTWLAN IDs are temporary values assigned by the base station.

The QoS management of a portable terminal is associated with the scheduling function of the base station. The QoS support in TUTMAC is discussed later in this chapter.

5.2.3 TUTMAC Channel Access

The design starting point for the TUTMAC channel access mechanism has not been to optimise the medium utilisation efficiency. Instead, the protocol targets support negotiated QoS enabled by the centralised TDMA, and to improve the dynamic performance of the used TDMA [103]. For future research purposes, the target has been to design a protocol platform suitable for experimenting with different MAC algorithms.

The TDMA slots repeating in each access cycle form fixed bandwidth channels. Thus, the TUTMAC protocol utilises four types of channels according to their purpose: *beacon*, combined *management/data*, *data*, and *contention* channels. These are presented in Figure 30.

Each access cycle begins with a beacon channel carrying a beacon frame. The frame contains the cell specific information, such as network ID, structure of the access cycle, and reservation status for data channels. A beacon frame is also used with portable terminals for acquiring synchronisation to the TDMA access cycle structure. Each reserved data slot is identified by the TUTWLAN ID of a terminal and the direction of data channel (uplink/downlink). The base station buffers data for terminals utilising the second mode power save mode. A terminal also receives an indication from buffered data by beacon frames. This wake up beacon for a terminal has been negotiated with the base station prior entering the power save mode.

A data channel carries two types of frames, MPDUs with user data payload (data frames) and ACK frames. A transmitted data frame is instantly acknowledged at the end of the same data slot by the receiving terminal if ARQ scheme is utilised. For ARQ, TUTMAC utilises both positive and negative acknowledgements. A missing acknowledgement is also interpreted as negative. An acknowledgement frame after a single data frame implies a simple stop-and-wait flow control protocol [109]. In

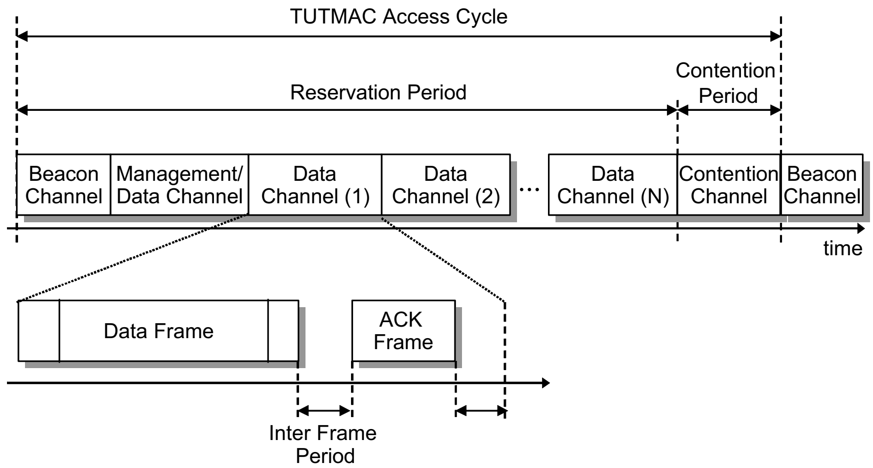


Figure 30. TUTMAC channel access cycle structure.

addition, the acknowledgement frames carry flow control information utilised by the scheduling function of the base station. This information consists of the amount and the priority of data that is currently queued for transmissions in a portable terminal.

The combined management/data channel following the beacon is mostly used for data frames, but can be reserved for broadcast transmissions and management messages used e.g. during authentication and registration procedures. The limiting of the control and management signalling to take place in this separate channel enables an undisturbed data exchange in the following data channels.

The uplink contention channel is required by a portable terminal for initiating the registration procedure with a base station and for channel reservation requests. The contention channel carries only short control frames, while longer management messages require the allocation of the management channel. The contention channel consists of a series of short contention slots, as depicted in Figure 31. TUTMAC introduces two alternatives for contention, slotted-ALOHA [112] and carrier sensing based schemes. The contention method is chosen according to the availability of the carrier sensing functionality of the radio subsystem. With a simpler slotted-ALOHA implementation, each slot in the contention channel lasts for the duration of the maximum size frame admitted in the channel. Consequently, a contending terminal chooses a slot randomly and transmits within a single slot period.

The carrier sensing increases the reliability against collisions. The contention slot is divided into short CSMA slots, corresponding to the time required by the radio layer carrier sensing to function. Before transmitting, a contenting terminal senses one or more contention slots depending on the priority of the control frame to be transmitted. Channel reservation requests have a higher priority against other messages.

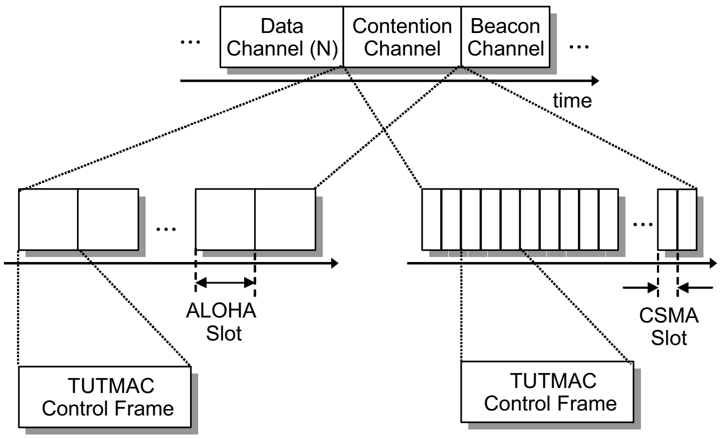


Figure 31. TUTMAC contention channel access.

5.3. TUTMAC QoS Support Architecture

TUTMAC introduces a three-layer QoS support architecture, consisting of *service*, *connection*, and *channel access* layers, as presented in Figure 32. The data transfer QoS requirements for the TUTMAC protocol originate from application requirements, QoS signalling of higher layer protocols, and from peer MAC protocols as presented in Chapter 3.

TUTMAC QoS support is controlled using three QoS parameter groups. These groups are called *service type*, *security type*, and *connection type*. Each type can consist of several parameters. The types and their main parameters are summarised in Table 11.

The higher layer QoS control information is input to TUTMAC as a service type ID parameter of the MSDU transfer request primitive. Secondly, the service type assignment for SAP MSDUs can be controlled by the management application entity through M-SAP. Native applications are TUTWLAN aware, and can thus control the QoS parameters for each N-SAP.

The service type ID parameter maps the MSDU into a TUTMAC service class (called service type). In link layer networking, the parameter carries the user priority acquired from a peer MAC. In a TUTWLAN cell, the data processing associated to a particular service type is configured by the management application of a base station, and the configuration information is transferred to registered portable terminals using management messages.

A service type is constructed with error recovery control and TUTMAC priority. Also, the acceptability of residual errors is an attribute of a service type. The error recovery attributes define the usage of ARQ and FEC schemes. The priority of MSDU inside the TUTMAC protocol is derived by two attributes: TUTMAC priority and ageing time. If an aging time of an MSDU expires during queuing, the data is discarded.

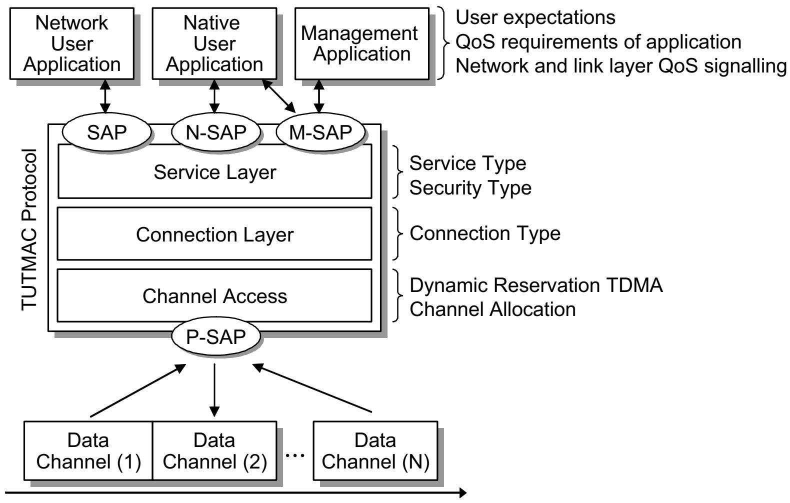


Figure 32. TUTWLAN QoS support architecture.

The *security type* defines the level of security utilised for protecting data transmissions against eavesdropping. The parameter is set by the local management, and it is associated with the supported encryption algorithms. TUTMAC data transfer can occur in three different security levels: open, cell, and secret key. The open security provides no cryptographic operations. The cell security defines a wired LAN equivalent level of security, i.e. terminals in the same cell share a common encryption key. The third security level, secret key, enables a secure transfer between two terminals of the same cell [102][39][37].

The connection type defines the requirements for a wireless connection between a portable terminal and a base station. In the current design, a single connection type definition is used for a terminal. The connection type aggregates the requirements of all traffic flows of a terminal originating from the network user SAP. N-SAPs are projected to support separate, user specific connection types in future research work.

The connection parameters are negotiated during the connection reservation procedure, and can be changed during the active mode. If required resources cannot be provided by the base station, the reservation is unsuccessful. The assignment of the connection type for the TUTMAC protocol is done by the management application entity that can be a user application, such as a native TUTWLAN application, or a higher layer control entity.

The parameters of the connection type are the required throughput, requested throughput, acceptability of changing throughput, and the permanency of the connection. The first two parameters define the minimum and the requested throughput by the number of data channels. The changing throughput state whether the connection throughput can vary between the required and requested. The connection permanency attribute affects to the automatic releasing of a reserved connection during idle periods. Delay parameters are currently not explicitly defined. The delay can be conducted from

Table 11.TUTMAC QoS support parameters.

Type / parameters	Description
Service type	Group of parameters defining the error control of user data transfer
Security type ID	Defines the utilised service type (can also assign only the priority)
Priority	TUTMAC priority for MSDU
ARQ	Defines the maximum number of retransmission attempts
FEC	Defines the used FEC algorithm
Ageing time	Defines the queuing time after the MSDU becomes obsolete
Connection Type	Group of parameters defining the connection requirements between a portable terminal and base station
Required throughput	Number of data channels required for a connection
Requested throughput	Number of data channels requested for a connection
Changing throughput	Acceptability of a changing throughput with the minimum of required channels
Permanency	The expected permanency of a connection
Security Type	Group of parameters defining the security level for authentication and data privacy
Authentication	Defines the need for encryption based authentication
Data security	Defines the user data privacy by security level. A security level is associated with the used encryption algorithm

the structure of the underlying TDMA structure and active connection type. In addition, the ageing time defines a delay bound for an MSDU transfer.

In the channel access, the connections are mapped onto the actual transmissions of MPDUs within TDMA slots. The number of allocated slots for a terminal can be changed dynamically by the base station scheduler function according to the connection type. A data channel can be allocated to a single portable terminal connection or shared with several portable terminals. Idle time slots are similarly allocated to existing connections, which increases the utilisation of the medium.

The structure of the access cycle can also be adapted for different operational environments and network data transfer requirements. The number of data channel slots and the length of the slots in a cycle can be changed, which results in the increase or decrease of the whole access cycle duration. Generally, the duration is minimised, which increases overhead but enables shorter access delays.

5.4. TUTWLAN AP

The TUTWLAN AP operates on top of TUTMAC and other connected MAC layer protocols, thus performing bridging by forwarding frames between connected LANs [P8][P10][79]. To save wireless bandwidth, AP also filters traffic. QoS support for the forwarded traffic has been designed for the TUTWLAN AP module. The functional architecture of the module is divided into four separate blocks: frame reception and sending functions, and forwarding and queuing processes. The functional architecture of the AP module and the internal packet structure are presented in Figure 33.

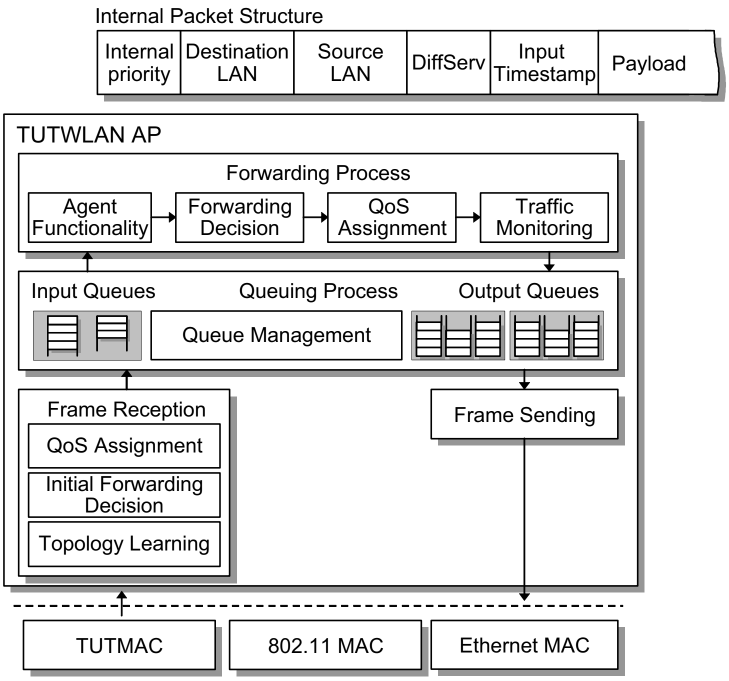


Figure 33. Functional architecture and the internal packet structure of the TUTWLAN AP module.

The frame reception makes the initial forwarding decision when a packet is received from one of the MAC protocols of the connected network interfaces. The network topology in TUTWLAN AP is constructed by the IP and MAC addresses of received frames and saved into the bridging table of AP. For the initial forwarding decision, the bridging table is searched for the destination MAC address. If the address is found but the network interface associated to it is the same as the source segment, the frame is discarded to prevent unnecessary traffic. Otherwise, the processing of the frame is continued.

When the frame is selected for forwarding, an internal packet is allocated for the frame. An internal packet contains the received MSDU and a number of AP specific header fields as presented in Figure 33. If the received frame is coupled with QoS signalling fields, such as the 802 user priority and the DiffServ value, the internal packet priority for the first queuing is assigned accordingly. The mapping can be configured by an AP management entity. In TUTWLAN, the management application of the TUTWLAN base station contains also the AP management entity.

The forwarding process bridges packets between network interfaces and performs traffic control functions. The first operation of the process is possible agent functionality. An agent of TUTWLAN AP operates as an intermediate node between two end terminals. The purpose is to decrease the load of wireless network, as an agent can appear and operate on a wired LAN on behalf of wireless terminals. The traffic monitoring function handles the collecting of AP management information.

If the destination address cannot be found from the bridging table at the frame reception, the decision of forwarding is done in the forwarding process. When topology information is not available, the frame must be relayed to all network interfaces, except the one it was received from.

If supported, the QoS fields of exchanged frames are usually assigned by network terminals. The TUTMAC protocol assigns the frame service type ID and higher layer entities can assign the DiffServ values. Also, tagged IEEE 802 frames may contain user priorities and other wireless technologies, such as Bluetooth and HIPERLAN/1 that contain their specific QoS signalling.

TUTWPLAN AP targets the adaptation of the QoS signalling and the corresponding support of different LAN segments. For TUTWLAN AP, different algorithms for assigning an internal priority and adapting the QoS signalling parameters will be researched in the future. The control of priority mappings has been designed to operate dynamically. Therefore, the actual numerical values depend on the local management configuration of AP, and on the configurations of LAN domains. The number of internal priority levels can be dynamically changed by the AP management. In Figure 34, one example for service naming and mapping between the IEEE 802 user priorities, TUTWLAN AP priorities, and TUTMAC service types is presented.

Internal packets are first queued between the frame reception and the forwarding process, and the second queuing takes place between the forwarding process and the sending function. For the first queuing, the internal priority based queues are shared by all frames. For the second queuing, separate internal queues according to supported priorities exist for each network interface.

The main task of the queue management function is to schedule the most suitable packet for the forwarding process and for the frame sending function. Currently, the scheduling is based on the packet priority first assigned by the receive handler and later updated by the forwarding process. The queue management can select packets for discarding, if AP load grows and the queues approach overflow. Also, each packet is associated with the

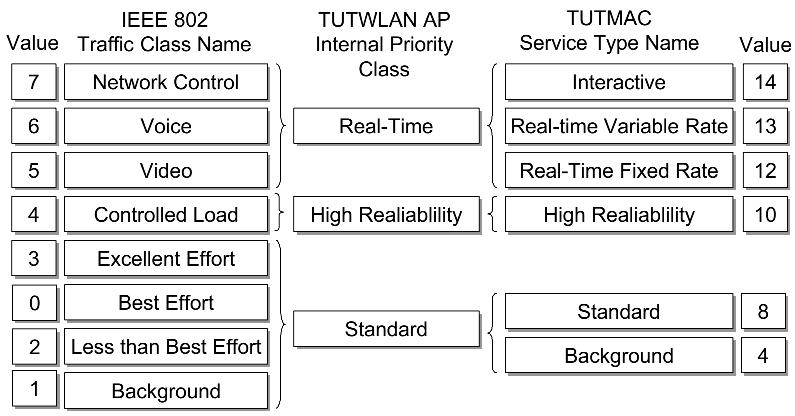


Figure 34. Example mapping of TUTWLAN AP priority with TUTMAC and 802 QoS signalling.

time it was input to the queuing process. If the queuing time exceeds a threshold for ageing, the packet is discarded.

5.5. TUTWLAN Test Case: a Wireless Video Demonstrator

TUTWLAN allows higher layer entities to control the QoS support functionality of the underlying wireless link. For testing and developing of the QoS support functionality of TUTWLAN and TUTWLAN AP, a wireless video demonstrator has been implemented [P11][P12]. The video demonstrator operates on application and transport layers, thus also enabling the development of a transport layer control protocol for real-time services. Another part of the demonstrator is the development of error resilience video encoding and decoding [82].

The wireless video demonstrator performs end-to-end video streaming. The layer architecture of the demonstrator is depicted in Figure 35. Its basic structure is a client-server system, in which the connection between the entities can be realised using different network technologies. The client is referred to as a video receiver and the server as video sender. The video receiver is a terminal with varying processing capabilities and network connection parameters. The video sender is considered a high capacity terminal.

The basic functions of the video encoder/decoder layer are to adapt a stream at the sender and to conceal occurred errors at the receiver end. The encoded stream is prepared for the wireless transfer by the video adaptation layer. It contains a video stream parser and assembly functions for enabling unequal error protection by FEC and retransmissions.

A wireless link can be directly available between a video sender and receiver, for example through TUTWLAN or Bluetooth technologies [P21]. On the other hand, the end-to-end transfer is in many cases created by concatenating different technologies, both wireless and wired. The configuration for video protection depends on the network/link technology.

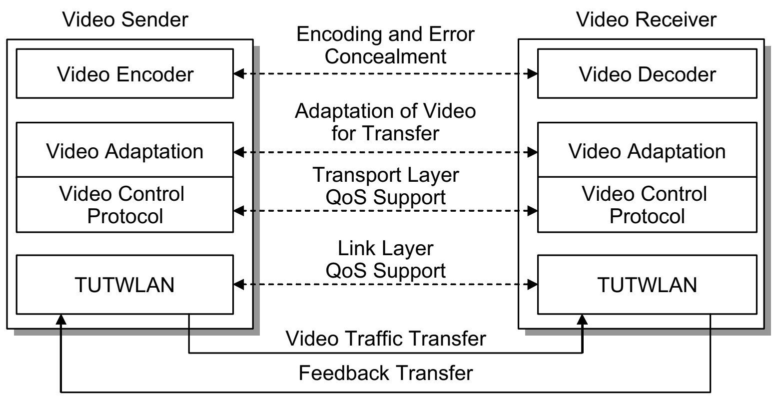


Figure 35. Layer architecture of the wireless video demonstrator on TUTWLAN.

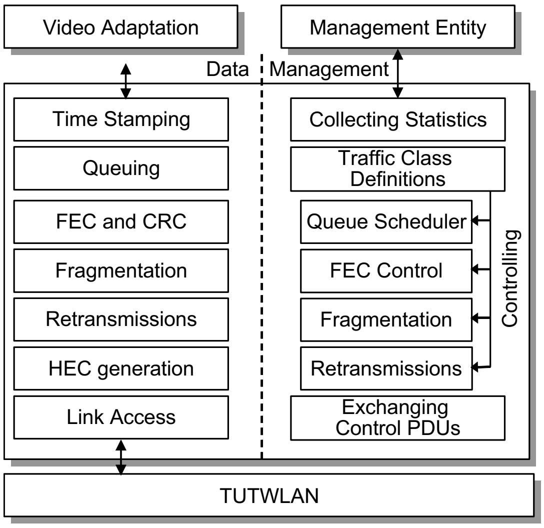


Figure 36. Functional architecture of VCP.

The control and support of video transfer over wireless links are managed by the Video Control Protocol (VCP) that is a central entity of the demonstrator. VCP contains its internal data processing functionality, and it can also control lower layer protocols and higher layer video encoding. The functional architecture of VCP is presented in Figure 36 [P12].

VCP has been divided into two planes for management and data processing functions. The central control module of VCP is the traffic class definition. The class defines what functional layers of the data processing are applied and controls their operations. VCP collects statistical information of its operation. The information is delivered to a higher layer management application for further processing, displaying, and storing. The connection establishment, synchronisation, protocol configuration, and measurements require the exchanging of control PDUs.

When a video packet is given to VCP, the packet is time stamped and put into a queue. After the queue scheduler takes the packet from the queue, it is prepared for sending by adding redundancy FEC coding, CRC fields, and by fragmenting the packet into smaller segments, if these are required by the used traffic class or by the underlying link. A separate Header Error Check (HEC) is calculated over the VCP packet header to verify an error free packet header despite the occurred payload errors.

5.6. TUTWLAN Implementation

The TUTWLAN design is being verified by implementation and measurements. The implementation covers hardware platforms, and software modules for the platform and for a host computer. The host computer for a TUTWLAN terminal is a Personal

Table 12.Main parts of the TUTWLAN implementation.

TUTWLAN hardware	SDL software	Host software
TUTWLAN platform hardware	TUTMAC protocol	TUTWLAN management application
Platform interface logic	TUTMAC protocol environment functions	Device driver for the TUTWLAN platform
Hardware accelerators for TUTMAC functions		TUTWLAN AP
		Wireless video demonstrator

Computer (PC) with Windows NT operating system. The main parts of the TUTWLAN implementation are summarised in Table 12.

The current TUTWLAN hardware platform contains an IEEE 802.11b compatible radio subsystem [2], a MAC processor for the TUTMAC protocol, and programmable logic [77]. Interface logic for connecting the different hardware blocks together has been implemented.

Specification and Description Language (SDL) [68][91] has been used for the TUTMAC protocol specification, simulations [P19], and implementations [40][42][41]. Automatic SDL to C code conversion has been utilised for generating an executable protocol [P15][P16][113]. In addition to SDL, logic designs for accelerating the most critical TUTMAC functions have been designed [36][37][39][38][35].

The SDL implementation of the TUTMAC protocol has been maintained platform independent. The adaptation of the generated protocol on the MAC processor of the TUTWLAN platform is done by environment functions [P17].

For connecting the TUTWLAN platform with a host computer, a Windows NT device driver has been implemented [P13][78]. Also, a protocol driver is used for enabling the communication between the device driver and TUTWLAN management application. The management application is used for both the local terminal control and TUTWLAN cell management [P14][P18][99]. The TUTWLAN AP has been implemented as a protocol driver in the Windows NT host computer [79][P22]. Also, the wireless video demonstrator software has been implemented for the Windows NT host computer environment.

The main parts of TUTWLAN implementation are operational, except for the hardware accelerators for TUTMAC functions that have recently been completed and are under testing. The current implementation state of the TUTWLAN platform has been verified to be operational with simplified (*lite*) versions of the TUTMAC protocol. Accurate synchronisation to the TDMA structure has been unavailable because of the uncompleted TUTMAC accelerator functions. However, the overall performance evaluation is now beginning. Further development will be carried out with the TUTWLAN integration and measurements.

The TUTWLAN AP module functionality is undergoing tests. Its performance is preliminarily estimated in [P10] and in [P22]. The wireless video demonstrator is operational and is being extended with new features. Several publications of its performance with different link technologies, including simulated links, are being prepared, but first results are already presented in [P11][P12].

6. Summary of Publications

This thesis consists of twelve publications that are based on the work of the author between the years 1997 and early 2002. The supplementary publications give further background information and details. The work concentrates on the various parts of the TUTWLAN research work. In this chapter, a summary of each publication is given and the contribution of the author clarified.

The publications can be divided into three sections. Publications [P1]-[P3] discuss the need to integrate QoS support for WLAN and WPAN technologies. Second, the designed QoS support and its implementations for TUTWLAN are presented in publications [P4]-[P10]. Supplementary publications [P13]-[P20] give further implementation details. Publications [P11]-[P12] present the design of the video transfer demonstrator application and discuss its use as a test application for QoS support. Supplementary publications [P21]-[P22] give further details on testing.

Publication [P1] evaluates the different available and emerging wireless data communication technologies, covering the WWAN, WMAN, WLAN, and WPAN fields. The publication gives perspective for the capabilities and future service possibilities of the wireless technologies. Evaluation of QoS support and the interoperability of QoS between different network technologies is the central theme of the publication.

The author provided the main contribution for the evaluation of technologies. Prof. Timo Hämäläinen co-authored the publication, and contributed especially to the classification of the different wireless technologies from the service point of view. Markku Niemi M.Sc provided technical details for the writing and improved the writing style with Prof. Jukka Saarinen.

Publication [P2] evaluates the effects of the IEEE 802.11 WLAN and Bluetooth coexistence and the resulting interference. The work contains the defining of a coexistence topology, interference evaluation with mathematical modelling, and practical measurements using spectrum analyser and WLAN and Bluetooth devices.

The paper was written by the author, who also designed the test arrangements and analysed the results. Mr. Tapio Rantanen and Jani Ruotsalainen M.Sc performed most of the practical measurements. Markku Niemi provided technical details for the work. Professors Timo Hämäläinen and Jukka Saarinen improved the writing style.

In *Publication [P3]*, the performance of the IEEE 802.11 MAC protocol under DCF is evaluated with mathematical calculations, simulations performed with an implemented SDL simulation model, and with practical measurements using commercial WLAN adapters. The QoS of the 802.11 data transfer under ad-hoc topology is projected for different application types.

The paper was written by the author. The performance calculations, implementation of the SDL model and performed simulations, planning of the practical measurements, and the analysing of the results were accomplished by the author. Prof. Timo Hämäläinen outlined the contents of the publication. Markku Niemi provided technical details for testing work.

Publication [P4] is the first publication of the author. The publication presents the developed architecture for TUTWLAN and the functionality of the TUTMAC protocol. The design work was mostly carried out in 1997, while the publication was written in early 1998.

The author has been the main architect of TUTWLAN and the TUTMAC protocol. Prof. Timo Hämäläinen, Prof. Jarno Knuutila, Jari Jokela M.Sc, and Juha Ala-Laurila M.Sc provided ideas for the new design. Ari Letonsaari M.Sc conducted parts of the SDL model programming for the TUTMAC protocol Prof. Jukka Saarinen improved the writing style of the publication.

In *Publication [P5]* the first version of the TUTWLAN hardware platform is described. The basic approach has been to design a platform capable for TUTWLAN terminal and base station implementations.

The publication was co-written by the author and Marjo Kari M.Sc. The author's contribution to the research was the design requirements for the platform that resulted from TUTWLAN and the TUTMAC protocol. The paper also presents the first description of the hardware accelerators of the TUTMAC protocol, who's functionality has been designed by the author. The platform hardware design was conducted by Marjo Kari and Prof. Timo Hämäläinen, and Prof. Jarno Knuutila provided practical hints for the design. Prof. Jukka Saarinen improved the writing style.

Publication [P6] describes the second version of the TUTWLAN hardware platform. Compared to the original platform presented in [P5], the advanced platform has been revised to meet better the requirements of the TUTMAC protocol implementation. Especially, automatic SDL to C code conversion and the need of hardware accelerators for the TUTMAC protocol have resulted in the development of the new platform.

The hardware design of the advanced platform was carried out by Kimmo Tikkanen M.Sc and Prof. Timo Hämäläinen. The design requirements for the new system were provided by the author, based on TUTWLAN and TUTMAC implementations. The paper was co-written by Kimmo Tikkanen and the author. Prof. Jukka Saarinen improved the writing style.

In *publication [P7]*, the hardware accelerators of the TUTMAC protocol are described and their implementation presented.

The paper was co-written by Mr. Tuomo Saari and the author, while Prof. Timo Hämäläinen revised writing style. The FPGA logic programming has been carried out by Tuomo Saari. The author specified the functional design of the hardware accelerators.

Publication [P8] describes the functional architecture of a TUTWLAN base station consisting of a TUTWLAN hardware platform and a Windows NT host computer. The publication concentrates on host software design and the functionality of the TUTWLAN management application.

The publication was written by the author. The Windows NT programming was carried out by Timo Vanhatupa M.Sc, while the requirements for the functionality were conducted by the author. Jussi Lemiläinen M.Sc provided technical hints for the contents. Professors Timo Hämäläinen and Jukka Saarinen improved the writing.

In *Publication [P9]* the SDL development flow for the TUTMAC protocol design and implementation is presented. The SDL development has proceeded by using the same SDL model that has been revised and extended towards implementation. Properties of the implemented protocol, as well as basic steps for performance improvements of the SDL implemented protocol are described.

The paper was written by the author. Parts of the SDL programming work and the SDL simulations for performance improvements were carried out by Ari Letonsaari. Professors Timo Hämäläinen, Jarno Knuutila, and Jukka Saarinen revised the writing.

Publication [P10] describes the design for the TUTWLAN AP module, and presents its implementation in a Windows NT host computer.

The paper was co-written by Mauri Kuorilehto M.Sc and the author. The contribution of the author was the requirements and functional design for the AP. Mauri Kuorilehto carried out the Windows NT implementation and programming related work. Professors Timo Hämäläinen and Jukka Saarinen, and Markku Niemi M.Sc revised the publication.

Publication [P11] presents the architecture of the wireless video demonstrator and its implementation. The reasoning for implementing the demonstrator is given and the basic functionality of each demonstrator component described.

The publication was co-written by the author and Olli Lehtoranta M.Sc. The contribution of the author was the development of the demonstrator architecture and the design of VCP functionality. The presented research goals were also defined by the author. Olli Lehtoranta was responsible for the parts related to video stream encoding and decoding. Mr. Jukka Suhonen carried out the programming of the video sender and receiver parts and the corresponding user interfaces for a Windows NT host computer. Markku Niemi and Prof. Timo Hämäläinen revised the writing.

Publication [P12] concentrates on the VCP of the wireless video demonstrator. The paper describes the detailed functionality of the protocol, including traffic class constructions, frame formats, and QoS support queuing management and scheduling. Implementation in Windows NT host computer has been addressed.

The publication was co-written by the author and Jukka Suhonen. The author's contribution was defining of the requirements and designing of the functionality for the protocol. The implementation of the VCP was carried out by Jukka Suhonen, while Olli Lehtoranta was responsible for the video related functionality. Mauri Kuorilehto contributed to the technical implementation of the protocol in Windows NT. Markku Niemi and Prof. Timo Hämäläinen revised the writing.

7. Conclusions

WLAN is becoming a de-facto technology for networking computers, replacing the need for wiring and extending the services of a LAN backbone. WLAN and WPAN technologies are being integrated into different personal devices, such as laptop computers, mobile phones, headsets, digital cameras, and remote control devices. Furthermore, WLAN/WPAN technologies are emerging in simpler applications like wireless sensors, thus forming a part of future ubiquitous wireless networks.

As WLAN technologies extend wired network services and provide ad-hoc networking, such as serial cable replacement, the different QoS requirements of the various applications becomes a challenge. Wireless medium itself is a limited and demanding resource, which complicates the design of QoS support for WLANs. A wired LAN level of QoS is difficult to achieve. The central component of a WLAN is the MAC protocol. Thus, a wireless MAC protocol must cope with shared broadcast medium, changing network topology, unreliability and insecurity of data transfer, and the lower available bandwidth compared to wired LANs.

The two basic WLAN topologies are centralised and ad-hoc. Ad-hoc networking is required by many application areas, and must be supported by a WLAN. QoS support must be available in both network topologies, as the projected applications are mostly the same.

The TUTWLAN research has produced results of how to design QoS support on the link and transport layers. The link layer QoS control is emphasised, as it is the operational base for higher layer protocols. TUTMAC protocol supports QoS by dynamic reservation TDMA protocol. The TUTWLAN TDMA frame structure has been designed to be flexible. Both the number of TDMA data slots, and the slot duration can be configured according to traffic requirements.

The centrally controlled network topology enables a more efficient support for QoS parameters in data transfer compared to distributed control. The QoS control is done by several parameters grouped into three parameter groups for service type, security type, and connection type. The service type defines the processing of user data for transfer error control. The security type specified the required level of confidentiality. The connection type defines the wireless connection between the TUTWLAN base station and a terminal in terms of TDMA data slot allocations.

The need to integrate QoS into WLANs has been recognised by WLAN standardisation and specification bodies, as well as by research institutions. Furthermore, it seems that the introduction of the IEEE 802.11 and HIPERLAN standards was a turning point in the research field. While the research proposals before the standards discussed proprietary systems for wireless Ethernet and wireless ATM, the recent WLAN research work mostly consists of improvements to the existing IEEE 802.11 standards. According to the author’s knowledge, no other similar proprietary WLAN system, containing a full scale development of hardware platforms, MAC protocols, and AP functionalities is currently being developed by research institutes.

References

[1] Aad I., Castelluccia C., "Differentiation Mechanisms for IEEE 802.11", IEEE Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001), Volume 1, pp. 209-218, Anchorage, Alaska, USA, April 22-26, 2001.

[2] Abrahams R. L., "2.4GHz 11Mbps MACless DSSS Radio HWB1151, User's Guide", Intersil Application Note: AN9835.1, 13 pages, May 1999.

[3] Adamou M., Khanna S., Lee I., Shin I., Zhou S., "Fair Real-Time Traffic Scheduling over a Wireless LAN", IEEE Real-Time Systems Symposium (RTSS 2001), pp. 279-288, London, UK, December 3-6, 2001.

[4] Ahmadi H., Krishna A., LaMaire R. O., "Design Issues in Wireless LANs", Journal of High Speed Networks, Volume 5, Issue 1, pp. 87-104, 1996.

[5] Ala-Laurila J., Mikkonen J., Rinnemaa J., "Wireless LAN Access Network Architecture for Mobile Operators", IEEE Communications Magazine, Volume 39, Issue 11, pp. 82-89, November 2001.

[6] Banchs A., Perez X., "Distributed Weighted Air Queuing in 802.11 Wireless LAN", IEEE International Conference on Communications (ICC 2002), Volume 5, pp. 3121-3127, New York city, USA, April 28 - May 2, 2002.

[7] Banchs A., Radimirsch M., Perez X., "Assured and Expedited Forwarding Extensions for IEEE 802.11 Wireless LAN", IEEE International Workshop on Quality of Service (IWQoS 2002), pp. 237-246, Miami, USA, May 15-17, 2002.

[8] Bantz D. F., Bauchot F. J., "Wireless LAN Design Alternatives", IEEE Network, Volume 8, Issue 2, pp. 43-53, March/April 1994.

[9] Bauchot F. J., Lanne F., "IBM Wireless RF LAN design and architecture", IBM Systems Journal, Volume 34, Issue 3, pp. 390-408, 1995.

[10] Bing B., "High-Speed Wireless ATM and LANs", Artech House, Boston, 268 pages, ISBN: 1580530923, February 2000.

[11] Bluetooth SIG, Specification of the Bluetooth System, Core, Version 1.1, 1082 pages, February, 2001.

[12] Bluetooth SIG, Specification of the Bluetooth System, Profiles, Version 1.1, 452 pages, February 2001.

[13] Border J., Kojo M., Griner J., Montenegro G., Shelby Z., "Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations", Request for Comments: 3135, available at: http://www.ietf.org/rfc/rfc3135.txt, June 2001.

[14] Braden R., "Requirements for Internet Hosts - Communication Layers", Internet Engineering Task Force, Request for Comments: 1122, available at: http://www.ietf.org/rfc/rfc1122.txt, October 1989.

[15] Braden R., Clark D., Shenker S., "Integrated Services in the Internet Architecture: an Overview", Internet Engineering Task Force, Request for Comments: 1633, available at: http://www.ietf.org/rfc/rfc1633.txt, June 1994.

[16] Bray J., Sturman C., "Bluetooth: Connect Without Cables", 2nd edition, Prentice Hall, 593 pages, ISBN: 0130661066, 2002.

[17] Cali F., Conti M., Gregori E., "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit", IEEE/ACM Transactions on Networking, Volume 8, Issue 6, pp. 785-799, December 2000.

[18] Campanella M., Chivalier P., Sevasti A., Simar N., "Quality of Service Definition", deliverables D.2 of the SEQUIN project (IST-1999-20841), 43 pages, March 2001.

[19] Chen K.-W., "Medium Access Control of Wireless LANs for Mobile Computing", IEEE Network, Volume 8, Issue 5, pp. 50-63, September/October 1994.

[20] Cisco Systems, Inc., "Internetworking Technologies Handbook", 3rd edition, ISBN: 1587050013, Chapter 49, 32 pages, January 2001.

[21] Doufexi A., Armour S., Butler M., Nix A., Bull D., McGeehan J., Karlsson P., "A Comparison of the HIPERLAN/2 and IEEE 802.11a Wireless LAN Standards", IEEE Communications Magazine, Volume 40 Issue 5, pp. 172-180, May 2002.

[22] Dugar A., Vaidya N., Bahl P., "Priority and Fair Scheduling in a Wireless LAN" IEEE Military Communications Conference (MILCOM 2001), Volume 2, pp. 993-997, Washington D.C., USA, October 28-31, 2001.

[23] EN 300 652 V1.2.1, Broadband Radio Access Networks (BRAN); High Performance Radio Local Area Network (HIPERLAN) Type 1; Functional specification, 105 pages, 1998.

[24] ETSI TR 101 031 V 2.2.1, Technical Report, Broadband Radio Access Networks (BRAN); High Performance Radio Local Area Network (HIPERLAN) Type 2; Requirements and architectures for wireless broadband access, 32 pages, January 1999.

[25] ETSI TR 101 683 V1.1.1, Technical Report, Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; System Overview, 19 pages, 2002.

[26] ETSI TR 101 856 V1.1.1, Technical Report, Broadband Radio Access Networks (BRAN); Functional Requirements for Fixed Wireless Access systems below 11 GHz: HIPERMAN, 34 pages, 2001.

[27] ETSI TS 101 493-2 V1.2.1, Technical Specification, Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Packet based Convergence Layer; Part 2: Ethernet Service Specific Convergence Sublayer (SSCS), 29 pages, December 2001.

[28] ETSI TS 101 761-1 V1.3.1, Technical Specification, Broadband Radio Access Networks (BRAN); HIPERLAN Type 2; Data Link Control (DLC) Layer; Part 1: Basic Data Transport Functions, 88 pages, December 2001.

[29] European Radiocommunications Committee (ERC), ERC Report 25, The European Table of Frequency Allocations and Utilisations Covering the Frequency Range 9 kHz to 275 GHz, 179 pages, January 2002.

[30] European Radiocommunications Committee, ERC Recommendation 70-03, Relating to the Use of Short Range Devices (SRD), 45 pages, February 2002.

[31] Ferguson P., Huston G., “Quality of Service - Delivering QoS on the Internet and in Corporate Networks”, Wiley Computer Publishing, 288 pages, ISBN: 0471243582, January 1998.

[32] Finkenzeller K., “RFID Handbook - Radio Frequency Identification Fundamentals and Applications”, Wiley & Sons, 322 pages, ISBN: 0471988510, October 1999.

[33] Ghanwani A., Pace W., Srinivasan V., Smith A., Seaman M., “A Framework for Integrated Services Over Shared and Switched IEEE 802 LAN Technologies”, Internet Engineering Task Force, Request for Comments: 2816, available at: http://www.ietf.org/rfc/rfc2816.txt, May 2000.

[34] Gidlund M., “An Approach for Using Adaptive Error Control Schemes in Wireless LAN with CSMA/CA MAC Protocol”, IEEE Vehicular Technology Conference, (VTC 2002/Spring), Volume 1, pp. 224-228, Birmingham, Alabama, USA, May 6-9, 2002.

[35] Hämäläinen P., Hännikäinen M., Hämäläinen T., Corporaal H., Saarinen J., “Implementation of Encryption Algorithms on Transport Triggered Architectures”, IEEE International Symposium on Circuits and Systems (ISCAS 2001), Volume 4, pp. 726-729, Sydney, Australia, May 6-9, 2001.

[36] Hämäläinen P., Hännikäinen M., Hämäläinen T., Saarinen J., “Configurable Hardware Implementation of Triple-DES Encryption Algorithm for a Wireless Local Area Network”, International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001), Volume 2, pp. 1221-1224, Salt Lake City, USA, May 7-11, 2001.

[37] Hämäläinen P., Hännikäinen M., Hämäläinen T., Saarinen J., “Hardware Implementation of the Improved WEP and RC4 Encryption Algorithms for Wireless Terminals”, The European Signal Processing Conference (EUSIPCO 2000), pp. 2289-2292, Tampere, Finland, September 5-8, 2000.

[38] Hämäläinen P., Hännikäinen M., Niemi M., Hämäläinen T., "Performance Evaluation of Secure Remote Password Protocol" IEEE International Symposium on Circuits and Systems (ISCAS 2002), Volume 3, pp. 29-32, Phoenix, USA, May 26-29, 2002.

[39] Hämäläinen P., Hännikäinen M., Niemi M., Hämäläinen T., Saarinen J., "Implementation of Link Security for Wireless Local Area Networks", IEEE International Conference on Telecommunications (ICT 2001), Volume 1, pp. 299-305, Bucharest, Romania, June 4-7, 2001.

[40] Hännikäinen M., Laitinen A., Hämäläinen T., Kaisto I., Leskinen K., "Architecture of a Passenger Information System for Public Transport Services" The IEEE Semiannual Vehicular Technology Conference (VTC 2001/Fall), Volume 2, pp. 698-702, Atlantic City, USA, October 7-11, 2001.

[41] Hännikäinen M., Laitinen A., Hämäläinen T., Kaisto I., Leskinen K. "Services of TUTPIS - A Passenger Information System for Public Transportation", IEEE Multimedia and Applications Conference (MTAC 2001), pp. 118-122, Irvine, California, USA, November 7-9, 2001.

[42] Hännikäinen M., Rekonius J., Hämäläinen T., Soininen R., "Real-Time Betting" IEEE Multimedia and Applications Conference (MTAC 2001), pp. 113-117, Irvine, California, USA, November 7-9, 2001.

[43] Heegard C., Coffey J. T., Gummadi S., Murphy P. A., Provencio R., Rossin E. J., Schrum S., Shoemake M. B., "High Performance Wireless Ethernet", IEEE Communications Magazine, Volume 39, Issue 11, pp. 64-73, November 2001.

[44] Held G., "Quality of Service in a Cisco Networking Environment", Wiley & Sons, 200 Pages, ISBN: 0470844256, May 2002.

[45] Ho H. J., Rawles M. S., Vrijkorte M., Fei L., "RF Challenges for 2.4 and 5 GHz WLAN Deployment and Design", IEEE Wireless Communications and Networking Conference (WCNC 2002), Volume 2, pp. 783-788, Orlando, Florida, USA, March 17-21, 2002.

[46] Homepage of the Bluetooth SIG, http://www.bluetooth.com/, April 26, 2002.

[47] Homepage of the ETSI BRAN Project, http://www.etsi.org/bran/, April 26, 2002.

[48] Homepage of the HomeRF Working Group Inc., http://www.homerf.org/, June 13, 2002.

[49] Homepage of the IEEE 802.11 Working Group, http://grouper.ieee.org/groups/802/11/, March 23, 2002.

[50] Homepage of the IEEE 802.15 Working Group, http://grouper.ieee.org/groups/802/15/, March 23, 2002.

[51] Homepage of the IEEE 802.16 Working Group on Broadband Wireless Access Standards, http://grouper.ieee.org/groups/802/16/, May 14, 2002.

[52] Homepage of the IEEE Wireless Standards Zone, http://standards.ieee.org/wireless/, June 13, 2002.

[53] HomeRF white paper, "Interference Immunity of 2.4 GHz Wireless LANs", 2001, http://www.homerf.org/, April 25, 2002.

[54] IEEE Draft P802.15.1/D0.9.2, Information technology - Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Draft Standard for Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Wireless Personal Area Networks (WPAN), 1149 pages, June 2001.

[55] IEEE Draft P802.15.3/D09, Draft Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks Specific requirements - Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPAN), 234 pages, December 2001.

[56] IEEE Draft P802.15.4/D12, Information technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks Specific Requirements - Draft IEEE Standard for Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low Rate Wireless Personal Area Networks (LR-WPANs), 133 pages, November 2001.

[57] IEEE Sdt 802.11 1st edition, Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and Physical layer (PHY) specifications, 512 pages, August 1999.

[58] IEEE Sdt 802-2001, IEEE Standard for Local and Metropolitan Area Networks: Overview and Architectures, 36 pages, December 2001.

[59] IEEE Std 802.11a-1999, Supplement to IEEE Standard for Information technology- Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, High-speed Physical Layer in the 5 GHz Band, 83 pages, 1999.

[60] IEEE Std 802.11b-1999, Supplement to IEEE Standard for Information technology - Telecommunications and information exchange between systems- Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band, 88 pages, September 1999.

[61] IEEE Std 802.11e/D2.0, Draft Supplement to Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and Physical layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS), 107 pages, November 2001.

[62] IEEE Std 802.11g/D1.1, Draft Supplement to Standard Information Technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band, 39 pages, January 2002.

[63] IEEE Std 802.1D, 1998 Edition, IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Common specifications - Part 3: Media Access Control (MAC) Bridges, 355 pages, 1998.

[64] IEEE Std 802.1Q-1998, IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks, 199 pages, December 1998.

[65] IEEE Std 802.3, 2002 Edition, IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks- Specific requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) access method and physical layer specifications, 379 pages, March 2002.

[66] IEEE Std 802.5, 1998 Edition, Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Speci?c requirements - Part 5: Token ring access method and Physical Layer speci?cations, 243 pages, May 1998.

[67] In-Stat/MDR Press Release, April 3, 2002, “WLAN Chips to Embark on Incredible Journey”, available at //www.instat.com/, May 13, 2002.

[68] International Telecommunication Union, CCIT Specification and Description Language (SDL), ITU-T Recommendation Z.100, 237 pages, 1993.

[69] Internet Engineering Task Force, “Internet Protocol”, Request for Comments: 791, available at: http://www.ietf.org/rfc/rfc791.txt, September 1981.

[70] ISO/IEC 7498-1:1994(E), Information Technology - Open Systems Interconnections - Basic Reference Model: The Basic Model, 59 pages, November 1994.

[71] Johnsson M., “HiperLAN/2 - The Broadband Radio Transmission Technology Operating in the 5 GHz Frequency Band”, HiperLAN/2 Global Forum, available at: http://www.hiperlan2.com/, 1999.

[72] Kabulepa L. D., Glesner M., "Design of Random Number Generators for the HIPERLAN/1 Channel Access Mechanism", IEEE International Conference on Electronics, Circuits and Systems (ICECS 2000), Volume 1, pp. 234-237, Beirut, Lebanon, December 17-20, 2000.

[73] Kamerman A., "Spread-Spectrum Techniques Drive WLAN Performance", Microwaves & RF, pp. 109-114, September 1996.

[74] Kamerman, A., Aben, G., "Net Throughput with IEEE 802.11 Wireless LANs", IEEE Wireless Communications and Networking Conference (WCNC 2000), Volume 2, pp. 747-752, Chicago, USA, September 23-28, 2000.

[75] Karaoguz J., "High-Rate Wireless Personal Area Networks", IEEE Communications Magazine, Volume 39, Issue 12, pp. 96-102, December 2001.

[76] Kilkki K., "Differentiated Services for the Internet", Macmillan technology series, 384 pages, ISBN: 1578701325, June 1999.

[77] Kukkala P., Kangas T., Hännikäinen M., Hämäläinen T., "SDL Generated Protocols in Seamless Co-Verification Environment" IEEE International Workshop on Design and Diagnostics of Electronic Circuits and Systems (DDECS 2002), pp. 158-165, Brno, Czech Republic, April 17-19, 2002.

[78] Kuorilehto M., Hännikäinen M., Hämäläinen T., "Suitability of Windows NT for Time-Critical Network Control Systems", International Conference on Telecommunications (ICT 2002), Volume 3, pp. 186-191, Beijing, China, June 23-26, 2002.

[79] Kuorilehto M., Hännikäinen M., Hämäläinen T., Saarinen J., "Bridging Network Traffic between Wireless and Wired LANs in Windows NT", European Signal Processing Conference (EUSIPCO 2000), Volume 1, pp. 267-270, Tampere, Finland, September 5-8, 2000.

[80] LaMaire R. O., Krishna A., Bhagwat P., Panian J., "Wireless LANs and Mobile Networking, Standards and Future Directions", IEEE Communications Magazine, Volume 34, Issue 8, pp. 86-94, August 1996.

[81] Lansford J., Stephens A., Nevo R., "Wi-Fi (802.11b) and Bluetooth: Enabling Coexistence", IEEE Network, Volume 15, Issue 5, pp. 20-27, September-October 2001.

[82] Lehtoranta O., Hännikäinen M., Suhonen J., Hämäläinen T., "Implementation of Unequal Error Protection of H.263 Video for a Wireless Video Demonstrator", International Conference on Telecommunications (ICT 2002), Volume 1, pp. 1024-1029, Beijing, China, June 23-26, 2002.

[83] Mettälä R., "Bluetooth Protocol Architecture", Version 1.0, Bluetooth White Paper, available at: http://www.bluetooth.org, August 1999.

[84] Mikkonen J., “Quality of Service in Radio Access Networks”, Doctor of Technology Thesis, Tampere University of Technology, 147 pages, May 1999.

[85] Mikkonen J., Aldis J., Awater G., Lunn A., Hutchison D., “The Magic WAND-functional overview”, IEEE Journal on Selected Areas in Communications, Volume 16, Issue 6 , pp. 953-972, August 1998.

[86] Miller B. A., Bisdikian C., “Bluetooth Revealed: The Insider’s Guide to an Open Specification for Global Wireless Communications”, Prentice Hall, 320 pages, ISBN: 0130902942, September 2000.

[87] Munoz L., Garcia M., Choque J., Aguero R., Mähönen P., “Optimizing Internet flows over IEEE 802.11b wireless local area networks: a performance-enhancing proxy based on forward error correction”, IEEE Communications Magazine, Volume 39, Issue 12, pp. 60-67, December 2001.

[88] Natkaniec, M., Pach, A.R., “An Analysis of the Backoff Mechanism Used in IEEE 802.11 Networks”, IEEE Symposium on Computers and Communications, (ISCC 2000), pp. 444-449, Antibes-Juan les Pins, France, July 3-6, 2000.

[89] Nichols K., Blake S., Baker F., Black D., “Definition of the Differentiated Services Field (DS Field) in the Ipv4 and Ipv6 Headers”, Internet Engineering Task Force, Request for Comments: 2474, available at: http://www.ietf.org/rfc/rfc2474.txt, December 1998.

[90] Ogawa M., Shimojima I., Hattori T., “CoS Guarantee Control for Wireless LAN”, IEEE Vehicular Technology Conference, (VTC 2002/Spring), Volume 1, pp. 50-54, Birmingham, Alabama, USA, May 6-9, 2002.

[91] Olsen A., Færgemand O., Møller-Pedersen B., Reed R., Smith J. R. W., “Systems Engineering Using SDL-92”, North-Holland, 480 pages, ISBN: 04448987271994, 1994.

[92] Pahlavan K., Levesque A. H., “Wireless Data Communications”, Proceedings of the IEEE, Volume 82, No. 9, pp. 1398-1430, September 1994.

[93] Pahlavan K., Zahedi A., Krishnamurty P., “Wideband Local Access: Wireless LAN and Wireless ATM”, IEEE Communications Magazine, Volume 35, Issue 11, pp. 34-40, November 1997.

[94] Partridge C., “A Proposed Flow Specification”, Internet Engineering Task Force, Request for Comments: 1363, available at: http://www.ietf.org/rfc/rfc1363.txt, September 1992.

[95] Passas N., Paskalis S., Vali D., Merakos L., “Quality-of-Service-Oriented Medium Access Control for Wireless ATM Networks”, IEEE Communications Magazine, Volume 35, Issue 11, pp. 42-50, November 1997.

[96] Qiao D., Choi S., “Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation”, IEEE International Conference on Communications (ICC 2001), Volume 7, pp. 1995-2000, St.-Petersburg, Russia, June 11-15, 2001.

[97] QoS Forum, “Technology Backgrounder - Quality of Service - Glossary of Terms”, acquired from http://www.qosforum.com, May 1999.

[98] QoS Forum, “White Paper - QoS protocols & architectures”, acquired from http://www.qosforum.com, July 1999.

[99] Rantanen T., Hännikäinen M., Niemi M., Hämäläinen T., “Implementation of a WWW-based MIB Browser with an ASN.1 Translator”, International Conference on Telecommunications (ICT 2002), Volume 3, pp. 202-207, Beijing, China, June 23-26, 2002.

[100] Roberts W., “Wireless LAN Design - Emerging Standards or Custom?”, Wireless Systems Design, January 1997, pp. 34-39.

[101] Rodriguez A., Gatrell J., Karas J., Peschke R., “TCP/IP Tutorial and Technical Overview”, 7th edition, Prentice Hall PTR, 980 pages, ISBN: 0130676101, available at: http://www.redbooks.ibm.com/, August 2001.

[102] Salli K., Hämäläinen T., Knuutila J., Saarinen J., “Security Design for a New Wireless Local Area Network TUTWLAN”, IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC 1998), pp. 1540-1544, Boston, USA, September 8-11, 1998.

[103] Sánchez J., Martinez R., Marcellin M. W., “A Survey of MAC Protocols Proposed for Wireless ATM”, IEEE Network, Volume 11, Issue 6, pp. 52-62, November/December 1997.

[104] Seaman M., Smith A., Wroclawski J., “Integrated Service Mappings on IEEE 802 Networks”, Internet Engineering Task Force, Request for Comments: 2815, available at: http://www.ietf.org/rfc/rfc2815.txt, May 2000.

[105] Shenker S., Wroclawski J., “General Characterization Parameters for Integrated Service Network Elements”, Internet Engineering Task Force, Request for Comments: 2215, available at: http://www.ietf.org/rfc/rfc2215.txt, September 1997.

[106] Sheu S.-T., Sheu T.-F., “A Bandwidth Allocation/Sharing/Extension Protocol for Multimedia over IEEE 802.11 Ad Hoc Wireless LANs”, IEEE Journal on Selected Areas in Telecommunications, Volume 19, Issue 10, pp. 2065-2080, October 2001.

[107] Sitaram D., Dan A., “Multimedia Servers - Applications, Environments, and Design”, Morgan Kaufmann, San Francisco, 297 pages, ISBN: 1558604308, 2000.

[108] Smulders P., “Exploiting the 60 GHz Band for Local Wireless Multimedia Access: Prospects and Future Directions” IEEE Communications Magazine, Volume 40, Issue 1, pp. 140-147, January 2002.

[109] Stallings W., "Data and Computer Communications", 6th edition, Prentice Hall, 810 pages, ISBN: 0130843709, 2000.

[110] Stallings W., "High Speed Networks: TCP/IP and ATM Design Principles", Prentice-Hall, ISBN: 0135259657, 576 pages, January 1998.

[111] Suzuki T., Tasaka S., "Performance Evaluation of Priority-Based Multimedia Transmission with the PCF in an IEEE 802.11 Standard Wireless LAN", IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2001), Volume 2, pp. 70-77, San Diego, USA, September 30-October 3, 2001.

[112] Tannenbaum A. S., "Computer Networks", 3rd edition, Prentice-Hall Inc., New Jersey, 750 pages, ISBN: 0133499456, March 1996.

[113] Telelogic AB, Telelogic SDL Suite v4.1, User's Manual, 2000.

[114] Toh C.-K., "Ad Hoc Mobile Wireless Networks", Prentice Hall, ISBN: 0130078174, 480 pages, December 2001.

[115] Walke B. H., Esseling N., Habetha J., Hettich A., Kadelka A., Mangold S., Peetz J., Vornefeld U., "IP Over Wireless Mobile ATM - Guaranteed Wireless QoS by HiperLAN/2", Proceedings of the IEEE, Volume 89, Issue 1, pp. 21-40, January 2001.

[116] Weatherall J., Jones, A., "Ubiquitous Networks and Their Applications", IEEE Wireless Communications, Volume 9, Issue 1, pp. 18-29, February 2002.

[117] Wireless LAN Association (WLANA), "Introduction to Wireless LANs", acquired from http://www.wlana.com, May 1998.

[118] Wu J.-L. C., Liu H.-H., Lung Y.-J., "An Adaptive Multirate IEEE 802.11 Wireless LAN", IEEE International Conference on Information Networking (ICOIN 2001), pp. 411-418, Beppu City, Japan, January 31-February 2, 2001.

[119] Xylomenos G., Polyzos G. C., Mähönen P., Saaranen M., "TCP Performance Issues over Wireless Links", IEEE Communications Magazine, Volume 39, Issue 4, pp. 52-58, April 2001.

[120] Yavatkar R., Hoffman D., Bernet Y., Baker F., Speer M., "SBM (Subnet Bandwidth Manager): A Protocol for RSVP-based Admission Control over IEEE 802-style networks", Internet Engineering Task Force, Request for Comments: 2814, available at: http://www.ietf.org/rfc/rfc2814.txt, May 2000.

Publications

Publication [P1]

Hännikäinen M., Hämäläinen T., Niemi M., Saarinen J., "Trends in Personal Wireless Data Communications", Computer Communications, Volume 25, Issue 1, pp. 84-99, January 2002.

Computer Communications 25 (2002) 84–99
computer communications
www.elsevier.com/locate/comcom

Tutorial

Trends in personal wireless data communications

Marko Hännikäinen[a,*], Timo D. Hämäläinen[a], Markku Niemi[b,1], Jukka Saarinen[a]

[a]*Digital and Computer Systems Laboratory, Tampere University of Technology, Hermiankatu 3 A, FIN-33720, Tampere, Finland*
[b]*Nokia Mobile Phones, ETSI BRAN Secretary, Tieteenkatu 1, FIN-33720, Tampere, Finland*

Received 22 December 2000; revised 29 May 2001; accepted 29 May 2001

Abstract

This paper gives an overview of the background, current status, and ongoing trends in wireless personal data communications. The driving force behind the future development is proven to be the convergence of multimedia content, the Internet, and wireless mobile devices. Network technologies from global coverage to short-range personal area networks are reviewed focusing on the most prospective standardisation and specification efforts. Future wireless services will be implemented using several network technologies, due to which the portability of services across networks is the most important issue. Especially the support for Quality of Service is emphasised in each wireless network. In a discussion, winners and losers are examined from the global convergence point of view.

Keywords: Personal wireless communications; Quality of service; Bluetooth; Wireless local area network; GSM; UMTS

1. Introduction

Personal wireless communication services have been available to the general public for only about 10 years, since the breakthrough of cellular phones. Strictly speaking, walkie-talkies and other simple devices have been used much longer, but they have not addressed the challenge of being connected wirelessly everywhere, any time, and with any type of information. Personal wireless communication has traditionally meant wireless speech service, but this is now changing rapidly.

The Internet as a source of content and as a medium for offering and delivering services is the driving force behind the increasing use of data communications among mass users. This phenomenon will be further fostered, as the multimedia content in numerous applications, services, and professions, such as electronic commerce, management, maintenance, education, and entertainment will become a common embedded feature in information systems and products. As a result of this development, the term multimedia as a separately addressed service category disappears.

Compared to speech and traditional data transmission, multimedia has more stringent requirements. A combination of several media components tends to increase the bandwidth requirement and the burstiness of transmissions. Typically, low error rates are also required and conversational multimedia services call for short delays in order to fulfil real-time requirements. Furthermore, different media components are synchronised in time. Multimedia is already merged in the Internet, as video and audio coding technologies have proceeded towards lower bit rates and robustness. At the same time, the capacity and support for Quality of Service (QoS) has increased in wired networks. The next logical step is to merge the Internet as a source of multimedia content and the wireless networking seamlessly.

The Internet has been growing at a tremendous speed in recent years. The growth in the number of nodes connected to the Internet, and especially the increase of the popularity of the World Wide Web (WWW) has been immense. Along with this trend wireless communication has rapidly gained wide acceptance. Especially the cellular systems, such as GSM, have experienced huge growth. Fig. 1 presents the expected number of fixed, mobile, and Internet subscribers globally. It is forecasted that around the year 2004 the numbers will converge [21]. After that point, the most common equipment to access Internet services will be a wireless mobile terminal, and the developed wireless networks will provide the same content as is available by fixed access. To a large extent, the digitalisation of the content has enabled this convergence of services.

* Corresponding author. Tel.: +358-3-365-2111; Fax: +358-3-365-4575.

E-mail addresses: marko.hannikainen@tut.fi (M. Hännikäinen), timo.d.hamalainen@tut.fi (T.D. Hämäläinen), markku.t.niemi@nokia.com (M. Niemi).

[1] Tel.: +358-50-511-7341; fax: +358-3-318-3690.

0140-3664/02/$ - see front matter
PII: S0140-3664(02)00345-0

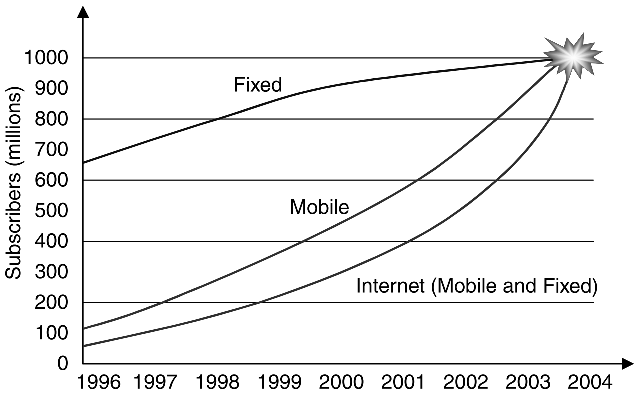


Fig. 1. Estimated subscriber growth [21].

The key technical factors for enabling the convergence of networks and various services are the Internet Protocols (IP) and the support for QoS. 'IP over everything and IP under everything' holds already true. The technical challenge is no longer how to combine heterogeneous data links but how to guarantee seamless QoS signalling and support in every network layer and in every network element for an end-to-end service. The issue is especially important in personal wireless networking. The individual needs differ depending on location, terminal type, and time of day. Thus, there is a clear need to tailor the services for each user individually. Furthermore, this possibility also opens up new interesting scenarios for users and application service providers.

This paper presents a survey of wireless network technologies that are intended for personal wireless networking and services. The background, current status, and trends are discussed, covering standardisation and specification work in various international bodies. The paper is divided into two parts. The first part contains a technical introduction to the different wireless networks, and the second part concentrates on the applicability of the networks in the presented personal wireless networking scenario. Networks are presented according to the area they cover; from wide area networks to very short-range networks. Frequency allocations are also discussed. Applicability issues include QoS and the accessibility of service contents to applications.

2. Network technologies

Fig. 2 presents various operational environments for personal wireless systems in terms of cell sizes and coverage areas [45]. Satellites provide global coverage and are suitable for urban and remote areas with low traffic density, and without access to terrestrial telecommunications networks. Terrestrial *macro cells* are typically situated in rural or suburban areas with low or medium traffic density. The cell radius has a range of several tens of kilometres. *Micro cells* are situated in urban areas with a cell radius of up to 1 km. Traffic density varies from medium to high, and the mobile speed remains moderate. *Pico cells* are predominantly situated indoors with a cell radius of less than 100 m. Their characteristics include low speed terminals, medium to high traffic density, and wideband or broadband services [44].

The final level of the hierarchy is *personal area cells* that refer to network connecting fixed, portable, and moving devices operating in a Personal Operating Space (POS). The cell radius is typically up to 10 m around a person, whether stationary or in motion. A typical cell would contain a limited number of devices (e.g. less than 10) with low traffic density and wideband services [11]. In the following, the most important networks from macro cells to personal area are presented. Tables 1–4 summarise the main technical features of the networks.

2.1. Macro cell networks

The most widespread macro cell network is Global System for Mobile Telecommunication (GSM) that operates at the 900 MHz frequency band. GSM can also be used in micro cell environments, as in urban areas the GSM systems increasingly operate around 1800 MHz frequencies with smaller cell sizes. In the following, the path from GSM to

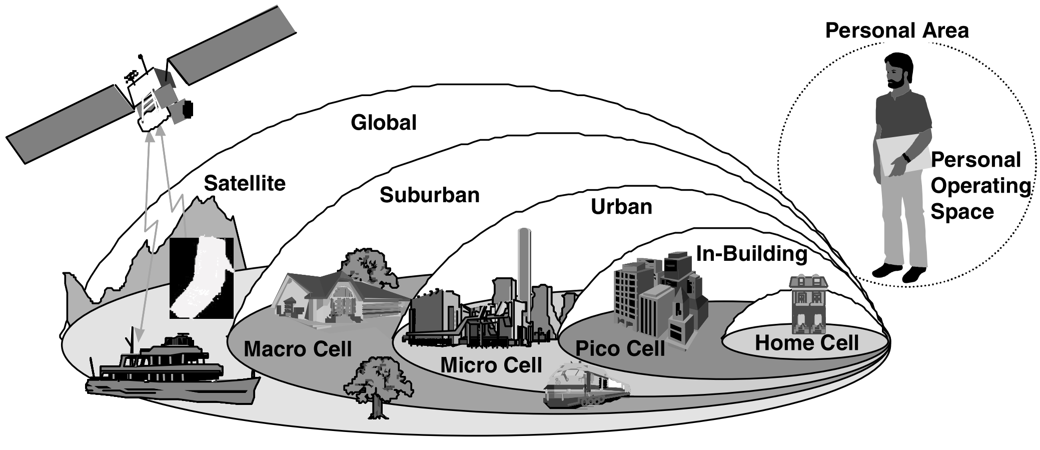


Fig. 2. Universal wireless network coverage [45].

Table 1
Macro cell networks

	3rd Generation UMTS	Digital cellular GSM 2.5 G
Frequency band	1.92–1.98 GHz 2.11–2.17 GHz	880–915 MHz 925–960 MHz 1710–1785 MHz 1805–1880 MHz
TX power	N/A	2 W
Access technique	W-CDMA	TDMA/FDMA/FDD
Cell radius	Global coverage using a variety of cell types	Large (macro cells) 0.5–30 km
User speed	Depends on the access type	< 250 km/h
User data rate	144 kbit/s–2 Mbit/s	9.6–348 kbit/s

third generation (3G) networks is explained. Universal Mobile Telecommunications System (UMTS) is highlighted as a 3G network.

2.1.1. Migration from 2G to 3G

The second generation (2G) digital cellular systems are currently available for mobile communications. Examples of the available systems are GSM in Europe, Interim Standards IS-54, IS-95 and IS-136 in the United States, and Personal Digital Communications (PDC) in Japan [30]. They have been developed and are mainly used for speech, but still posses potential also for data transmission.

GSM has experienced rapid growth since it appeared on the market in 1993. At the end of 2000, it was used in over 140 countries with 400 million subscribers. In addition, 71 new networks were planned, which indicates continuing growth [9,27]. The later extension to the 1800 MHz frequency band, earlier allocated for the Digital Cellular System (DCS), has provided more capacity and thus has helped the growth [24]. Currently, most of the GSM terminals on the market support the dual-band operation.

GSM represents the latest 2G development. The standardisation is currently referred to being in phase 2.5G, which means an intermediate phase where the original digital voice network is enhanced for a better support of data services. High Speed Circuit Switched Data (HSCSD) and General Packet Radio System (GPRS) belong to the GSM 2.5G releases 1996 and 1997 from ETSI. The GSM development has been recently transferred from European Telecommunications Standard Institute (ETSI) to 3GPP (Third Generation Partnership Project) [19]. 3GPP develops globally applicable specifications for 3G systems. The first specification release will be based on the GSM core network.

3GPP is also continuing the effort started in ETSI towards higher bit rates and new services in GSM. First changes include an enhanced mobile data rate of 14.4 kbit/s compared to the basic 9.6 kbit/s, still using a single Time Division Multiple Access (TDMA) timeslot. The higher bit rate is achieved by reducing the number of bits used for forward error correction in the 22 kbit/s raw data rate GSM channels. The rate enhancement is relatively small for a single slot usage, but becomes a significant factor when multiple TDMA timeslots are used for data transfer.

HSCSD is an enhancement to the current circuit switched GSM data enabling the reservation of multiple TDMA timeslots for a single user connection. With the maximum of four time slots, the symmetric rate is 57.6 kbit/s, using 14.4 kbit/s per time slot. Thus, the data rates increase to the level of fixed telephone networks, namely between the level of a standard V.90 analog modem and the B-channel of ISDN (Integrated Services Digital Network). Some operators have already launched HSCSD services. The first user terminals on the market use a maximum of three timeslots, thus reaching data rates of 43.2 kbit/s [13].

GPRS brings the GSM system much closer to legacy data networks by providing packet access for a GSM terminal and packet switching based routing in the GSM infrastructure [27]. The data rates depend on the availability of channels in use per cell, giving a maximum rate of up to 171.2 kbit/s. The uplink and downlink data rates are asymmetric.

GPRS is a major improvement over the voice oriented circuit switched network. The spectrum can be utilised more

Table 2
Micro cell networks

	Digital cordless DECT	Fixed wireless access IEEE 802.16	HIPERACCESS
Frequency band	1.88–1.9 GHz	Several	Several, 32–40 GHz
TX power	250 mW	100 mW	100 mW
Access technique	TDMA/FDMA/TDD	Dynamic TDMA/TDD and FDD	TDMA/TDD
Cell radius	Medium (micro cells) 200 m	Line of sight required	Line of sight required
User speed	50 km/h	Stationary	Stationary
User data rate	22–552 kbit/s	2–155 Mbit/s	> 36 Mbit/s

Table 3
Pico cell networks

	IEEE 802.11	802.11b	802.11a	HIPERLAN/1	HIPERLAN/2	MMAC (several variants)
Frequency band	2.4 GHz	2.4 GHz	5 GHz	5 GHz	5 GHz	5 GHz
TX power	100 mW (EU)	100 mW (EU)	200 mW (EU)	200 mW	200 mW	200 mW
Topology and access technique	Distributed with contention or centralised with polling /TDD	Distributed with contention or centralised with polling /TDD	Distributed with contention or centralised with polling /TDD	Distributed with contention /TDD	Centralised with dynamic TDMA/TDD	Several alternatives of TDMA
Cell radius	50 m	50 m	30–50 m	30–50 m	30–50 m	30–50 m
User speed	Walking speed (10 m/s)					
User data rate	< 2 Mbit/s	11 Mbit/s	54 Mbit/s	19 Mbit/s	54 Mbit/s	> 20 Mbit/s

effectively since idle voice channels can be dynamically assigned to serve packet-based connections. GPRS allows a user to maintain a continuous virtual connection to the network, without the need for slow dialling procedures before data transfer or connection costs of idle data channels. This facilitates several types of variable data rate services, including IP based applications for mass markets.

ETSI and now 3GPP have also been working on a new modulation scheme and protocols for the GSM radio interface. The Enhanced Data Rates for GSM Evolution (EDGE) technical specification for GSM/EDGE Radio Access Network (GERAN) was released in 1999 [35]. The EDGE system provides up to 348 kbit/s for HSCSD and GPRS data services, extending the GSM bit rates close to 3G systems.

The introduction of EDGE does not require major technical investments from operators, since the technology is strongly based on GSM. EDGE is compatible with the current GSM architecture in terms of core network and cell coverage. EDGE uses the same TDMA frame structure and the 200 kHz carrier spacing as GSM and can thus coexist with the current GSM networks at frequencies around 900 and 1800 MHz. The higher bit rates of EDGE will reduce the maximum velocity of a mobile user. Dynamic link adaptation, by utilising different modulation and coding schemes at different carrier to interface ratios must be supported for a continuous service support. EDGE is planned to be widely commercially available in 2001 [1].

The development of GSM towards 3G is depicted in Fig. 3. The Figure summarises the most important technology steps and the timetable of the services based on these technical advantages.

2.1.2. UMTS — 3G system

UMTS is one of the major 3G mobile systems that is being developed and standardised by 3GPP within the framework of International Mobile Telecommunications 2000 (IMT-2000). IMT-2000 is further defined by the International Telecommunications Union (ITU)[45]. The basic standard of UMTS, the Release-99, was finalised in the spring of 2000. A further developed standard will be available by the beginning of 2002, whereas the service provision is expected gradually between 2002 and 2005 [27].

UMTS aims for high capacity, high data rates, and for global mobility, incorporating both terrestrial radio and satellite communications. Thus, UMTS merges several access technologies into a single service interface. This allows a user to roam e.g. from a private, micro cell network (e.g. a Wireless Local Area Network) to a 2.5G or 3G cellular network, and finally to a satellite access network without interrupting an ongoing communication. Other system

Table 4
Personal area networks

	IEEE 802.15 TG1	802.15 TG3	802.15 TG4	Bluetooth	Bluetooth 2
Frequency band	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz
TX power	1–100 mW	< 100 mW	< 20mW	1–100 mW	1–100 mW
Topology and access technique	Master station polling/TDD	Master station polling/TDD	Master station polling/TDD	Master station polling/TDD	Master station polling/TDD
Cell radius	< 10 m (personal cell)	< 10 m (personal cell)	Depends on application environment	< 10 m (personal cell)	< 10 m (personal cell)
User speed	Network moves with a user Walking speed	Network moves with a user Walking speed	Low/ stationary	Network moves with a user / walking speed	Network moves with a user / walking speed
User data rate	< 1 Mbit/s	> 20 Mbit/s	< 200 kbit/s	< 1 Mbit/s	< 10 Mbit/s

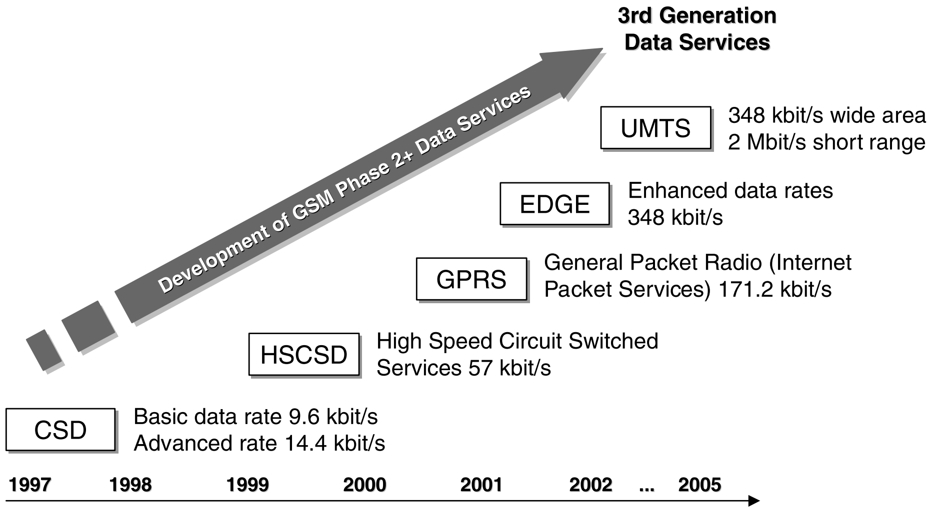


Fig. 3. Development of GSM towards 3G services.

requirements for UMTS include multimedia services that can be customised according to user's preferences. The user should also experience a consistent set of services while roaming outside home network [45].

Terrestrial UMTS operates in the assigned 2 GHz frequency band. In addition, there are a number of possible frequency bands for the terrestrial UMTS/IMT-2000 [44]. Terrestrial UMTS utilises new efficient radio access architecture called Universal Terrestrial Radio Access (UTRA), which combines TDMA and Wideband Code Division Multiple Access (W-CDMA). The band is compatible with the GSM radio band and allows these network types to coexist. The access system will provide at least 144 kbit/s for full mobility applications in all environments, up to 348 kbit/s in macro cell and micro cell environments, and up to 2 Mbit/s short-range coverage in micro and pico cell environments with limited mobility.

The satellite component of UMTS will utilise the Mobile Satellite Service (MSS) frequency allocations and provide a global coverage with service compatible to terrestrial UMTS systems. However, the satellite systems will remain supplementary to the terrestrial systems in special areas and for special users.

2.2. Micro cell networks

Micro cell networks cover smaller, restricted areas like a block of buildings. The mobility of the users is expected to be fairly low, relating to slow speed vehicles. The cell size is also small, and frequent handovers take place when the user moves faster than walking speed. Two example networks are presented here: fixed wireless access networks and digital cordless phones.

2.2.1. Fixed wireless access

The last hop between an Internet Service Provider (ISP) and a customer can be implemented using wireless access instead of fixed telephone lines or cable TV modem. Similarly to all wireless access systems, a line of sight is required for operation. Satellite systems are already used for this purpose, but in the following only the terrestrial systems are discussed. The main standardisation work takes place in the IEEE 802.16 working group and in the HIPERACCESS project of ETSI BRAN [5]. Both attempt to define a fixed Broadband Wireless Access (BWA) comparable to the traditional wired access networks, such as Digital Subscriber Line (DSL), Integrated Service Digital Network (ISDN), or digital cable TV.

IEEE 802.16 will specify the physical and MAC layers for BWA systems [22,40]. The network topology contains point-to-point or point-to-multipoint connections carrying data, video, and voice. 802.16 is expected to provide a peak capacity from 2 to 155 Mbit/s for a single connection. The MAC protocol should be able to support also higher bit rates. The frequency band is 30 GHz, but the specification allows scalability from 10 to 66 GHz. The 802.16 BWA system is also expected to provide services for authentication, authorisation, and data privacy.

HIPERACCESS supports a large variety of networks including the UMTS core network, Asynchronous Transfer Mode (ATM), and IP based networks. Multimedia requirements were included in the development from the start. The HIPERACCESS standards will specify the physical and Data Link Control (DLC) layers. Several different licensed frequency bands will be supported, such as the 32 and 40 GHz regions. The data rates required from the system are over 36 Mbit/s. HIPERACCESS functional specification is expected at the end of 2001, and test and conformance specification at the end of 2002 [3].

2.2.2. Digital cordless telephone

A residential cordless telephone provides a wireless counterpart to a legacy wired phone. However, digital cordless systems with enhanced functionality are able to support more sophisticated applications and provide relatively high data rates within pico cell environments. Examples of such systems are Digital Enhanced Cordless Telecommunications (DECT) in Europe, Personal Handyphone System in Japan, and Personal Access Communications Services (PACS) in the United States. There are a number of similarities among these cordless systems with respect to data rates, frequency bands, and coverage areas. In the following, the European DECT system is taken as an example [24].

The standard for the DECT Common Interface (CI) was published by ETSI in 1992, and the second edition of CI was finalised in 1996. The first DECT products, residential cordless phones, started to appear in the market a year later. So far, phones and cordless PBXs (Private Branch eXchange) have been the main product group utilising the system. Other applications have been Local Area Network (LAN) cards utilising DECT for wireless access to a fixed network [26]. The DECT systems operate in a spectrum band of 1.88–1.9 GHz, which is reserved for the system in Europe. The radio band is divided into ten carriers, each containing 24 TDMA timeslots. The raw data rate for each TDMA channel is 32 kbit/s and after error protection the rate is 22 kbit/s. Multiple channels can be dynamically reserved by a single terminal, thus resulting into 288 kbit/s birectional and 552 kbit/s unidirectional data rates.

DECT contains functionality for handovers, encryption of radio transmissions, and authentication, which makes it a complete system platform for phone applications. However, for local data communications Wireless Local Area Networks (WLAN) offer better performance.

2.3. Pico cell networks

Wireless LANs (WLAN) can be used as an extension or as an alternative to fixed LANs. First products were limited to the former and enabled special portable terminals to access fixed network resources, for example databases in warehousing and health care applications. Currently, whole local area networking can be set up very conveniently by WLAN. WLANs are primarily targeted for high data rate applications in pico cells, but the data rate has not been previously comparable to fixed LANs. However, WLANs are rapidly gaining significance as mobile devices are replacing personal desktop computers and major advances in the WLAN technology are taking place. The overall trend is towards higher bit rates, interoperable networking with other systems, international standards, and interoperable products between different WLAN product manufacturers. WLAN will be a very important local access method for UMTS based 3G services and is therefore intensively developed in the USA, Europe, and Japan.

2.3.1. IEEE 802.11

IEEE initiated the 802.11 working group in 1990 for developing a WLAN standard for the 2.4 GHz unlicensed ISM (Industrial, Scientific, and Medical) frequency band. Reasons for the band selection were its global availability (with some national and regional limitations) and maturing radio technology. The IEEE 802.11 standard was approved in 1997, but first products appeared already before that and were based on a draft standard.

The IEEE 802.11 standard specifies the physical layer and the MAC protocol layers. The network architecture supports both an infrastructure based topology and a distributed topology with independent peer-to-peer connections. In the former, a master station that can also operate as an access point to wired networks controls the network. The standard includes support for authentication of peer terminals, data encryption, and power saving functionality. Both asynchronous and time-bounded data transfer services are specified.

Currently, the IEEE 802.11 family of standards includes specifications for five types of physical layers. Four of these are based on radio technologies and one on diffuse infrared technology. Still, all physical layer alternatives utilise the same MAC protocol. Three of the radio standards utilise the 2.4 GHz frequency band and one is designed for the higher 5 GHz frequency band.

The original 802.11 and the newer 802.11b standards specify 2.4 GHz radios that utilise spread spectrum technologies: Frequency Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS). The original 802.11 data rates specified for the physical layers are 1 and 2 Mbit/s, thus being lower than in legacy fixed LANs. The modulation is Gaussian Frequency Shift Keying (GFSK) and Binary Phase Shift Keying (BPSK) for FHSS and DSSS, respectively. In addition, the DSSS physical sublayer uses the Digital Quadrature Phase Shift Keying (DQPSK) modulation in the 2 Mbit/s mode. In low network traffic load conditions, the 2 Mbit/s link capacity is enough to accommodate legacy LAN type of traffic characterised by data bursts with no real-time requirements [23].

As new applications require higher data rates and the capacity of fixed LAN and fixed access networks is increasing, the IEEE 802.11 standard has been updated for higher link speeds. IEEE 802.11b increases the data rates with 5.5 and 11 Mbit/s physical data rates. The used modulation is Complementary Code Keying (CCK) that is based on the DSSS technology. The multirate control of the 802.11 MAC protocol allows a terminal to switch from the 11 Mbit/s link speed down to 5.5 and further to 2 and 1 Mbit/s, if the radio channel quality cannot accommodate higher data rates [10].

A new task group 'g' (TGg), founded in March 2000, aims at further increasing of the data rates, but still utilising the same 2.4 GHz band. The purpose of TGg is to investigate 802.11b interoperable technologies that can support higher than 20 Mbit/s data rates and other performance improvements to the existing 802.11b standard [10].

IEEE 802.11 has also developed a physical layer for the 5 GHz band [6]. The 802.11a standard is based on the Orthogonal Frequency Division Multiplexing (OFDM) data modulation. The physical layer achieves data rates from 6 to 54 Mbit/s and is expected to be utilised in products already during the year 2001.

2.3.2. HIPERLAN

WLAN standardisation began in ETSI in 1992 by HIPERLAN Type 1 (HIPERLAN/1). The work area was expanded in 1996 to cover wireless ATM for short-range access (HIPERLAN/2), and remote fixed wireless access (HIPERACCESS). The HIPERLAN/1 functional specification was published in 1996. In the following year, ETSI began a new standardisation project called Broadband Radio Access Networks (BRAN) that continues the development work for wireless access systems [7]. In all of these activities, QoS and multimedia support in general has been an essential subject right from the beginning, contrary to IEEE 802.11.

HIPERLAN Type 1 (HIPERLAN/1) specifies a fully distributed WLAN architecture with peer-to-peer connections and one-to-many broadcast. A data forwarding functionality of HIPERLAN/1 terminals extends the coverage area of the network. HIPERLAN/1 can be utilised as an extension to a wired LAN, but is most convenient for forming ad-hoc network for different types of applications. HIPERLAN/1 specifies a MAC protocol and a physical layer operating in the 5 GHz frequency band assigned by CEPT. The MAC layer functionality contains a power consumption control, security, and multi-hop routing for forwarding terminals. HIPERLAN/1 supports both isochronous and asynchronous data transfer services. The modulation used is Gaussian Minimum Shift Keying (GMSK) resulting in a channel bandwidth of 23.5294 Mbit/s. The achieved user data rate depends on the packet payload length, reaching up to 19 Mbit/s [2,34].

HIPERLAN Type 2 (HIPERLAN/2) provides high speed communications between portable computing devices and interacts with existing wired networks, especially broadband IP and ATM networks [25,36]. The system is also meant to be a complementary broadband access mechanism for UMTS. Other projected application areas are broadband Internet access, as well as future real-time video and multimedia services. As an example for this, adaptation to the IEEE 1394 high performance serial bus is being specified in ETSI BRAN.

The fundamental idea of HIPERLAN/2 has been to develop a generic radio access network which is able to support QoS and which can be connected to different types of core networks via specific Convergence Layers. Unlike HIPERLAN/1, HIPERLAN/2 has a fully infrastructure based topology where terminals communicate with an access point that is consequently connected to a core network. The HIPERLAN/2 standard addresses the physical layer, the DLC layer including both Logical Link Control (LLC) and MAC sublayers and core network specific convergence layers above DLC. The physical layer and DLC specifications were finalised in the beginning of 2000.

HIPERLAN/2 utilises the 5 GHz radio band. The modulation method is OFDM with physical bit rate up to 54 Mbit/s and user data rates up to 36 Mbit/s. There are 19 frequency channels available in Europe, each channel being divided into 52 subcarriers. This provides a high performance in dispersive radio environments, such as in office environments with a high number of signal reflections. In addition, the physical layer supports several subcarrier modulation (BPSK, QPSK, QAM) and forward error coding alternatives for accommodating different radio path characteristics.

The HIPERLAN/2 MAC is based on dynamic TDMA with Time Division Duplex (TDD). The TDMA frame has a fixed duration (2 ms) and contains separate time slots for uplink and downlink data as well as for control channels. The uplink and downlink periods are dynamically divided according to current traffic density for efficient use of radio resources.

HIPERLAN integrates into the standard a number of control and management functions that are not present in 802.11. User mobility with handovers between access points while maintaining established connections is supported. As in 802.11, battery power conservation, link level authentication, and data security by encryption have been standardised. An advanced feature is an automatic frequency selection, in which access points can dynamically select the most appropriate radio channel for communicating according to interference measurements.

Currently, there are no HIPERLAN/1 products available, although the standard has been available for several years. The main reasons are that in 1996 markets were not ready for wireless multimedia in general and the 5 GHz technology was not mature enough. Today, networks with similar capabilities, such as broadband data rates and ad-hoc networking, have been presented for the 2.4 GHz band. 802.11a and HIPERLAN/2 on the 5 GHz frequency band will consequently overtake HIPERLAN/1. This shows the importance of the right timing of the standardisation efforts. Because of the advanced support for QoS, HIPERLAN/2 products are expected to appear for home and office multimedia applications. As an example, there are major companies producing consumer electronics that are planning video over HIPERLAN/2 for home applications [8].

2.3.3. MMAC

MMAC (Multimedia Mobile Access Communication) [37] is a consortium of mainly Japanese companies and universities to develop high-speed wireless access. MMAC is planned to be launched in 2002. MMAC comprises of specifications for several physical layers at 5, 25, 40, and 60 GHz, and upper layer specifications according to the application areas: high-speed WLAN, or fixed wireless access. The 5 GHz WLAN type includes protocols resembling 802.11 and HIPERLAN/2. As the name implies, multimedia requirements have been considered from the beginning. Wireless ATM has been one of the goals, which also resembles the HIPERLAN approach.

The highest speed is 156 Mbit/s for non-mobile terminals and 20–30 Mbit/s for slow mobile terminals supporting handovers. The modulation technology is OFDM. The MMAC, 802.11, and HIPERLAN/2 physical layers at 5 GHz will be harmonised, which in theory allows for a global coverage [33] and will ease the utilisation of the same type of physical components in different terminals. However, for roaming between these access network types, multi-mode terminals will be needed due to the differences in the upper layer protocols.

2.4. Personal area networking

Wireless Personal Area Network (WPAN) refers to very short-range systems with only few users and personal devices around a single human user. The area is therefore also called the Personal Operating Space (POS). POS typically extends up to 10 m in all directions enveloping a person whether stationary or in motion. Currently, the Bluetooth consortium and the IEEE 802.15 working group are actively developing and standardising WPANs.

2.4.1. Bluetooth

Bluetooth aims for a global standard targeted for a low cost, small size, and very low power consumption applications. The vision is to connect laptops, printers, Personal Digital Assistants (PDAs), cellular phones, headsets, keyboards, and any devices that benefit from a 'wireless serial cable connection'. The Bluetooth 1.0 specification was finalised in the beginning of 1999 and first mass products are expected to be available in 2001.

The Bluetooth specification covers the physical layer and data link layer protocols as a basic set. In addition, Bluetooth includes several network layer protocols that are used upon the basic bearer. The Bluetooth profiles define various ways to utilise the wireless communication in applications. Adopted profiles include a LAN access, data synchronisation, service discovery application, dialup networking, serial port emulation, and headset application profile, to mention a few. These enable legacy services such as TCP/IP networking over Bluetooth, cordless phone applications, and Wireless Application Protocol (WAP) [46] over Bluetooth. A unit does not have to support multiple profiles and can thus contain only the minimum functionality. The specification also includes link level authentication and data security by encryption [17,41].

A Bluetooth network has a centralised channel access control, but an ad-hoc type network (piconet) can be automatically created between a master unit and slave units when arriving within range. Bluetooth supports point-to-point and point-to-multipoint connections. Both asynchronous connectionless data traffic and synchronous voice connections are available.

The nominal link range is 10 cm–10 m, but can be extended to more than 100 m by increasing the transmit power. The number of active slave units is limited to seven per one piconet, but more units can remain synchronised to the network while being in a power save mode. Also, a Bluetooth unit can be a slave in another overlapping piconet and forward traffic between the two networks. Bluetooth operates in the 2.4 GHz unlicensed ISM band and utilises the FHSS technology in the radio interface. The GFSK modulation is used, resulting into a gross link speed of 1 Mbit/s and the maximum of 721 kbit/s for user data rate in asymmetric link. The voice channel bandwidth is fixed 64 kbit/s.

Recently, new Bluetooth 2.0 activity has been started. The scope contains several different items including a new, backwards compatible physical layer that will offer bit rates of up to 10 Mbit/s. Furthermore, extensive work is being done on new profiles, such as audio and video (multimedia) communications, in-car communications, and enhancing the PAN support in Bluetooth. Bluetooth will then expand from a serial cable replacement to a general-purpose short-range network with features overlapping those of WLANs.

2.4.2. IEEE 802.15

The IEEE P802.15 working group for WPANs was formed in March 1999, with a goal similar to Bluetooth [11]. The purpose of the group is to develop physical layer and MAC layer specifications for WPANs. The system will also address QoS for supporting various traffic classes. The 802.15 WPAN standard will target for interoperability between WPAN device and devices meeting the IEEE 802.11 standard.

802.15 WPAN aims at very low power consumption (average of 20 mW or less) with operation range of 10 m. The user data rate required is between 19.2 and 100 kbit/s. The frequency band is the same as in the 802.11 WLAN, i.e. 2.4 GHz. Therefore, the coexistence with other networks, either another 802.15 WPAN, a 802.11 WLAN, or other systems operating in the same physical area, is required [39].

The 802.15 working group contains four task groups targeted for different application areas. The task group 1 (TG1) utilises directly the available Bluetooth specification. As in other 802 standards, only the physical and MAC layers are defined. The emerging specification is in practise the same as the Bluetooth 1.0, concentrating to the two lowest protocol layers [11].

The task group 3 (TG3) is defining a new standard for high data rate personal networking. The required capacity is over 20 Mbit/s, which makes the network suitable for portable digital imaging and multimedia. The physical layer remains in the 2.4 GHz frequency band. A draft standard from TG3 is expected at the beginning of 2002[11].

A new task group 4 (TG4) was formed in November 2000. TG4 develops ultra low complexity and ultra low cost applications and solutions where battery lifetime is the key issue. The application areas contain wireless sensors, interactive toys, smart badges, remote control,

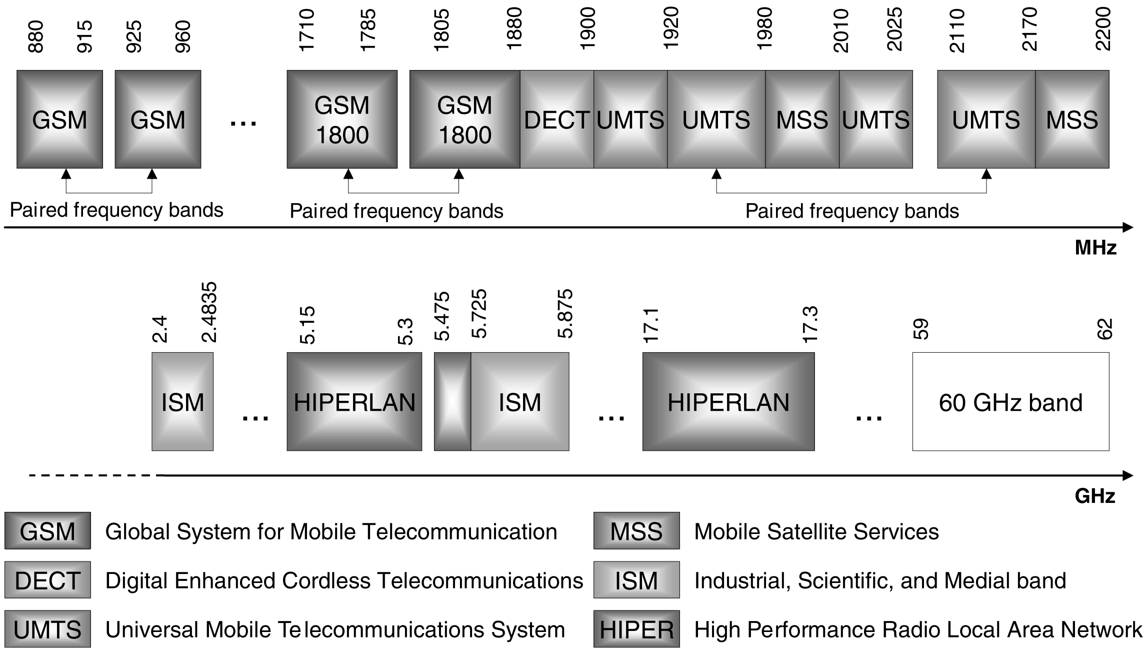


Fig. 4. Frequencies allocated for wireless systems in Europe [44]. The ISM and HIPERLAN bands are unlicensed.

and home automation. The targeted battery lifetime is from months to several years. The data rates are below 200 kbit/s, which is expected to be sufficient for the target applications.

In addition to the above mentioned groups, the task group 2 (TG2) is developing practises to facilitate the coexistence of the 802.15 WPANs and 802.11 WLANs on the 2.4 GHz band. Furthermore, in order to maintain the continuing interoperability with Bluetooth, the IEEE 802.15 Radio 2 Study Group (R2SG) is following the activities of the Bluetooth Radio 2 working group and will recommend a course of actions to the Bluetooth working group.

2.5. Frequency allocation

The traditional cellular systems operate on licensed frequency bands. Operators are given a piece of spectrum for their sole use from the total allocation of a particular system. This scheme works nicely with systems covering wide areas, such as GSM. However, given the limited amount of spectrum this means that only a few operators can have licenses on a geographical area.

For pico cell and personal area networks licensed frequencies are clearly not a feasible solution. By their nature, these systems are operated by companies, other organisations, or individuals. Furthermore, in order to be able to provide bandwidths over 2 Mbit/s these systems will demand a relatively large frequency band.

Typically, pico cell and personal area networks operate on an unlicensed spectrum. This means that a certain piece of spectrum is available for a set of systems, provided they fulfil the agreed requirements. This arrangement causes the operating environment, and the coexistence related problems to be very different compared to those of traditional cellular networks. For example, traditional network planning is not possible because anyone has the right to install base stations and therefore co-ordination of their physical locations would be very difficult.

A summary of both the licensed and unlicensed frequency bands is given in Fig. 4. The 2.4 and 5 GHz bands will be discussed in the following since they are the most essential bands because of global availability and due to the fact that the complexity and the cost of technology goes up with the frequency.

2.5.1. 2.4 GHz ISM frequency band

The 2.4 GHz Industrial, Scientific and Medical (ISM) band is currently the most popular band for wireless short-range devices. It is globally available, although there are some exceptions (e.g. less channels available) in some countries. However, the recent developments in Japan and France indicate that the strong market pressure is leading to the harmonisation of the regulation also in these countries.

The application category utilising the 2.4 GHz band is quite wide, including high-frequency heaters, telemetry and paging systems, wireless LANs, cordless phones, and microwave ovens, just to mention a few. From this point of view, it is likely that the band is quickly getting fully crowded. Especially the introduction of Bluetooth devices and the increased interest in WLANs is likely to foster this development. This potentially leads to worse interference, which in turn decreases the performance of WLAN and Bluetooth devices. Especially, the interference may hinder real time services with guaranteed QoS as both systems are targeted to the same applications. In the future, the 2.4 GHz band will be left for low-cost, short-range, and moderate

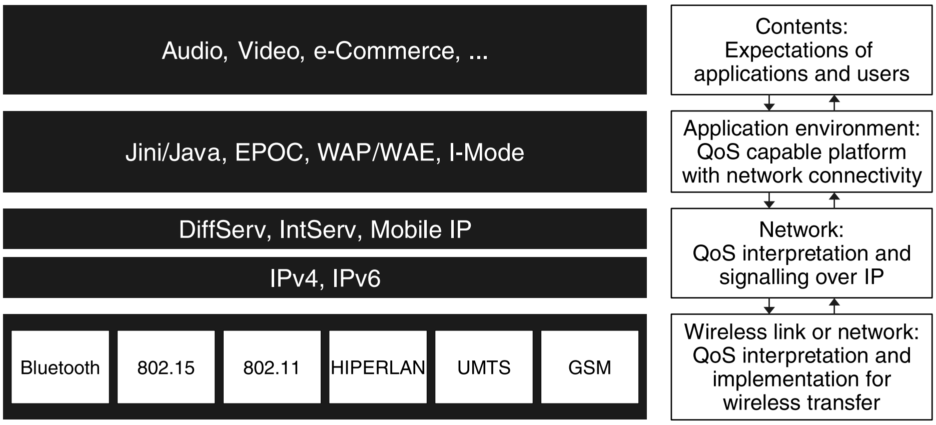


Fig. 5. Coexistence of wireless networks and applications.

data speed devices. Higher performance WLAN devices will gradually move to the 5 GHz band, which will follow the popularity of the 2.4 GHz band. The 5 GHz band is highlighted in the following. The region specific allocations and transmission power restrictions are covered.

2.5.2. 5 GHz frequency band

In Europe, the ERC decision (ERC/DEC/(99)23) designates the frequency bands 5150–5350 and 5470–5725 MHz (455 MHz in total) to HIPERLANs. For the first band between 5150–5250 MHz the maximum mean EIRP (Equivalent Isotropically Radiated Power) is 200 mW, and only indoor usage will be allowed. For the band 5470–5725 MHz the maximum mean EIRP is 1 W, and both indoor and outdoor usage will be possible. Furthermore, dynamic frequency selection is mandatory to provide a uniform loading of HIPERLANs across the minimum of 330 or 255 MHz in the case of equipment used only in the second band of 5470–5725 MHz. As well, transmitter power control is required to ensure a mitigation factor of at least 3 dB for uplink and downlink.

Currently the 5 GHz band is available for unlicensed devices in the United States, where the Federal Communication Commission (FCC) has allocated the Unlicensed National Information Infrastructure (U-NII) band. U-NII contains 300 MHz in three bands: 5150–5250, 5250–5350, and 5725–5825 MHz. The transmission power limits are 200 mW, 1 and 4 W EIRP, respectively.

In Japan, the 5150–5250 MHz band is currently allocated for radio LAN use. The band is restricted for indoor use only to protect MSSs feeder links. Furthermore, the transmission rate should be equal to or greater than 20 Mbit/s. The transmission power shall be less than 10 mW/MHz. In the future, the 5150–5250 MHz band will be allocated for WLAN services. To facilitate frequency sharing, Japan requires that the spectrum shall not be monopolised by a single system. This consequently requires a carrier sense method in medium access technologies.

As a summary, the frequencies of 5 GHz band allocated in each region overlap, which allows inter-operability of devices. Harmonisation of the physical layers in 5 GHz WLAN technologies also promotes this development. Unfortunately, it seems that the upper layer protocols of the 5 GHz systems will lack functional compatibility.

3. Applicability of personal wireless networks

Personal wireless communications can and will usually be implemented using a combination of the previously presented networks. For example, a personal area network moves everywhere with the user and can communicate with the outside world through a WLAN when the user is in a building or in public WLAN hot spot service area. In a car, a cellular 2G/3G network would be the best choice. When walking on a street, all of the networks may be used depending on the current location.

A user naturally prefers to use the network that currently provides the best services and lowest costs, but roaming from one network to another should be totally transparent to the user. In addition, the desired services should be easily available and up-to-date, independent of the user's current location. Quality of Service, inter-operability of the services, and accessibility of the content are the key issues discussed in the following.

3.1. Service quality and interoperability

In this paper, the QoS means the capability of a network to provide services, and to fulfil the user's expectations. The service performance requirements contain general dimensions such as service operability, accessibility, integrity, security, and retain ability. In addition, there can be a number of other requirements that are specific to each service [21].

The QoS support must be implemented in all the protocol layers of a multimedia network, as depicted in Fig. 5. For the seamless interoperability of heterogeneous networks on the service level, the QoS must be extended across different networks. This feature is generally missing from the current, widely used network protocols and applications. However,

Table 5
Summary of the traffic classes, QoS parameters, and implementation approaches

	User traffic classes	Main QoS parameters	Implementation approach
Differentiated Services	Standardised service classes and local classes inside a DiffServ domain, e.g. premium service, standard service, best-effort service, and virtual private line service	Service class indicated by the DiffServ field of the IP header	Different combinations of forwarding, scheduling, classification, queue management, and dropping disciplines at each network hop
Integrated Services	Guaranteed service Controlled load service Best effort service	Token bucket depth, bucket rate, peak rate, minimum polished unit, maximum datagram size	Resource Reservation Protocol (RSVP) and admission control for allocating resources for a stream. Reservation made at each network hop, combinations of forwarding, scheduling, classification, queue management, and dropping disciplines
802.1p (IEEE LAN family)	Interactive voice, Interactive multimedia, Streaming multimedia, Excellent effort, Standard, Background, and Best effort	Priority value (0–7)	Depends on the type of network. Priority used for signalling the service requirements
GPRS/UMTS	Conversational, Streaming, Interactive, and Background	Maximum bit rate, guaranteed bit rate, delivery order, service data unit size (SDU), SDU error ratio, transfer delay, traffic handling priority	Base station control
802.11 WLAN	Asynchronous and time-bounded data services	Channel access priority by different inter frame spaces	Polling algorithm of an access point controlling contention and contention free traffic
802.11 Tge	802.1p classes (expected)	802.1p priority (expected)	Priority based contention, polling algorithm of an access point
HIPERLAN/1	Isochronous and asynchronous services	User priority (high/low) and packet lifetime	Priority based contention and traffic forwarding
HIPERLAN/2	Expected support for: ATM 802.1p RSVP (IntServ), and DiffServ	Connections for separate data streams having different requirements for bandwidth, transfer delay, delay variance, bit error rate	Access point controlled connections
Bluetooth (802.15 TG1)	Synchronous connection and asynchronous connectionless links	Token rate, peak bandwidth, latency, and delay parameters	Master station controlled connections

the specification and standardisation of QoS related signalling is currently taking place. The utilisation of this will gradually be included into the networks in the near future. It should be noted that currently the service specifications might not be viable to be implemented, as bit rates are still low and current applications mostly cope with the best effort services.

The QoS management must allow an application to request or negotiate for the level of QoS. These requirements first correspond to the functionality required from the application's operation environment, such as embedded operating systems like EPOC [15], Java Virtual Machine with Jini networking [31], and Wireless Application Environment (WAE) of WAP. The adaptation between user requirements and the application environment takes place inside a user device. On the other hand, the network QoS signalling is realised using manufacturer independent signalling protocols and parameters. Regardless of the application types or network technologies, the IP layer will commonly connect different parties. The mapping of the requirements into IP QoS signalling is therefore one of the most critical issues for achieving a seamless end-to-end service.

As IP packets are transferred over wireless links, the QoS related control information is used for packet classification and service support. Wireless links may contain local QoS class definitions and support that are used for differentiating the payload IP packets. On the other hand, the IP layer QoS information may be directly used for assigning the performance parameters requested from the link layer. In the following, QoS is considered in more detail for IP and the QoS support in different wireless networks is discussed. Table 5 summarises the QoS classes, parameters and implementations.

3.1.1. QoS on the internet

Recently, the wireless ATM access has been a widely studied subject combining both a wireless network access and QoS issues. Despite of numerous proposals for wireless ATM, the problem of adopting the strict QoS of ATM over wireless links still remains. The interoperability with ATM core network has been targeted in the emerging BRAN and IEEE 802.16 standards, while the ubiquitous ATM networking, which has been the original vision behind wireless ATM and BRAN, seems to lose the battle to the ubiquitous IP networking [34].

Legacy Internet applications have not generally contributed to the QoS of the network but have themselves adapted to the only available type of service: the best effort service with varying bandwidth and delay characteristics. This approach is not suitable for time critical multimedia applications. In order to improve the situation, the Integrated Services (IntServ) and Differentiated Services (DiffServ) working groups of the Internet Engineering Task Force (IETF) [18] have proposed two different alternatives for including QoS into the Internet.

The IntServ approach utilises an end-to-end control signalling and resource allocation, while the DiffServ aims at specifying the Type of Service (IPv4) and QoS Class (IPv6) IP header fields so that packets can be handled separately. Only the information of these header fields is used for deciding the QoS support at each network node.

Currently, it can be seen that differentiated type of services will be the most dominant alternative in IP based access networks due to low complexity compared to IntServ. DiffServ does not strictly guarantee an end-to-end QoS by reservations. Instead, services are classified and different traffic classes handled separately at each network inter-node. DiffServ scales well for various application types, being adequate for most requirements as it provides nearly constant bit rates and delay variations when there exists enough bandwidth to accommodate all traffic classes.

The IntServ approach is more suitable for trunk networks and for environments where the bandwidth is the most restrictive resource, e.g. for wireless links that cannot support for a certain service without explicitly denying an access from lower priority services. IntServ has a scaling problem as the number of reservation on a network increases, and the complexity of signalling and functionality makes it a less appealing alternative. However, IntServ was originally developed for real time video and audio multicast services, and can be utilised for supporting a limited number of these types of applications, thus complementing the general DiffServ QoS architecture. To summarise, it is important to remember that these QoS technologies do not increase the total bandwidth of a network but instead divide it more efficiently between users.

The new generation IPv6 develops the Internet networking towards the mobile infrastructure. First, it provides enough address space for all the mobile or portable user devices, and for all the embedded systems that benefit from network access. Secondly, IPv6 enforces the security of the IP networking, as it has been included into the specification. Thirdly, IPv6 also includes the mobility of terminals as an integrated function. The mobility has already been specified over IPv4 as Mobile IP extensions, thus allowing a user or a terminal to move seamlessly outside a home network. Finally, QoS is a central target for building the new IP standard and the Internet 2 [12,28].

There has been active development in incorporating QoS into fixed LANs that have so far been the predominant access technology to Internet services. The new IEEE 802.1D standard for MAC bridges incorporates 802.1p traffic class expediting and dynamic multicast filtering. 802.1p provides a method for allowing preferential queuing and access to media resources by seven different *traffic class* priorities (3-bit priority value). The priority information is assigned by a user or higher layer protocols and signalled across LAN. IEEE 802.1Q is another supplementary standard for Virtual LANs (VLAN) to the 802 families. 802.1Q also defines an additional tag header field to the basic MAC frame format. The tag header is used to distinguish different VLANs but the field also transfers the 3-bit priority information for 802.1p [38]. Generally, fixed LANs provide enough bandwidth to accommodate all wireless access services, and the QoS support is therefore primarily needed in wireless networks. However, as the application requirements and traffic volumes increase, the LAN as a backbone network serving wireless access points will demand QoS control to accommodate end-to-end services.

3.1.2. GSM 2.5G and UMTS

The GSM 2.5G enhancements HSCSD and GPRS are capable of supporting a variety of service types. In HSDCSD, throughput and delays for user data can be bound because switched channels are used as a data bearer. The system is suitable e.g. for constant bit-rate video transmissions, but the set-up delays establishing a new connection, and the possible over or under allocation of resources are problematic. These challenges are addressed by GPRS that supports variable bit rate data services with session based QoS parameters [29].

GPRS share the same QoS definitions with UMTS. The QoS is based on a service classification of four classes: *Conversational*, *Streaming*, *Interactive*, and *Background*; each class corresponding to different application requirements. The implementation of the service class differentiation are based on a set of parameters, e.g. bit rate definitions, delivery order, error ratios, and packet sizes.

The conversational class sets the highest requirements for QoS, while the background class has the lowest requirements and can be seen as a best-effort traffic class. The conversational class is capable for demanding applications that require data rates up to 384 kbit/s, a transfer delay under 150 ms, and a delay variance below 1 ms. In the interactive class the bandwidth and delay requirements are lower, but the reliability of the transfer must be high. Applications include e-commerce type of services and voice messaging. The streaming class applications include video and audio streaming. The class has high bandwidth requirements but tolerates longer transfer delays (up to 10 s) [43].

UMTS will benefit from the work of the IETF [18], which extends the IP based networking to cover also mobile communications. In addition, the IP QoS approaches allow an UMTS service provider to enhance the end user services [45].

3.1.3. IEEE 802.11

In the IEEE 802.11 standard, the priority of medium access is selected according to the frame type (not directly according to the traffic type). The priority is based on inter-frame spaces, which means that a higher priority frame can be sent earlier on an idle medium than a lower priority frame. 802.11 contains three inter-frame spaces, where the highest priority (shortest space) is reserved for control frames and for the following segments of the same fragmented MAC Service Unit (MSDU). The lowest priority is for a distributed control architecture (ad-hoc network), where all competing terminals utilise the same access priority. The inter-frame space is followed by another contention window for resolution. The purpose is to provide fairness of access among all competing terminals.

The actual support for user data related access parameters (time-bounded services) is implemented only in the centrally controlled network topology of 802.11. A controlling terminal (access point) uses the middle channel access priority to seize the control of the channel. The terminal may then generate contention-free periods by frequently polling other terminals, always acquiring the channel control due to the higher access priority compared to the distributed control. The channel is reserved to the polled party for the transmission of a single frame.

In IEEE 802.11, a MAC user may select between two traffic types: *contention* and *contention free*. A defined poll repetition rate for contention free time-bounded data will provide shorter transfer delay variations than with full contention based data. This depends on the implementation of the used polling algorithm in the controlling access point [38].

As the bit rates and services develop, the need for the link layer QoS for WLANs has been taken into consideration by IEEE. The IEEE 802.11 task group e (TGe) is currently defining the QoS into the WLAN standard. The implementation of the new functionality is still open, although the IEEE 802.1p type of QoS signalling is likely to be adapted also by the IEEE WLANs.

3.1.4. HIPERLAN/1

HIPERLAN/1 is a completely distributed-control network. The support for various services is based only on channel access priorities that terminals use for contention. HIPERLAN/1 supports five priority levels during the resolution phase for channel access. The purpose of the resolution is to provide an access to the terminal with the highest priority data, and in case of several equal priority terminals, fair access between them.

A HIPERLAN/1 user can assign two QoS related parameters for MSDU, an *MSDU lifetime* (0–16,000 ms) and a *user priority* (0–1). The third QoS related parameter of the system is the *residual MSDU lifetime* which is a timer value storing the remaining lifetime of a packet. This value, together with the user assigned priority is then used when calculating the channel access priority at each node.

3.1.5. HIPERLAN/2

The members of the BRAN standards aim at providing wireless access to different core networks, such as IP, ATM, and UMTS, and supporting their applications and QoS. Especially ATM contains a rich QoS concept which must be considered when designing the link level access to the network [4]. The communication in HIPERLAN/2 is connection-oriented, which enhances the service differentiation and the support for application requirements. Each HIPERLAN/2 connection between an access point and a mobile terminal can be assigned with a specific QoS negotiated during the establishment of a connection. The QoS parameters are e.g. connection bandwidth, transfer delay, delay variance, and bit error rate. For simpler applications, the system also supports an approach in which connections can be assigned using only priority levels in relation to other connections.

For HIPERLAN/2, the ATM and 802.1p QoS signalling and service support functionality are likely to be acquired. This will accommodate seamless networking with different types of core networks, such as ATM and Ethernet. Furthermore, the support for IntServ and DiffServ should be taken into the specification for enabling service adaptation with any type of IP network. Also IEEE 802.16 is targeted for similar use, such as offering broadband access to the ATM network and should therefore preserve its QoS assignments.

3.1.6. Bluetooth

Bluetooth supports two types of links: synchronous connection oriented and asynchronous connectionless link. The former is a voice connection and the latter is used for other types of data. Voice links are primarily targeted for telephone related applications, e.g. for transferring voice between a mobile phone and a wireless headset. The links have a fixed 64 kbit/s throughout, but the error coding can be altered according to radio path properties.

For data links, Bluetooth includes QoS capabilities for supporting user oriented bandwidth allocation and latency control. The master of the piconet uses a polling list to determine when a slave is queried and what is the time interval between consecutive queries. These parameters can be negotiated during the link establishment. The Bluetooth Host Controller Interface (HCI) contains a rich set of QoS related parameters that are likely to have actual meaning only in the future versions of the system with higher bit rates. The specified HCI functions allow an application to assign the token rate, peak bandwidth, latency, and delay requirements.

3.2. Accessibility of content

Wireless connectivity has already brought many benefits to users, but is at the same time also raising new problems. A familiar problem for mobile users is that personal and business data is not synchronised between a portable terminal and e.g. a home network personal computer. The problem also relates to the synchronising of data with other users. SyncML [16] is one of the newest attempts to address this challenge. SyncML is a protocol that makes use of Extensible Markup Language (XML) at the highest level. Supported protocols include IP, WSP (Wireless Session Protocol used in WAP), and OBEX (Object Exchange Protocol in Bluetooth, IrDA, and other local connectivity). SyncML can also be deployed over SMTP, POP3, IMAP as well as proprietary wireless communication protocols. SyncML should function equally in Virtual Private Networks (VPNs) as well as in non-trusted networks. First SyncML-compatible products will be launched in 2001.

Another problem is the accessibility of area or time-dependent information that is filtered out from all of the available data on the Internet. Reservation of a hotel room, unlocking a door, payment of a parking fee, or getting more information from a piece of art in a museum would be convenient to do online using a wireless terminal. Personalised, area or time-dependent services can be best implemented if the location of the user is known. The user could be located using existing cellular networks, but in some countries this is prohibited or the accuracy is not sufficient. Global Positioning System (GPS) cannot be used indoors. Pico cell and personal area networks best combine the accurate localisation with a high-speed Internet connection. From the beginning, the Bluetooth concept has approached the interactivity with the environment. However, it is suitable only for services that do not require lots of raw data. WLANs are needed for full multimedia content services and in the future, WLANs could also cover personal area network applications [20].

Even though QoS was properly signalled over all network technologies from end-to-end, the last hop wireless link would still limit the accessible content. For example, a video-on-demand in TV size would not even be reasonable to be viewed on a small portable display. WAP was the first initiative to scale the contents of the Internet to fit the terminals. The most important feature is that the content have to be adjusted according to the terminal's features, and only the necessary data in compact form is transferred to the device. Unfortunately, WAP is currently able to support only text and very simple graphics and has thus been a disappointment to everyday Internet users. WAP will benefit from GPRS, but might lose the battle when portable devices in general are able to handle web contents. The same idea of content scaling is found in NTT DoCoMo's iMode [14] terminals, but by restricting the contents already in a server. The benefit is the usage of standard Internet protocols, but the same content is still not advanced enough for e.g. a PC browser. The idea of content scaling is reasonable also for future higher speed wireless networks, but WAP and iMode as today are not yet the best solutions.

4. Discussion

Being connected wirelessly everywhere, any time, and with any source of information is taking place with the efforts put on network technology, interoperability, and content accessibility. Convergence of multimedia, the Internet, and mobility is not yet reality, although it appears to be technically feasible within months. Commercial and other non-technical issues mostly dictate the pace of the convergence. For example, too greedy charging of the services will slow down the penetration. On the other hand, proprietary, non-standard systems and applications will also slow down the overall take-off.

An interesting contest will take place between 3G (UMTS) and 2.5G cellular networks. Huge investments to the 3G licenses might not pay back as expected, because similar services and quality can be achieved using EDGE and GPRS [45]. Especially new frequencies will extend the lifetime of GSM significantly all over the world. However, it should be noted that with the wide take off of the new GSM services we are likely to see capacity limitations.

On the other hand, UMTS is much more than just a W-CDMA access technology. The concept includes roaming between different networks and mobile services previously presented in this paper. In a very general view, 3G or UMTS is an umbrella that spans over all macro, micro, and pico cell networks as well as personal area networks. Independent of the term used, personal wireless networking will consists of all the network types in the future.

In our opinion, the winners are the GSM based 2.5G networks, fixed wireless access networks, IEEE 802.11 and HIPERLAN/2 WLANs, and Bluetooth, as well as the IP protocol that enables the convergence. Fixed telephone copper wires are owned by monopolies and thus scaleable fixed wireless access networks will be a competitive solution. That ensures a reasonably priced information highway also to homes. In restricted areas like a building, WLANs and Bluetooth offer an attractive solution for local access, and at that time, the capacity of cellular networks is saved. HIPERLAN/2 is especially suited for wireless multimedia networks, which is why consumer electronics vendors are interested in it [8].

The losers will be DECT, HIPERLAN/1, HomeRF [32] and other proprietary access technologies. DECT has been popular for a cordless phone. In the future, mobile phones will support both cellular network and WLAN or Bluetooth, which means that DECT alone is not viable any longer. Dual mode GSM/DECT phones may still exist, but DECT is not reasonable for a pico cell network with multimedia content [26]. HIPERLAN/1 failed as a wireless access technique due to the poor timing of the specification. In general, all systems that require a licensed frequency band and are still being managed by end users are not reasonable. The IP QoS signalling supported by applications and by underlying networks, provides near the same level of service that

ATM was designed to offer. HomeRF could be seen as an intermediate stage combining rather old technologies and, therefore, have only minor importance in the future.

The question marks are the IEEE 802.15 family, if it diverges far from Bluetooth, and in the fixed wireless access field, HIPERACCESS and IEEE 802.16 will continue to evolve and hopefully converge also in the upper layer protocols. A question mark is also how long the unlicensed 2.4 GHz frequency band will be usable, since lots of devices operating on this band will interfere with each other, at least locally. This, in turn, promotes the use of 5 GHz ISM band for high-performance WLANs. In the future, the 2.4 GHz band might be dedicated for low-price, low data bandwidth devices.

In a wider perspective, as the 3G terminals and services become more common, the spectrum allocated for 2G systems can be seen as a natural resource for 3G applications and can be migrated to UMTS/IMT-2000 [44]. Factors promoting UMTS over the advanced 2G systems are the improved spectrum efficiency, the possibility to manufacture lower-cost terminals, and the technical capability to offer data rates over 2 Mbit/s. However, due to the limited amount of spectrum available it will be possible to provide only a few 2 Mbit/s connections per cell. Due to the same reason, the end user price of such a connection is likely to be rather high. In the light of this, the pico cell networks are a natural complementary technology being able to provide higher bit rates with cheaper price at hot spots like airports and railway stations. The overall success of each of the discussed systems will depend on the support of QoS, ease of use, pricing, and content accessibility.

References

[1] A. Furuskar, S. Mazur, F. Muller, H. Olofsson, EDGE: enhanced data rates for GSM and TDMA/136 evolution, IEEE Personal Communications 6 (3) (1999) 56–66.

[2] EN 300 652 (1998–07). Broadband Radio Access Networks (BRAN); High Performance Radio Local Area Network (HIPERLAN) Type 1; Functional specification, 1998.

[3] ETSI BRAN, Terms of Reference, published in http://www.etsi.org/BRAN/BRAN_ToR.htm.

[4] ETSI, Broadband Radio Access Networks (BRAN); High Performance Radio Local Area Network (HIPERLAN) Type 2; Requirements and architectures for wireless broadband access, TR 101 031 (1999–01), 1999.

[5] ETSI, Broadband Radio Access Networks (BRAN); Requirements and architectures for broadband fixed radio access networks (HIPERACCESS), Technical report 101 177, 1998.

[6] H. Li, G. Malmgren, M. Pauli, Performance Comparison of the Radio Link Protocols of IEEE802.11a and HIPERLAN/2, IEEE, Vehicular Technology Conference (VTC'2000), Tokyo, Japan 2000 pp. 2185–2191.

[7] Homepage of ETSI BRAN project, http://www.etsi.org/bran/.

[8] Homepage of EU Eureka 1549 COMMEND (Consumer MultiMEdia Networks in Digital) project, http://www.commend.org/.

[9] Homepage of GSM Association, http://www.gsmworld.com/.

[10] Homepage of IEEE 802.11 Working Group, http://grouper.ieee.org/groups/802/11/.

[11] Homepage of IEEE 802.15 Working Group, http://grouper.ieee.org/groups/802/15/.

[12] Homepage of Internet2 Consortium, http://www.internet2.org/.

[13] Homepage of Nokia phones, http://www.nokia.com/phones/.

[14] Homepage of NTT DoCoMo I-Mode, http://www.nttdocomo.com/i/.

[15] Homepage of Symbian, http://www.symbian.com/.

[16] Homepage of SyncML consortium, http://www.syncml.org/.

[17] Homepage of the Bluetooth Special Interest Group, http://www.bluetooth.com/.

[18] Homepage of The Internet Engineering Task Force, http://www.ietf.org/.

[19] Homepage of the Third Generation Partnership Project, http://www.3gpp.org/.

[20] Homepage of the TUTWLAN research project, http://www.tkt.cs.tut.fi/research/tutwlan/.

[21] Homepage of UMTS Forum, http://www.umts-forum.org/.

[22] IEEE 802.16 Working Group on Broadband Wireless Access Standards, Broadband Wireless Access System Requirements, Draft 4, 10.8.1999.

[23] ISO/IEC 8802-11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 1999.

[24] J.E. Padgett, C.G. Günther, T. Hattori, Overview of wireless personal communications, IEEE Communications Magazine 33 (1) (1995) 28–41.

[25] J. Khun-Jush, G. Malmgren, P. Schramm, J. Torner, Overview and Performance of HIPERLAN Type 2, IEEE, Vehicular Technology Conference (VTC'2000), Tokyo, Japan 2000 pp. 112–117.

[26] J. Knuutila, J. Hämäläinen, T. Sipilä, Data Possibilities of DECT/GSM Dualmode Terminals, IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC'6), Taipei, Taiwan 1996.

[27] J. Knuutila. On the Development of Multimedia Capabilities for Wireless Terminals, Doctorate Thesis, Tampere University of Technology, 1999.

[28] J. Mikkonen, L. Lehtinen, J. Lahti, S. Veikkolainen, Radio Access Network Architecture for IP QoS, IEEE International Workshop on Mobile Multimedia Communications (MoMuC'98), Berlin, Germany 1998 pp. 109–119.

[29] J. Mikkonen, M. Turunen, An Integrated QoS Architecture for GSM Networks, IEEE, International Conference on Universal Personal Communications (ICUPC'98), Florence, Italy 1998 pp. 403–407.

[30] J. Rapeli, UMTS: targets, system concepts, and standardisation in a global framework, IEEE Personal Communications 2 (1) (1995) 20–28.

[31] J. Waldo, Alive and well: Jini technology today, Computer 33 (6) (2000) 107–109.

[32] K.J. Negus, A.P. Stephens, J. Lansford, HomeRF: Wireless Networking for the Connected Home, IEEE Personal Communications 7 (1) (2000) 20–27.

[33] K.-J. Jamshid, P. Schramm, U. Wachsmann, F. Wenger, Structure and performance of the HIPERLAN/2 physical layer, IEEE, Vehicular Technology Conference (VTC'99), Amsterdam, The Netherlands 1999 pp. 2667–2671.

[34] L. Taylor, HIPERLAN Type 1 Technology Overview, White Paper, 1999, available at http://www.hiperlan.com/.

[35] M. Eriksson, A. Furuskar, M. Johansson, S. Mazur, J. Molmo, C. Tidestav, A. Vedrine, K. Balachandran, The GSM/EDGE radio access network-GERAN; system overview and performance evaluation, IEEE, Vehicular Technology Conference (VTC'2000), Tokyo, Japan 2000 pp. 2305–2309.

[36] M. Johnsson, HiperLAN/2 — The Broadband Radio Transmission Technology Operating In the 5 GHz Frequency Band, 1999, White Paper, available at http://www.hiperlan2.com/.

[37] M. Umehira, A. Ohta, O. Kagami, H. Hojo, T. Kobayashi, A 5 GHz-band advanced wireless access system for mobile multimedia

applications, IEEE, Vehicular Technology Conference (VTC'2000), Tokyo, Japan 2000 pp. 2300–2304.
[38] P. Ferguson, G. Huston, Quality of Service, Wiley, New York, 1998.
[39] R.F. Heile. WPAN Background, IEEE 802.15 project report P802.15-99/027r1, 1999.
[40] R.B. Marks, The IEEE 802.16 working group on broadband wireless, IEEE Network 13 (2) (1999) 4–5.
[41] Specification of the Bluetooth System, v1.0 B, December 1, 1999, available at http://www.bluetooth.com/.
[43] UMTS Forum, Enabling UMTS/Third Generation Services and Applications, White Paper, 2000.
[44] UMTS Forum, Report on Candidate Extension Bands for UMTS/IMT-2000 Terrestrial Component, Report No. 7, 1999.
[45] UMTS Forum, The Path towards UMTS — Technologies for the Information Society, White paper, 1998.
[46] WAP Forum, Wireless Application Protocol, White Paper, 1999, available at http://www.wapforum.org/.

Publication [P2]

Hännikäinen M., Rantanen T., Ruotsalainen J., Niemi M., Hämäläinen T., Saarinen J., "Coexistence of Bluetooth and Wireless LANs", IEEE International Conference on Telecommunications (ICT 2001), Volume 1, pp. 117-124, Bucharest, Romania, June 4-7, 2001.

Coexistence of Bluetooth and Wireless LANs

Marko Hännikäinen[1], Tapio Rantanen[1], Jani Ruotsalainen[1], Markku Niemi[2], Timo Hämäläinen[1], and Jukka Saarinen[1]

[1] Tampere University of Technology
Digital and Computer Systems Laboratory
Hermiankatu 12 C, FIN-33720 Tampere, Finland
Tel. +358 3 365 2111, Fax +358 3 365 4575
marko.hannikainen@tut.fi

[2] Nokia Mobile Phones
Sinitaival 5, FIN-33721 Tampere, Finland
Tel. +358 50 511 7320, Fax +358 10 505 7606
markku.t.niemi@nokia.com

ABSTRACT

High-density Bluetooth environments will appear in near future, as the Bluetooth technology is being integrated into variety of personal and mobile devices. At the same time, Wireless Local Area Networks (WLAN) are already emerging as a common solution for both corporate and small office/home environments. Both systems operate on the same unlicensed 2.4 GHz (Industrial, Scientific, Medical) frequency band. This paper evaluates the coexistence of the Bluetooth and WLANs. According to theoretical modelling and spectrum measurements, the collision probability causing throughput reductions is a noticeable factor between the two networks, depending on the distances between interfering devices and the traffic loads of the networks. Practical data throughput measurements using WLAN products and Bluetooth development kits also indicate rising error probabilities on high traffic loads. However, when evaluating example application scenarios of Bluetooth, the practical effect of the interference can still remain non-significant.

I. INTRODUCTION

The unlicensed Industrial, Scientific, and Medical (ISM) frequency band at 2.4 GHz (2.4000 GHz - 2.4835 GHz) is an attractive and globally available alternative for a number of wireless communication systems. The two main systems utilising this band are Wireless Local Area Networks (WLAN) and the emerging Wireless Personal Area Networks (WPAN). Bluetooth [1] will be the first technology on the market in the WPAN field. WLAN provides wireless data access to a corporate backbone wired network and is becoming a commonly used solution also for small office and home environments. At the same time, Bluetooth will be integrated into a variety of devices benefiting from the wireless data transfer. Example applications range from desktop and laptop PCs to Personal Digital Assistant (PDA) devices and mobile phones, and further to wireless headsets and wireless sensors. Consequently, the WLAN and Bluetooth systems will be operating in a close proximity to each other, and often both wireless interfaces are integrated into the same device.

In order to prevent interference between systems and therefore to make the transmissions more robust against demanding radio medium, low transmission powers and spreading of transmission spectrum are required when operating on the ISM band. Most of the WLAN systems available on the market meet the IEEE 802.11 standard [2] for Direct Sequence Spread Spectrum (DSSS) or Frequency Hopping Spread Spectrum (FHSS). Similarly, the Bluetooth utilises the FHSS scheme. As the ISM band gets more crowded by WLAN and Bluetooth applications, interference between these two systems will occur.

This paper studies the coexistence of the Bluetooth and WLAN system by theoretical modelling, spectrum measurements, and by practical traffic measurements. The measurement results are compared with theoretical models taken from the referenced research work. Functionality for enhancing the coexistence and interoperability is proposed. The paper is organised in the following way. In the next section, the systems operating on the 2.4 GHz ISM band are shortly described. The coexistence architecture of WLAN and Bluetooth is studied in Section 3, covering both the radio and the link layer protocol characteristics. In Section 4, the coexistence is studied by theoretical models, in Section 5 by radio transmission spectrum measurements, and in Section 6 by transfer error measurements using Bluetooth development kits and wireless LAN adapters. Before conclusions, interference management functions are studied in Section 7.

II. OVERVIEW OF SYSTEMS ON THE ISM BAND

Currently, the 2.4 GHz ISM band is utilised by the IEEE 802.11 WLAN systems, HomeRF [12], microwave ovens, baby monitors, cordless phones, etc., depending on the geographical area. There are also a number of advanced versions of WLANs and WPANs being developed and standardised, which will increase the loading of the band and the effects of co-system interference.

A. IEEE 802.11

The IEEE 802.11 specifies the physical layer with 2.4 GHz radios and the MAC protocol for the data link layer. The network architecture supports both infrastructure based and ad-hoc networking. The infrastructure based topology having Access Point (AP) units for controlling the wireless Stations (STA) and for providing a gateway to wired networks is the most common alternative in practise.

The original IEEE 802.11 data rate of 2 Mbit/s has now been extended up to 11 Mbit/s in the new radio standard 802.11b [2], thus making the data rate comparable to fixed LANs. In the original radio, the modulation is Gaussian Frequency Shift Keying (GFSK) for FHSS. DSSS uses 11 bit spreading code and Binary Phase Shift Keying (BPSK) modulation for 1 Mbit/s and Digital Quadrature Phase Shift Keying (DQPSK) modulation for 2 Mbit/s rate. The IEEE 802.11b increases the data rates by the Complementary Code Keying (CCK) modulation. CCK is based on DSSS and the spectrum is spread similarly to the basic rate DSSS systems.

To further increase the data rates in the 2.4 GHz ISM band, the IEEE 802.11 has founded a new Task Group g (TGg) for investigating techniques to support data rates over 20 Mbps [15]. This is likely to increase the loading of the frequency band in future, as IEEE 802.11 WLAN systems increase their attractiveness by increasing data rates.

B. Bluetooth

Bluetooth is specified by the Bluetooth Special Interest Group (SIG) [16]. The system targets for providing voice and data transfer services over short distance wireless links. The Bluetooth 1.0 specification was finalised at the beginning of 1999 and first products are expected to be available during the year 2001. The current version of the specification is 1.1 published in February 2001 [1].

Bluetooth network has a centralised channel access control, but the network (piconet) is constructed automatically in ad-hoc fashion between master unit and up to 7 slave units. In addition, a number of more slaves can be associated to the same piconet while being in a park mode. Several piconets can form a scatternet, as a Bluetooth unit can participate into several piconets at the same time and forward traffic between these networks.

Bluetooth utilises GFSK (Gaussian Frequency Shift Keying) modulation with 1 Mbit/s gross data rate and FHSS with 1 MHz channels. The user data rate is dependent of the utilised link type. The asynchronous connectionless (ACL) data link provides up to 721 kbit/s asymmetric data rate. The synchronous connection oriented (SCO) voice link has 64 kbit/s rate, and up to 3 calls can exists in parallel.

Recently, activity for Bluetooth radio version 2 has been started for developing a physical layer having up to 10 Mbit/s data rates. First, a modified version of the current radio is being defined, providing 2 Mbit/s link speed by more advanced modulation. Both new radios are compatible with the original one and will operate on the same band [16].

C. IEEE 802.15

The IEEE 802.15 working group for WPANs was formed on March 1999, with destination similar to the Bluetooth [14]. The 802.15 Task Group 1 (TG 1) is, in fact, developing a standard based on the Bluetooth specification. In addition, the utilisation of the 2.4 GHz ISM band is expected to increase also by the work done by 802.15. First, a new group TG3 is developing a standard for achieving over 20 Mbit/s data rates. The work destines for WPAN applications. Another task group, TG4, targets to a new low data rate solution with ultra low complexity and power consumption. The data rates are below 200 kbit/s and can be scaled down to 10 kbit/s, which is still adequate for home automation applications such as wireless sensors.

IEEE 802.15 working group has also established a new task group (TG2) for developing practises to facilitate the coexistence of the 802.11 WLAN and 802.15 WPAN/Bluetooth. Similarly, the coexistence has been taken into concern by Bluetooth SIG that has established a working group to evaluate the effects of other 2.4 GHz technologies on Bluetooth and vice versa. The results will be published by best practises guidelines, or formally by improvements made to the Bluetooth specification or by creating new Bluetooth profiles.

III. COEXISTENCE ENVIRONMENT

Because of the extensiveness of potential applications areas of Bluetooth and WLAN it is impossible to define a single typical coexistence network topology. The picocell created by a Bluetooth device reaches 10 cm - 10 m around a Bluetooth device, while IEEE WLAN systems cover ranges up to 100 m in an office environment. According to radio transmissions powers, the 100 mW transmit power (+20 dBm) is used by the IEEE WLANs. Most Bluetooth devices operate with 1 mW (0 dBm) transmit power, thus being class 3 devices according to the Bluetooth specification. Also class 2 (0.25-2.5 mW) and class 1 (1-100 mW) devices are defined, with the class 3 requiring a power control. Consequently, Bluetooth picocells will be surrounded by higher rate and higher transmission power WLAN cells in real operation environments.

A single WLAN AP is capable for serving a large number of wireless stations. According to the Bluetooth topology, it can be estimated that several Bluetooth devices and piconets are operating on near distance from WLAN stations. This topology is illustrated in Figure 1. The density of these overlapping cells will greatly depend on the operation environment. A practical estimate in a small office/home environment is the overlapping of several independent Bluetooth cells with two or three WLAN cells. These WLAN cells will be using different frequency sub-bands in DSSS or different hopping patterns in FHSS.

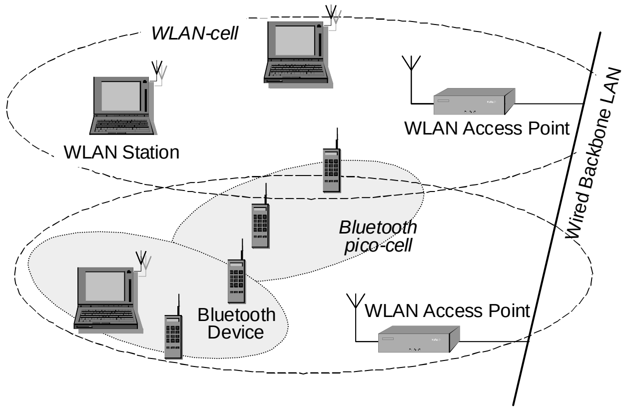


Figure 1. WLAN-Bluetooth coexistence topology.

A. Spectrum Spreading

The ISM band available between 2.4000 GHz and 2.4835 GHz is divided into 79 sub-bands 1 MHz each. There are national and regional exceptions to the use of this band, but efforts have been made to harmonise the regulation on this band globally [1].

The IEEE 802.11 FHSS hopping rate is dependent on the regulatory domain. 50 hop/s rate is commonly required. IEEE 802.11 defines 78 hopping sequences (each with 79 hops) grouped in three sets of 26 sequences. Sequences from the same set encounter minimum collisions and they should be selected for co-located systems. Theoretically, 26 FHSS systems may be co-located, but because synchronisation among independent systems is forbidden, the practical number of coexisting IEEE 802.11 FHSS systems is estimated to be around 15 [13].

In the IEEE 802.11 DSSS, the transmission spectrum is distributed over 22 MHz band around the central frequency. The maximum number of spatially overlapping, non-disturbing DSSS cells is thus three or in practice four independent cells can operate in the same space without significant disturbance.

In the Bluetooth, the hopping rate is significantly higher than required in the 802.11 FHSS system. A Bluetooth unit performs 1600 hop/s. The hopping is usually done over the whole frequency band, as 79 frequency channels are used in pseudo random order. The hopping pattern is based on the device address of the master units of a Bluetooth piconet, and each piconet utilises a different pattern. The Time Division Multiple Access (TDMA) slot structure in Bluetooth is synchronised to these frequency hops, as master units send in every second slot and slave units in the others. In addition, there are procedures for finding other Bluetooth units and creating a connection with them, during which the hopping rate is doubled.

B. Interference Parameters

There are a number of parameters that affect to the actual interference experienced by Bluetooth in the presence of WLAN transmissions nearby, and vice versa. First, the type of the WLAN system has to be considered. On the radio band, the DSSS type WLANs are more disposed to the Bluetooth interference than FSHH systems and the DSSS transmissions are also likely to cause more interference to Bluetooth [5]. This is because a wide band DSSS transmission is more probable to collide with narrow band Bluetooth than similarly hopping narrow band FHSS system. Another, very important reason for the significance of DSSS for the coexistence evaluation is the current popularity of 802.11b DSSS systems that provide a higher data rate.

Secondly, as the collision between different systems is caused by the same frequency used at the same time, the traffic loading on both Bluetooth and WLAN networks is an important factor. The main parameters in this case are the probability of transmissions, length of a WLAN packet, and the time difference between the start of transmissions of WLAN and Bluetooth.

It is expected that a collision results into the lost of packets. This is directly true in WLAN, since the after a single symbol error resulting into one or more converted bits, the whole packet will be rejected because of a Cyclic Redundancy Check (CRC) sum mismatch. Therefore, the length of a colliding Bluetooth packet does not probably affect the overall Packet Error Rate (PER) of WLAN. For Bluetooth, there are several packet alternatives that are used according to the application type. Data packets may have different lengths (1, 3, or 5 time slots) and the payload can be protected by Forward Error Correction (FEC) coding (2/3 rate shortened Hamming 10,15 code can be used for data). For voice, the single slot packets can be protected by 1/3 or 2/3 rate FEC. For simplicity, it is estimated that Data - High Rate (DH1) packets will be used in Bluetooth cells. With DH1 packets, a collision leads to increase in PER in Bluetooth correspondingly to WLAN as no FEC is used. DH1 packets are single time slot data packets and the radio hops into a new frequency sub-band after each packet. The minimum packet interval is 625 μs, which corresponds to traffic load of 100% in a piconet. Figure 2 presents the relations between lengths of Bluetooth and WLAN DSSS packets.

For DSSS type systems the packet length is more important variable in order to avoid mutual disturbance. The probability of shorter packets to collide is lower, and thus shorter packets with higher data rate are more efficient. The recommended value for maximum payload length in an FHSS PHY system is 400 octets at 1 Mbit/s and 800 octets at 2 Mbit/s, which corresponds to frame duration of less than 3.5 ms.

The distance between Bluetooth and WLAN transmitters from corresponding receivers is the third main factor affecting to the mutual interference. The topology studied in this paper assumes that the Bluetooth devices are located near WLAN stations and relatively far from APs. Therefore, downlink transmission from AP to STAs is more affected by the presence of Bluetooth than the uplink from STA to AP. On the other hand, transmissions from STA to AP are a more significant source of interference to the Bluetooth.

IV. THEORETHICAL MODELLING

The work for the IEEE standardisation for 802.11 WLANs and 802.15 WPANs has resulted into theoretical models for simulating the coexistence of Bluetooth and WLAN systems. These models study the effect of Bluetooth transmissions on an IEEE 802.11b DSSS network. The network topology presented in [4] assumes a commonly used WLAN installation, where WLAN STAs are evenly distributed within 20m range from AP. Bluetooth piconets are co-located with each STA. The model also takes the propagation loss into account. The path loss for line of sight signal in decibels is expressed as:

$$L = 20\log\left(\frac{4\pi r}{\lambda}\right), \quad r \le 8m \qquad (1)$$

where r is the distance and λ is the free space wavelength at 2.450 GHz (0.1224 m). Over 8 m distances, the path loss increases as r^n, where n has a value of 3.3:

$$L = 58.3 + 33\log\left(\frac{r}{8}\right), \quad r > 8m \qquad (2)$$

An approximation for the minimum Signal-to-Interface Ratio (SIR) for successful WLAN DSSS operation in the presence of narrow band Bluetooth signal colliding into the same band is 10 dB. This threshold has been verified using laboratory testing [4]. Results for applying the formulas (1) and (2) are presented in Table 1. The table presents the signal strengths of received WLAN transmissions at different distances from AP and the resulting maximum not-disturbing strength of a simultaneous Bluetooth signal. Also, the minimum (safe) distances required to the nearby transmitting Bluetooth device for achieving these not-disturbing signal strengths are given.

Table 1. Interference distances between WLAN and Bluetooth [4].

Distance to AP/m	Received WLAN signal /dBm	Maximum simultaneous Bluetooth signal/dBm	Safe distance to Bluetooth /m
4	-32.3	-42.3	1.27
10	-41.4	-51.4	3.66
20	-51.4	-61.4	9.95

For estimating the effect of Bluetooth on WLAN, a probability model for overlapping transmissions is presented in [4]. In this case, the SIR is expected to be less than 10 dB between the overlapping transmissions. The probability varies according to the loading of the Bluetooth piconet, difference of the start of transmission times of the Bluetooth and WLAN frames, and the length of a WLAN packet. The probability of Bluetooth piconet to hop inside the WLAN DSSS band (P_{hop}) is expected to be 25%. This results from the relation between the ISM frequency band size and the frequency spectrum size of the DSSS radio. The Bluetooth piconet loading (P_{load}) varies between 7% and 100% full loading. A special procedure used for establishing a Bluetooth piconet, inquiry, is also included. During the inquiry procedure, a Bluetooth master transmits short ID-packets using a double (3200 hops/s) hopping rate. This is likely to cause more interference to WLAN.

Since the two networks are not synchronised, the start of WLAN frame transmissions is expected to be a uniformly random variable in accordance to the phase of the Bluetooth TDMA slots. The start time and the duration of the WLAN transmission results into different overlapping (but not necessary collision) probabilities (P_{n_over}). The probabilities are depicted in Figure 2. For example, with 1500 bytes payload WLAN frame, the probability of overlapping with two Bluetooth 366 µs transmission slots (P_{2_over}) is 48%, and with 3 slots (P_{3_over}) 52%. During the inquiry procedure when Bluetooth has a double hopping rate, the probabilities of several overlapping transmissions increase.

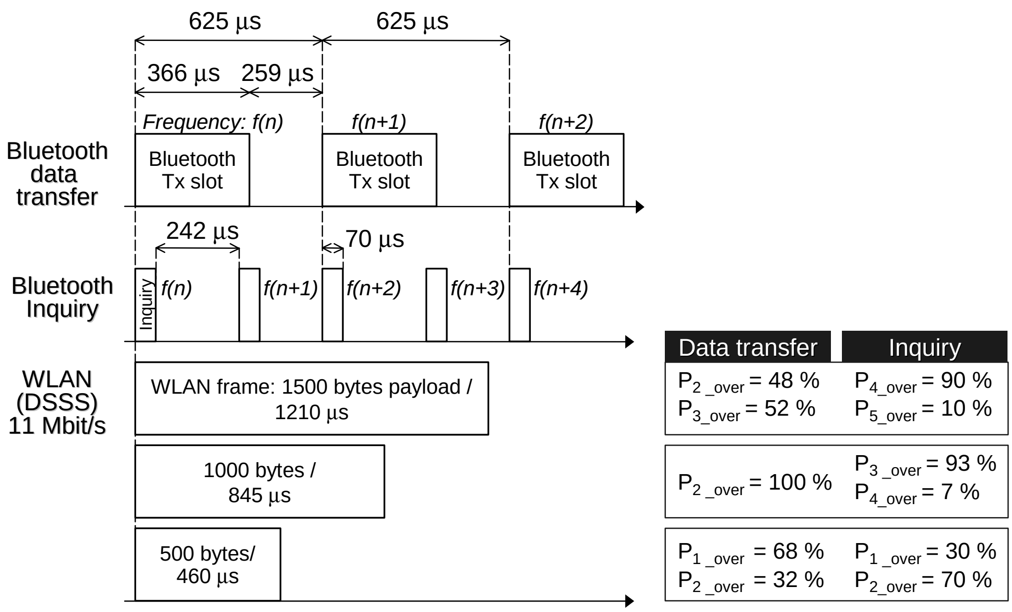


Figure 2. Frame lengths of Bluetooth and WLANs with probabilities of overlapping.

The probability of a collision in a single case with n overlapping frame transmissions can thus be acquired as [4]:

$$P_{coll}(n) = (1-(1-(P_{hop}P_{load}))^{n}) \quad (3)$$

and the total collision probability as:

$$P_{total} = \sum_{i=1}^{n} P_{n\_over} P_{coll}(n) \quad (4)$$

The collision probabilities between the IEEE 802.11b WLAN and Bluetooth with different WLAN payload sizes and Bluetooth loading levels are estimated in Figure 3 by applying the formulas (3) and (4) with the presented presumptions. The figure presents how the length of the WLAN radio frame and the loading of the Bluetooth increases the collision probability between the two networks. By combining the collision probabilities with the interference distances of Table 1, error probabilities of a given WLAN/Bluetooth topology can be estimated.

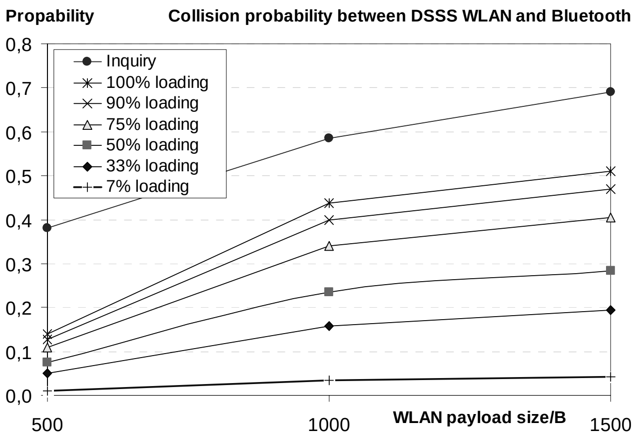


Figure 3. Collision probability between the 802.11b and Bluetooth frames with different WLAN payload sizes and Bluetooth loadings.

V. RF SPECTRUM MEASUREMENTS

A similar topology as theoretically studied was first taken to the subject of spectrum measurements. A Bluetooth device was originally located near the STA while the distance between STA and AP was increased in the steps of 4, 10, and 20 m. Also, the distance to the peer Bluetooth device was increased, away from the STA towards the AP. In addition, the RF spectrum of a microwave oven was measured.

The Bluetooth devices used for the spectrum and throughput tests were Ericsson's Bluetooth development products. First, the Bluetooth slave was the Bluetooth Application Tool Kit [6] that hosts the Ericsson's Bluetooth module. The Bluetooth master device for the testing was the Bluetooth Development Kit from Ericsson [10]. Both kits are based on the Bluetooth 1.0 specification. The kits contain host interfaces through serial port or USB, and using the host PC software it is possible to count the bit/packet errors for Bluetooth transmissions. The basic 1 mW transmission power (0dBm) was used for testing. The WLAN systems used for DSSS testing were Nokia WLAN products [11]. The WLAN access point was Nokia A020 for 2 Mbits/s link and A032 for 11 Mbit/s link speed. Correspondingly, C020 and C110 WLAN PCMCIA card adapters were used in WLAN stations (STA). The transmissions power of the WLAN is 100 mW (20dBm). The analyser used for the measurements was Hewlett-Packard 8593A (9kHz - 22GHz) spectrum analyser. The receiving antenna of the analyser was a simple dipole antenna.

The measured transmission energy spectrums for the Bluetooth and IEEE 802.11 DSSS (2 Mbit/s) are shown in Figure 4. First (a) and (b) present the measured signal spectrum at very close range (below 0.5 m). For a Bluetooth device, this is a predicted to be common operating range. WLAN is designed for longer transmissions periods, but the (a) presents the interference of uplink transmissions from STA on a Bluetooth device attached to the same terminal. This WLAN transmission is likely to envelop Bluetooth signals within same frequencies. In (b), the measurement is collected over a longer period (using the maximum hold mode that freezes the highest measured value), thus different hop frequencies over the whole spectrum are shown.

Another source of interference for both the Bluetooth and WLAN are microwave ovens. Figure 4 (c) presents the measured spectrum of a commonly used 800 W oven (Rosenlew RP117M) at 4 m distance. A water container was placed inside the oven during heating, which was presumed to absorb most of the energy. The energy of the interference signal seems to be quite high, the same level as a Bluetooth device transmitting in near proximity. The interference remains statically in the frequencies shown in the figure. For a DSSS WLAN, the interference caused by the microwave is likely to depend highly on the frequency sub-band used.

In figures (d), (e), and (f) the spectrums of simultaneous Bluetooth and WLAN signals are presented at different distances from the receiver. Compared to Table 1, the WLAN signal seems to be weaker than expected, but the Bluetooth applies more accurately to the theoretical propagation loss model. According to the measured spectrum, it can be estimated that the Bluetooth signal is more robust to WLAN interference, also over longer distances. WLAN will clearly suffer from the narrowband Bluetooth interference when studying the frequency spectrum. However, as the spread spectrum is de-spread at the WLAN receiver, the processing gain (G) achieved by the 11 bit spreading sequence can be calculated as [17]:

$$G = 10 \log(11) = 10.4 \text{ dB} \quad (5)$$

The gain will decrease the error rate of WLAN.

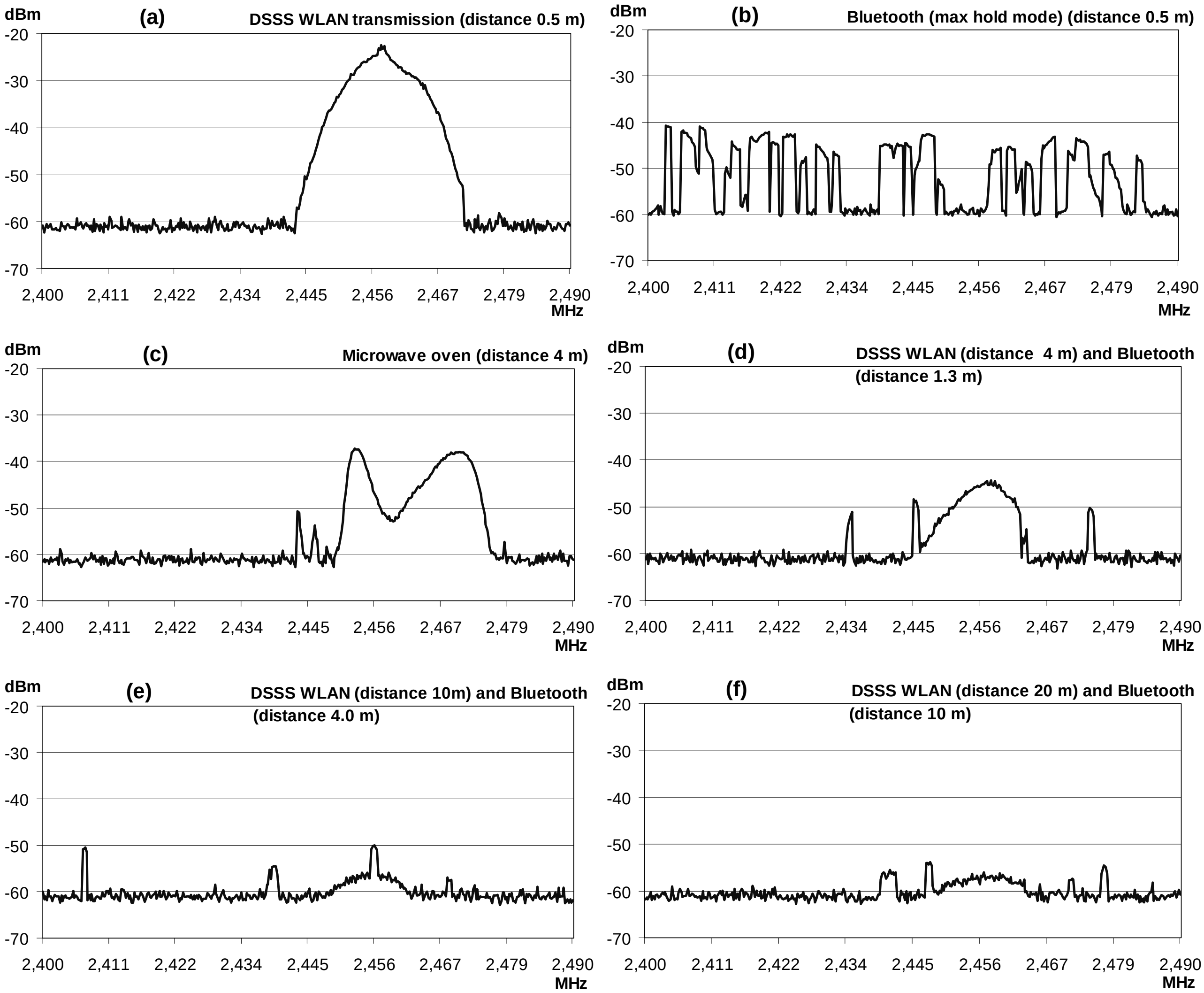


Figure 4. Transmission energy spectrum measurements of Bluetooth and WLAN.

VI. TRANSFER ERROR MEASUREMENTS

The probabilities of data transfer in WLAN and Bluetooth were also estimated using practical measurements with the test equipment. The topology used during the test is presented in Figure 5. The topology consists of IEEE 802.11b WLAN transmitting from AP to STA and a Bluetooth piconet that is located between the AP and STA or right next to the STA. The distances between the STA, AP, and the Bluetooth piconet are modified as presented in the figure. In addition, two different distances between the master and slave device of the Bluetooth piconet were used.

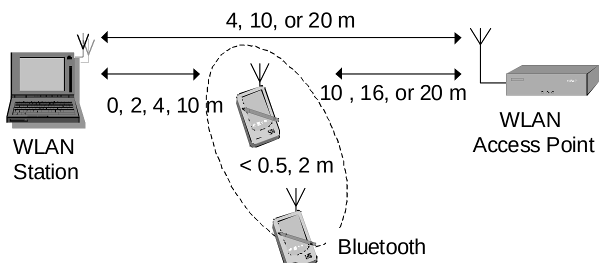


Figure 5. Topology used in transfer error measurements.

Figure 6 (a), (b), and (c) present the effect of Bluetooth piconet transmissions on a WLAN transmission. The distance between WLAN AP and STA is increased from 4 m, to 10 m, and further to 20 m. Also, the Bluetooth piconet range in grown from 0.5 m, to 4 m, and finally to 10 m. The total number of WLAN frames transmitted at each case was 6200. The Figure 6 (a), (b), and (c) present the portion of retransmissions (resulting from transfer errors) that are required to transfer this amount of frames correctly.

The importance of the traffic load in the Bluetooth piconet is a clear factor affecting to the WLAN error rate, which can be seen in the figures. Bluetooth load is presented in inquiry procedure and in a low traffic load (7%). With low Bluetooth loading the number of resulting WLAN retransmissions is also very low, but noticeable. For example, in the Figure 6 (b) the retransmissions of 1500 payload packets increase from 0.27% to 2.4%, when there is a Bluetooth device at 0.5 m distance transmitting with 7% piconet load. An inquiring Bluetooth device at same distance resulted into 24% retransmission probability.

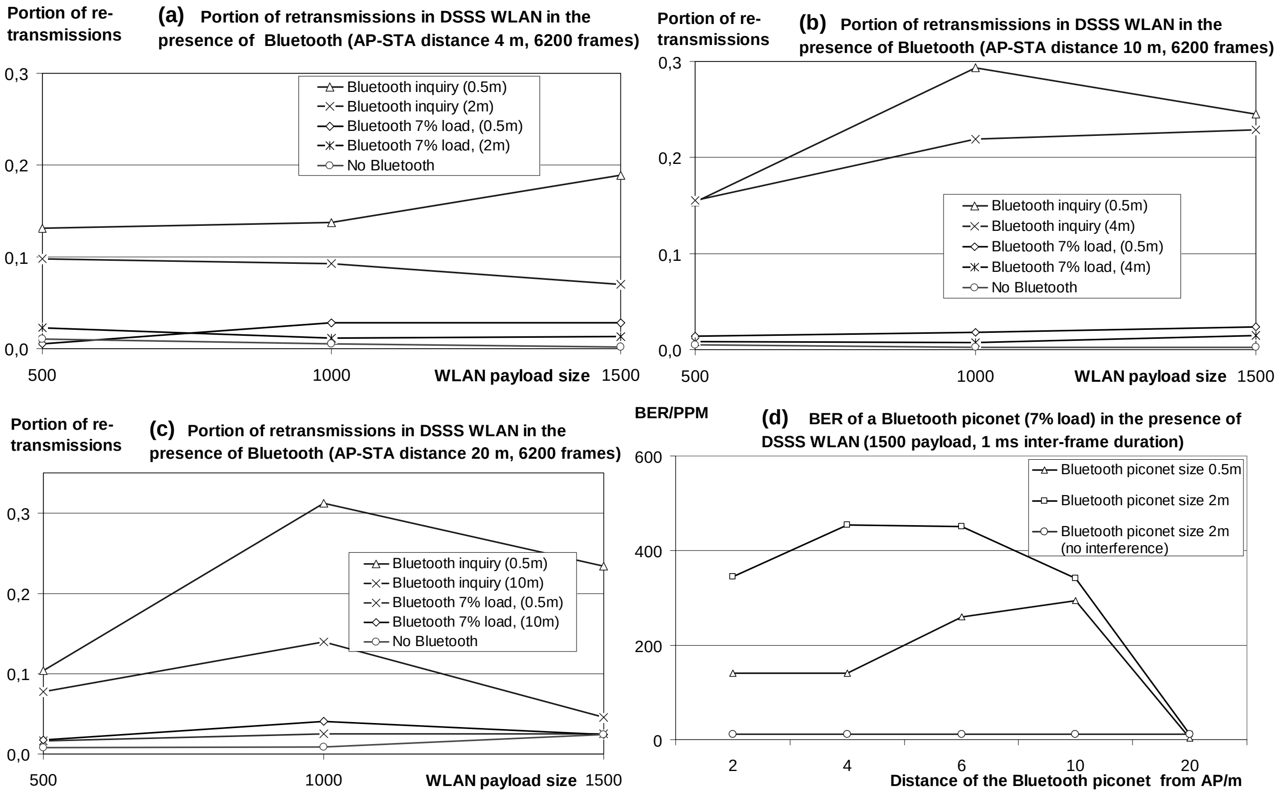


Figure 6. Data transfer error measurements with coexisting Bluetooth piconet and IEEE 802.11b WLAN transmissions.

Generally, as the Bluetooth traffic load and the WLAN packet lengths are increased, the presence of Bluetooth has a clear effect on the error probabilities. The amount of retransmissions remains higher, even when the distance between STA, Bluetooth, and AP is increased. There are also accidental variations shown in the graphs, as measurements are not averaged. Therefore the graphs should be taken as informative, not as exact values.

In Figure 6 (d) the Bit Error Rate (BER) in Parts Per Millions (PPM) of a Bluetooth piconet is presented under the interference of IEEE 802.11b DSSS WLAN. The size of the Bluetooth piconet is also changed, and seems to have a clear effect on the BER. Also, WLAN transmissions seem to result into errors until the distance from AP grows over 10 m. The average BER without interference is measured to be about 10 PPM in a 2 m size piconet.

VII. INTERFERENCE MANAGEMENT

It is expected that in future a single terminal (e.g. a laptop) is likely to have both WLAN and Bluetooth network interfaces. These two systems can also operate at the same time, as no dependencies between them necessarily exist. If there is no isolation between the two radio transmitters, serious interference is likely to occur and the transmission range and capacity of both networks will be reduced. This two-radio problem can be approached by careful antenna design, thus providing isolation to the transmitters by directive radiation beams.

A solution to improve the coexistence in general case between Bluetooth and WLAN is to utilise adaptive interference management. The interference management functionality, usually residing in an access point or in a master device of a network, has a four basic step procedure [7].

1. *Monitoring of the RF environment* - Master or AP can apply several methods for measuring the link quality. For all the sub-channels, a retransmission counter can be maintained. Large number of retransmissions in some frequencies, detected by the CRC-errors of the received frames, relates to an interference or fading. The received signal level can also be measured and a fixed or moving threshold value maintained. Errors in a known bit sequence, for example in the training sequence of the WLAN frame pre-ample header or in the channel access code of the Bluetooth frame, inform of the poor channel quality.

An efficient alternative is also to implement new functionality into the system. For example, BER testing with dedicated frame types that are designed for testing the link quality are a suitable solution [9]. Also, active scanning of the radio environment in order to detect other networks is an efficient alternative for interference management. The controller station can thus dynamically adapt to the current spectrum characteristics.

2. *Analysing the type of interface found* - The interface might cover the whole frequency band used or just one or few sub-channels. The duration of the interference is also an important parameter, as short temporal fading occur in the radio environment.

3. *Performing a corrective action* - The corrective action depends of the type of the interface found. For FHSS systems, the controlling station might choose to replace some used frequency sub-bands or if errors occur in a large number of used hopping frequencies, the phase of the hopping pattern. However, the required minimum amount of hop frequencies has to be maintained. The replacing of some frequency sub-bands is an effective alternative against static narrow-band interferes. But, as the spreading of transmissions decrease, the probability of interference from other Bluetooth or FHSS WLAN system increases.

For DSSS systems, the changing of the channel can be a suitable solution. However, there is little room in the ISM frequency band and finding of a free channel might be difficult. Still, partly overlapping frequencies in DSSS networks usually operate adequately. Also, the higher speed 802.11b WLANs are capable of reducing the link speed from 11 Mbit/s to 5.5, and further to 2 and 1 Mbit/s in poor radio environment.

4. *Informing network devices* - The final action after the correction is to notify all the registered terminals of the changes in the radio medium. The information can be broadcast in the control fields of frame headers, or using dedicated management frames.

The IEEE 802.15 TG2 for Bluetooth/WLAN coexistence divides the interface management roughly into two groups. In *collaborative mechanisms* there is communication between the WLAN and Bluetooth (WPAN). Thus the two networks are able to negotiate and agree on how to share the bandwidth in a fair way. In *non-collaborative mechanisms* there is no communication between the networks. Therefore the aim of these methods is to search for techniques that minimise the interference caused from a network to another. The functionality presented above belongs to the latter category [14].

VIII. DISCUSSION AND CONCLUSIONS

WLAN and Bluetooth wireless systems operate with the presence of each other, with similar data networks, with other communication devices, and with other non-communicational radio devices. In addition, the properties of the frequency-selective fading wireless medium vary all the time.

The WLAN/Bluetooth interoperability was studied by the means of mathematical modelling, transmission spectrum measurements, and by practical transfer error measurements. The results indicate that clear error probabilities result from coexisting networks. The interference is highly dependent on distances between transmitting devices and on the loading of the networks. A distance increase of few meters is in many cases enough to reduce the interference below a level that has any of practical significance to the end user.

Furthermore, the network loading can be quite low. For example, a user could exchange 20 email messages (average 15 kbytes per message) during a workday. The total amount of data to be transmitted is 300 kbytes, which takes about 13.5 seconds to transfer using the DH1 Bluetooth packets. Consequently, the total time of functional coexistence will be quite short. On the other hand, in future there will be more demanding application, such as constant bit rate audio/video and the transfer of large files. In these cases, the transfer delay can grow rapidly because of constant errors and the users of an audio application experience reduced voice quality. Also, the density of Bluetooth applications in home or office environment can also grow very high, as Bluetooth modules are targeted to be embedded into wide range of systems benefiting from wireless data communications.

Spread spectrum is the solution for averaging the effect of disturbance among all the devices. However, as the unlicensed frequencies become more crowded, also other types of methods have to be implemented. These methods refer to functions for active interference management.

REFERENCES

[1] Specification of the Bluetooth System, v1.1, February 22, 2001, available at http://www.bluetooth.com/

[2] ISO/IEC 8802-11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 1999.

[3] B. Bing, *High-Speed Wireless ATM and LANSs*, Artech House Publishers, London, 2000.

[4] J. Zyren, "Reliability of IEEE 802.11 Hi Rate DSSS WLANs in a High Density Bluetooth Environment", technical paper of Harris, 1999, available in [14].

[5] G. Ennis, "Impact of Bluetooth on 802.11 Direct Sequence", IEEE doc. P802.11-98/319, available in [14].

[6] Homepage of Sigma Comtec, http://www.comtec.sigma.se/, April 5, 2001.

[7] F.J. Bauchot, F. Lanne, "IBM Wireless RF LAN design and architecture", IBM System Journal Vol. 34, No. 3, 1995.

[8] T. Sizer, "Bluetooth SIG Coexistence Working Group Liaison Report", IEEE doc. P802.15-00/387r0, 2000, available in [14].

[9] H. Gan, B. Treister, "Adaptive Frequency Hopping Implementation Proposals for IEEE 802.15.1/2 WPAN", IEEE doc. 802.15-00/367r0, 2000, available in [14].

[10] Documentation for the Ericsson Bluetooth Development Kit (EBDV), issue 2.5, 2000.

[11] Homepage of Nokia WLAN, http://www.nokia.com/corporate/wlan/, April 4, 2001.

[12] Homepage of HomeRF, http://www.homerf.org/, April 2, 2001.

[13] Homepage of BeezeCom, http://www.breezecom.com/TechSupport/, January 3, 2001.

[14] Homepage of IEEE 802.15 working group, http://grouper.ieee.org/groups/802/15/, April 2, 2001.

[15] Homepage of the IEEE 802.11 working group, http://grouper.ieee.org/groups/802/11/, April 2, 2001.

[16] Homepage of the Bluetooth SIG, http://www.bluetooth.com, April 3, 2001.

[17] R. LaMaire, et.al., "Wireless LANs and Mobile Networking: Standards and Future Directions", IEEE Communications Mag., Aug. 1996, pp. 86-94.

Publication [P3]

Hännikäinen M., Niemi M., Hämäläinen T., “Performance of the Ad-hoc IEEE 802.11b Wireless LAN”, International Conference on Telecommunications (ICT 2002), Volume 1, pp. 938-945, Beijing, China, June 23-26, 2002.

Reprinted from the proceedings of ICT 2002.

Performance of the Ad-hoc IEEE 802.11b Wireless LAN

Marko Hännikäinen[1], Markku Niemi[2], Timo Hämäläinen[1]

[1] Tampere University of Technology
Institute of Digital and Computer Systems
Korkeakoulunkatu 1, FIN-33720 Tampere, Finland
Tel. +358 3 3115 2111, Fax +358 3 3115 3095
marko.hannikainen@tut.fi

[2] Nokia Mobile Phones
Sinitaival 5, FIN-33721 Tampere, Finland
Tel. +358 50 511 7320, Fax +358 10 505 7606
markku.t.niemi@nokia.com

Abstract

The performance of the IEEE 802.11b Wireless LAN (WLAN) is evaluated under Distributed Coordination Function (DCF) ad-hoc topology. A number of stations with different types of applications are used in the tests. The evaluated application types are maximum rate data transfer, H.263 video streaming, and Voice over IP (VoIP). UDP/IP protocols are exploited on transport and network layers. The performance results are acquired by simulations using a Specification and Description Language (SDL) simulation model and by practical testing. The results indicate that the ad-hoc WLAN divides the available bandwidth nearly equally with contenting stations. The average achieved throughput remains predictable as the number of active stations increases. For application requiring constant bit rate, such as VoIP, the transfer delay variance becomes a challenge when aggregate traffic load is increased on WLAN. For enabling these demanding applications, modified ad-hoc contention algorithms for differentiating Quality of Service (QoS) according to the traffic type seem potential.

1. Introduction

Current Wireless LANs (WLAN) provide broadband data transfer capacity without cable restrictions. The technology has matured due to the stabilising standardisation and available products. Current products mostly meet the IEEE 802.11 standard for WLANs [1], providing up to 11 Mbit/s radio link speed by the 802.11b radio standard [5].

IEEE WLAN has been targeted as a wireless replacement or extension for a traditional IEEE 802.3 (Ethernet). Wired Ethernet has advanced since the basic 10 Mbit/s shared bus technology. The link capacities have increased to current 100 Mbit/s and gigabit products are now emerging. Also, at the Medium Access Control (MAC) protocol layer, the current wired Ethernet utilises powerful switches that separate the end stations from each other providing a collision and contention free medium access. In addition, modern switches implement Quality of Service (QoS) based network frame processing based on different user data priorities [7].

The design of IEEE WLAN follows the distributed and stable behaviour of the shared bus Ethernet, leading to reliable operation in varying wireless medium characteristics, network topologies with increasing number of stations, and different traffic loadings. This best effort service type is adequate for the non-realtime data services of traditional computer data networks, for example in office environments.

Multimedia services are emerging in a variety of application areas: education, work, and entertainment. In addition to proprietary applications, Internet as a source of content and as a medium for delivering multimedia services is driving the use of personal data communications among mass users [6]. Seamless merging of the Internet as a source of multimedia content with wireless networking is the next step to be taken.

The combinations of several media components of multimedia tend to increase the bandwidth requirement and the burstiness of data transfer. Typically, low error rates are also required, depending on the application. In addition, conversational services, such as Voice over IP (VoIP) and H.263 video conferencing, call for a short transfer delay and predictable delay variance to fulfil their realtime requirements. Furthermore, different media components must remain synchronised in respect of time. WLAN is projected to become a complementary broadband access mechanism for third generation cellular data network services. In order to enable this, WLAN must develop to manage increasing and highly varying number of users with different QoS requirements.

This paper evaluates performance of the IEEE 802.11b WLAN and its applicability of for multimedia applications. The evaluated applications are maximum rate data transfer, H.263 video streaming, and VoIP. User Datagram Protocol and Internet Protocol (UDP/IP) are exploited as transport and network layer protocols. The performance evaluation is first carried out by numerical calculations and simulations. Second, practical measurements with a commercial WLAN card with increasing number of stations are carried out. The paper is organised in the following way. In Section 2, the IEEE 802.11 WLAN technology is introduced, concentrating on the MAC layer and access in ad-hoc topology. Section 3 evaluates the WLAN performance by calculations and simulations. The practical performance measurements and results are presented in Section 4. Conclusions are given in the final section.

2. IEEE 802.11 WLAN

The IEEE 802.11 for wireless LANs is currently the most successful WLAN standard. The work began in 1990 when IEEE initiated the 802.11 working group to

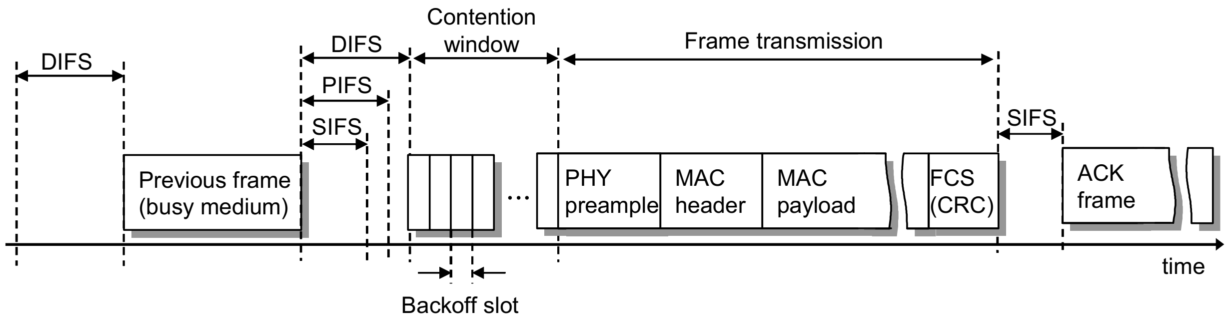


Figure 1. Basic MAC operation of the IEEE 802.11 WLAN.

develop a WLAN standard for the 2.4 GHz ISM band [4]. The first IEEE 802.11 standard was approved in 1997. First products that appeared before the approval were based on draft standard versions.

IEEE 802.11 specifies different physical layers and a MAC protocol for the data link layer. Consequently, the same MAC protocol is used on all physical layer alternatives. Currently, the IEEE 802.11 family of standards includes specifications for a number of physical layers; one is a diffuse infrared technology, while the rest are radio technologies. Three current and one emerging radio PHY standards utilise the 2.4 GHz frequency band for Industrial, Scientific, and Medical (ISM) devices. One physical layer standard is currently available for the 5 GHz frequency band (for ISM/HIPERLAN devices).

The original 802.11 and the current 802.11b standards specify 2.4 GHz radios that utilise spread spectrum techniques: Frequency Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS). 802.11 data rates specified for the physical layers are 1 Mbit/s and 2 Mbit/s, thus being lower than in legacy fixed LANs. In low network traffic load conditions, the 2 Mbit/s link capacity is enough to accommodate legacy LAN type of traffic characterised by data bursts with no real-time requirements.

As the applications require higher data rates the IEEE 802.11 standard has been updated for higher link speeds. The IEEE 802.11b increases the data rates with 5.5 and 11 Mbit/s physical data rates [5]. The used modulation is Complementary Code Keying (CCK) that is based on the DSSS technology. The multirate control of 802.11 MAC protocol allows a terminal to switch from the 11 Mbit/s link speed down to 5.5 and further to 2 and 1 Mbit/s if the radio channel quality cannot accommodate higher data rates.

The 802.11 work is advancing by a new 2.4 GHz physical layer. The 802.11g is extending the capacity of the system to reach 22 Mbit/s using Packet Binary Convolutional Code (PBCC) modulation, and further to 54 Mbit/s by applying Orthogonal Frequency Division Multiplexing (OFDM) modulation [1]. The 22 Mbit/s PBCC system will backwards be compatible to the 11 Mbit/s CCK modulation of 802.11b. PBCC is expected to become an optional modulation scheme for the new version of the standard.

IEEE 802.11a is the specification for the 5 GHz band radio. The 802.11a standard is based on Orthogonal Frequency Division Multiplexing (OFDM) data modulation. The physical layer achieves link rates from 6 Mbit/s up to 54 Mbit/s [4].

2.1 MAC Protocol

The IEEE 802.11 network architecture supports both an infrastructure-based topology called Point Coordination Function (PCF) and ad-hoc topology with independent peer-to-peer connections known as *Distributed Coordination Function (DCF)*. This research concentrates on the ad-hoc topology with DCF. However, PCF and DCF are not independent, as PCF operates on top of DCF, and a WLAN utilises both control types also in the infrastructure based topology.

In the IEEE 802.11 standard, the priority of medium access is selected according to the frame type (not directly according to the traffic type). The fundamental channel access is called Carrier Sensing Multiple Access/Collision Avoidance (CSMA/CA). The access is based on the function for sensing ongoing transmissions on the channel (carrier sensing or energy detection). With carrier sensing, different *Inter-Frame Spaces (IFS)* and a random *Backoff* time is utilised. A higher priority frame can be sent earlier by exploiting a shorter IFS, when a medium is sensed idle during the channel access contention.

802.11 contains three basic IFSs, where the highest priority (shortest space) is reserved for short control frames (such as acknowledgements) and for the following segments of a same fragmented MAC Service Unit (MSDU). The lowest priority is for ad-hoc topology under DCF, where all competing stations utilise this access priority for data frame transfer. Actually, there exists also a longer IFS called Extended IFS (EIFS). EIFS is used for recovering from possible abnormal network situations detected by an unsuccessful data transmission.

The basic access cycle under DCF is presented in Figure 1. When a WLAN *Station (STA)* wishes to access the medium for transmitting a single frame, it first waits for the duration of *DCF IFS (DIFS)* after the medium has been released by the previous transmission. DIFS is then followed by a *Contention Window (CW)* for resolving a winning STA. The purpose of CW is to prevent collisions when several STAs start to transmit simultaneously and to provide a fair access among all competing STAs.

The actual support for user data originated access parameters (time-bounded services) is implemented only in the PCF topology. An Access Point (AP) operating as a point coordinator uses the PCF IFS (PIFS) to seize the control of the channel. AP can then generate contention-free periods by frequently polling other terminals. AP can always acquire the channel control by the higher priority compared to the distributed control. The channel is reserved to the polled party for the transmission of a single frame.

Following the transmission of a data or management frame, the receiving STA acknowledges an error free reception immediately by and acknowledgement (ACK) frame. ACK is transmitted using the *Short IFS (SIFS)*. Thus the medium access is acquired with the highest priority.

In addition to the basic access, IEEE 802.11 specifies two complementing access functions for reaching an error-free frame transfer. Virtual carrier sensing is exploited by broadcasting the required time for the current transfer attempt in frame headers. All other STAs receiving this header information prevent from channel access for that time. Contention free polling based data transfer sequences are fully protected by virtual sensing.

Also, terminals may change short Request-To-Send (RTS)/Clear-To-Send (CTS) control frames broadcasting the time information before the transmission of a longer data frame. These frames contain the timing information for virtual carrier sensing over the following data frame transfer [1].

2.2 Backoff algorithm

The medium access between competing terminals under DCF after the DIFS period has passed is based on CW that is a random backoff time. The backoff time consists of an integer number of short *backoff slots*, which duration is fixed, but specific separately for each physical layer. The selection of backoff time is dynamic, as the average value changes according to network traffic load. The backoff time is selected using

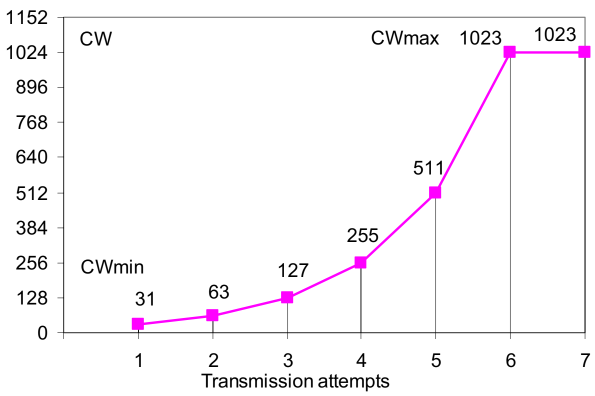


Figure 2. CW selection algorithm of the IEEE 802.11b.

Table 1. Timing of the IEEE 802.11b access cycle.

Field	Bytes	Duration /µs
DIFS		50
SlotTime		20
BackoffTime (variable)		-
PHY preample (normal, 1 Mbit/s)		192
MAC header	30	21.8
MAC payload (Ethernet frame)	1500	1090.1
MAC FCS	4	2.9
SIFS		10
PHY preample		192
MAC header	10	7.3
MAC payload (ACK frame)	0	0
MAC FCS	4	2.9
Total without Backoff time		1569.8
Utilisation (payload time vs. total time, no backoff overhead)		69.5 %

the function:

$$BackoffTime = INT[Random(0,1) * CW] * SlotTime \quad (1)$$

Thus, the number of backoff slots is selected using an equal distribution between 0 and the *CW* number of backoff slots. As the contention proceeds, the selected backoff time is decreased after each idle sensed backoff slot. When the counter reaches zero, the transmission can begin. Also, in case the STA loses the contention by sensing channel activity, which is due to transmission of another STA having a smaller backoff time, the value of the backoff time is frozen and the countdown continues during the next cycle [13].

CW takes values between *CWmin* and *CWmax*. CW is increased exponentially by a STA from the initial CWmin after each unsuccessful retransmission attempt, until the maximum CWmax value is reached. After that point, the value stays in CWmax. The increasing of CW in the IEEE 802.11b radio layer is illustrated Figure 2.

Continuing retransmission attempts, however, are limited by two variables: *short retry limit* for frames not utilising RTS/CTS procedure and *long retry limit* for frames sent with the RTS/CTS procedure. When the retransmission limit is reached, the CW is reset back to CWmin and the data frame is discarded.

3. Theoretical Performance Evaluation

The theoretical performance of the IEEE 802.11 ad-hoc WLAN is first estimated by numerically calculating the utilisation efficiency of the MAC algorithm. The structure of a data frame is presented in Figure 1 and the duration of the different fields of the 11 Mbit/s 802.11b radio in Table 1 [5]. The maximum network utilisation with 1500 B MSDU (MAC payload) without including the backoff time is 69.5%. This corresponds to the total data throughput of 7.6 Mbit/s. In a point to point data transfer between two STAs, assuming an error free radio environment (CW equals CWmin, i.e. 31 slots) the average size of backoff is 15 backoff slots. This results into utilisation of 58.3% and throughput of 6.4

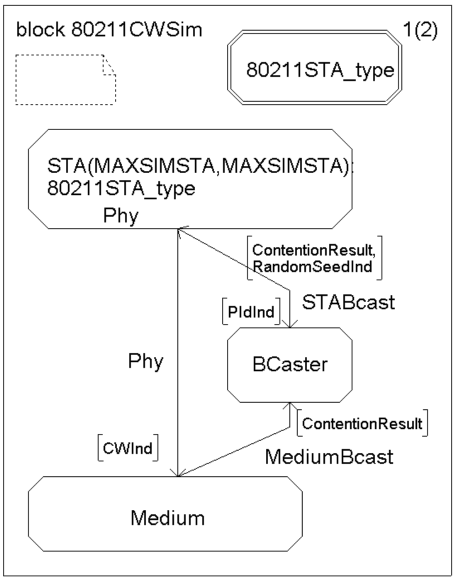


Figure 5. SDL simulation model for the backoff algorithm.

Mbit/s. The maximum backoff time consisting of CWmax number of slots leads to the throughput of only 0.5 Mbit/s. The effect of contention window size on the throughput is thus significant.

3.1 Throughput Simulation

The performance of the 802.11 CSMA/CA backoff algorithm in an ad-hoc network was next evaluated by a simulations. A simulation system was constructed using the Specification and Description Language (SDL) [2]. The architecture of the implemented SDL model (the 802.11CWSim block) is shown in Figure 5. SDL is a high abstraction layer specification language that is efficient for specifying real-time control systems, such as communication protocols [2]. The language provides a clear graphical interface that facilitates the design of complex systems. Also, a formal SDL model can be automatically converted into an executable application code, which was exploited for generating an executable simulation for the WLAN performance.

The built SDL model contains a *STA* process that implements the CW algorithm of a 802.11 STA, a *Medium* process that selects the winning STA among contenting STAs and detects collisions, and a *Bcast* process that emulates the broadcast medium and carrier sensing. The STA process is dynamic. Thus, any number (defined by the constant MAXSIMSTA) of WLAN STAs can be created in a simulation. An additional logging process has been implanted for the SDL model. The logging process collects the selected backoff time for each STA, occurred collisions, and successful transmissions. The process writes the logged information into a text file that is later processed by spread sheet programs.

In the SDL model, the radio medium was assumed to be ideal. Therefore, no transmission errors due to the unreliable radio path were simulated. However, retransmissions occur after collisions.

The SDL simulation results for the total network throughput and for the throughput of a randomly selected single STA (marked as *STA A*) in an ad-hoc network up to 128 STAs are presented in Figure 3. The simulation lasted for the duration of 1000 access cycles, i.e. the contending for the transmission of a 1500 B payload MAC frame was done as many times. All STAs were competing for the access on all cycles Therefore, the maximum network capacity was searched.

The results indicate that the basic DCF access algorithm remains stable also with a high number of active stations, as the total network throughput decreases about 3 Mbit/s while the number STAs increases to 128. The STA A that is a single simulated WLAN STA loses throughput as shown in the lower curve of Figure 3.

A slightly modified backoff algorithm is simulated for referential results in Figure 4. In this case all STAs detect a collision (unsuccessful attempt to transfer) of a single STA and perform the increasing of their CW. Consequently, the stability of the network improves as seen from the increased total network throughput with high number of STAs.

Table 2 lists the simulated average and maximum access delays, and the mean deviation of the delay of a single contending STA. As can be seen, the mean delay and delay variation increases rapidly with the number of parallel STAs. This is further illustrated in traffic profiles of Figure 9. Each figure presents the access of a

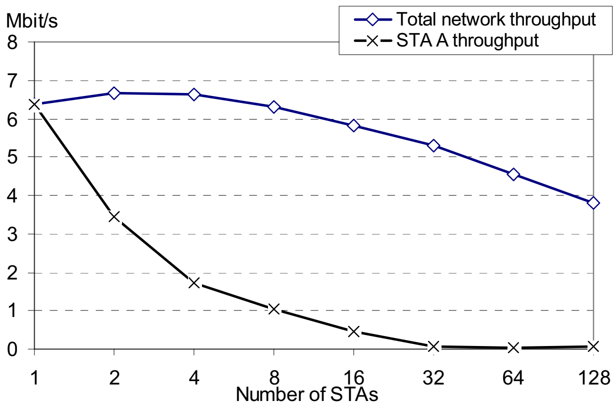


Figure 3. Network throughput simulation.

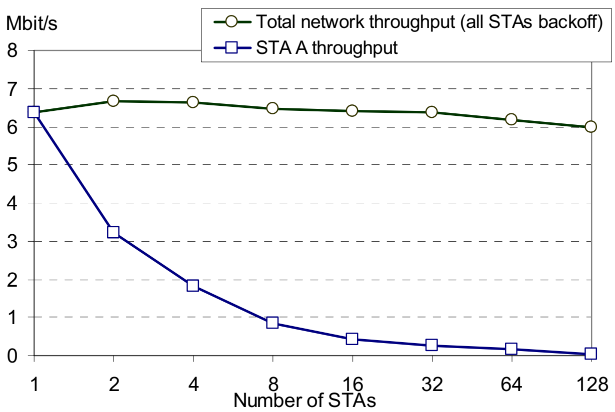


Figure 4. Network throughput simulation (non-standard approach when all STAs backoff).

single STA. The height of a bar gives the access delay for a won contention, and a gap between presents a lost contention.

Table 2. Access delays for a single STA (STA A).

Number of STAs	Access delay /ms Average	Max	Mean deviation
1	1.88	2.19	0.16
2	3.69	18.66	1.50
4	7.13	49.69	4.05
8	11.83	60.03	7.98
16	57.89	341.42	55.03
32	89.17	681.09	100.43
64	725.78	1368.31	642.53

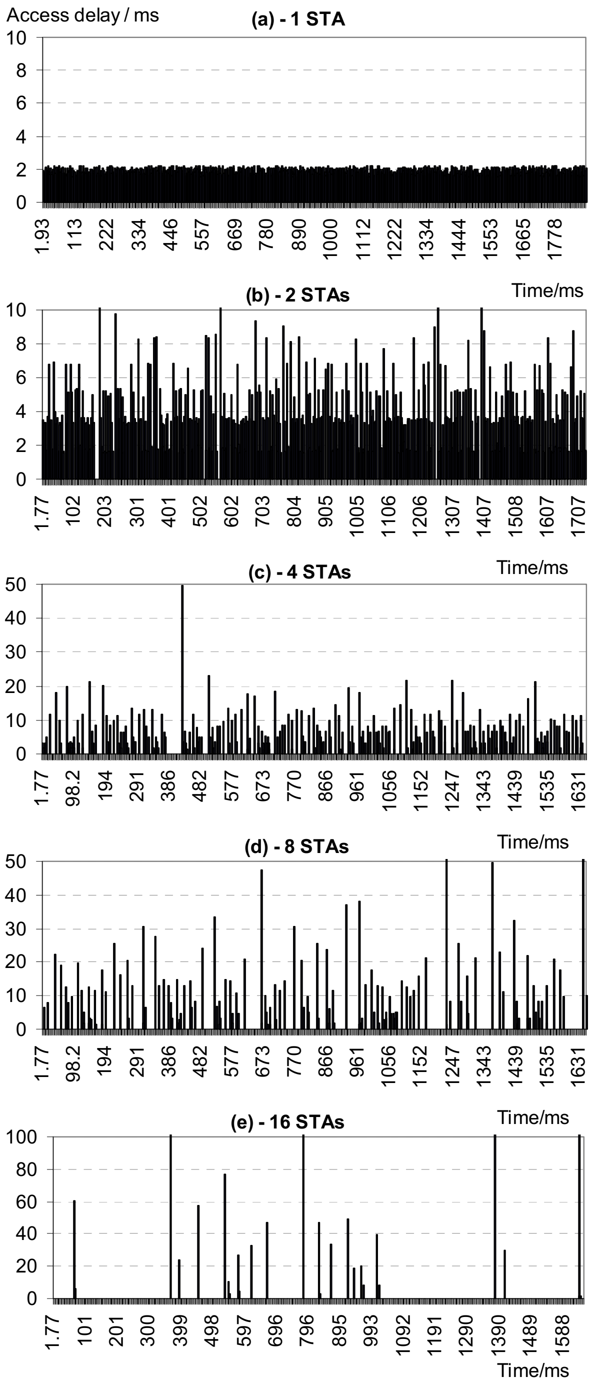


Figure 9. Channel access delays for a single STA with increasing number of contenting STAs.

The length of an average access cycle shortens as the number of contenting STAs increases. This is due to two factors. First, with a high number of STAs a low value for backoff time is more likely to be selected. Second, an acknowledgement is not returned after an unsuccessful (colliding) transmission attempt. Thus, the average length of an access cycle decreases by a high count of collisions. A missing acknowledgement is concluded as negative acknowledgement by the transmitting STA. When no acknowledgement frame emerges after the DIFS period following a previous transmission, the channel is can be accessed under regular DCF. The simulated length of an average winning backoff time and the percentage of occurred collisions are presented in Figure 6. Also, the simulated

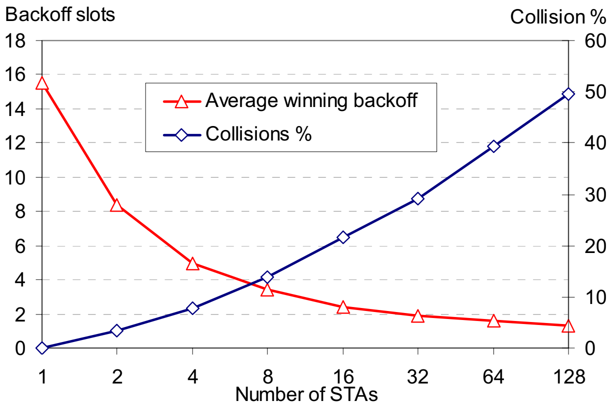


Figure 6. Average winning backoff and collision percentage.

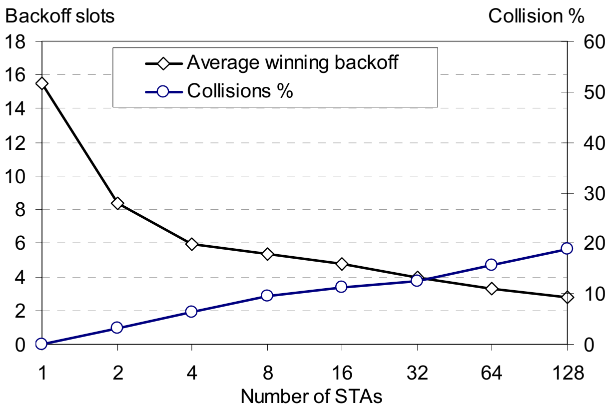


Figure 7. Average winning backoff and collision percentage (non-standard approach when all STAs backoff).

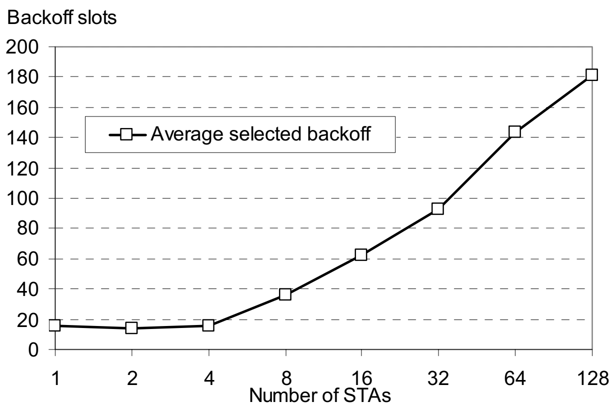


Figure 8. Average selected backoff of all STAs.

average backoff of all STAs is plotted in Figure 8. Despite of the rapidly rising length of average selected backoff, the probability that one STA has selected a very small backoff remains high, as can be seen from Figure 6.

In Figure 7 presents the simulated results for the non-standard approach, where all STAs perform backoff. The average length of the winning backoff period is near the standard simulation in Figure 6. However, the portion of collisions remains significantly lower. The drawback of this better channel utilisation performance is increased access delay.

4. Performance Measurements

The practical performance measurements of 802.11b have been carried out using commercial WLAN adapters. The used products were Nokia C110 PCMCIA cards [3]. The card was configured with default ad-hoc network parameters (independent WLAN) expect that the short retry limit for retransmissions was configured to value 1 using the card's management software. Thus, no retransmission attempts are made by the MAC protocol. This simplifies the testing and gives more practical results from the DCF operability. Also, the results are more comparative to the SDL simulations. The RTC/CTS procedure was not used for transfer.

For the performance measurements, altogether 14 PC computers were used, equipped with Windows NT and Windows 2000 operating systems. The PCs were desktop computers with PCMCIA adapters for WLAN cards, and laptop computers. All test PCs had a PII or higher processor platform. The computers were placed in close proximity to each other. Therefore, the STAs were expected to be equal during the channel access contention. The topology for the measurements is depicted in Figure 10.

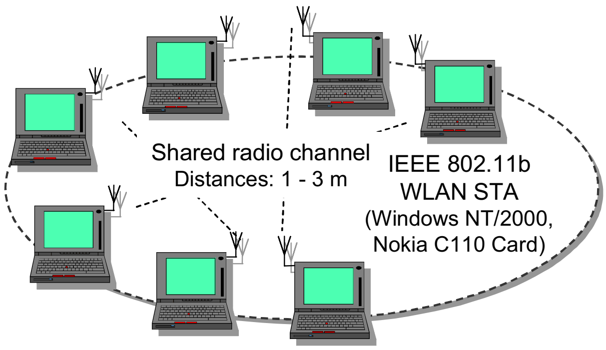


Figure 10. Ad-hoc topology for measurements.

4.1 Maximum Data Transfer

A data transfer testing application called Traffic GEnerator (TGE) has been implemented for throughput testing. Asymmetric data streams can exist between the TGE processes running in sender and receiver PCs. TGE operates over UDP/IP, which makes the throughput measurements straightforward and transparent, as no transport layer flow control and retransmissions are exploited (as performed by Transmission Control Protocol, TCP). TGE allows a user to select the transmitted packet size and the interval between separate transmitted packets. In addition, the tool can be set to generate a packet burst and a variable bandwidth packet stream.

TGE has its own headers containing a time stamp, running packet number, traffic type information, and a footer CRC field that verifies the correctness of the padded payload. By using these fields, TGE measures the network throughput between the sender and receiver PCs, missing packets, and transfer delays. For delay measurements, the program performs a synchronisation procedure for determining the clock offset between transmitter and receiver computers.

As the WLAN throughput varies constantly, the measured values are averaged over time (about 15s). Also, the measurements were repeated three times with changed STA locations and transmitting and receiving PC pairs, and the results again averaged. The STAs were configured to generate 1500 B MAC payload (MSDU) frames, corresponding to the theoretical SDL simulations. The generation rate for packets were set to maximum by configuring the packet interval to 1 ms. This is significantly more than the network is able to transfer and the excess frames overflow in transmission buffers. However, the received stream bandwidth for each transmitting STA was accurately measured, giving the actual realised throughput of the network. For balancing the processing load, each STA was receiving and transmitting at maximum only a single traffic stream.

The measurement results are presented in Figure 11 that again gives the total network throughput with increasing number of contenting STAs and the achieved throughput of one station. As can be seen, the throughput decreases more rapidly than with the theoretical modelling on ideal channel. However, the stability of the MAC protocol remains within the tested size topology. The maximum total network throughput is not reached until 4 STAs are transmitting. This effect is expected to result from PC/WLAN card performance. With 4 STAs, the performance of the network is very close to the theoretical limit.

4.2 VoIP

VoIP traffic has a stable traffic profile, as constant length packets are send on fixed intervals. The packet size and interval depends on the type of voice codec and buffering. The basic G.711 PCM codec generates 240 B audio payload with 30 ms packet intervals. VoIP utilises Real Time Protocol (RTP) over UDP/IP, which generates additional overhead for WLAN payloads. Altogether, 280 B MSDUs are transferred, requiring about 75 kbit/s throughput. However, for VoIP applications, the transfer delay and delay variance are more stringent requirements [8].

For the WLAN performance tests, a VoIP application was estimated by using the *ping.exe* tool of Windows 2000 for measuring the roundtrip times of VoIP size packets. The performance of a single VoIP connection was estimated by pinging, while the number of parallel contending STAs, transmitting 1500 B MSDU packets with maximum rate, was increased.

The ping measurement results are presented in Figure 12. The curves present the percentage of average roundtrip times below the thresholds of 20 ms, 100 ms, 500 ms, and 1000 ms. As can be seen, the 20 ms threshold average (10 ms per direction as symmetric link is estimated) can be reached with only 4 parallel transmitting STAs. 100 ms threshold, which is still usable for VoIP applications, is reached with up to 5 disturbing STAs.

4.3 Video Stream

Video streaming application was tested with an actual H.263 demonstrator system. A *wireless video demonstrator* has been implemented at TUT [11]. The system has a dedicated Video Control Protocol (VCP) that controls the video packet transfer and can accurately measure the throughput or a video stream, lost packets, delay variance, and also protect the video against transmission errors [12]. The error protection was not utilised with the WLAN measurements. Similar to TGE, VCP operates over UDP/IP and contains its own header and footer structure.

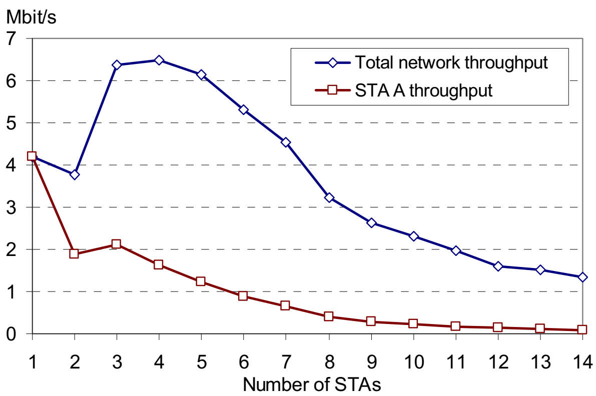


Figure 11. Measured network throughput.

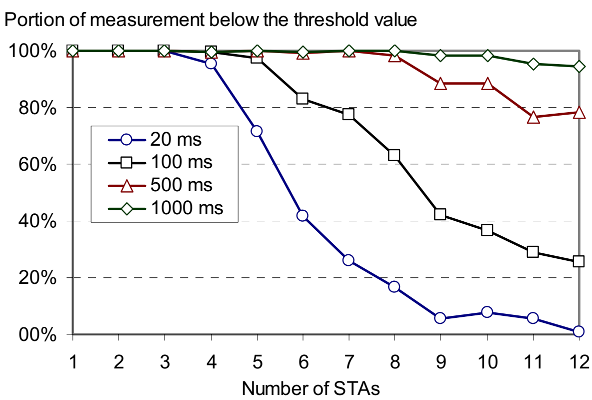


Figure 12. VoIP ping delay measurements.

For the video measurements, a 1Mbit/s bit rate H.263 (v.1) video stream was encoded. The bit rate was controlled by a rate control, which minimised throughput variation. The stream was CIF size (352 x 288) and the frame rate was 25 frames per second (f/s). This resulted into approximate 5000 B video packets, with additional UDP/IP headers. This application layer packet is fragmented by the IP layer into 1500 MSDU frames. This results to four WLAN MAC frames that are sent as fast as possible after IP layer has received the video packet. As with ping measurements, the number of parallel STAs transmitting 1500 B MSDU packets with maximum rate was increased parallel to a single video stream. The measurements results are presented in Figure 13 and Figure 14. Figure 13 gives the realised throughput for the 1 Mbit/s video stream and the portion of missing video packets. In Figure 14, the average and maximum video packet transfer delays and the mean deviation are presented.

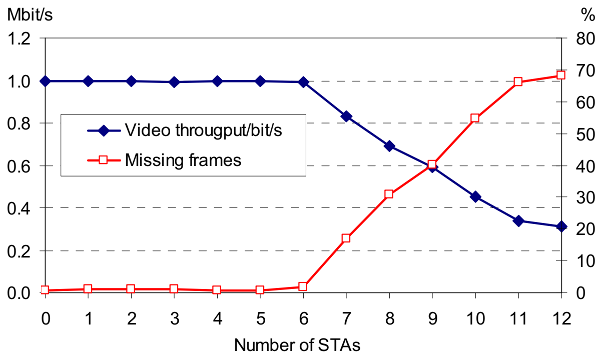


Figure 13. Single video stream measurements.

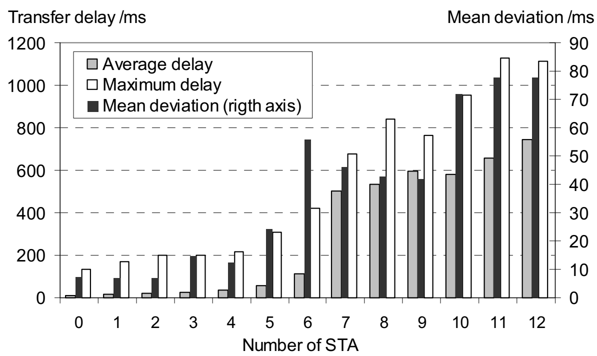


Figure 14. Video packet transfer delay.

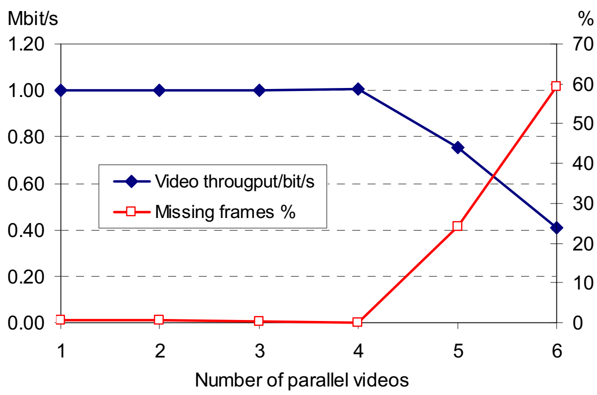


Figure 15. Parallel video streams measurements.

Another similar video test series, in which the number of parallel video streams was increased on the WLAN, was performed. In this case no disturbing data transmissions existed. These results for realised throughput and experienced packet loss are shown in Figure 15.

Overall, the video stream transmission with constant bit rate seems to survive well while there is enough capacity available. The rate of missing video frames remains low until 6 parallel data STAs or altogether 4 parallel videos, which results into usable video quality. Also the video transfer delay parameters remain manageable below this STA number threshold. After this point, both the delay and the average delay and frame loss increase and make the video quality poor. The missing frames in most cases results from the buffer overflow at the transmitting STA, due to low WLAN throughput. The frame error rate on the wireless channel is low.

5. Conclusions

An analysis of the ad-hoc (distributed) topology IEEE 802.11b WLAN was performed by SDL simulations and by practical testing. Overall, the ad-hoc WLAN divides the available bandwidth nearly equally between contenting STAs. The average achieved throughput remains predictable as the number of active STAs in a network increases. This can be seen from both simulated and measured results. For applications requiring time bounded transfer, the access delay and delay variance growing rapidly with the increasing aggregate traffic load becomes a challenge. This is especially shown with the VoIP ping tests. For video streams, the stability of a stream with increasing number of parallel data STAs or parallel video streams was measured. Rather than decreasing gradually, the video quality drops rapidly only when the total network capacity is increased.

The effect of a modified backoff algorithm, where all STAs took a cautious approach for channel access, illustrated how the stability of network can be improved in high traffic loads. The utilising of configurable backoff algorithms for differentiating QoS for WLAN data transfer seems potential. The target would be to maintain the statistical stability for preventing collisions but giving a higher probability for channel access to a higher priority frame. For the IEEE 802.11 standard WLAN, these types of enhancements for the MAC protocol QoS are being developed in the IEEE 802.11e group.

References

[1] ISO/IEC 8802-11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 1999.

[2] The user manual for Telelogic Tau SDL Suite, version 4.1, Telelogic AB, Sweden, 2001.

[3] Homepage of Nokia WLAN, http://www.nokia.com/corporate/wlan/, Jan. 4, 2002.

[4] Homepage of the IEEE 802.11 working group, http://grouper.ieee.org/groups/802/11/, April 2, 2001.

[5] IEEE Std 802.11b-1999, (Supplement to ANSI/IEEE Std 802.11, 1999 Edition), Higher-Speed Physical Layer Extension in the 2.4 GHz Band, September 1999.

[6] M. Hännikäinen, T. Hämäläinen, M. Niemi, J. Saarinen, "Trends in Personal Wireless Data Communications", Computer Communications, Elsevier Science, Volume 25, Issue 1, 1 January 2002, pp. 84-99.

[7] IEEE P802.1Q D11, IEEE Standard for local and metropolitan area networks: Virtual Bridged Local Area Networks, IEEE Standards Department, June 30, 1998.

[8] J. Feigin, K. Pahlavan, M. Ylianttila ”Hardware-Fitted Modelling and Simulation of VoIP over Wireless LAN”, IEEE International Vehicular Technology Conference (VTC’2000), Boston, MA, USA, September 24-28, 2000, vol.3, pp. 1431-1438.

[9] M. G. Arranz, R. Agüero, L. Muñoz and P. Mähönen, “Behavior of UDP-Based Applications over IEEE 802.11 Wireless Networks”, IEEE International Symposium on Personal Indoor and Mobile Radio Communication, PIMRC 2001, San Diego, September 2001, pp. F-72-77.

[10] A. Kamerman, G. Aben, “Net throughput with IEEE 802.11 wireless LANs”, Wireless Communication and Networking Conference (WCNC'2000), Chicago, September 2000, pp. 747-752.

[11] M. Hännikäinen, et.al. “Architecture of a Wireless Video Transfer Demonstrator”, International Zurich Seminar on Broadband Communications (IZS’2002), Zurich, Switzerland, February 19-21, 2002, pp. 53-1 - 53-6.

[12] J. Suhonen, M. Hännikäinen, O. Lehtoranta, M. Kuorilehto, M. Niemi, T. Hämäläinen, “Video Transfer Control Protocol for a Wireless Video Demonstrator", International Conference on Information Technology, Las Vegas, USA, April 8-10, 2002, pp. 462-467.

[13] M. Natkaniec, A. R. Pach, “An Analysis of the Backoff Mechanism Used in IEEE 802.11 Networks”, Symposium on Computers and Communications, (ISCC’2000), Antibes, Juan les Pins, France, July 3-6, 2000, pp. 444 -449.

Publication [P4]

Hännikäinen M., Knuutila J., Letonsaari A., Hämäläinen T., Jokela J., Ala-Laurila J., Saarinen J., "TUTMAC: A Medium Access Control Protocol for a New Multimedia Wireless Local Area Network", IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC 1998), Volume 2, pp. 592-596, Boston, USA, September 8-11, 1998.

TUTMAC: A Medium Access Control Protocol for A New Multimedia Wireless Local Area Network

Marko Hännikäinen[1], Jarno Knuutila[2], Ari Letonsaari[1], Timo Hämäläinen[1], Jari Jokela[2], Juha Ala-Laurila[2] and Jukka Saarinen[1]

[1]Tampere University of Technology, Signal Processing Laboratory
Hermiankatu 12 C, FIN-33720 TAMPERE, Finland
Tel: +358-3-365 2111, Fax: +358-3-365 3095,
Email: markoh@cs.tut.fi, aril@cs.tut.fi, timoh@cs.tut.fi, jukkas@cs.tut.fi

[2]Nokia Mobile Phones, Wireless Data
Sinitaival 5, FIN-33720 TAMPERE, Finland
Tel: +358-10-505 6800, Fax: +358-10-505 6888, Email: jarno.knuutila@nmp.nokia.com, jari.jokela@nmp.nokia.com,juha.ala-laurila@nmp.nokia.com

ABSTRACT

This paper presents a Medium Access Control (MAC) protocol called TUTMAC for a new wireless local area network (TUTWLAN). The design objective has been to develop a simple, multimedia service capable protocol that provides sufficient medium utilisation efficiency and guarantees QoS (Quality of Service) parameters. The developed system utilises a centralised (base station controlled) network architecture. A limited number of portable stations can be associated with the same base station, i.e. in the same TUTWLAN cell. TUTMAC is connection oriented: the bandwidth is allocated deploying constant bit-rate TDMA based data channels that are reserved by exchanging short control messages. The connection parameters can be dynamically altered during the data exchange session. Currently, a TUTWLAN prototype is being developed comprising both TUTMAC software and platform hardware modules. The prototype will support up to eight simultaneous data-transfer connections each having 64 to 512 kbit/s data transmission bandwidth.

I. INTRODUCTION

Wireless local area network is a practical choice for accessing data services in limited geographical areas, such as home, office and public areas like airport lounges and libraries. In offices there is a common need for wireless access to various office equipment, such as printers and passage supervision systems. In residential environment benefits of wireless connections can be found in the area of surveillance and wireless remote control applications. In public areas one potential application will be wireless real-time information access to databases. WLAN technology is possibly the only solution when traditional wired networks are impossible or inconvenient, for example for temporary ad-hoc networks or for the networking of old buildings.

In the design of a wireless LAN a common question is how to provide a cost-effective system with adequate performance and support for mobility, multi-user, and QoS characteristics. Emerging WLAN standards such as IEEE 802.11 [1] and High Performance Radio Local Area Network (HIPERLAN) [2] are good examples of this kind of general approach. However, the universal purpose of the system often leads to too complex and too expensive design. Furthermore, the design can turn out to be too general to support required user specific service parameters, such as constant data-rate wireless links.

The TUTWLAN is a new wireless local area network being designed and implemented in Signal Processing Laboratory at Tampere University of Technology (TUT). In the first phase of the work a wireless Medium Access Control protocol (TUTMAC) has been specified and designed for a prototype system. The prototype hardware, illustrated in Figure 1, contains MAC processor module, radio module and interface modules for the radio and a host computer.

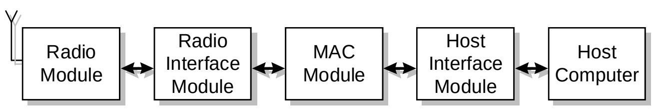


Figure 1. Prototype architecture.

This paper presents the essential design characteristics and functional features of the TUTWLAN system. The paper is organised as follows. First the TUTWLAN design requirements and network architecture are presented. Next the architecture of the TUTMAC protocol, as well as utilised channel access, quality of service and management schemes of the system are described. Finally, prototype characteristics and performance estimates are given before conclusions in the last section.

II. TUTWLAN REQUIREMENTS

TUTWLAN is targeted for an indoor, on-premises environment. The system utilises radio link as the physical medium. Target application areas are related to low-mobility or stationary wireless network terminals ranging from multimedia laptops to wireless sensors, thus introducing various transmission bandwidth and QoS needs. The requirements for the TUTWLAN system have been derived from the demands of these target environments and applications, while also concerning the characteristics of current radio technology. The key design objective has been to develop a simple system supporting wireless real-time, multimedia capable, data-services. The selected channel access mechanism does not achieve the medium allocation efficiency of a contention based protocol on low traffic loads, but is able to guarantee negotiated QoS parameters for reserved connections. On the other hand, the system should be significantly less complex and costly compared to wireless ATM access networks that must support a large amount of various traffic classes and connection parameters.

The general requirements for TUTWLAN are:

- The *low-cost* is targeted by simple implementation and the use of standard multipurpose modules and components. Modularity and reconfigurability in all stages of system design are key elements to fulfil this requirement.
- The *QoS requirements* for the data-transfer service of the TUTMAC protocol include support for user defined traffic types and connection parameters. The protocol must be capable for real-time data-transfer services.
- The TUTWLAN can be used both as an extension and as an alternative to a wired LAN. Therefore, for *interoperability requirements* the changing topology of a wireless network, insecurity and unreliability of the medium and protocol specific management functionality, such as power saving, must be hidden from the network user, e.g. from legacy TCP/IP applications [3].
- Wireless medium does not provide the same level of confidentiality and user identification as a wired counterpart. A wireless coverage area cannot be reliably defined nor restricted. Actions at the MAC-layer must be taken to provide a *secure data-transfer* service [3].
- An *unlicensed* and globally available *frequency band* must be selected for the system.
- The architecture of the TUTMAC protocol should follow a master-slave hierarchy as *the centralised control* and management enables an easy and efficient support QoS parameters and a natural access point for outside network resources [4].
- In order to guarantee the low-cost, efficient resource management and guaranteed QoS, the *number of simultaneous users* in a single TUTWLAN cell has been restricted according to the target environment.
- The requirement for *low power consumption* follows from the usage of battery powered portable network equipment, laptops in most cases. A wireless network adapter should not significantly shorten the operating time of a portable terminal. Therefore, the TUTMAC protocol should be capable of turning off the transceiver during idle periods without missing any relevant transmissions [1].

III. NETWORK ARCHITECTURE

The TUTWLAN consists of two types of separate network elements: *Portable Stations* (PS) and a *Base Station* (BS). The service coverage area of a BS creates a single radio cell. All PSs belonging to a same cell are under a common centralised BS management. The access to a shared radio medium is controlled by the exchange of short control messages. The network management and maintenance information is distributed similarly from the BS to PSs within the cell area.

A PS can appear in two separate operational modes according to the network topology. A PS that has registered its presence with a BS resides in an associated mode while an unregistered PS remains non-associated. The associated mode is further divided into an active and a non-active modes. A non-active PS is able to exchange control and management messages and only to receive broadcast data. However, the non-active PS has no data channel reserved as does an active mode station. The TUTWLAN architecture is illustrated in Figure 2.

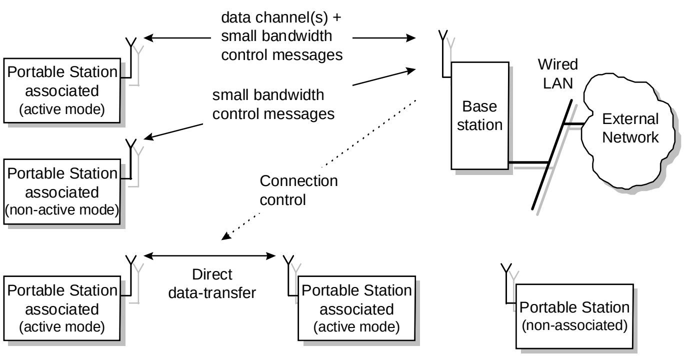


Figure 2. TUTWLAN Architecture.

IV. TUTMAC PROTOCOL ARCHITECTURE

The TUTMAC protocol introduces a functional module architecture, illustrated in Figure 3. The architecture contains interface modules for data-transfer service and management users, and for the access to physical layer. The data processing, as well as control and management functionality are located in separate functional entities.

TUTMAC protocol provides two types of data-transfer Service Access Points (SAP): network and native. The network SAP offers an access to a legacy network protocol (e.g. IP). The native SAP provides an extended service interface that may be utilised by custom network protocols or user applications capable of fully exploiting the TUTMAC specific QoS parameters within the cell service area.

The data processing block converts the user data into a more suitable form for the wireless medium. Encryption is performed for confidentiality while fragmentation and Forward Error Correction (FEC) coding functions are added for better protection of the data against transmission errors. The frame queuing and Automatic Repeat Request (ARQ) retransmissions are controlled according to assigned QoS.

The control and management functionality consists of state machines that adapt to the inputs from the management interface and data-processing modules while producing a proper output according to current state of the system. All the operational parameters are stored into a Management Information Base (MIB) that can be accessed and modified through the station management user interface.

Both the PS and the BS functionality are assembled using the same functional modules. A BS functionality is constructed by adding the base specific modules (data-processing and control and management) on top of a PS functionality. In this way, a certain set of BS functions can be included into a PS capable of executing them (e.g. a laptop). Thus an ad-hoc networking is enabled if no permanent BS service is available.

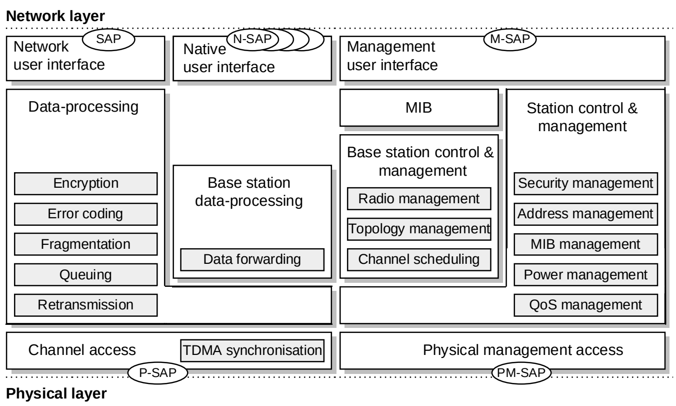


Figure 3. TUTMAC protocol architecture.

V. CHANNEL ACCESS

TUTMAC is a connection oriented wireless MAC protocol that utilises reservation based Time Division Multiple Access (TDMA) scheme with the shared medium. The medium access cycle, illustrated in Figure 4, is divided into time slots that form constant bit-rate channels. Four types of channels are distinguished by their purpose, direction and bandwidth: *data*, *contention*, *control* and *beacon* channels. The data channel includes also *acknowledgements*.

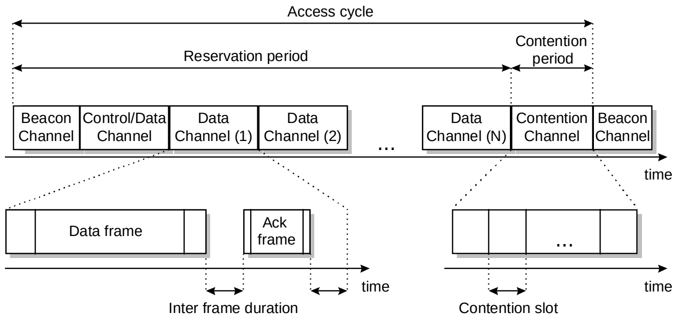


Figure 4. TUTMAC channel access cycle.

A data channel can be reserved by a PS for uplink transmission of user data. In a default case all data is forwarded by the BS. However, also a direct data-transfer between two PSs, under a common BS control, is optionally possible. Data channels stay reserved while data exists to be transmitted and are released by the request of the PS or, in case of an idle reserved channel, automatically by the BS. The reservation and release procedures are illustrated in Figure 5.

Another uplink control channel is formed by short acknowledgement messages that follow each unicast (destined to a single station) transmission. TUTMAC exploits a simple stop-and-wait flow control scheme in order to enable short delays with retransmissions and fast adaptation to the varying quality of radio link. Besides acknowledging a successful or unsuccessful reception, the acknowledgements carry control information for bandwidth requirements. This consists of the amount and priority of data queued in a PS for transmission. The information is used by the channel scheduling function of a BS to determine the direction (uplink/downlink) of reserved data channels and a possible requirement for an extra bandwidth for each PS.

PS transmits uplink control messages, such as channel reservation and association requests, in a contention based channel at the end of the access cycle. The contention channel is constructed by a series of short contention slots that are monitored for a signal carrier or energy for Carrier Sense Multiple Access (CSMA) based transmissions. The amount of idle contention slots to be detected before transmitting also enables various control message priorities.

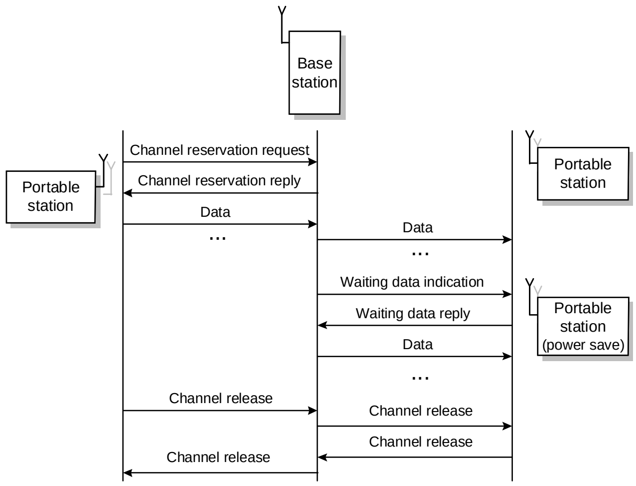


Figure 5. Channel reservation and release.

The BS transmits downlink control messages in the control and beacon channels. The control channel is actually a data channel that can be reserved for control and management information transfer. Otherwise the channel is used as a variable bit-rate data channel. The beacon channel is used by the BS only for beacon messages. A beacon broadcasts the current channel reservation state for the following access cycle. Beacons also carry cell specific information, such as a cell identification, structure of the access cycle and indications for a required confidentiality with the association and data-transfer. A beacon frame indicates the beginning of the access cycle thus providing a TDMA cycle synchronisation for PSs. Furthermore, the beacon carries indications for buffered data to those PSs that utilise power save functionality. These stations power on their receivers only periodically to receive the possible announcement with the beacon.

VI. TUTMAC SERVICE PARAMETERS

The TUTMAC protocol introduces three QoS parameters referred to as QoS types that together define the data-transfer service. The defined parameters are: *connection type*, *service type* and *security type*. Each type consists of a set of attributes grouped together for a more convenient management. The connection type and security type parameters are negotiated between a PS and BS during the connection establishment, i.e. data channel reservation. If the required resources cannot be provided the reservation is considered unsuccessful. The supported service types are defined by the BS and delivered to other stations during the association procedure.

A. Connection Type

The connection type contains a set of attributes specific for each wireless connection. The available attributes are: required channels, requested channels, changing bandwidth and connection permanency. The first two announce the minimum and a desired number of data channels for a reserved connection. The changing bandwidth informs whether the connection throughput can vary between the required and requested. With the connection permanency attribute the portable station may request a connection that remains reserved also over idle periods.

B. Service Type

The service type parameter can be assigned separately for each transferred MAC Service Data Unit (MSDU) in order to define a suitable processing for the data. By defining new service types, the TUTWLAN management can adapt the data-transfer service according to the operational environment of the network and custom data characteristics.

The service type consists of attributes for error recovery and for data priority. The error recovery attributes define the usage of ARQ and/or FEC schemes. Retransmissions can be denied or limited and the strength of the FEC coding changed. If bit errors still remain after decoding the data may still be delivered to the user. The priority of the user data in each service type is determined by two independent attributes: data priority and aging time. If an aging time of an MSDU expires during the queuing for a transmission the data is concluded obsolete and discarded.

C. Security Type

Security type defines the level of security utilised in the protecting of data transmissions against eavesdropping. The parameter is associated with the supported encryption algorithms and keys. Data-transfer can occur in three different security levels: open, cell-wide and secret key. The open security provides no cryptographic operations while the cell security defines a wired LAN equivalent level of security, i.e. stations in the same cell share a common encryption key. The third security level, secret key, enables a secure transfer between two stations in the same cell [7].

VII. TUTMAC MANAGEMENT

BS is the central point of control and management in a TUTWLAN cell. The BS management functionality contains channel scheduling, traffic forwarding, topology and radio management functions and a network management application. Security, address, MIB, power save and QoS management functions are included in all TUTWLAN stations.

The topology and address management functions keep track of associated PSs in the cell. Since there exists a limit to the number of simultaneous users, idle PSs will be disassociated by the BS. To reduce the overhead produced by the transfer of 48-bit standard OSI MAC addresses these are converted into TUTWLAN cell specific 8-bit TUTMAC station IDs. IDs are assigned by the BS during association and the changed address conversion information is broadcast to all PSs in the cell.

Various control and management parameters are distributed to the associated PSs via broadcast messages or unicast transmissions. By these messages the BS can dynamically control the operation of the PSs in the cell, for example when adapting to a changed radio environment by exchanging the carrier frequency or to an increased amount of PSs by adding the number of data channels in the access cycle.

VIII. PROTOTYPE CHARACTERISTICS AND PERFORMANCE

Currently a prototype of the TUTWLAN is being implemented including a realisation of the TUTMAC protocol software and a demonstrator platform hardware. The radio module utilised in the prototype is PRISM1RC-EVAL (no MAC processor included) manufactured by Harris Semiconductors [6]. The module is implemented meeting the IEEE 802.11 wireless LAN standard for physical layer specification for 2.4 GHz DSSS (Direct Sequence Spread Spectrum) [1].

The prototype utilises 1 Mbit/s link speed of the radio module. The number of simultaneously associated portable stations is restricted to ten, while four or eight of them being simultaneously in an active mode. Thus the access cycle in the prototype has two configurations: four and eight data channels. No FEC coding, further to the coding gain achieved by the DSSS, is used at this phase. The TUTMAC data frame and physical frame structure [1,6] of the prototype are presented in Figure 6.

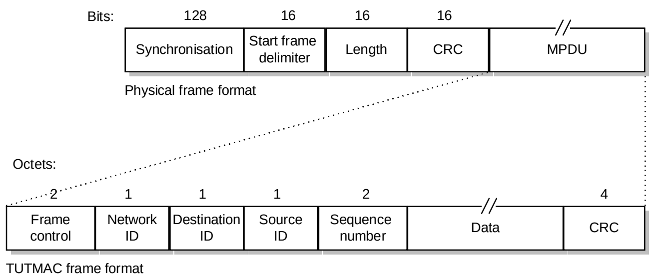


Figure 6. Physical and MAC frame formats of the prototype.

Figure 7 presents the relation between the data channel throughput, radio layer frame length and estimated medium allocation efficiency. The purpose is to maximise the throughput while maintaining the lengths of an access cycle and transmitted frames short in order to minimise transfer delays and transfer error probabilities. In the prototype, a 128 kbit/s channel throughput has been chosen for the four data (plus a control) channel configuration. This implies short lengths for a data frame (approx. 200 octets) and access cycle (ca. 10 ms) and results to an adequate medium allocation efficiency (over 60 %).

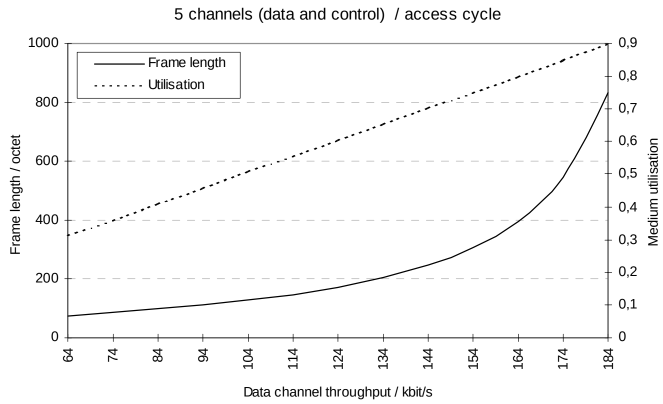


Figure 7. Physical frame length and theoretical efficiency.

IX. CONCLUSIONS

TUTWLAN has been developed to provide a wireless data-transfer service for office and home environments and for restricted public areas. The most important system requirements have been on a reduced-complexity system, supporting real-time, multimedia type of services, thus providing suitability for various applications ranging from wireless sensors to portable multimedia terminals. Currently, a prototype containing a network adapter platform and a TUTMAC protocol software is being implemented. The development continues by testing and verification of the prototype in the target operational environments and using the results for further improvement of the system.

REFERENCES

[1] IEEE P802.11 D5.0, IEEE Standard for local and metropolitan area networks: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE Standards Department, 19.7.1996

[2] ETS 300 652, Radio Equipment and Systems (RES); High Performance Radio Local Area Network (HIPERLAN), Type 1; Functional specification, October 1996

[3] F. J. Bauchot, F. Lanne, “IBM Wireless RF LAN design and architecture”, IBM System Journal, Vol. 34, No. 3, 1995, pp. 390-408

[4] D. F. Bantz, F.J. Bauchot, “Wireless LAN Design Alternatives”, IEEE Network, March/April 1994, pp. 43-53

[5] K.-C. Chen, “Medium Access Control of Wireless LANs for Mobile Computing”, IEEE Network, September/October 1994, pp. 50-63

[6] Harris Semiconductor, “Wireless Handbook”, 1997

[7] K. Salli, T. Hämäläinen, J. Knuutila, J. Saarinen, “Security Design for A New Wireless Local Area Network TUTWLAN”, accepted to the IEEE PIMRC’98

Publication [P5]

Kari M., Hännikäinen M., Hämäläinen T., Knuutila J., Saarinen J., "Configurable Platform for a Wireless Multimedia Local Area Network", International Workshop on Mobile Multimedia Communications (MoMuC 1998), pp. 301-306, Berlin, Germany, October 12-14, 1998.

Proceedings of 5th Intl. Workshop on Mobile Multimedia Communication *MoMuc'98*, October 12-14 1998, Berlin

Configurable Platform for a Wireless Multimedia Local Area Network

Marjo Kari[1], Marko Hännikäinen[1], Timo Hämäläinen[1], Jarno Knuutila[2], Jukka Saarinen[1]

Abstract -- **This paper presents a configurable demonstrator platform for a new wireless local area network TUTWLAN. The network is purposed for limited service areas with stationary or portable terminals ranging from simple wireless sensors to multimedia laptops, for which Quality of Service parameters is guaranteed. The design objective has been to construct a platform that enables a seamless development of the system from a protocol design to implementation. Consequently, several network design alternatives and system configurations can be supported by the platform. The platform architecture is divided into functional and structural modules that can be reused to form different terminal types as well as reconfigured at run-time for desired applications. Software configurability is due to a formal, high abstraction Specification and Description Language (SDL) producing executable codes. Configuration for dedicated hardware blocks is obtained by VHDL synthesis. A TUTWLAN demonstrator, while implementing a single wireless network terminal, consists of a distinct wireless network adapter card connected to a host computer. The demonstrator also includes the protocol related software modules.**

Index Terms—**Configurable platform, SDL, VHDL, WLAN**

I. Introduction

A Wireless Local Area Network (WLAN) provides flexible data-services with fast and easy network management for a limited geographical area. WLANs have been emerging to markets for several years while still not been able to replace or even compete with traditional wired LANs. Several vendor specific solutions [1] have gained popularity especially in service sector where the movement of users is required. However, the WLAN technology has been suffering from low performance, high costs, limited communication range, and especially lack of standardisation. New WLAN standards IEEE802.11 [2] and High Performance Radio Local Area Network (HIPERLAN) [3] are now available but few products meeting these specifications exist. The new standards are good examples of solutions where high performance, mobility and multi-user characteristics have been taken into specification. However, these general-purpose systems could be too complex and expensive for simple, stationary wireless applications. Consequently, a new low-cost, reduced complexity TUTWLAN system is being designed and implemented in Signal Processing Laboratory at Tampere University of Technology (TUT) [4].

In the TUTWLAN development the goal is to achieve a seamless design flow from a high-level requirements specification into the realisation of a physical device. For a TUTWLAN prototype, a Medium Access Control (TUTMAC) protocol has been designed and implemented with a formal, high abstraction level Specification and Description Language (SDL) specified by ITU-T [5]. The protocol is also verified in the SDL development tool environment in a workstation. A C-language code is generated automatically from the SDL description, compiled into an executable code, and ported into the demonstrator platform.

Careful design of the platform provides a convenient embedding for the ported protocol code as well as for the testing and benchmarking of various protocol configurations and applications in practice. The scope of this paper is on the implementation of the platform including software and hardware modules. The next section presents the architecture of TUTWLAN and operation of TUTMAC. Also the most important design aspects are introduced. In Section 3, first an overview of the platform architecture is given followed by a presentation of the software and hardware modules. Section 4 states the conclusions summarising the key development experiences and design issues.

II. Overview of TUTWLAN

The TUTWLAN architecture, as illustrated Figure 1, consists of two types of terminals: *Base Station* (BS) and *Portable Terminal* (PT). A base station is the central point of control and management of its coverage area, referred to as a wireless cell. All portable terminals within the cell area communicate via the base station, although also an optional direct data-transfer between two portable terminals is specified. However, the direct transfer function is available only under the base station control. A base station operates

[1]Tampere University of Technology,
Signal Processing Laboratory
Hermiankatu 12 C, FIN-33720 TAMPERE, Finland
Tel: +358-3-365 2111, Fax: +358-3-365 3095
E-mail: marjok@lenkkari.cs.tut.fi, markoh@cs.tut.fi, timoh@cs.tut.fi, jukkas@cs.tut.fi

[2]Nokia Mobile Phones, Wireless Data
Sinitaival 5, FIN-33720 TAMPERE, Finland
Tel: +358-10-505 6800, Fax: +358-10-505 6888
E-mail: jarno.knuutila@nmp.nokia.com

also as an access point to other networks. A portable terminal can appear in two separate operational modes according to the network topology: A portable that has registered its presence with a base station appears in an *associated mode* while an unregistered portable terminal remains *non-associated*. The associated mode is further divided into *active* and *non-active modes* according to reserved wireless connections. A non-active terminal is only able to exchange control messages and receive broadcast data, but has no data channel reserved, as does an active mode terminal.

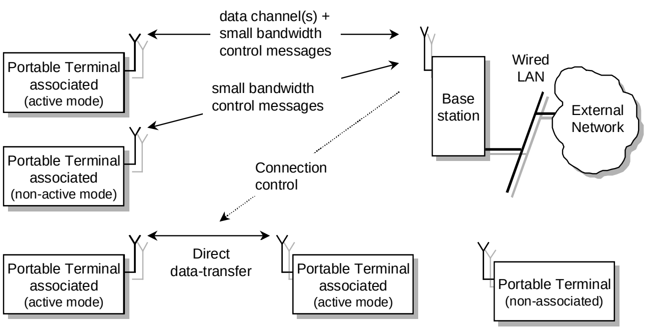


Figure 1. TUTWLAN architecture

The key design objective of the TUTMAC protocol has been to develop a simple system capable of supporting multimedia services, such as real-time audio and video. TUTMAC is a connection oriented wireless MAC protocol that utilises a reservation based Time Division Multiple Access (TDMA) scheme with the shared radio channel. This kind of MAC protocol requires a centralised control and an accurate TDMA frame synchronisation over the wireless link.

The structure of the TDMA frame of the TUTMAC is illustrated in Figure 2. The frame is divided into time slots that form constant bit-rate channels. Four types of channels are distinguished by their purpose, direction and bandwidth: *data*, *contention*, *control* and *beacon* channels. The data channel includes also *acknowledgements*. A data channel can be reserved by a portable terminal for uplink data transmission via a short reservation request frame transmitted in the contention channel at the end of the TDMA frame. The contention channel is only used for short control messages while longer control and management information messages are exchanged through a reserved control channel. The control channel is otherwise used as a normal data channel.

Another uplink control channel is formed by short acknowledgement messages that follow each data frame transmission. Consequently, TUTMAC exploits a simple stop-and-wait flow control in order to enable short delays with retransmissions and fast adaptation to the varying quality of the radio link. The beacon channel is used only by the base station for beacon messages. A beacon broadcasts the current channel reservation state for the following data channels. Beacons also carry various cell specific information. A beacon frame indicates the beginning of the TDMA frame thus providing a synchronisation for the portable terminals.

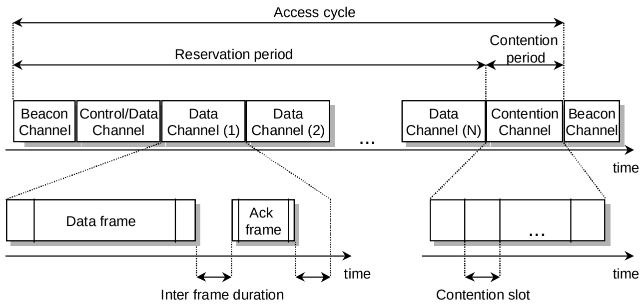


Figure 2. TUTMAC TDMA frame

The block architecture of the demonstrator platform is depicted in Figure 3. The platform consists of a *Radio sub-system*, *Radio Interface*, *MAC*, and *Host Interface* modules. This platform can be attached to a PC that contains the actual data applications or utilised as a stand-alone system. In the latter case the applications are executed in the MAC module while the required peripheral devices, such as keyboards, displays and censors, are attached to the platform via the host interface.

The modularity of the system architecture allows different configurations according to a desired terminal type. For example, the radio module can be replaced with another type of radio or an infrared transceiver by reconfiguring the radio interface. Furthermore, even two different transmitters can be used simultaneously. Configurations for each module can be downloaded from the host computer. The functions and the structure of each module are presented in the following sections.

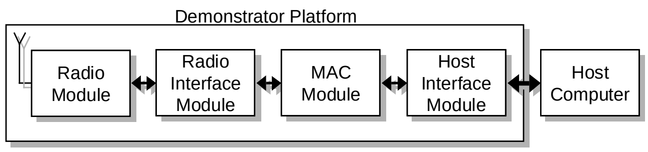


Figure 3. Demonstrator platform HW modules

III. DEMONSTRATOR PLATFORM MODULE ARCHITECTURE

The demonstrator platform modules are constructed using reconfigurable hardware together with downloaded software configurations. The complete HW/SW architecture of the platform, together with the demonstrator applications of the host computer, is presented in Figure 4.

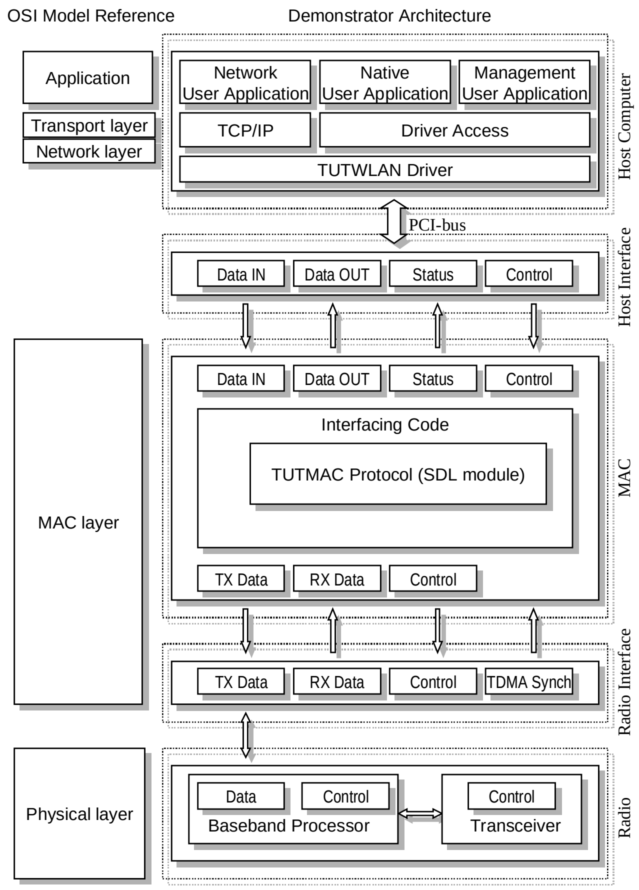


Figure 4. Demonstrator Platform HW/SW architecture

A. Host Computer

A host computer is utilised for downloading system configurations into the demonstrator platform. It is also convenient when verifying the operation of the demonstrator and measuring the performance of the system. Furthermore, interoperability with legacy Internet applications is achieved. The host computer contains several TUTWLAN specific software blocks as depicted in Figure 4. The *TUTWLAN driver* is a Windows NT device driver for the demonstrator platform card. The driver supports all hardware control, for example the initialisation of the MAC module at system reboot. The *driver access* is a component required for the user mode application to access the device driver without the *network protocol* stack, such as TCP/IP.

Three kinds of applications are supported in the host computer: A *network user application* refers to an Internet application, e.g. Web browser, and thereby requires standard Internet protocols. A *native user application* presents a TUTWLAN system specific software that does not require any pre-installed network protocol. Instead it utilises the functions of the driver, and thereby the services of the TUTMAC protocol, directly through the TUTWLAN driver access block. The third application alternative is a *management user application*. This application is implemented for controlling, managing and observing the operation of the TUTMAC protocol and for gathering network management information.

B. Host Interface

PCI (Peripheral Component Interconnect) bus was chosen for the interconnection between the host and MAC modules due to very high communication bandwidth, low-latency and support for PCs, workstations and various embedded systems (industrial PCI). By reconfiguring the host interface module, it is possible to connect several MAC modules to the host system, thus realising a multiterminal with several independent MAC protocols and physical connections.

The host interface module is implemented with a Field Programmable Gate Array (FPGA) circuit of type Xilinx XC4010E [6]. It offers sufficient logic resources, speed and fulfils the tight electrical requirements of the PCI-bus [7]. The logic inside the interface is designed with Very high-speed integrated circuit Hardware Description Language (VHDL) which ensures convenient high-level description of hardware functions and fast reprogramming.

The communication between the host interface and MAC module consists of data reads, writes and interrupts. For these tasks the host interface includes FIFO (First In First Out) buffers and registers for status and control information. These registers are used for handshaking for the lowest-level data transfers between hardware modules. Also, higher-level control information, associated for example with the TUTMAC protocol service primitives, is exchanged through these registers.

C. MAC Module

The MAC module, illustrated in Figure 4, consists of a DSP processor, in which the TUTMAC protocol software module is embedded. The TUTMAC protocol is implemented in a separate development environment using SDL. SDL is an efficient tool to be used during the whole development cycle of a system, from requirement specification into design and implementation. The formality of the language allows the system model to be verified and simulated in all development stages. Similarly, the original requirements of the system can be easily traced back and verified. Furthermore, SDL provides an intuitively clear and simple graphical presentation and is thereby easy to understand also for non-engineers.

The formal SDL specification can be automatically converted into C-language code, which decreases the implementation period of the development cycle significantly. In addition, the automatic code generation enables faster prototyping during the development phase and facilities the maintenance and evolution of the system.

The high level abstraction of the design language introduces drawbacks in end-systems with hard real-time requirements. The performance of the automatically generated C-code is not optimised according to the properties of the end-system hardware platform. For another, SDL lacks the support for writing complex structured algorithms, since constructs such as loops and structured conditionals are not included into current versions of the language.

To improve performance, the most complex and time critical algorithms of the system can be implemented using external optimised C-functions that are appended to the SDL description. Furthermore, several SDL development tools [8] include features for performance optimisation, such as setting priorities for each system process and construction of data-types for passing parameters.

The generation process for the executable platform or simulator code is illustrated in Figure 5. The converted C-code from the SDL description is compiled using the platform specific C-compiler. Next the compiled code is linked with the *Function Library* containing the coding for SDL constructs and types for the platform.

The platform specific features are isolated from the SDL generated code by a simple *Interfacing Code*. The interfacing code connects the input and output signals and passed parameters of SDL into the corresponding platform operations. The interface coding ensures that the porting of an SDL description can be done platform independently also to other processors.

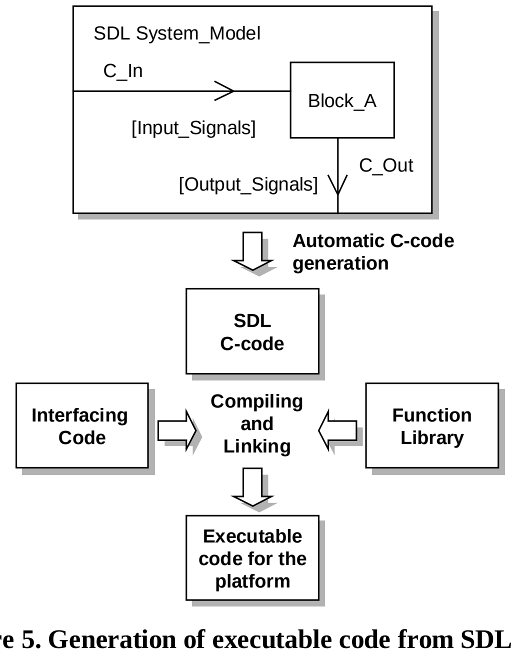


Figure 5. Generation of executable code from SDL

In the TUTWLAN demonstrator platform, a *light integration* [8] method for the porting of the SDL specified system is utilised. In the light integration, no services from the underlying operating system are required. Consequently the compiled code can be ported to a naked platform without an operating system. For this reason, the executable code contains an own, non-pre-emptive, scheduler for allocating processor time for SDL processes.

Another possibility for porting is to use the services of an operating system, which in this case would require a real-time operating system for the MAC processor thus increasing the costs and complexity. On the other hand, operating systems usually support pre-emptive scheduling that enables a more efficient implementation of systems with strict real-time requirements.

The SDL development tool used in the TUTMAC protocol development is the SDL Development Tool (SDT) by Telelogic AB [8]. The tool includes an integrated simulator and a Message Sequence Chart (MSC) generator useful for the verification of the model.

The DSP chosen for the MAC module is Analog Devices's ADSP-21062 processor that contains a large 256 Kbytes on-chip memory, distinct I/O processor, 40 MIPS processing power and good multiprocessor interfacing capabilities, which allows experimenting of various design choices and system configurations [9]. In addition to DSP, only an optional external memory bank and few discrete components are required for the implementation.

The demonstrator platform provides substantial amount of processing capacity. The processing resources are needed for the testing and implementing of various real-time data processing operations, such as security related encryption/decryption algorithms, traffic scheduling functions in the base station, and forward error correction coding.

D. Radio Interface

The radio interface is a dedicated hardware block utilised to separate the MAC and radio modules in a way that allows one or both to be changed. In addition, it is used for accelerating operations having critical timing requirements, such as calculation of a Cyclic Redundancy Check (CRC) sum over a received frame. These operations are not viable to be computed on the system processor because of additional delays.

The most important task of the module is to connect the signal lines and buses between the MAC processor and radio sub-system. This function adapts the non-synchronised data streams by buffering and manages the control line handshaking between the two modules. If a radio module is changed, a new configuration described in VHDL is loaded into the interface chip, which allows a flexible method for testing various WLAN radio modems with the TUTMAC protocol.

For TUTMAC, the radio interface contains also MAC layer related functionality utilised for adapting the TUTMAC protocol into the physical layer, i.e. to the radio sub-system. Firstly, this functionality provides an efficient and simple functional interface to be used by the TUTMAC protocol. This means that various control procedures of the radio sub-system, such as placing the radio module into a power saving mode and changing of the used radio carrier frequency, can be initialised by a single control message.

Secondly, the radio interface contains a link level synchronisation function required by the TDMA framing. For this, the radio interface contains an accurate timing information of the phase of the current TDMA frame cycle in a special control register (*TDMA frame counter*). This register value is reset, as depicted in Figure 6, by a received beacon frame at the beginning of the TUTMAC channel access cycle presented in Figure 2. The counter value can be accessed by the TUTMAC to obtain an exact reception time of a radio frame.

Similarly the TUTMAC protocol can utilise the frame counter to assign an exact transmission time for a frame. The exact time is stored into another control register called *transmission time*. This register value is compared to the phase of the TDMA frame (in the TDMA counter register) and the transmission of a radio frame is initialised at the time when values of the two registers agree.

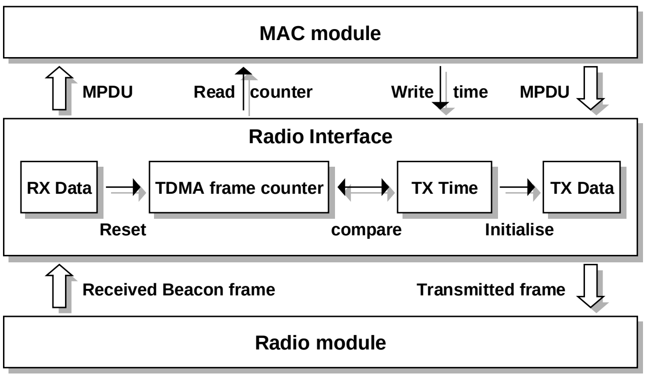


Figure 6. TDMA frame synchronisation in radio interface

A Xilinx XC4010E FPGA chip [6] is used to implement the radio interface. The chip is configured using VHDL. This FPGA chip is re-programmable through SPROM (Serial Programmable Read Only Memory).

E. Radio Module

The radio module in the prototype platform utilises a 2.4 GHz frequency band with a Direct Sequence Spread Spectrum (DSSS) technique. Currently a 1 Mbit/s link speed with Differential Binary Phase Shift Keying (DBPSK) carrier modulation is used. Also other types of physical transmission channels could be used by modifying the platform software, i.e. applying a new configuration to the radio interface. The radio module chosen for the demonstrator platform is PRISM1RC-EVAL [10], manufactured by Harris Semiconductors. The radio is implemented meeting the emerging IEEE 802.11 [2] wireless LAN standard for DSSS physical layer specifications. In addition, the radio module has capabilities that go beyond the 802.11 specification. These are especially useful for testing purposes.

IV. CONCLUSIONS

TUTWLAN has been developed to provide a wireless data-transfer service for office and home environments and for restricted public areas. The most important system requirements have been on a reduced-complexity system, supporting real-time, multimedia type of services, thus providing suitability for various applications ranging from wireless sensors to portable multimedia terminals.

Easy configuration and the reuse of software and hardware modules have been the key design drivers for implementing the presented demonstrator platform. The goal has been achieved at three levels: In the specification and design phase, SDL offers a flexible way to modify the desired features. These modifications can be effectively verified by downloading the corresponding software modules into the platform. An initial, or system boot-up time, configuration is performed when hardware modules are plugged into each other. At run-time, the embedded TUTMAC software can also change the functionality of hardware by modifying control register parameters.

Currently a prototype system containing two wireless stations is under construction and will be in function in the fall 1998. The development continues by testing and verification of the prototype in the target operational environments and using the results for further improvement of the system. In the future, current board-level modularity of the hardware will be included into a single chip. This chip-level implementation requires configurable processors that include FPGA-like portions on the same chip. In order to enable the system level simulations, also the VHDL hardware blocks will be described in the SDL format.

V. REFERENCES

[1] Nathan J. Muller, "Wireless Data Networking",Norwood, Artech House, 1995

[2] IEEE P802.11 D5.0, IEEE Standard for local and metropolitan area networks: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE Standards Department, 19.7.1996

[3] ETS 300 652, Radio Equipment and Systems (RES); High Performance Radio Local Area Network (HIPERLAN), Type 1; Functional specification, October 1996

[4] M. Hännikäinen et al., "TUTMAC: A Medium Access Control Protocol for A New Multimedia Wireless Local Area Network", Accepted to IEEE PIMRC'98 Conference, 1998
[5] CCITT Specification and description language, ITU-T recommendation Z.100 (03/93)
[6] Xilinx, "XC4000E and XC4000X Series Field Programmable Gate Arrays", Product Specification, 1997
[7] K. Kuusilinna, "FPGA Based PCI Interface Design", Tampere University of Technology, Technical Report, 5-1997
[8] Telelogic Tau 3.2 User's Manual, Telelogic AB, 1997
[9] Analog Devices Inc., "ADSP-2106X SHARC User's Manual", 1997
[10] Harris Semiconductor, "Wireless Handbook", 1997

Publication [P6]

Tikkanen K., Hännikäinen M., Hämäläinen T., Saarinen J., “Advanced Prototype Platform for a Wireless Multimedia Local Area Network", The European Signal Processing Conference (EUSIPCO 2000), Volume 4, pp. 2309-2312, Tampere, Finland, September 5-8, 2000.

Reprinted from the proceedings of EUSIPCO 2000.

ADVANCED PROTOTYPE PLATFORM FOR A WIRELESS MULTIMEDIA LOCAL AREA NETWORK

Kimmo Tikkanen, Marko Hännikäinen, Timo Hämäläinen, and Jukka Saarinen
Digital and Computer Systems Laboratory, Tampere University of Technology
Hermiankatu 3 A, FIN-33720 Tampere, FINLAND
Tel: +358 3 365 2111; Fax: +358 3 365 4575
e-mail: kimmot@cs.tut.fi, markoh@cs.tut.fi, timoh@cs.tut.fi, jukkas@cs.tut.fi

ABSTRACT

This paper presents an advanced version of a configurable demonstrator platform developed for a new wireless local area network called TUTWLAN. TUTWLAN is targeted for limited service areas with stationary or portable terminals. Applications range from simple wireless sensors to multimedia laptops. The network supports the different Quality of Service (QoS) requirements of these applications. The improved development platform has been designed because of the restrictions discovered in the first prototype. The new platform provides better testing environment for developing Medium Access Control (MAC) protocols for TUTWLAN and for designing embedded stand-alone applications. Furthermore, various other designs can be tested, for example hardware implementations of encryption algorithms. Both the new and the old prototypes consist of a Digital Signal Processor (DSP), external memory modules for the DSP, and a Field Programmable Gate Array (FPGA) circuit. The platform is connected to a radio module and can be attached to a host computer using Peripheral Component Interconnect (PCI) bus. Compared to the original platform, the new prototype contains more memory, a faster and larger FPGA, and a higher bit-rate radio.

1 INTRODUCTION

A Wireless Local Area Network (WLAN) provides flexible data-services with fast and easy network management for private areas, such as homes, offices, and for limited public access areas, such as libraries and airport lounges. WLANs have been emerging to the markets for several years, but have still not been able to compete with traditional wired LANs. WLAN solutions have gained popularity in specific sectors, such as service and education, where the movement of users is required. However, the WLAN technology has been suffering from low performance, high costs, limited communication range, and lack of compatibility [1].

WLAN standards IEEE 802.11 [2] and High Performance Radio Local Area Network (HIPERLAN) [3] are now available and the specification work proceeds towards higher bit-rates. However, these general-purpose systems could be too complex and expensive for simple, wireless applications and on the other hand too general for specific user services, such as real-time video. Consequently, a new low-cost, reduced complexity TUTWLAN system is being designed and implemented in Digital Systems Laboratory at Tampere University of Technology (TUT) [4] [5]. TUTWLAN is targeted for a broad range of services, including multimedia and real-time circuit switched data. In the TUTWLAN development the goal is to achieve a seamless design flow from high-level requirements to the realisation of a physical device. For TUTWLAN prototyping, a Medium Access Control (TUTMAC) protocol has been designed and implemented with a formal, high abstraction level Specification and Description Language (SDL) [6].

In the TUTWLAN development project, a demonstrator platform has been designed to provide a testing environment for the TUTMAC protocol testing. Currently, the first platform has been implemented and TUTMAC testing is being performed. An advanced version of the platform has now been designed. The scope of this paper is on the implementation of the hardware for this advanced prototype.

This paper is organised as follows. First, an overview of the first platform architecture is presented. After that the advanced platform architecture is depicted and compared with the first platform version. Arguments for design choices made are stated. Furthermore, the most important design considerations are presented. Conclusions summarising the key development experiences and design issues are given in the last section.

2 ORIGINAL DEMONSTRATOR PLATFORM ARCHITECTURE

The block architecture of the original demonstrator platform is depicted in Figure 1 together with components used for implementing these functional blocks. The platform consists of a *radio sub-system*, *radio interface*, *MAC*, and *host interface* modules [5]. The modularity of the system architecture allows different configurations according to desired terminal properties. For example, the radio module can be replaced with another type of radio or with an infrared transceiver by reconfiguring the radio interface.

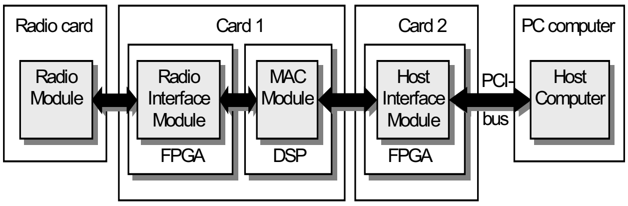


Figure 1. Hardware modules of the first platform.

A host computer is utilised for downloading system configurations into the various modules of the platform. The host computer is also used for verifying the operation of the demonstrator and for measuring the performances of tested configurations. The host computer is a PC with Microsoft Windows NT operating system. Furthermore, interoperability with legacy Internet applications and interconnection to wired LANs are achieved via the host computer. The host computer contains several TUTWLAN specific software blocks, such as device drivers, protocol drivers, and management software [5].

PCI bus was chosen for interconnecting the host computer and the MAC module due to its very high communication bandwidth, low-latency, and support for PCs, workstations, and various embedded systems (industrial PCI). The host interface module is implemented using Xilinx XC4010E FPGA [7]. The logic inside the interface is designed with Very high-speed integrated circuit Hardware Description Language (VHDL), which ensures convenient high-level description of hardware functions and fast reprogramming.

The MAC module consists of a DSP processor, in which the TUTMAC protocol software module is embedded. DSP chosen for the MAC module is Analog Devices' ADSP-21062 processor that contains 2 Mbits on-chip memory and 40 MIPS of processing power [8].

The most important task of the radio interface module is to interconnect the signal lines and buses between the MAC processor and the radio sub-system. This function adapts the non-synchronised data streams by buffering and manages the control line handshaking between the two modules. If a radio module is changed, a new configuration described in VHDL is loaded into the interface chip, which allows testing various WLAN radio modems with the TUTMAC protocol.

In addition, the radio interface module is used for accelerating MAC operations having critical timing requirements. Examples of these tasks are the calculation of a Cyclic Redundancy Check (CRC) sum over a received frame, Forward Error Coding (FEC), synchronisation to the used Time Division Multiple Access (TDMA) of the medium, and data encryption algorithms. These operations are not viable to be computed on the system processor because of the resulting delays. A Xilinx XC4013E FPGA chip is used to implement the radio interface [7].

The radio module in the prototype platform utilises a 2.4 GHz frequency band with a Direct Sequence Spread Spectrum (DSSS) technique. Currently, a 1 Mbit/s link speed is used. The radio module chosen for the demonstrator platform is PRISM1RC-EVAL, manufactured by Intersil [9].

3 NEW DEMONSTRATOR PLATFORM ARCHITECTURE

An advanced prototype platform for TUTWLAN testing is being developed because of certain testing challenges were encountered. These challenges are discussed in the following sections. Furthermore, new improvement ideas were revealed during the development and testing of the first TUTWLAN prototype. For example, the lack of testing points in the first prototype was noticed. Furthermore, programs generated using SDL require larger program memory on the MAC processor. Also, the FPGAs of the first platform should have more capacity and performance.

The first prototype was implemented using two Printed Circuit Boards (PCB). In order to lower manufacturing costs these two PCBs were integrated on a single PCB for the advanced prototype. Architecture and the used components of the new platform are depicted in Figure 2. The figure also illustrates the placement of the original prototype's functional blocks as shaded blocks.

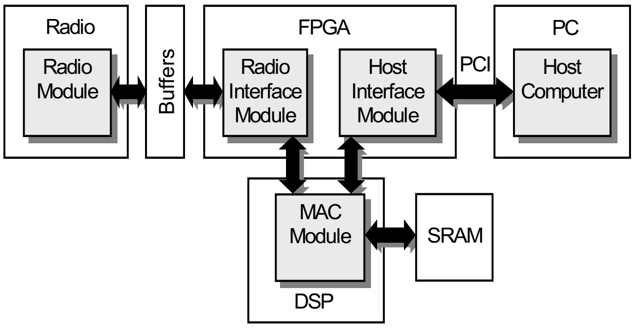


Figure 2. Components and functional blocks of the advanced platform.

3.1 FPGA of the Advanced Platform

In the new prototype, the host interface and the radio interface functional modules are located on a single FPGA compared with the two separate FPGAs of the first prototype. In the new prototype, these functional blocks are implemented with a larger and faster Xilinx XCV800 FPGA [10]. Because the new prototype was designed using a single PCB, two separate FPGAs were found to be too large in their physical dimensions to fit on a same PCB with other needed components in the new prototype.

More FPGA capacity was needed because the host interface module took about 80% of the logic cells on the old FPGA and the radio interface module needed about 75% of its FPGA's capacity. Consequently, it was impossible to add any new functional blocks, such as encryption modules, in the old FPGAs. For example, a well-known encryption algorithm RC4 [14] implemented with a Xilinx 4013E FPGA takes about 40% of its logic cells.

Implementation of more reliable and more secure Triple-DES [14] encryption algorithm is not even possible using this FPGA, since it requires more logic cells than the FPGA provides. The total number of logic cells on the new FPGA is about twenty times larger than the number of the logic cells on the two old FPGAs together.

Performance of the old FPGAs has been a restriction in the old prototype. Written VHDL-models had to be optimised carefully in order to achieve the desired performance level. The new FPGA is considerably faster than the older ones. Therefore, the performance of the FPGA should not prevent the testing of TUTMAC protocol with complete functionality.

The new FPGA also has a lower 3.3V supply compared with the original platform. However, the radio module still requires the same 5V signal levels. New challenges of design were faced when interfacing the radio sub-system to the FPGA. As a result, buffers between these components have to be added. Furthermore, the use of several supply voltages in a same PCB requires adding of several separate power supply layers to the PCB.

Logic inside the new FPGA is designed using VHDL-codes of the first prototype. Only minor changes to the top hierarchy level are needed, e.g. pin mappings have to be changed and component names have to be updated to correspond component names in the new VHDL-libraries of FPGA.

3.2 TUTMAC Protocol Requirements

The TUTMAC protocol is implemented in a separate development environment using SDL. The formal SDL description can be automatically converted into C-language for implementation, but this generated code is usually not space optimised.

In TUTWLAN research, the design tool for SDL has been SDT (versions 3.4 and 3.6) manufactured by Telelogic [11]. SDT belongs to the Telelogic Tau product family. The Cadvanced code generator of the tool has been used for generating the TUTMAC protocol application. SDT also contains the Cmicro code generator that produces a further optimised code and is therefore more suitable for programming embedded systems. However, not all the SDL properties, such as remote procedure calls, enabling conditions, continuous signals, and declaring of infinite number of process instances can be utilised with Cmicro.

The TUTMAC protocol implemented using SDL requires more memory from the target DSP than the original platform provides. The compiled protocol for a TUTMAC base station protocol needs 490 kbytes memory for the program code. In addition, 61 kbytes are required for static variables and approximately 100 kbytes for dynamic memory allocations.

3.3 Processor and Memory Selection

Because of high memory requirements of the generated TUTMAC protocol, DSP chosen for the MAC module of the advanced platform is Analog Devices' ADSP-21060L processor, that is fully compatible with the old platform processor [8]. The new DSP contains 4 Mbit on-chip memory, which is twice as much as the processor in the first prototype. The selected DSP has also 3.3V supply voltage which lowers the power consumption and makes the interfacing with the selected FPGA easier.

In addition to the large internal memory, a large external memory for DSP has been added on the prototype PCB. The selected memory is White Electronic Designs EDI8F32512V-MZ static RAM (SRAM) modules [12]. This memory type was chosen because of its fast access time and high storage capacity. Each memory module provides a 512k x 32 bit memory block. Furthermore, each SRAM-module has two selection signals (cs0 and cs1). By using these signals the module can be used as two separate blocks, both having 512k x 16 bit capacity. The width of the DSP data path is 48 bits. Therefore, three separate memory modules connected to form a 1M x 48-bit memory are used on each PCB. Thus, the total capacity of the DSP program memory is now 6 Mbytes.

Connections between DSP and SRAM-modules are depicted in Figure 3. DSP uses two logically separate external memory blocks. Selection signals (MSX0 and MSX1) are used to choose the corresponding SRAM-modules. The first external memory block of the DSP consists of the SRAM-module 1 and half of the SRAM-module 2. Correspondingly, the second memory block consists of SRAM-module 3 and half of the SRAM-module 2. The advanced platform is expected to provide enough memory capacity to enable testing of embedded applications in addition to the TUTMAC protocol.

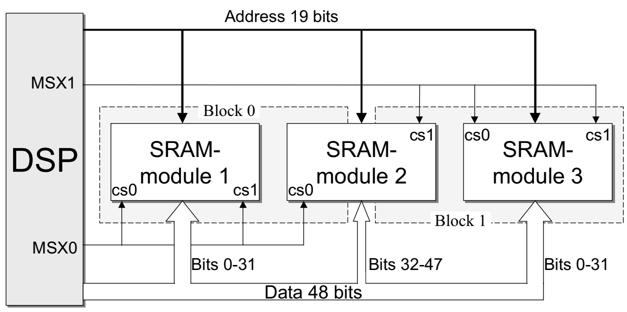


Figure 3. Connections between DSP and SRAM-modules

3.4 Radio Module

Radio used in the advanced prototype is changed to HWB1151-EVAL from Intersil [13], because of its higher 11 Mbit/s data rate compared with the original 1 Mbit/s rate. The radio of the previous prototype can also be used, since the signals and the connector of the new radio are compatible with it. Only the logic inside the radio interface module must be modified. The 11 Mbit/s data rate results into challenges for the prototype design since the transfer speed between the radio interface module and the MAC module might be a bottleneck. However, increasing the transfer clock rate between the radio interface module and the MAC module should solve this problem.

In the future, the radio module might be replaced with another type of radio sub-systems, such as Bluetooth system [18]. This requires the reconfiguration of the radio interface module.

3.5 Design Considerations

Due to the higher transfer clock rate between the radio module and the radio interface, the electromagnetic compatibility (EMC) design rules must be followed [15]. For example, crosstalk, power-supply noise, and transmission line reflections should be taken into account. Several separate PCB ground planes are used to reduce the problem of crosstalk. Power-supply planes are also used to reduce the power-supply noise. Furthermore, several decoupling capacitors are located near every component that is expected to use large amount of electric current. For example, 33 capacitors are located near the FPGA package according to recommendations of the manufacturer [16]. Signal reflections are reduced using termination resistors for critical signals. EMC design rules are taken into account also in the other parts of the PCB design.

Testing of the first prototype revealed that more test points are needed in PCB to make testing and debugging easier. Therefore, several pin headers are added to the new prototype. For example, all the signals between DSP and FPGA can be measured with a logic analyser system [17]. Test points also enable the using of the chip-on-board emulation of the DSP.

4 CONCLUSIONS

TUTWLAN system has been developed to provide a wireless data-transfer service for office and home environments and for restricted public areas. The most important system requirement has been to achieve a reduced complexity system for supporting real-time, multimedia type of services.

The advanced prototype platform has been developed for providing a general-purpose wireless development platform. Besides the main TUTMAC protocol, by using this platform it is possible to test different types of communication protocols and systems, such as other WLANs and Bluetooth. Through test pins it is also possible to add other kinds of components to the advanced prototype, for example memory can be added near the FPGA that can consequently use this memory without the DSP. With this configuration, various processor models can be tested inside the FPGA.

Currently, the first network demonstrator containing two wireless platforms is under testing and verification. The second prototype platform is under construction and will be completed and operational during the spring 2000. The development continues by porting the VHDL codes of the first prototype to the improved platform. Next, the second prototype is tested in target operational environments. In the future, different MAC protocols and embedded applications generated using SDL are tested in the platform. It is expected that with the new prototype platform, wireless systems with quite high data rates can be implemented.

REFERENCES

[1] Nathan J. Muller, "Wireless Data Networking", Norwood, Artech House, 1995.

[2] IEEE P802.11 D5.0, IEEE Standard for local and metropolitan area networks: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE Standards Department, 19.7.1996.

[3] ETS 300 652, Radio Equipment and Systems (RES); High Performance Radio Local Area Network (HIPERLAN), Type 1; Functional specification, October 1996.

[4] M. Hännikäinen, J. Knuutila, A. Letonsaari, T. Hämäläinen, J. Jokela, J. Ala-Laurila, J. Saarinen, "TUTMAC: A Medium Access Control Protocol for A New Multimedia Wireless Local Area Network", Proceedings of the Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'98), USA, September 8–11, 1998, pp. 592-596.

[5] M. Kari, M. Hännikäinen, T. Hämäläinen, J. Knuutila, J. Saarinen, "Configurable Platform for a Wireless Multimedia Local Area Network", Proceedings of the 5th International Workshop on Mobile Multimedia Communications (MoMuC'98), Germany, October 12-14, 1998, pp. 301-306.

[6] CCITT Specification and description language, ITU-T recommendation Z.100, March 1993.

[7] Xilinx, "XC4000E and XC4000X Series Field Programmable Gate Arrays", Product Specification, 1997.

[8] Analog Devices Inc., "ADSP-2106X SHARC User's Manual", 1997.

[9] Harris Semiconductor, "Wireless Handbook", 1997

[10] Xilinx, "The Programmable Logic Data Book 1999", 1999.

[11] Telelogic AB, "Telelogic Tau 3.6 User's Manual", 1999.

[12] White Electronic Designs Corporation, "EDI8G32512V Data Sheet", 1999.

[13] Intersil, "2.4GHz 11Mbs MACless DSSS Radio HWB1151 User's Guide", 1999.

[14] Bruce Schneier, "Applied Cryptography: Protocols, Algorithms and Source Code in C", 2nd edition, John Wiley & Sons, Inc., 1996.

[15] Henry W. Ott, "Noise Reduction Techniques in Electronic Systems", John Wiley & Sons, 1988.

[16] Xilinx, "Powering Virtex FPGAs", 1999.

[17] Agilent Technologies, "HP 16600A and 16700A Series Logic Analysis System Mainframes", 1999.

[18] Specification of the Bluetooth System – Core, v1.0 B, December 1999.

Publication [P7]

Saari T., Hännikäinen M., Hämäläinen T., “Hardware Acceleration of Wireless LAN MAC Functions”, IEEE International Workshop on Design and Diagnostics of Electronic Circuits and Systems (DDECS 2002), pp. 398-401, Brno, Czech Republic, April 17-19, 2002.

HARDWARE ACCELERATION OF WIRELESS LAN MAC FUNCTIONS

Tuomo Saari, Marko Hännikäinen, Timo Hämäläinen
Tampere University of Technology, Institute of Digital and Computer Systems
Korkeakoulunkatu 1, FIN-33720 Tampere, Finland
tuomo.saari@tut.fi, marko.hannikainen@tut.fi, timo.d.hamalainen@tut.fi

Abstract. *This paper presents the hardware acceleration of the most demanding Medium Access Control (MAC) protocol functions for a proprietary Wireless LAN system (TUTWLAN). Developed functions contain channel synchronization, forward error correction coding, encryption, and CRC-sum calculation. Implementation with an FPGA is shown to yield a powerful platform without the need of high-performance processor.*

1 Introduction

A proprietary Wireless Local Area Network (WLAN) called TUTWLAN [1] has been implemented in Institute of Digital and Computer Systems at Tampere University of Technology for developing WLAN Quality of Service (QoS) and security. TUTWLAN includes a Medium Access Control (MAC) protocol named TUTMAC [2] and a hardware platform. The demand of reliability, high capacity and security of WLANs requires high processing power. This paper presents how hardware acceleration is utilized in TUTWLAN to implement a powerful WLAN platform without a high-performance processor.

Figure 1 presents the architecture of the TUTWLAN platform. The host is a Windows NT PC containing several TUTWLAN specific software blocks, such as TUTWLAN device driver, protocol drivers, and management software [3]. The radio sub-system is IEEE 802.11b compatible 2.4 GHz 11Mbps MACless radio of type HWB1151 from Intersil [4]. The platform board contains *Radio Interface* (RI) and *Host Interface* (HI) implemented on a single Xilinx XCV800 FPGA chip [5] and the MAC protocol implemented by a digital signal processor ADSP 21060L from Analog Devices [6] running at 40MHz.

The hardware acceleration presented in this paper is performed on the FPGA together with the RI module. It contains functions for channel synchronization, Forward Error Correction (FEC) coding, encryption, and CRC-sum calculation. Before presenting the implementation and performance analysis, the requirements for the functions are first discussed.

2 Requirements for the Hardware (HW) MAC Functions

The RI separates the MAC processor and radio module in a way that allows one or both to be changed. The basic task of the RI is to connect the signal lines and buses between the MAC processor and the radio and synchronize data streams between the two modules.

In software implementation on the DSP, the most demanding data-processing functions of TUTMAC are data encryption, FEC coding, fragmentation, fast retransmissions, CRC-calculation, and synchronization to the TDMA frame structure. For improving the system performance, the RI configuration now implements the following MAC related functions on RI hardware:

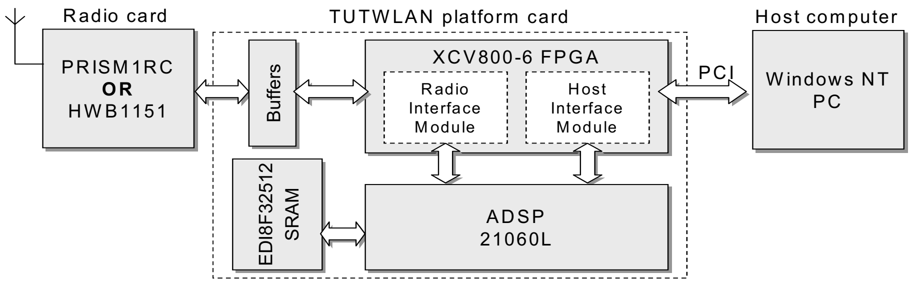


Figure 1. The components of the demonstrator.

- **Adapting the TUTMAC protocol** with the radio sub-system by creating a simple and efficient functional interface for the protocol.
- **Link level synchronization** is required by TUTMAC for the TDMA framing. For this, the RI contains a special counter register for timing the phase of the TDMA cycles.
- **Encryption of data** is required to ensure privacy. The default encryption algorithm for TUTMAC is IWEP [7] that has been developed for the TUTWLAN.
- **FEC coding and CRC calculation** are required for protecting the data against transmission errors.

3 Implementation of RI Acceleration Functions

The structure of the RI is shown in Figure 2. First, the design is divided into *Transmission* (TX) and *Reception* (RX) portions. Both parts consist of data processing and control blocks. Furthermore, *Control Register, Synchronization, Beacon Counter, Status Register and Baseband Control* blocks have been implemented. For the communication between the MAC processor and the RI two link ports (*Link Port 0* and *Link Port 1*) and two serial ports (*SPORT0* and *SPORT1*) of the processor are used. The signals between the radio and the RI include serial type control signals and Rx and Tx lines.

Frame Transmission - The data flow through the transmission part of the RI block can be seen in Figure 2. The *TX_serial_parallel* block works as a DSP interface that generates the TFS0 (frame) and TCLK0 (clock) signals for the DSP. Furthermore, this block receives the data from SPORT0 in serial and transforms it to 64-bit parallel form for the encryption. A 32-bit CRC is counted for the data in the *IWEP CRC* block before encryption so that the receiver can check if the right encryption key is used. A 32-bit CRC is also calculated over the data field of a transmitted frame in the *D_CRC* block.

The encryption is performed in *TX_IWEP* block. The first 8 bytes of data frames are reserved for TUTMAC frame header and are not encrypted. The header propagates through the *TX_DFF* block that is a 64 bits wide data flip-flop. The header is loaded from the *TX_DFF* to the *64bit_serial* and *72bit_serial* blocks. An 8-bit CRC is then calculated for the TUTMAC frame header in the *H_CRC* block. *FEC* coding is used for the header and the header CRC to correct possible transfer errors. A simple triple transmission with interleaving is used. *TX_Buffer* is the block that feeds the data into the radio.

Transmission Control - The *Control Register* block contains signals for the Transmission and Synchronization blocks. The main control signals of the register are the length of the TDMA cycle (*Beacon_Interval*), the used encryption key, the next transmission start time (*Tx_Time*), and the length of the next transmitted data frame. TUTWLAN base stations transmit beacon frames to synchronize the TDMA cycles. In

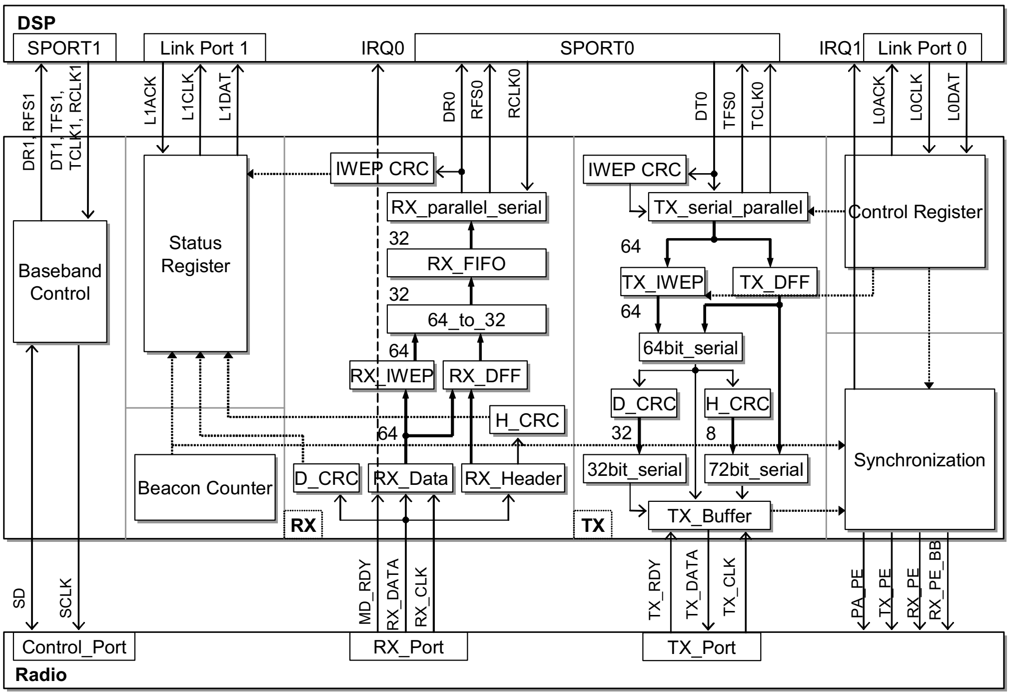


Figure 2. The structure of the radio interface.

every station there is a special counter, *Beacon Counter* that counts how many microseconds have elapsed from the beginning of the current TDMA cycle. The *Synchronization* block controls the power enable signals of the radio module. When the value of the *Beacon Counter* equals *Tx_Time*, the *Synchronization* block initializes transmission by setting the radio module in transmission mode.

Frame Reception - The FEC coded header and the header CRC are received and decoded in the *RX_Header* block. The header CRC enables to verify a correctly decoded header. Then the header is loaded through *RX_DFF* into the *64_to_32* block, where the data is converted from 64 to 32 bits wide for the *RX_FIFO*. This data width conversion is needed because the *SPORT0* receives 32 bits at a time. The depth of the *RX_FIFO* is 16 and its purpose is to buffer the reception data in case the DSP is busy. From *RX_FIFO* header is loaded into the 32 bits wide *RX_parallel_serial* block that feeds it to *SPORT0* in serial.

The TUTMAC frame data is loaded into the *RX_DATA* block and into the *D_CRC* block that checks if the data has been corrupted during the transmission. From the *RX_DATA* block the data is loaded in the *64_to_32* block through *RX_IWEP* or *RX_DFF*, depending whether the data has been encrypted or not. When the TUTMAC frame data is loaded into the *SPORT0 f*rom the *RX_parallel_serial* block, it is also loaded into the *IWEP CRC* block that detects if the right encryption key has been used in the *RX_IWEP*.

Status Register – The *Status Register* is a 48 bit wide register used for collecting information about the received frame. The field *Data_Length* tells the length of the data, *Beacon* indicates if the frame is a beacon frame, and *Encrypted* indicates if the frame data is encrypted or not. These fields are extracted from the header of the received frame. The fields *Data_Error*, *Enc_Error* and *Header_Error* indicate possible errors detected in *D_CRC*, *IWEP CRC* and *H_CRC* blocks accordingly.

Table 1. The processing times in nanoseconds for header and data in the transmission part of the RI.

Clock rate		Delay
40 MHz	data	2025
	header	3850
63 MHz	data	1285.7
	header	3266

4 Testing and Performance

The RI occupies 1656 out of 9408 slices (17%) and 62 out of 166 user I/O -pins on the Virtex FPGA. The equivalent gate count is 34,437. The theoretical propagation delays in the RI for the header and data are calculated separately in Table 1. The waiting times due to the difference between the speeds of the RI and radio are not included, but the delay is from the time the first bit is loaded into the RI to the time the first bit is ready to be read by the radio. The speed of the SPORT0 is currently configured to be 10 MHz. For the RI, the Xilinx Alliance 2.1i synthesis tool [8] gives 63 MHz as an estimate of a maximum clock. The actual operating frequency of the chip is 40 MHz.

The speed of the reception part of the RI is evaluated by calculating the time from when the header or 64 bits of data have been loaded into the RI from the radio to the time when the first bit is ready to be read by the DSP through SPORT0. This processing time is 100 ns for both header and data when RI operates at 40 MHz.

5 Conclusions

Hardware acceleration of WLAN MAC protocol functions has been implemented for TUTWLAN. The purpose has been to develop a platform supporting real-time secure and reliable communications without the need for a high performance MAC processor.

The implemented acceleration module fulfils the current requirements for the TUTWLAN platform, providing real-time operation with 11 Mbit/s radio link with 40 MHz FPGA clock rate. Furthermore, the design can be scaled to higher clock rates and communication port speeds. The area and I/O capabilities of the exploited FPGA are also sufficient for the design, and allow the future implementation of new MAC functions.

References

[1] Tikkanen, K., Hännikäinen, M., Hämäläinen, T., Saarinen, J.: Advanced Prototype Platform for a Wireless Multimedia Local Area Network. In: Proceedings of European Signal Processing Conference (EUSIPCO’2000), 2000, pp. 2309-2312.

[2] Takko, A., Hännikäinen, M., Knuutila, J., Hämäläinen, T., Saarinen, J.: Embedding SDL implemented Protocols into DSP. In: Proceedings of International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES'2000), 2000, pp. 48-56.

[3] Kari, M., Hännikäinen, M., Hämäläinen, T., Knuutila, J., Saarinen, J.: Configurable Platform for a Wireless Multimedia Local Area Network. In: Proceedings of the 5th International Workshop on Mobile Multimedia Communications (MoMuC'98), 1998, pp. 301-306.

[4] Intersil: 2.4GHz 11Mbs MACless DSSS Radio HWB1151 User’s Guide, 1999.

[5] Xilinx: The Programmable Logic Data Book 1999, 1999.

[6] Analog Devices Inc.: ADSP-2106X SHARC User’s Manual, 1997.

[7] Hämäläinen, P., Hännikäinen, M., Hämäläinen, T., Saarinen, T.: Hardware Implementation of the Improved WEP and RC4 Encryption Algorithms for Wireless Terminals. In: Proceedings of European Signal Processing Conference (EUSIPCO'2000), 2000, Finland, pp. 2289-2292.

[8] Alliance Series 2.1I Software Documentation: http://toolbox.xilinx.com/docsan/2_1i/, 2002.

Publication [P8]

Hännikäinen M., Vanhatupa T., Lemiläinen J., Hämäläinen T., Saarinen J., “Windows NT Software Design and Implementation for a Wireless LAN Base Station”, ACM International Workshop on Wireless Mobile Multimedia (WoWMoM 1999), pp. 2-9, Seattle, USA, August 20, 1999.

Windows NT Software Design and Implementation for a Wireless LAN Base Station

Marko Hännikäinen, Timo Vanhatupa,
Timo Hämäläinen, Jukka Saarinen
Tampere University of Technology, Signal Processing Lab.
Hermiankatu 12 C, FIN-33720 TAMPERE, Finland
Tel. +358-3-365 2111
markoh@cs.tut.fi

Jussi Lemiläinen
Nokia Wireless Business Communications
P.O.Box 88 (Tieteenkatu 1)
FIN-33721 TAMPERE, Finland
Tel: +358-50-521 0607
jussi.lemilainen@nokia.com

ABSTRACT

This paper introduces the architecture of a base station designed for a new Wireless Local Area Network (WLAN). The emphasis is on the software but also backgrounds for the design choices made are discussed. The main design objective has been to develop a simple and low-cost system while maintaining suitability for a large variety of applications, ranging from wireless sensors to multimedia workstations. A base station is implemented by attaching a custom WLAN adapter card into a PC. The network adapter consists of a custom wireless Medium Access Control (MAC) protocol and a standard low-power radio subsystem. The PC hosts other wireless network related functions, such as management of a wireless cell and bridging of data between the wireless network and a wired LAN. Currently, a demonstrator network is being implemented. Components implemented for a base station contain the network adapter card, a Windows NT device driver, and a management application that is a user mode program needed for configuring and managing the network interface hardware and the MAC protocol. In addition, a Windows NT protocol driver has been developed for implementing the data exchange between the device driver and the management application.

Keywords

Wireless LAN, Demonstrator platform, Windows NT

1. INTRODUCTION

Wireless Local Area Networks (WLAN) are currently emerging as a mature technology by having an increasing range of products and vendors on the market. Also, wireless LAN standardisation has recently proceeded into available specifications, such as IEEE 802.11 [9] and ETSI High Performance Radio Local Area Network (HIPERLAN) [5], which has brought interoperability to the previous rather divergent set of product alternatives.

WLAN combines the freedom of movement and simple network management with numerous utilisation possibilities of local networks. WLAN is a suitable alternative for network connectivity in limited geographical areas, such as offices and semi-public places like airport lounges and libraries, where wiring is inconvenient because of the constantly changing network topology and the movement of users. Furthermore, a low-cost wireless link can be utilised in simpler applications as a replacement for a cable connection.

Despite of its advantages and the availability of products, the wireless connectivity has not been able to successfully compete with traditional wired counterparts. Main reasons for this lack of popularity have been high costs, low performance, and the previously missing standardisation. Also, the new IEEE 802.11 and HIPERLAN standards are targeted for general applications with high performance, mobility, and multi-user characteristics in mind, which can make them too complex and expensive alternatives for simple wireless appliances. Consequently, there are a number of ongoing efforts for promoting wireless connectivity also in this field; Bluetooth [16] and HomeRF [18] being among the most important.

Traditional wired LANs rely on the reliability and stability of the physical link. However, these assumptions do not generally apply in a wireless environment, where the most significant constraints are the limited bandwidth, high error probability, and insecurity of the medium together with the movement of terminals. In order to meet the interoperability requirements with existing higher-layer protocols and applications, a wireless link must appear similar to the wired counterpart. It should provide a reliable data-transfer service and support for other Quality of Service (QoS) related parameters with compatible service interfaces.

Thus, functionality for managing the medium and terminals must be included into wireless link layer protocols. At the same time, most of these additional control and management functions can be automated and thereby hidden from the above layers by the link level implementation. On the other hand, the management should be configured according to the characteristics of the operational environment, which states the need for another management entity requiring an access to link protocols. The management entity must also be provided with adequate operational information as feedback, for example transfer error statistics and signal to noise levels, and current network topology.

A new low-cost and reduced complexity wireless LAN system is being designed and implemented in the Signal Processing Laboratory at Tampere University of Technology (TUT). This *TUTWLAN* is targeted for indoor, on-premises, home, and office environments with a wide range of target applications from multimedia laptops to wireless sensors. The system utilises a low-power radio transceiver in the 2.4 GHz ISM band (Industrial, Scientific, Medical) for the physical layer connectivity. The work consists of a proposed wireless Medium Access Control (MAC) protocol called *TUTMAC*, demonstrator platform hardware, and a set of software modules, which are used for implementing a demonstrator network. The hardware platform is implemented as a Network Interface Card (NIC) that can be attached to a host computer (PC). Software parts contain a Windows NT application that controls and manages the platform, TUTMAC protocol, and consequently the wireless network. In addition, the required device and protocol drivers for transferring data between the NIC and user applications have been implemented.

This paper presents the structure of a TUTWLAN demonstrator base station emphasising on the software architecture, especially to the management application. The paper is organised as follows. In the next section, the topology of TUTWLAN, functionality of the TUTMAC protocol, and the architecture of the demonstrator platform hardware are briefly introduced. Next, an overview of the Windows NT networking is given followed by the description of each software module of the base station. Finally, the essential elements of the work are summarised and concluded shortly.

2. TUTWLAN OVERVIEW

TUTWLAN has centrally controlled network architecture, as illustrated in Figure 1. A *base station* is the central point of control and management inside its coverage area referred to as a wireless *cell*. As most of the processing requirements are centralised into the base station, *portable terminals* can be simpler units with varying levels of intelligence.

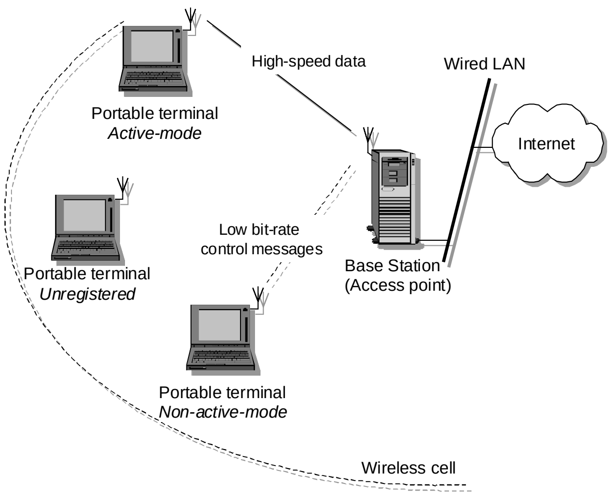


Figure 1. TUTWLAN architecture.

All portable terminals must first register with the base station in order to authenticate their identity and to exchange network management related information. Registered terminals within the cell area communicate via the traffic forwarding base station. A registered portable terminal that has an active data-transfer connection resides in an active mode. The terminal can also switch to non-active mode if there exists no user data to be transmitted. A non-active terminal remains synchronised to the network and can exchange low-bandwidth network management messages, while being able to minimise its power consumption. Data transfer can continue without a new registration after a fast channel reservation procedure.

Since WLAN can be utilised both as an alternative and as an extension to a wired LAN, the wireless system must also be compatible with an exiting LAN infrastructure. The simplest way to realise this is through a base station containing interfaces for both the wireless and wired networks and bridging functionality for forwarding data between the two networks.

2.1 Demonstrator Platform

The block architecture of the demonstrator platform is depicted in Figure 2. The platform consists of the *radio subsystem*, *radio interface*, *MAC*, and *host interface* modules. The card can be attached to a *host computer* via the host interface using the PCI-bus (Peripheral Component Interconnect.

The hardware of the adapter is assembled utilising off-the-shelf modules. The host and the radio interfaces are implemented as Field Programmable Gate Array (FPGA) circuits, which offers full re-configurability. The programming of the logic has been realised using Very high-speed integrated circuit Hardware Description Language (VHDL) that is a convenient tool for high-level description of hardware functions. The MAC module consists of a Digital Signal Processor (DSP), into which the TUTMAC protocol software module is embedded. The processor chosen for the module is ADSP-21062 (Analog Devices). It contains a large 256 kB on-chip memory, distinct I/O-processor, and 40 MIPS processing power, which allows experimenting with various design alternatives and system configurations. High processing capacity is needed for supporting several real-time data-transfer requirements, such as security related encryption algorithms, Forward Error Correction (FEC) coding, and traffic scheduling function in the base station. The TUTMAC protocol is implemented in a separate development environment using Specification and Description Language (SDL) [3]. SDL is a formal specification language with high abstraction level providing a clear graphical presentation of the system. The formality allows the system to be verified and simulated in system level during the development. Furthermore, the SDL model of the TUTMAC protocol is automatically converted into C-language and ported into the processor platform.

The radio subsystem module of the demonstrator platform utilises the Direct Sequence Spread Spectrum (DSSS) technique. Currently, 1 Mbit/s link speed with Differential Binary Phase Shift Keying (DBPSK) carrier modulation is used. Other types of physical transmission channels could also be used by reconfiguring the radio interface. The radio subsystem of the demonstrator platform is PRISM1RC-EVAL manufactured by Harris Semiconductors [6]. The radio is implemented meeting the DSSS physical layer specification of the IEEE 802.11 wireless LAN standard. This radio module has also additional capabilities that are especially useful for testing purposes. More information about the TUTWLAN platform HW architecture can be found from [10].

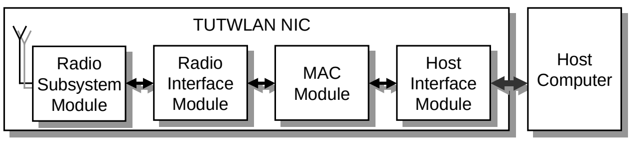


Figure 2. Demonstrator platform hardware modules.

3. WINDOWS NT NETWORKING

In the host computer of the TUTWLAN demonstrator Windows NT v4.0 operating system is used. The I/O system of NT contains a build-in networking functionality, such as device drivers (also referred to as NIC drivers) accessing network adapters and transport drivers implementing networking protocols. Other important architectural components of the system are networking Application Programming Interfaces (API) and interface layers providing uniform internal interfaces between various protocol layers. The most important are Network Driver Interface Specification (NDIS) and Transport Driver Interface (TDI). The overall networking architecture, with various types of layer interfaces, is presented in Figure 3.

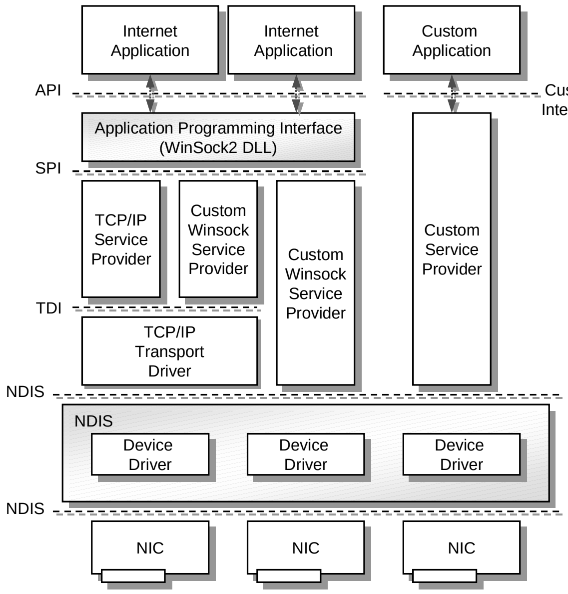


Figure 3. Windows NT networking architecture.

Windows NT provides several APIs for network applications, NetBIOS and Windows Sockets (WinSock 2 DLL) being the most important. WinSock 2 API is a Dynamic Link Library (DLL) that separates the implementation of underlying protocols from the actual client application, thus providing a high degree of portability. Several network protocols, implemented as transport drivers, can be used via WinSock. Service providers also seen in Figure 3 offer a Service Provider Interface (SPI) to the WinSock 2 DLL while being connected to lower protocol layers via the TDI or NDIS interface.

Via TDI several network protocol drivers can be connected together, for example existing transport drivers can be exploited in this way for new services. In the upper interface, a transport protocol driver may implement an application specific interface (through TDI actually) for custom applications.

NDIS is a standardised interface that allows multiple network adapters and multiple network protocols from various vendors to coexist. On the other hand, NDIS is more than an interface; it is an operational environment for network drivers. NDIS specifies and implements the most common functions for all drivers and offers these via a standardised interface to network driver programmers. Consequently, only the hardware and protocol specific functions have to be implemented separately. At the same time, the available set of functions is tied, since NDIS does not provide any proprietary way to access a wrapped driver through NDIS or by passing it.

4. BASE STATION ARCHITECTURE

The complete TUTWLAN base station hardware and software architecture is presented in Figure 4. The main functions of the station are:

- ❑ Controlling of the radio channel access (MAC protocol),
- ❑ Forwarding and processing of frames inside the cell,
- ❑ Bridging of frames between the wireless cell and a wired network,
- ❑ Managing of a wireless cell and registered terminals,
- ❑ Exchanging of protocol messages with other base stations for supporting mobility, and
- ❑ Providing support for standard network management protocols, such as Simple Network Management Protocol (SNMP).

These functions are assigned to the various functional blocks of the architecture. Some tasks are also shared or hierarchically divided between separate modules. The shaded blocks in Figure 4 refer to the modules being part of the TUTWLAN development project. The user mode portion of the implemented system has been divided into two parts: the actual management application that contains the user interface, and the management API that is implemented as a DLL and correspondingly named as *TUTDLL*. The management service provider is implemented as an NDIS protocol driver and named as TUTPROD and similarly the NDIS device driver for the TUTWLAN NIC is called TUTNICD. The *access point* module refers to bridging functionality. The purpose is to implement the module as an NDIS intermediate driver.

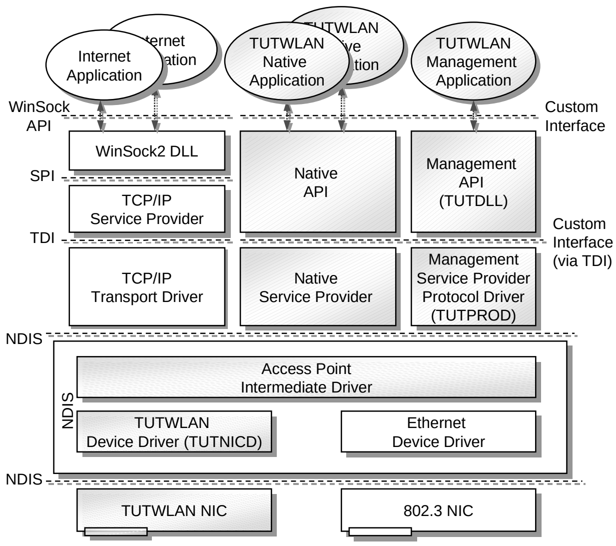


Figure 4. TUTWLAN base station architecture.

5. TUTMAC PROTOCOL

TUTMAC protocol is responsible for controlling the access to the shared radio channel. The protocol consists of two entities, as the channel is accessed by exchanging control messages between a base station and a portable terminal. The channel access is based on dynamic reservation Time Division Multiple Access (TDMA).

The protocol destines for supporting negotiated QoS parameters for the data transfer. The link level communication is connection oriented with point-to-point connections, which makes the separation of various traffic classes more efficient. In addition, the TUTMAC protocol provides functionality for managing the special challenges of the wireless medium and moving terminals.

The TUTMAC protocol architecture together with the most important functions is illustrated in Figure 5. The architecture consists of interfaces to other protocol layers and entities for the data-processing and protocol management. In addition, the Management Information Base (MIB) is referred to as a separate module. For simpler design, both the portable terminal and the base station configurations utilise the same functional architecture. A portable terminal implementation consists of a reduced set of functionality compared to a base station.

The protocol supports two types of data transfer users: a single network user and several native user applications. The network user *Service Access Point (SAP)* offers an access for legacy network protocols (e.g. TCP/IP). *Native-SAPs (N-SAP)* provide an extended interface for each custom TUTWLAN application. However, the N-SAP interface or native applications have not been currently implemented in the TUTWLAN demonstrator.

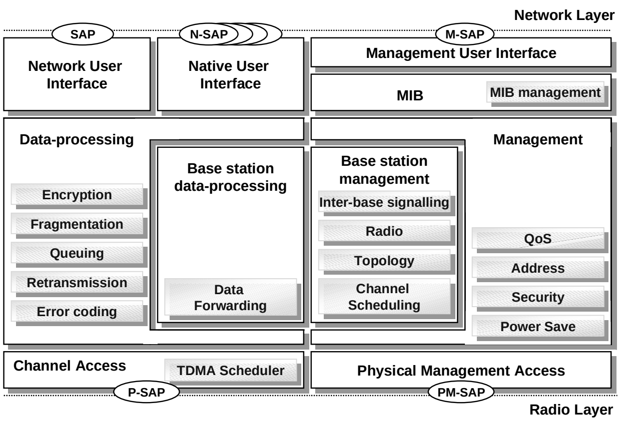


Figure 5. TUTMAC protocol architecture.

For QoS, all user originated MAC Service Data Units (MSDU) are processed according to the *Service Type* class of the TUTMAC protocol. This class contains a set of parameters defining the transfer service for each MSDU according to priority, ageing time and error protection. During the connection establishment procedure, the parameters for reserved wireless data channels are negotiated using a *Connection Type* class parameter. This class defines the characteristics for the actual connection throughput and duration. The third QoS related parameter class of the protocol is the *Security Type* that defines the utilised level of confidentiality for the transfer.

The data transfer functionality is controlled by the management side of the protocol that contains QoS definition, addressing, security, and power save related management functions. In addition, a base station is responsible for managing the changing topology and radio link, as well as scheduling the TDMA channels for reserved connections. A base station also exchanges messages with other base stations being connected to the same backbone wired LAN segment.

The protocol management functionality must be initialised and configured from the management application. For this purpose, another *Management-SAP (M-SAP)* interface for controlling the TUTMAC protocol and the NIC hardware is available. The M-SAP control signalling contains values for various configurable parameters stored into TUTMAC MIB or in the control registers of the radio subsystem. MIB holds also a number of traffic counters, of which values can be periodically transferred to the management application for continuous observation of operations and protocol states. Furthermore, the initialisation of networking operations is managed through the M-SAP interface. Examples of these are the registering to and leaving of a wireless network (by a portable terminal), creating and dissolving a network cell (by a base station), and changing the radio channel frequency of the radio subsystem. M-SAP supports three types of protocol service primitives. The management application may access variable values by a *request* primitive, while the requested value is returned by TUTMAC with *confirm* primitive. *Indication* primitives on the other hand are issued independently by TUTMAC when required. More information about the TUTMAC protocol and TUTWLAN can be found in [7] and [15].

6. DEVICE AND PROTOCOL DRIVERS

TUTNICD controls the TUTWLAN NIC and together with TUTPROD manages the data exchange between a host and the host interface. TUTNICD is wrapped inside NDIS, as can be seen in Figure 4. When implementing a NIC driver, the data exchange with a network protocol driver can be easily implemented through NDIS. However, NDIS does not allow a user mode application to freely exchange data with a NIC driver, which is required by the TUTWLAN management application. To overcome this restriction the control information can be multiplexed with the network data. This can be done before the TCP/IP protocol driver, in which case the management application can use the Winsock API similarly to other network applications. As a second alternative, another system specific network protocol driver can be used for accessing the device driver via the NDIS interface. As mentioned, this custom protocol driver can implement an application specific interface for the TUTMAC management application. In the TUTWLAN demonstrator, however, existing NDIS functions are utilised as carriers for the custom information.

For exchanging request-confirm primitives between the management application and the device driver, the demonstrator utilises the *MiniportQueryInformation* function of NDIS. Although this function is designed for only acquiring control data from a NIC driver, it is possible to use it in two-way communication, since NDIS (v4) uses *Direct I/O* [4] for passing function parameters. This means that the caller and the called driver share a common memory space, which is reserved by the caller to be used by the called driver for the return data. Since in this case the TUTWLAN management application reserves the memory, it is thus protected and can be initialised with the input data (request primitive) that can consequently be accessed by TUTNICD. The driver may then use the same memory for the return values (confirm primitive). As NDIS does not allow a NIC driver to accept new function calls until the last one is finished, no sequence numbering is required for associating a request primitive to the correct confirm. Any new primitive calls are buffered by NDIS while waiting the previous one to complete. The call does not either prevent the operation of the NIC driver, as NDIS allows the MiniportQueryInformation to return immediately while leaving the call pending until a reply is received from TUTMAC. The call can then be asynchronously completed by the *NdisMQueryInformationComplete* function.

The transfer of M-SAP indication primitives requires also special solutions. These primitives are issued by TUTMAC at occasional times to the management application. However, in Windows NT a NIC driver is not allowed to call user-mode application code. Therefore, TUTNICD utilises another build-in NDIS function for carrying the control information to the opposite direction, the *NdisIndicateStatus*. When this function is called, NDIS successively calls the *ProtocolStatus* function that carries the information to the TUTPROD driver. The TUTDLL contains a separate thread that calls TUTPROD in a loop for retrieving indications. The call is actually made before an indication arrives from the NIC driver and thus the open call can return immediately with the indication data as a parameter. A message sequence chart presenting the exchange of control information is shown in Figure 6. Detailed information of the implementation of TUTNICD and TUTPROD can be found from [8].

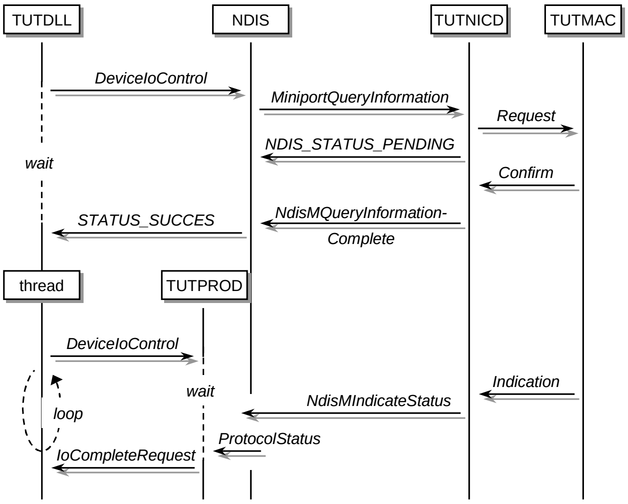


Figure 6. Control information exchange in M-SAP.

7. ACCESS POINT MODULE

The access point module has been specified as a kernel-mode intermediate NDIS driver, which can be seen in Figure 4. The driver forwards frames between a TUTWLAN cell and an external wired LAN (Ethernet in case of the demonstrator network), thus providing bridging functionality. The module is currently in the implementation phase and the specified requirements for the module and various implementation alternatives are given in the following.

An access point interconnects two networks that usually have different link capacities. Therefore, the module should provide enough buffering capacity for avoiding congestion, and also include functionality for recovering from overloaded data buffers. The wireless link is the most probable bottleneck in the data path. However, in a home environment the base station may be connected to an external LAN via an analog modem connection, which switch the location of the bottleneck.

For more than copying of data payloads, an access point has to adapt the MAC-level control signalling between the two networks. In case of an Ethernet frame, only a single *packet type* field is used with destination and source MAC-level addresses. Furthermore, an access point module must be capable of screening the traffic in order to prevent any unnecessary loading of the wireless medium. Therefore, the module should be aware of the MAC addresses of the terminals being currently registered with the base station and forward only frames destined to these stations (and broadcast frames) over the air interface. All data between portable terminals belonging to the same cell is forwarded by the TUTMAC protocol directly, thus not reaching the access point module.

In order to minimise the amount of signalling overhead, and also for security related reasons, the TUTMAC protocol utilises a custom addressing scheme for network terminals. In this approach, the 48-bit IEEE MAC addresses are supported by the TUTMAC protocol at the network layer interface, but TUTMAC converts these into temporary TUTWLAN terminal identifiers (ID). One octet size ID value is assigned during the registration of a portable terminal and only this value is used for addressing a TUTWLAN terminal over the air interface.

The use of shorter addresses must be transparent for an interconnection to another LAN. The system provides two alternatives for the address conversion depending on the size of the external LAN. First, the TUTMAC may assign temporary ID values also for the wired terminals that are connected to the same wired LAN as the base station. Therefore, the mapping between an IEEE 48-bit address and a TUTWLAN ID is managed by the TUTMAC protocol. In this solution, the IEEE MAC addresses of the fixed network are dynamically registered by TUTMAC. Therefore, the address learning procedure resembles the spanning tree routing that is a widely utilised technique in conventional bridges [17]. However, the TUTWLAN architecture is more limited as a portable terminal can exchange data frames only with a single base station at a time. Therefore, the terminal cannot operate as a wireless bridge.

The second alternative for the address management is to convert the MAC-level addresses inside the access point module according to network layer IP addresses. In practise, this means that the access point module must be capable of performing Address Resolution Protocol (ARP) for determining the correct MAC level address according to the destination IP address. When this solution is used, the base station may have multiple IP addresses for its single MAC address. The base station also replies to ARP requests on behalf of a registered portable terminal with proxy ARP messages. Consequently, wired network stations use the MAC address of the base station when transmitting to the IP address of any of the portable terminals being currently registered into the wireless cell. When the base station receives an IP datagram, the access point module checks from the destination IP address whether the packet should be given to the local TCP/IP protocol entity or forwarded into the wireless cell.

8. MANAGEMENT APPLICATION

The TUTWLAN management application is a Windows NT user mode program that accesses the NIC via TUTPROD and TUTNICD. The application has several functional purposes, as has been presented in the previous sections. Furthermore, during the development and testing phase of the demonstrator platform, the management application has been used for debugging and testing. For example, the application has capabilities for directly accessing data and control registers of the TUTWLAN NIC, thus

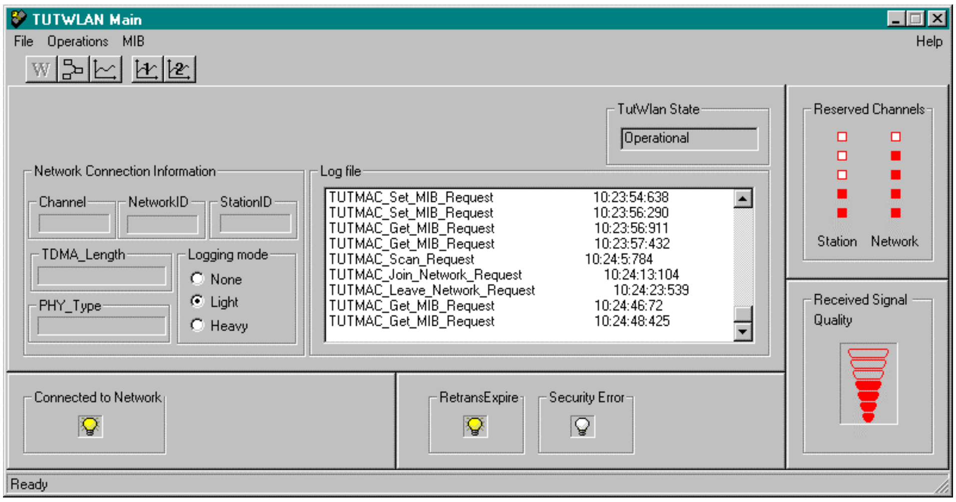


Figure 7. Main window of the management application.

allowing the verifying of register values. Similarly, the application can generate valid service primitives for the TUTMAC protocol consequently appearing as a network protocol. In addition, the initialising of the MAC processor in the NIC is done by the application as a user may choose a pre-complied boot-file to be downloaded into the processor.

At the moment, the support for SNMP has not been included into the demonstrator as the system is in the implementation phase. There are several possible locations for an SNMP agent, which task is to manage a standard MIB II database parallel with the custom TUTWLAN MIB. Both MIBs can be accessed over the wired network using SNMP client. The custom TUTWLAN MIB available for the SNMP client will also include a copy of the TUTMAC protocol MIB. A suitable location for the agent software in the demonstrator base station is in conjunction with the TUTDLL. This allows the agent to access the TUTWLAN MIB and the radio control registers in a similar way as the management application when replying to the requests received by SNMP messages.

In order to avoid any unnecessary message exchange between the TUTMAC protocol and the management application, the required system attributes are retrieved from the TUTMAC MIB only when needed by the management application. This procedure refers to the exchange of request-confirm primitives through the M-SAP interface. In addition, the management application can subscribe any set of TUTMAC MIB variables to be transmitted automatically to the application when their values are updated by the TUTMAC protocol. In this case, new variable values are carried by TUTMAC indication primitives. In order to minimise the signalling, several MIB variable values can be read or written by a single primitive.

The TUTWLAN management application contains several types of windows for user interfaces. First, the *Main* window, presented in Figure 7, has been implemented for the initialisation of the most common operations of a base station. The Main window presents the main parameters of a cell, such as the current utilisation of data channels (reserved channels), number of registered terminals, and various cell parameters like the channel frequency, link speed, link type, and network ID. The averaged quality of the radio link is also available. A user can optionally view the exchanged TUTMAC primitives in a log window area and also save the log into a text file. There are two alternatives for logging accuracy. The light-mode shows only the names of the assigned TUTMAC service primitives together with the time of occurrence. In the heavy-mode, also the parameters of each primitive are saved.

Several network management dialogues are associated to the main window. Among these is the scanning of radio environment in order to discover other transmitters or error sources. Also, dialogues for the changing of power save attributes, configuring of the QoS related parameters of the TUTMAC protocol according to the operational environment, and creating and dissolving a TUTWLAN cell are included. Important interfaces for the security management are the assignment of portable terminal specific secret encryption keys and the creation of an access list according to the MAC layer addresses of portable terminals. In practice, the network access of a certain terminal can be denied or limited according to the time, location, or the type of terminal. As an example, the *create network* and the *configure service type* dialogues are presented in Figure 8. The create network window allows a user to start a TUTWLAN cell with suitable parameters. With the service type dialogue, the service class parameters are assigned for a certain service type number according to the operational environment (i.e. link quality) and type of traffic.

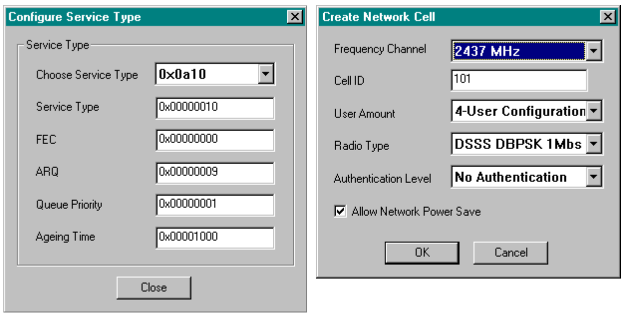


Figure 8. Create network and configure service type dialogues.

Some of the most critical warnings can also be presented in the main window. In the example interface of Figure 7, the warnings for expired retransmission attempts (meaning a lost network connections) and for security error are visible. The security warning may be caused by an intruder in the network.

The next window type of the TUTWLAN management application is used for the graphical presentation of operational parameters and variables. Any variables of the TUTMAC MIB, for example counter values for transmitted and received frames and retransmission attempts, can be subscribed from the TUTMAC management block and presented as a real-time statistical graph. The update frequency of the graph can be chosen by a user. An example of such *statistics* window is presented in Figure 9. The application also allows a user to save the graph and the corresponding data in plain text format, and reload it for a later presentation. In addition to traffic counters and other similar types of variables, the statistics interface is also suitable for event logging as the interface produces a clear notification of the occurred event and stores the time of the occurrence. The number of parallel statistics windows is not limited in the demonstrator base station and each window can present several MIB variables. However, the presentation of several constantly changing values with high refresh rate may affect the performance of the NIC driver and to the TUTMAC protocol.

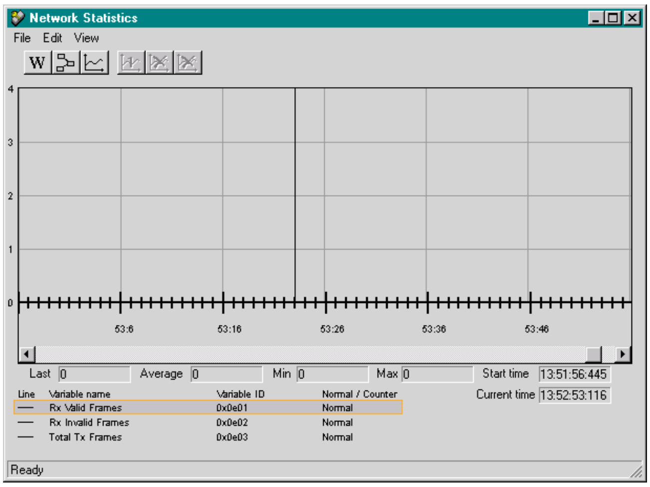


Figure 9. Statistics window.

The management application contains a *MIB browser* interface for the lowest level configuration access to the TUTMAC protocol and NIC hardware. Each TUTMAC MIB variable value and the control registers of the radio subsystem can be viewed and changed by the browser. TUTMAC MIB is divided into groups as can be seen from a browser view presented in Figure 10. Each group contains associated configuration parameters or traffic counter variables.

All TUTMAC MIB and the radio register variable values can be saved into a configuration profile file. The file can be later reloaded to the browser, and at the same time, the predefined values can be assigned for the TUTMAC MIB variables and for the radio registers. For the construction of a MIB profile, a *profile manager* dialogue has been included into the MIB browser. The profile manager interface is presented in Figure 11. The profile manager allows a user to easily select groups, subgroups, or single variables when building or changing a profile. Similarly, a profile can be used when starting a statistic window, as the variables to be presented are identified in the profile.

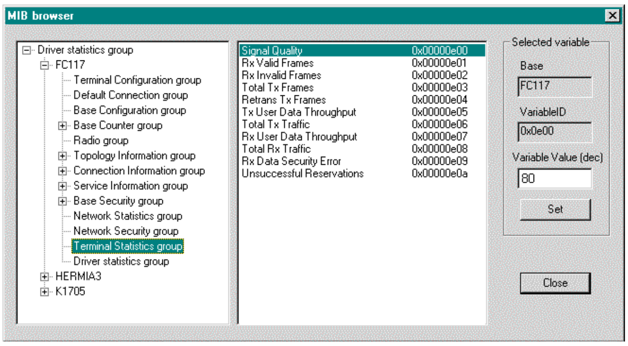


Figure 10. MIB browser interface.

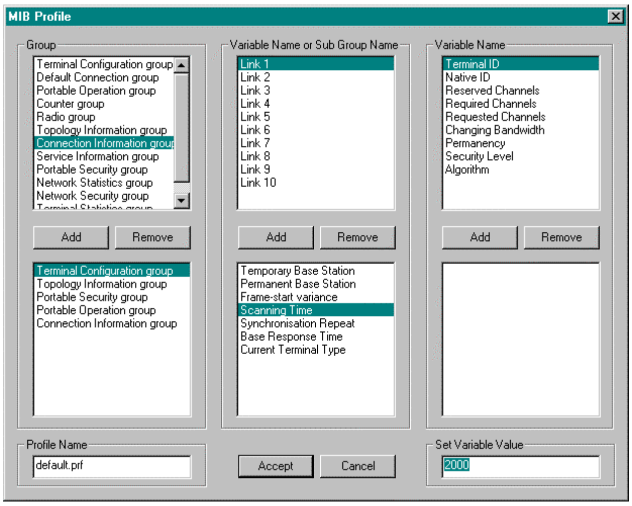


Figure 11. MIB profile manager.

9. CONCLUSIONS

The constantly changing characteristics of wireless links and the movement of terminals require additional management functionality from various protocol layers of a wireless terminal. Physical link should contain registers for power save control, link quality measurement, radio channel selection, etc. Similarly, a wireless MAC protocol should automate the mobility and error control as much as possible for maintaining compatibility with higher protocol layers and applications. For initialising and controlling the lower layers, a separate management entity is required. This entity must also provide a user interface for network management.

The TUTWLAN base station architecture has also resulted into more complicated functionality of the inter-layers between the management application and TUTMAC. A custom protocol driver has been implemented and available NDIS functions are utilised for piggybacking the control information as a parameter. For the interconnection between the wireless and a wired LAN, an intermediate NDIS driver has been designed for the bridging of frames.

The used software architecture is relatively simple to implement and as the standard NDIS interface is utilised, this solution maintains the interoperability and reliability of the system.

10. REFERENCES

[1] Bantz D. F., Bauchot F. J., Wireless LAN Design Alternatives, IEEE Network, March/April 1994, 43-53

[2] Bauchot F. J., Lanne F., IBM Wireless RF LAN design and architecture, IBM Systems Journal, Vol. 34, No. 3, 1995, 390-408

[3] CCITT Specification and description language, ITU-T recommendation Z.100 (03/93)

[4] Custer H., Inside Windows NT, Microsoft Press, 1992

[5] ETS 300 652, Radio Equipment and Systems (RES); High Performance Radio Local Area Network (HIPERLAN), Type 1; Functional specification, October 1996

[6] Harris Semiconductors, Wireless Handbook, 1997

[7] Hännikäinen M., Knuutila J., Letonsaari A., Hämäläinen T., Jokela J., Ala-Laurila J., Saarinen J., TUTMAC: A Medium Access Control Protocol for A New Multimedia Wireless Local Area Network, Proceedings of the Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'98), Boston, USA, September 8–11, 1998, 592-596

[8] Hännikäinen M., Vanhatupa T., Lemiläinen J., Hämäläinen T., Saarinen J., Design and Implementation of a Wireless LAN Interface Card Driver in Windows NT, Accepted to The International Conference on Telecommunications (ICT'99), Cheju, Korea, June 15–18, 1999

[9] IEEE P802.11 D5.0, IEEE Standard for local and metropolitan area networks: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE Standards Department, July 1996

[10] Kari M., Hännikäinen M., Hämäläinen T., Knuutila J., Saarinen J., Configurable Platform for a Wireless Multimedia Local Area Network, Proceedings of the Fifth IEEE International Workshop on Mobile Multimedia Communications (MoMuc'98), Berlin, Germany, October 12-14, 1998, 301-306

[11] Microsoft, Win32 Software Development Kit (SDK), Microsoft Corporation, 1997

[12] Microsoft, Windows NT Device Driver Kit (DDK), Microsoft Corporation, 1997

[13] Mikkonen J., Lehtinen L., Lahti J., Veikkolainen S., A Radio Access Network Architecture for IP QoS, Proceedings of the Fifth IEEE International Workshop on Mobile Multimedia Communications (MoMuc'98), Berlin, Germany, October 12-14, 1998, 109-119

[14] Muller, N. J., Wireless Data Networking, Artech House, Boston/London, 1995

[15] Salli K., Hämäläinen T., Knuutila J., Saarinen J., Security Design for A New Wireless Local Area Network TUTWLAN, Proceedings of the Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'98), Boston, USA, September 8–11, 1998, 1540-1544

[16] Specification of the Bluetooth System, v0.7, October 1998

[17] Tannenbaum A. S., Computer Networks, third edition, Prentice-Hall Inc., New Jersey, 1996

[18] URL=http://www.homerf.org/, January, 1999

[19] Wesel E. K., Wireless Multimedia Communications, first edition, Addison-Wesley Longman Inc., Reading, Massachusetts, 1998

Publication [P9]

Hännikäinen M., Knuutila J., Hämäläinen T., Saarinen J., "Using SDL for Implementing a Wireless Medium Access Control Protocol", IEEE International Symposium on Multimedia Software Engineering (MSE 2000), pp. 229-236, Taipei, Taiwan, December 11-13, 2000.

Using SDL for Implementing a Wireless Medium Access Control Protocol

Marko Hännikäinen[1], Jarno Knuutila[2], Timo Hämäläinen[1], and Jukka Saarinen[1]

[1] Tampere University of Technology
Digital and Computer Systems Laboratory
Hermiankatu 12 C, FIN-33720 Tampere, Finland
Tel.+358 3 365 2111, Fax +358 3 365 4575
marko.hannikainen@tut.fi

[2] Nokia Mobile Phones
Sinitaival 5, FIN-33721 Tampere, Finland
Tel. +358 50 511 7320, Fax. +358 10 505 7606
jarno.knuutila@nokia.com

Abstract

Specification and Description Language (SDL) is a high abstraction level system design language with a clear graphical notation. Because of formal presentation, an SDL model can be automatically converted into source C code for implementation. However, the high abstraction level creates a conceptual gap between a general SDL model and its implementation in a final operational platform. This paper studies the SDL development of an embedded Medium Access Control (MAC) protocol for a Wireless Local Area Network (WLAN) demonstrator. The SDL design flow for the protocol is first started by architectural design without target platform dependencies. Functionality is added to the model using the top-down design approach. Functional simulations are used for verifying the operation of the protocol. Next, the performance is estimated using performance simulations in a workstation environment. Performance improvements can be achieved by optimising the SDL model.

1. Introduction

In the field of personal wireless communications, full service coverage is currently achieved by digital cellular systems. In addition, high-speed data can be locally provided by Wireless Local Area Networks (WLAN) and communication between handheld user equipment is expected by the wireless Bluetooth technology. For continuous service coverage, higher bandwidth, and lower communication costs several network interfaces can be integrated into a single terminal. Also, evolving markets and technological advances drive for a larger variety of applications while increasing user expectations call for new product features and services. Consequently, the lifetime of a product is shortening. This affects to all phases of the development flow from initial requirement analysis to final implementation.

Until recently, the main programming tools for embedded systems have been assembler languages. The increasing complexity has demanded more sophisticated software design tools, such as optimised compilers, debuggers, and development environments with simulators and verification tools. A solution to accelerate the design flow is to utilise higher abstraction and formal languages. This facilitates the comprehending of complex systems and enables hardware/software (HW/SW) co-design.

Specification and Description Language (SDL) is emerging as an efficient tool for general software development [1], [3]. The use of SDL in embedded application development has been studied with *TUTWLAN*. TUTWLAN is a multimedia WLAN demonstrator being designed and implemented at Tampere University of Technology (TUT). The demonstrator contains both hardware and software parts, i.e. demonstrator platform hardware with embedded applications, and Windows NT software applications and drivers. The main embedded application is the Medium Access Control (MAC) protocol (TUTMAC) that has been designed and implemented using SDL. The protocol is currently being ported into the Digital Signal Processor (DSP) of the demonstrator platform. An executable protocol application for the DSP is generated automatically from the SDL model.

The high abstraction level of SDL provides advantages during the specification and design phases but results into challenges when converting the design into a final application. This paper presents the SDL design flow used for the TUTMAC protocol and discusses how the SDL model was optimised for achieving higher performance for implementation. The paper is organised as follows. First, SDL and the TUTWLAN demonstrator platform are introduced. Next, the SDL design flow for the TUTMAC protocol is presented followed by the optimisation of the TUTMAC SDL model. Performance simulations are presented before the concluding section.

2. Specification and Description Language

SDL can be considered as a high-abstraction level programming language with a graphical user interface. The language is especially beneficial in specifying real-time systems and generally systems with a discrete stimuli-response type of behaviour, such as communication protocols. SDL eases the verification of functionality, allowing e.g. to achieve high reliability with little target testing. It is possible to verify the system functionality at early states of the design by simulations.

SDL is standardised by International Telecommunication Union (ITU-T). According to the requirements that ITU-T placed for SDL, the language should [11]:

- Provide a well defined set of concepts to be used for constructing a model for the specified systems,
- Provide an unambiguous, clear, precise, and concise specification, which is especially important for avoiding mistakes and misunderstandings between separate bodies contributing to the system specification,
- Enable the analysing of the specification for completeness and correctness, and
- Provide possibility to utilise computer-based tools in order to create, maintain, analyse, and simulate specifications.

SDL should also abstract the specification from the implementation in order to:

- Provide basis for determining the conformance of implementation to specification,
- Give a clear overview of a complex system, and
- Enable the postponing of implementation decisions, while not excluding any valid implementation techniques.

The system behaviour in SDL is based on communicating extended finite state machines that are executed concurrently. State machines are presented by SDL *processes*. Processes communicate with each other and with the system environment by exchanging asynchronous *signals* that may carry any number of parameters. Signals are usually trigger-type, although also continuous signals can be used. SDL contains also *timers* that can be configured to generate signals at defined moments. Each process in an SDL system contains a FIFO (First-In-First-Out) input buffer into which the received signals are queued. When a process is in a *state*, i.e. not currently performing a transition, it is able to receive a signal from the queue. Signal reception triggers a state transition.

SDL architecture contains hierarchy that allows the dividing of large systems into more comprehensible structures. The hierarchical architecture is depicted in Figure 1. The highest level, system level, consists of *blocks* and *channels*. Channels connecting separate blocks and system environment, as well as *routes* between processes, are used as signal carriers. Each block may contain any number of sub-blocks, with similar block and channel architecture. The lowest level sub-block contains the actual processes. A process may have local variables and *procedures* as attributes. Procedures are the lowest level in the functional hierarchy containing their own local scope.

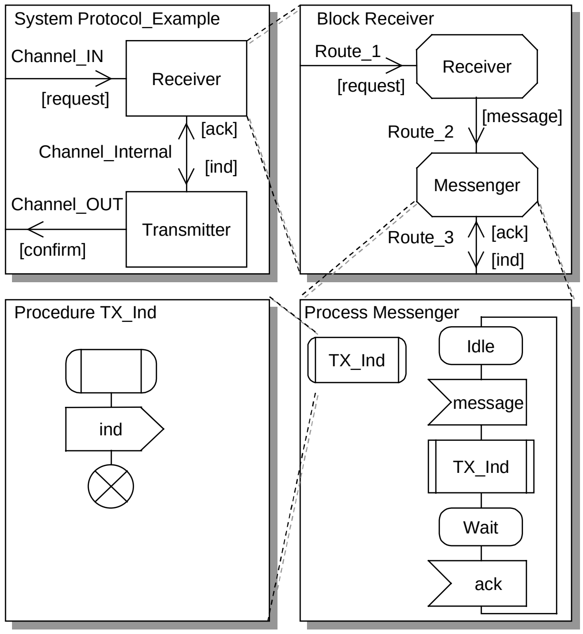


Figure 1. SDL hierarchical architecture.

SDL utilises Abstract Data Types (ADT). Most operations on a type can be applied without regarding the actual implementation. For example, there is no limit to the size of an integer or length of an array. In addition to conventional data types, SDL includes specific types such as time and duration. By defining new types, the definitions and algorithms of other programming languages can be reused. In practice, SDL programming tools may include a set of vendor defined additional data types adequate for most design cases.

SDL is standardised in the ITU-T recommendation Z.100 and its addenda [2]. A number of related recommendations are available defining the use of SDL with Message Sequence Chart (MSC) and Abstract Syntax Notation 1 (ASN.1). The most resent version of the language SDL-2000 is now emerging.

The SDL tool used in the TUTMAC development is SDT (versions 3.4 and 3.6) that is a part of the Telelogic Tau product family [10]. SDT provides a complete development environment containing an analyser, simulator, C-code generators, and an integrated MSC tool.

3. TUTWLAN and the TUTMAC Protocol

The TUTWLAN architecture consists of two types of terminals: *base stations* and *portable terminals*. A base station is the central point of control and management of its coverage cell area. All portable terminals within the cell area communicate via the base station. A base station also operates as an access point to other networks while being connected to a backbone wired LAN.

The current version of the TUTWLAN demonstrator platform is presented in Figure 2. The platform consists of radio sub-system, radio interface, MAC, and host interface modules. The platform can be attached to a Host Computer via the host interface using the PCI-bus (Peripheral Component Interconnect). The host and the radio interfaces are implemented using a re-programmable Field Programmable Gate Array (FPGA) circuit. The TUTMAC protocol is embedded into the MAC module DSP (Analog Devices' ADSP-21060) [8]. 6 Mbytes of external memory is available for program and data. The radio subsystem module of the demonstrator utilises the 2.4 GHz frequency band with 1 Mbit/s link speed [9].

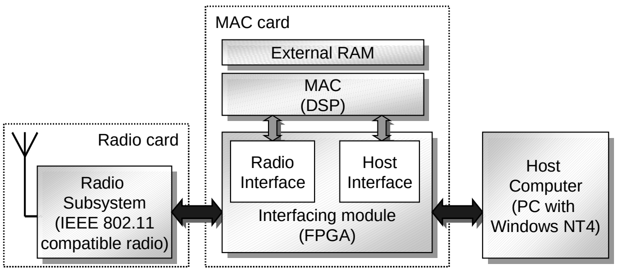


Figure 2. Demonstrator architecture.

TUTMAC is a dynamic reservation Time Division Multiple Access (TDMA) based MAC protocol that destines for supporting negotiated Quality of Service (QoS) parameters for the data-transfer. In addition to access control, the wireless MAC protocol must manage the challenges of wireless network, such as the dynamically changing network topology and unreliable and insecure wireless link.

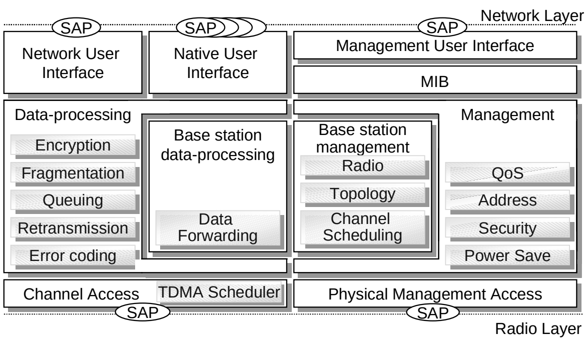


Figure 3. TUTMAC protocol architecture.

The reference architecture of the protocol presenting the most important functions is shown in Figure 3. High processing capacity is needed for supporting the real-time data-transfer functions, such as security related encryption algorithms and traffic scheduling. Especially the TDMA air interface yields to demanding momentary real-time processing requirements. One of the most demanding tasks is to generate and send an acknowledgement frame immediately after a received data frame. More information of the demonstrator platform and TUTMAC protocol can be found from [5] and [6].

4. SDL Design Flow for TUTMAC

A system engineering process is commonly divided into few separate design phases between requirement analysis and implementation. A general design flow is depicted in Figure 4 and combined with SDL design phases used with the TUTMAC protocol development. The use of SDL facilitates a continuous design flow through the whole system development. By extending and adding more details into a system SDL model, the output document of a previous phase can be directly used as input for a next one. At the same time, the maintenance and redesign of the system is simplified as the formal model can be used as documentation. Logic analysis, simulations, automatic MSC generation, and performance estimations by real-time simulations can be applied in early phases for verifying the design.

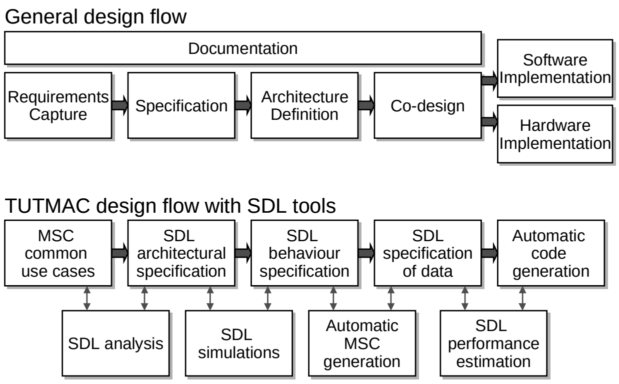


Figure 4. TUTMAC design flow with SDL tools.

4.1. Requirements Capture

The first task of the SDL design flow is to capture, formalise, and analyse requirements for the system to be developed. The purpose of this design phase is to understand the problem domain. The TUTMAC system behaviour was specified informally from the user's point of view. MSCs proved to be a useful graphical notation during this design phase. Consequently, the service interfaces of the TUTMAC system was sketched by defining the service primitives and their informal parameters.

A valuable result from the use of MSCs was a set of most common *use cases* of the protocol. These can be later reused for design verification and simulations [11]. As an example, messages exchanged between a portable terminal and a base station during the network registration are presented in Figure 5. It is also possible to generate SDL model templates using MSCs, but this feature was not used with TUTMAC. [4].

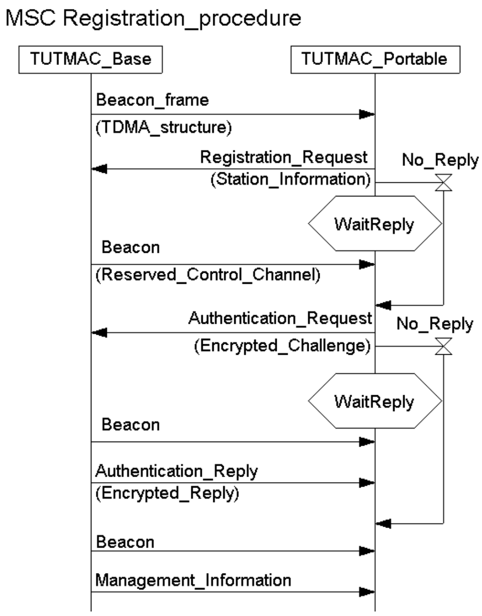


Figure 5. TUTMAC MSC example.

4.2. SDL Specification Phases

During the architectural and behavioural specification phases, the functional model of the system is build using the top-down design approach. These phases also correspond to block level and process level hierarchical layers of the SDL architecture. The specification abstraction was maintained for keeping the TUTMAC design implementation independent. The SDL model can thus be realised with any combination of HW and SW.

For the use of SDL in system specification, ITU-T has provided stepwise recommendations for producing an SDL specification [11]. For specifying and testing using SDL, ASN.1, and MSC techniques, the European Telecommunications Standards Institute (ETSI) has defined its own guidelines. In addition, SDL tool vendors have developed strategies for efficient SDL design [7].

The TUTMAC SDL specification was started by defining the boundaries (interfaces) between the protocol and its environment. This was based on the MSCs acquired in the previous design phase. Next, the specification proceeded by further partitioning the system into SDL blocks. Dependencies between blocks were defined by connecting them with channels. The block architecture partitioning was started from the system level and blocks were subsequently decomposed into sub-blocks. Channels inside sub-blocks are also defined during this phase.

Next, the functional behaviour of the TUTMAC protocol was specified in means of active objects (processes and procedures) and passive objects (data types). Processes describe how signals and the data exchanged between blocks are handled. Most complex data types, timers, and different kinds of exceptional behaviour situations were mostly defined informally by natural language, while the design concentrated to describe the common use cases acquired during the requirements capture phase. Procedures with informal specifications can be effectively used to hide unnecessary details. As the behaviour description proceeded, also these detailed elements were gradually included into the TUTMAC model.

During the SDL data specification, a formal definition for data types used in SDL processes was provided and the signal definitions were completed. For a clearer specification, a large number of new data types were constructed. No implementation details of the types where considered at this phase of design.

The system level of the developed TUTMAC SDL model is presented in Figure 6. The main blocks are *user interface*, *data processing*, *control*, and *channel access*. The control block is further divided into sub-blocks while the other system level clocks are constructed directly using processes. There are four channels between the protocol and its environment: *user channel*, *physical channel*, *physical management channel*, and *station management channel*. As the naming reveals, the signalling is divided into data and control.

Both the base station and portable terminal protocols are developed using the same SDL model. This means that common processes and procedures to both models are static while all the station configuration specific processes are dynamically created during the protocol initialisation. As an example, the processes inside the data processing block are presented in Figure 6. The block contains a dynamically created process called *base routing*, which is present only in a base station configuration of the TUTMAC protocol.

The TUTMAC SDL model for a base station contains the four system level blocks, four sub-blocks inside the control block and altogether 24 processes and 75 procedures. Each process contains generally 2 to 5 states. Since processes are parallel state machines, the amount of processes results into a high number of possible states for the whole protocol, although not all the state combinations of different processes are possible. The number of internal signals used in the model is 96, while 45 different signals are used for communicating with the protocol environment.

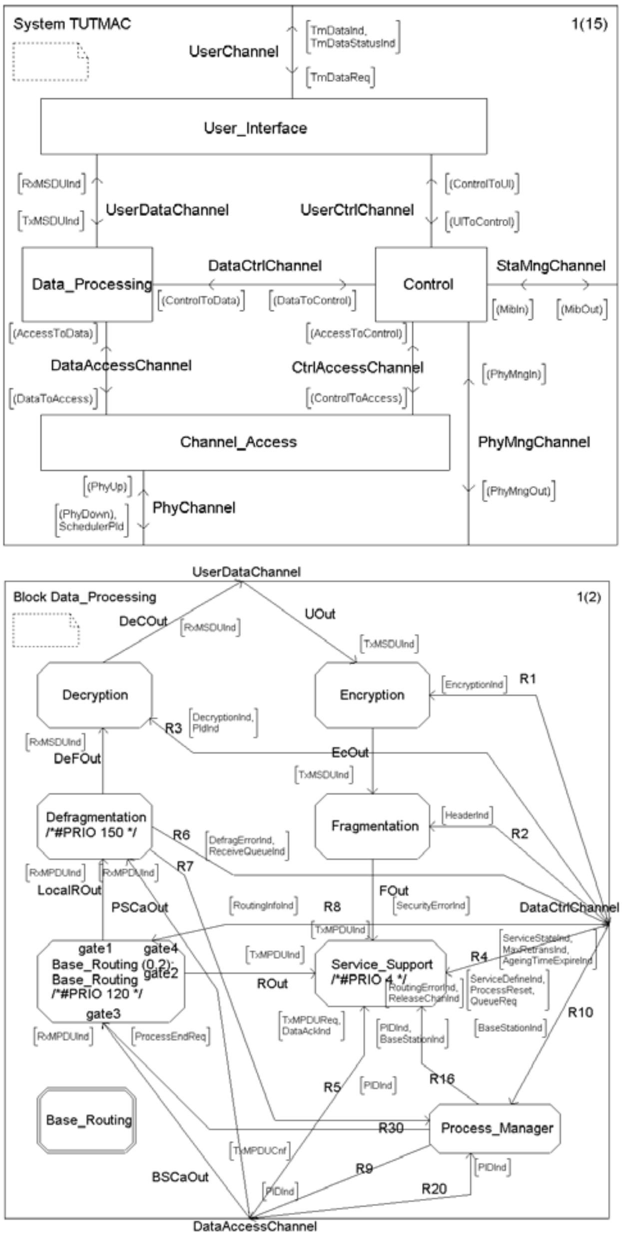


Figure 6. TUTMAC SDL model extracts.

4.3. Simulating of the SDL System

One of the most significant advantages of SDL is that system functionality can be simulated and verified in early phases of design. Consequently, errors and misunderstandings are detected and corrected before proceeding into the implementation phase, which increases the quality of design and can significantly save time and resources.

As the system development proceeds, the focus of simulation and testing can be transferred towards target platform to facilitate the implementation of final application. Since the same specification is used throughout the whole design flow, system simulations enable gradual steps towards implementation by allowing easily evaluating the effects of each design step.

As in any software design testing is deeded for ensuring the error free operation of an SDL model. The operation of all blocks, processes, and data type definitions must be verified also in exceptional or error situations. A convenient way used with TUTMAC testing was the building of test environments in SDL. Additional SDL blocks and processes were constructed imitating the interfaces of real operating environment.

4.4. Architecture Selection

Generally, all digital systems can be realised using both full hardware and full software implementations. The software implementation here refers to a single, general purpose and high-capacity processor into which the system code is embedded. This approach is practical and cost-effective in complex systems with lower real-time requirements. Application Specific Integrated Circuits (ASIC) and FPGA chips based HW implementation can be selected for lower-complexity systems with critical time requirements.

The HW/SW partitioning of the TUTMAC protocol was done between the DSP and FPGA of the demonstrator platform. The most time critical MAC functions are implemented using the HW. These functions are the synchronisation to the TDMA channel access, data encryption, and the calculation of CRC sums for error detection. FPGA logic was traditionally implemented using Very high-speed integrated circuit Hardware Description Language (VHDL). There have been also research activities for performing automatic SDL-to-VHDL conversions [12], [13].

4.5. Implementation

The procedure for building an executable program for the target platform or for simulation in the workstation environment is illustrated in Figure 7. The C source code for the TUTMAC protocol is generated from the SDL specification by using the *Cadvanced* generator of the SDT tool [4]. Next, the code is compiled using the target processor C compiler. The compiled system is linked with a runtime function library for building an executable program. This runtime library contains the coding for SDL constructs and types on the target platform, implementation of the SDL operations, and functions used for scheduling. No other operating system for TUTMAC implementation is used. External C functions can be accessed from the SDL system, e.g. for reusing existing code.

As no information concerning the operational environment is included into the SDL model, the platform specific features are isolated from the SDL code by environment functions. These manually implemented functions connect the SDL input and output signals and their parameters with the corresponding events of the physical environment.

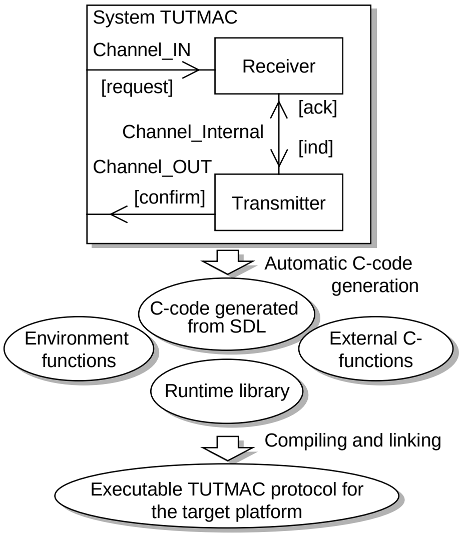


Figure 7. Application conversion.

4.6. Properties of the Implemented System

The resulting size of the protocol and its performance were generally not considered during the specification of TUTMAC SDL model. The purpose was to develop a clear system design that can also be used for documentation.

The size of the C code generated automatically from the TUTMAC SDL model can be estimated by counting the semicolons (;) found from the code. Thus, all comment lines and directives are excluded and only the expressions and function calls are counted. The generated TUTMAC C code contains altogether over 31 000 expressions and function calls. In addition, the generated header fields count for over 2600 expressions. On top source C files directly converted from the SDL model, the SDT runtime library, external C functions, and the environment functions need to be appended for implementing the final protocol. These library files are compiled and linked with the SDL generated C files. The compiler used for the target processor of the TUTWLAN demonstrator is currently cc21k provided by Analog Devices [8]. The compiled TUTMAC base station protocol requires 490 kbytes memory on the target processor for the program code, 61 kbytes for static variables, and approximately 100 kbytes for dynamic memory.

5. Optimisation of the SDL model

The high abstraction level of SDL leads to a conceptual gap between the formal description and its physical implementation. Each SDL construct can be translated into the target platform implementation using a number of ways, each way having a different implementation strategy. The performance of the automatically generated C code is usually not optimised according the properties of the end-system platform. The ITU-T SDL recommendation does not either concern the implementation of SDL syntax or semantics.

SDL contains properties that separate it from real-world systems:

- The operation of SDL processes is parallel, while most systems utilise a single or few processors for multitasking. In addition, the target languages of the automatic code generation are usually also sequential.
- SDL description assumes that infinite system resources are available. First, each process contains an unlimited size FIFO buffer for incoming signals. Second, SDL uses abstract and infinite size data types. Third, no processing time is assumed for state transitions.
- A target platform is usually synchronised while the SDL behaviour is mainly based on asynchronously communicating processes. Perhaps the main drawback in real-time systems is the lack of interrupt support.
- The concept of communication via signals is not supported by conventional programming languages.

5.1. Process scheduling

The basic scheduler of the SDT run-time library used for implementation is non-preemptive, which means that a complete SDL state transition is executed at a time without any interruptions. In TUTMAC protocol, it is essential that certain tasks of concurred processes are executed in a correct order. The SDT scheduler supports process priorities, which allows a designer to distinguish the most time critical processes of the system. SDT also allows a designer to prioritise signals enabling a process with multiple input signals to select the most time critical task to execute. This is an incompatible feature with the SDL recommendation but effective tool for real-time system design.

Generally, an accurate examination of the execution strategy during simulations enables a designer to choose an optimum prioritisation order. In TUTMAC, all processes associated directly to the radio channel access require higher operating priorities than other processes. This is because channel access synchronisation has to be maintained.

In the TUTMAC protocol, an instant reaction to a received frame is essential for an efficient radio channel utilisation. The reaction time is dependent on the frequency that the input signal queues of processes are polled, which again depends on the duration of a preceding state transition. It is therefore important to avoid performing time consuming algorithms or other functions as a single state transition. Despite of careful optimisation, individual state transmissions can still reserve system resources too long, thus delaying the execution of a higher priority process. One solution is to divide these transitions, especially in the lower priority processes, into several parts by adding artificial states. This allows the scheduler to switch to another process having a higher priority. If there are no other processes waiting for execution, the transition continues by an immediate input signal. A convenient way for adding states is to use continuous signals with an enabling condition that is always true. In this case, no actual signal passing is involved and the state transition continues after shorter delay.

5.2. Signal addressing

In an SDL system, signals are guided between the sending and receiving processes according to the names of processes, channels, and signal routes connecting them. While the size of the system increases, the execution of the addressing algorithm takes more time and increases the signalling delay. SDL contains unique Process Identifiers (PId) that are assigned for a process instance when it is created. In order to minimise signalling delay, the PIds should be used for explicit addressing of signals between processes.

Explicit addressing is also needed in order to configure the TUTMAC protocol between the base station and portable terminal modes. The dynamically created processes are attached to the rest of the SDL model by the exchange of PId-values with the neighbouring processes.

5.3. Continuous signals

In the recommendation of ITU-T, the concept of continuous signals and enabling conditions is based on repetitive signals, which is an unacceptable implementation strategy because of excessive signalling overhead. In an actual implementation, the system should consider when the expression associated with a continuous signal actually can change its value, and ignore the signal in other situations. In SDT, the expression is recalculated after state transmissions and timer outputs.

5.4. Data types

The implementation of the infinite and abstract data types of SDL is problematic. For example, SDL contains generators for array and string types but unlike most programming languages there are no restrictions on the length (index sort) of such types. An unlimited array has to be implemented using dynamic memory management. On the other hand, if the index sort is finite the array can be converted into a corresponding static array of the C language. Thus, the selection of used data types is a significant factor affecting to the implementation. An accurate selection of data types, especially the types used as signal parameters, also affects to the overall system performance. Basic modelling of an SDL system requires that signal parameters are copied from the sender process to the receiver process. In order to avoid excessive copying delays in signal exchange, smaller size parameters are preferred. SDL tool vendors support pointer types that are not present in the current standard of SDL.

5.5. Algorithms

SDL lacks the support for writing complex structured algorithms. To improve performance, the most complex and time critical algorithms of the system can be implemented using external functions, which can be optimised C functions. TUTMAC implements the strongest encryption algorithm using external C functions.

6. Performance simulations

In the simulation of the generated protocol, the purpose was to estimate the performance of the generated code in a workstation environment. At this time, a series of SDL model optimisation steps presented in the previous section were carried out.

A base station configuration of the TUTMAC SDL model was selected for the performance testing. Because of the additional base station functions, it demands more processing power than a portable terminal. Simulations were made using a Sun SPARCstation 5 workstation that contains a 110 MHz Sun microSPARC CPU. The operating system was SunOS 5.5.1 and the C code generated was compiled using the Sun Workshop Compiler 4.2.

The performance on the complete TUTMAC protocol was evaluated in a practical manner by measuring the achieved data throughput of one TUTMAC data channel. The TUTMAC TDMA structure contains altogether two control channels and 5 similar data channels that can be allocated for a single user. The physical link speed was tied to 1 Mbit/s correspondingly to the radio sub-system of the demonstrator platform. Traffic load was added by implementing a test block presenting the wireless medium that generated traffic corresponding to portable terminals in the TUTWLAN cell. Also, test blocks for protocol users were implemented. The simulated SDL model is presented in Figure 8.

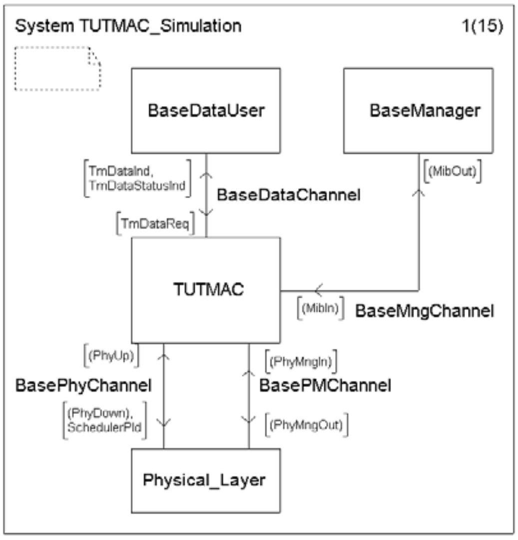


Figure 8. SDL model for performance simulations.

During the first simulations only 80 kbit/s throughput was reached for a data channel. This results into the maximum user bandwidth of 400 kbit/s with five data channels. The performance was improved by the use of priorities, redesign of most critical processes, explicit addressing, reselection of data types, and by partitioning long transitions. The optimisation results are presented in Table 1. The total improvement for the user data throughput is over 50 per cent. For the 1 Mbit/s link speed of the demonstrator platform, the optimisation increases the maximum bandwidth for a user from 400 kbit/s to 615 kbit/s.

Table 1. Optimisation of the TUTMAC SDL.

Optimisation steps for the SDL model	Throughput of one data channel	Incr.
Original model	80 kbit/s	-
Setting process priorities	96 kbit/s	20.0%
Re-designing channel access processes	107 kbit/s	11.5%
Explicit addressing of signals	112 kbit/s	4.7%
Changing used data types	117 kbit/s	4.5%
Dividing long transitions	123 kbit/s	5.1%
Total improvement	**43 kbit/s**	**53.8%**

7. Conclusions

In this paper, the SDL design flow for implementing a wireless MAC protocol was presented. SDL is an efficient graphical tool for system specification and design. Its formality enables the analysing and simulating of the model at early design phases and automatic conversion of the model into C source code for implementation. The automatically generated code is generally not optimised for a target platform. However, performance improvements can be obtained by optimising the SDL model based on system simulations. The optimisation includes careful selection of used data types, assignment of process priorities, utilisation of explicit addressing, and partitioning long state transitions.

8. References

[1] Olsen, A., Færgemand, O., Møller-Pedersen, B., Reed, R., Smith, J.R.W., Systems Engineering Using SDL-92, North Holland, 1994.

[2] ITU-T Recommendation Z.100, 1993.

[3] Automatic code for embedded systems based on formal methods, Technical paper, Telelogic AB, Sweden.

[4] The user manual for Telelogic Tau, versions 3.4 and 3.6, Telelogic AB, Sweden, 1999.

[5] M. Hännikäinen et. al., TUTMAC: A Medium Access Control Protocol for A New Multimedia Wireless Local Area Network, Proceedings of the Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Boston, Massachusetts, September 8–11, 1998, pp. 592-596.

[6] Kimmo Tikkanen et. al., Advanced Prototype Platform for a Wireless Multimedia Local Area Network, Accepted to The European Signal Processing Conference (EUSIPCO'2000), September 5-8, Tampere, Finland.

[7] Homepage of the SDL-Forum, http://www.sdl-forum.org/, 1.3.2000.

[8] Analog Devices Inc., ADSP-2106X SHARC User's Manual, 1997.

[9] Harris Semiconductors, Wireless Handbook, 1997

[10] Homepage of Telelogic AB, http://www.telelogic.com, 3.3.2000.

[11] ITU-T Recommendation Z.100. Appendices I and II, 1994.

[12] I. Bonatti, R. Figueiredo, Stoht-an SDL-to-Hardware Translator, Design Automation Conference, Aug. 29-Sept. 1, 1995, pp. 33 – 36.

[13] B. Lutter, W. Glunz, F. Rammig, Using VHDL for simulation of SDL specifications, Design Automation Conference, Sept. 7-10, 1992, pp. 630 – 635.

Publication [P10]

Kuorilehto M., Hännikäinen M., Niemi M., Hämäläinen T., Saarinen J., "Design for a Wireless LAN Access Point Driver", IEEE International Conference on Telecommunications (ICT 2001), Volume 3, pp. 167-173, Bucharest, Romania, June 4-7, 2001.

Design for a Wireless LAN Access Point Driver

Mauri Kuorilehto[1], Marko Hännikäinen[1], Markku Niemi[2], Timo Hämäläinen[1], and Jukka Saarinen[1]

[1] Tampere University of Technology
Digital and Computer Systems Laboratory
Hermiankatu 12 C, FIN-33720 Tampere, Finland
Tel. +358 3 365 2111, Fax +358 3 365 4575
mauri.kuorilehto@tut.fi , marko.hannikainen@tut.fi

[2] Nokia Mobile Phones
Sinitaival 5, FIN-33721 Tampere, Finland
Tel. +358 50 511 7341, Fax +358 3 318 3690
markku.t.niemi@nokia.com

ABSTRACT

Wireless Local Area Networks (WLAN) are commonly used access technologies to backbone LANs. Bluetooth technology will also provide access to LAN services in near future. When connecting the different types of access networks, the adapting of their services while maintaining the Quality of Service (QoS) is required. This paper presents the functionality and design of an access point (AP) for bridging traffic between Ethernet LAN, a proprietary WLAN called TUTWLAN, and an IEEE 802.11 compatible commercial WLAN. Further extensions for Bluetooth and new generation WLANs are projected. The AP supports QoS management for bridged traffic. The support is implemented by identifying the bridged traffic flows, interpreting and utilising the QoS related signalling of each wireless interface, and by using internal queuing that is based on dynamic priorities. The driver is being implemented as a protocol driver in a Windows NT workstation. Driver implementation in the Windows NT environment is discussed.

I. INTRODUCTION

Wireless Local Area Networks (WLAN) are emerging as a commonly used network access technology in various environments, such as companies, educational institutes, limited public places like airports and libraries, and increasingly also at homes. Compared to mobile data networks, WLANs can provide higher data rates at lower price.

The number of different types of LANs has resulted into challenges when connecting the networks together. Traditionally, the alternatives for LANs have been low costs bridges that only forward traffic between two or more networks on a best-effort service basis. LAN switches are now widely used because of their high performance. In wireless network, the Access Point (AP) term is used because the network element provides an access for wireless terminals to other networks. APs usually contain more sophisticated traffic management functionality than just forwarding. This is because the wireless bandwidth is a scarce resource and must be protected. Also, unreliability and insecurity of the medium cause new functional requirements.

Multimedia is emerging as an integrated service of the Internet, and is also increasingly used in person-to-person communications. These types of applications require support for Quality of Service (QoS) from the underlying networks. According to bandwidth, which is generally the most important QoS parameter seen by users, core networks generally provide high capacity. In user access networks, wired LAN technologies are currently approaching 1 Gbit/s data rates. WLANs reach currently 11 Mbit/s data rates. From this point of view, the WLAN access is the main bottleneck of communication path.

Bandwidth is only one part of the solution for enabling end-to-end multimedia applications. The QoS support in all the nodes of the communication path and the adaptation of the different QoS control functionalities of the underlying networks together for constructing an end-to-end service are required. In wireless networks, the QoS support with accurate bandwidth control becomes more important because of the low bandwidth.

This paper presents the functionality and design for QoS capable LAN access point for connecting wireless access networks to wired backbone LANs. The AP is implemented as a protocol driver in Windows NT environment. Section 2 gives a brief overview of the Windows NT networking environment and protocol layers. In Section 3, the high level architecture of the driver and the connected LAN types are presented. Section 4 presents the functional architecture and services for the AP driver design. Section 5 discusses the driver implementation in the Windows NT environment. Finally, conclusions are given and future work projected.

II. WINDOWS NT4 NETWORKING

The Windows NT operating system is divided into user mode and kernel mode portions. Applications and Application Programming Interfaces (API) are located in user mode, while system service and internal routines in kernel mode. Network drivers are also embedded in the kernel mode portion, in which the communication with other drivers and devices is implemented through I/O manager routines [1].

The network stack in Windows NT is related to the seven-layer Open System Interconnection (OSI) reference model. However, not every layer in OSI model is implemented in Windows NT and also the implementation of layers can be combined. The network stack is mostly implemented by network drivers. User mode applications are connected to the network stack by APIs. The physical layer and parts of the MAC layer are implemented in hardware in the Network Interface Card (NIC). The architecture of Windows NT networking and the OSI reference model are given in Figure 1.

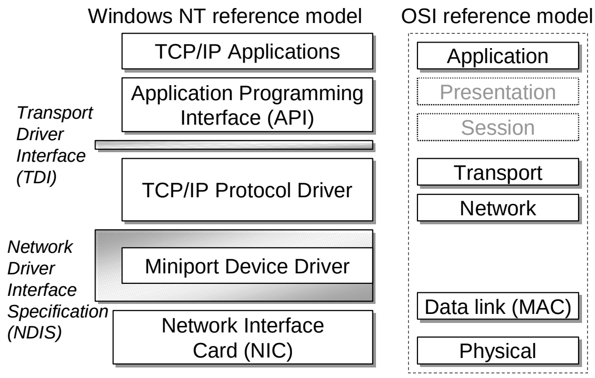


Figure 1. Windows NT networking layers and the OSI model

In Windows NT the interfaces between different levels of network drivers are implemented in Network Driver Interface Specification (NDIS). Thus, several internal and network related support routines, as well as routines implementing hardware interfaces, are provided by this function library.

The interface between user mode APIs and protocol drivers is implemented in Transport Driver Interface (TDI). As in case of NDIS, several interface and internal support routines are provided. The TDI can also be replaced by a custom made interface between protocol drivers and APIs.

III. AP ARCHITECTURE

The purpose of the developed AP driver is to connect different types of wireless local access technologies. At the first phase, the AP will be utilised for three types of networks. A proprietary multimedia WLAN called *TUTWLAN* is the first one. TUTWLAN is implemented at Tampere University of Technology (TUT) [2], [3]. The second wireless network is IEEE 802.11 standard WLAN [7], [9]. The third LAN is wired backbone Ethernet.

The AP protocol driver is located above all NIC device drivers, at the same level as standard protocol drivers such as Transmission Control Protocol/Internet Protocol (TCP/IP). The high level architecture of the system is illustrated in Figure 2.

One of the benefits on the Windows NT protocol driver implementation is the ease in the management provided by higher level implementation. The management of the AP protocol driver is performed by a user mode application that has been previously implemented for TUTWLAN management. The communication between the AP driver and user mode management application is achieved through a custom made API as depicted also in Figure 2. Desired management and monitoring functions can therefore be implemented.

The extensibility is considered in the AP protocol driver design. The amount and type of connected adapters and device drivers are generally not restricted, as the NDIS interface is met. However, in case of large number of different types of adapters, the service adaptation between them can become a computationally extensive task.

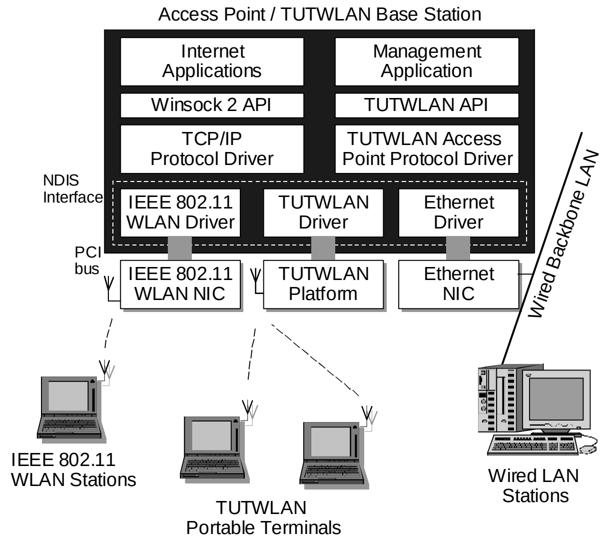


Figure 2. AP architecture and different LAN types.

A. *Properties of Bridged LANs*

TUTWLAN demonstrator platform provides currently up to 2 Mbit/s data rate, and the system is being updated by 11 Mbit/s radios. TUTWLAN utilises the unlicensed 2.4 GHz frequency band. The TUT Medium Access Control (TUTMAC) protocol shares the radio channel between TUTWLAN portable terminals and a base station. The TUTMAC protocol is embedded into a Digital Signal Processor of the demonstrator platform (TUTWLAN NIC). TUTMAC supports several service classes with different QoS requirements [4], [5].

The original IEEE 802.11 standard specify WLANs with 2 Mbit/s data rate, and the newer IEEE 802.11b provides up to 11 Mbit/s data rate radio using same 2.4 GHz frequency band as TUTWLAN. The IEEE is currently standardising new technologies extending the data rates to 20 Mbit/s in the same band and has specified a new radio at the 5GHz region with data rates over 50 Mbit/s.

The Ethernet LAN is referred to being a backbone network for wireless APs. The data rates currently used are 10/100 Mbit/s. The Ethernet segment is further connected to an Ethernet switch, which provides undisturbed connection to the AP. In future, switches with QoS related priority support will be studied.

B. *Packet Types and QoS Signalling*

The support for QoS is gradually being appended into LANs. In addition to the support, QoS signalling for defining the service for payload data is required. The QoS signalling related packet/frame header fields for IP, Ethernet, and TUTWLAN are illustrated in Figure 3. An Ethernet frame (as packets are commonly referred to as frames below the network layer) consists of an Ethernet header, payload data and Cyclic Redundancy Check (CRC) fields. 48-bit source and destination MAC addresses are located in the beginning of the header. The original IEEE 802.3 specification defines a 16-bit length field as the last field of the header [8]. The Ethernet that is generally always used in practice specifies this field as Ethernet Type and uses it to define the upper protocol. The IEEE 802.1Q standard adds an optional 32-bit *tag header* field to the frame. This field contains Virtual LAN information and also a 3-bit IEEE *802.1p priority* field [6].

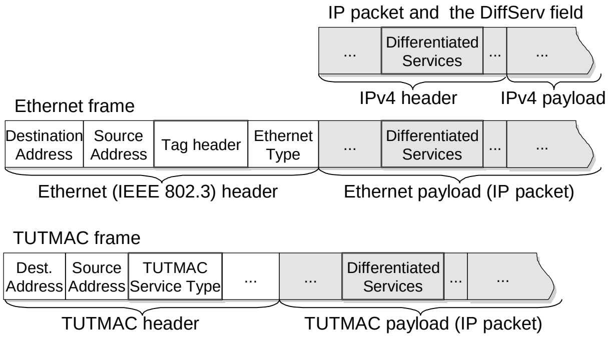


Figure 3. IP, Ethernet, and TUTMAC QoS related header fields.

The IEEE 802.11 WLAN frame structure is similar to the IEEE 802.3/Ethernet, but contains more header fields. This is because more complex management and control functionality and signalling are required for efficiently exploiting the complex nature of the wireless medium with its impairments. QoS has not been integrated into the current standard. However, the IEEE 802.11 working group E is currently defining the QoS extensions to the system. It is expected that the IEEE 802.1p type of priority based signalling will be adopted.

The TUTMAC frame structure contains short (1 byte) addresses that are mapped to the 48-bit IEEE addresses in the TUTMAC protocol. The frames carry a 1 byte TUTMAC *Service Type* field that contains the service class of the payload data. The QoS related communication between TUTWLAN device driver and TUTMAC is performed using the service type field.

Usually, the protocol above Ethernet or WLAN is the Internet Protocol (IP). The IP packet consists of a 20-byte header and upper layer protocol data. The second byte in IP header is the *Type of Service* (ToS) field. It is generally not utilised in the current Internet.

There are two working groups of Internet Engineering Task Force (IETF) targeting to integrate QoS into the Internet. The Integrated Services (IntServ) approach destines to end-to-end reservation based virtual connections. The Differentiated Services (DiffServ) working group, on the other hand, destines for a lighter QoS functionality [10]. In DiffServ, the ToS field (renamed to DiffServ field) of the IP header is used for signalling the traffic class information between network elements, such as routers. This is specified by IETF in the Request For Comments (RFC) 2474 [11]. In DiffServ, the handling decisions for packets are done locally, per hop based, according to the ToS class. No end-to-end reservation exists, which is a similar situation as with LAN frames priorities.

IV. AP DRIVER FUNCTIONAL DESIGN

Generally, the main function of the AP driver is the forwarding of the frames between different network interfaces. Because of the lack of bandwidth in wireless networks, AP filters unnecessary part of the traffic. Thus, a frame is forwarded only if the destination station can be reached. In addition to frame forwarding functionality, the QoS maintenance for the data flow is implemented.

The functional design of the driver is divided into four separate modules: *receive handler*, *forward process*, *send handler*, and *queuing process*, as illustrated in Figure 4. The receive handler makes the initial forwarding decision when a packet is received from one of the network interfaces, maintains the topology information, and builds an internal packet for the driver. The forward process forwards frames/packets between network interfaces and performs a number of control and management functions. The sending of frames to the connected network adapters is done in the send handler. The queuing process manages data exchange between these modules.

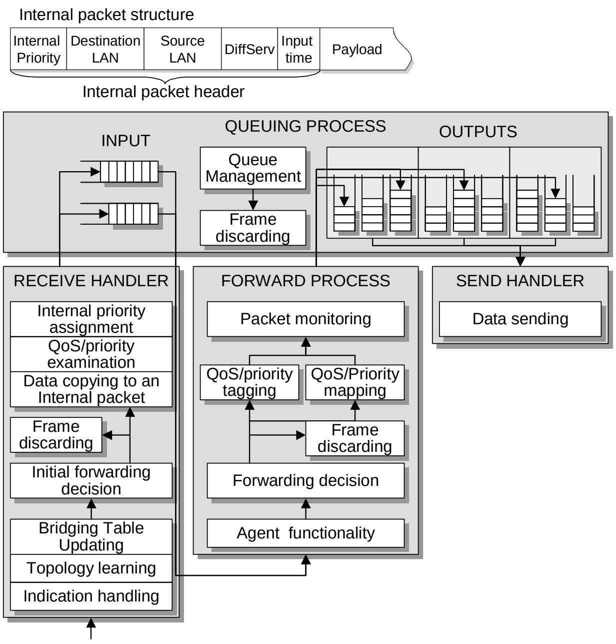


Figure 4. Functional architecture and internal packet format of AP.

A. Receive Handler

The receive handler is executed every time an indication of a new packet is received from one of the connected network adapters. The receive handler functions are performed in a device driver thread. Thus, the thread exists only during the frame receiving.

The necessary information of the network topology is gathered by the topology learning function. The network topology is constructed through the IP and MAC addresses of received frames. The address information of frame headers is examined and the source addresses saved to into the bridging table. The table stores the topology information. Each address is also associated with the interface it was received from. During the operation, the topology database dynamically expands to produce necessary information for most cases. The information is also updated regularly, as wireless stations can rapidly disappear from the network and roam to a new AP.

The complexity of the system increases as several APs are connected to a same backbone network. Before forwarding, the network topology containing several bridges or APs must be investigated in order to avoid multiple routes between two end stations. A channel for this kind of functionality, as well as for other management messages, must be available. Bridging Protocol Data Units (BPDUs) with a suitable inter-bridge protocol can be used for signalling. Multiple alternative routes can also be utilised for better fault tolerance.

For the initial forwarding decision, the bridging table is searched for the destination MAC address. If the address is found and the network interface associated to it is the same as the source segment, the frame is discarded to prevent unnecessary traffic. Otherwise, the processing of the frame is continued.

When the frame is selected for forwarding, an internal packet is allocated for the frame, and the frame data is copied into it. The frame is searched for the QoS fields, Ethernet priority field and the DiffServ value, and the internal packet priority for the first queuing is assigned accordingly. If no predefined priority information is available, the priority is assigned according to the previous frames of the same stream. If this information is not available, the internal priority is set to the best effort (default) value. Depending on the internal priority level, the frame is queued for the forward processing.

B. Forward Process

The forward process, as well as the send handler, is executed in system thread that is created during the initialisation phase of the protocol driver. Therefore, these modules are executed continuously in parallel.

The first operation of the forward process is the agent functionality. Agents can be optionally implemented if required. An agent operates as an intermediate node between two end stations. Thus, the communication that is performed between the end stations is actually performed with one end station and the agent of the AP driver. The purpose is to decrease the load of wireless network, as an agent can appear and operate in the wired LAN side on behalf of wireless station. For example, an agent can reply to Address Resolution Protocol (ARP) requests on behalf of wireless stations, since their MAC and IP addresses are available in the bridging table.

The initial forwarding decision is already made in the receive handler, but if the destination address cannot be found from the bridging table the decision is delayed until here. When topology information is not available, the frame must be relayed to all network interfaces, except the one it was received from. This is done by duplicating the internal packet for all the interfaces. To avoid unnecessary memory usage and copying, the actual frame data is not duplicated. Internal packets contain only pointers to frame data.

Furthermore, the topology of the network constructed by multiple AP drivers is considered here. The network topology is constructed by exchanging BPDUs. Some of the adapters connected to the AP driver must be closed, if another AP is connected to same network segment. Thus, if the AP driver receives a frame targeted to a closed adapter, the frame is discarded.

After the forward decision, the operation depends whether the priority/QoS information is available in the forwarded LAN frame or IP packet. The information has already been searched by the receive handler, and is stored into the header of the internal packet. The QoS/priority assignment in the receive handler is based on the management signalling or LAN priority, which in many cases is not available. In the forward process, a more throughout analysis of the packet contents is therefore performed.

If supported, the QoS fields of exchanged packets or frames are usually assigned at the ingress node of the network. In TUTWLAN portable terminals, TUTMAC protocol assigns frame service types and the TUTWLAN device driver assigns the DiffServ values according to the application and the management configuration. Also, the Ethernet frames may contain the 802.1p priorities and the IP payload the DiffServ information. If the QoS signalling and support by the source and destination network segments differ, the QoS fields are mapped by AP into the destination network specific parameters and signalling fields, so that similar QoS is maintained across bridged segments.

In case no IP DiffServ priority/traffic class exists but the priority is available in the LAN header (IEEE 802.1p priority), the IP DiffServ field can be updated accordingly. On the other hand, existing DiffServ information can be applied for assigning the missing LAN priority. Destination LAN frames are tagged with the IEEE 802.1p priority if the destination network supports this.

When no QoS/priority information is available, the AP driver can examine the MAC and IP addresses, and the utilised network and transmission layer protocols (IP/TCP/UDP) and transmission layer port numbers from the header fields of the payload protocols. The assignment of the IP DiffServ field can then be done according to this information and by using the local configuration set by a management user. The local configuration associates a priority class to the IP address and common transmission layer ports. When the IP header is altered, the header checksum must be recalculated in order to retain correct operation.

After QoS/priority assignment and possible tagging, the packet monitoring function updates the management information collected from the driver. The monitoring concept is divided into four levels. In the first level, light monitoring, such as byte and frame counting, is performed. The AP driver's internal states, such as queue and bridging table sizes, are monitored in the second level. The third level contains meters for certain type of frames and for each priority level. In the fourth level, contents of a stream can be monitored. This level is used only for debugging purposes.

C. Queue Process

The frames in internal packets are first queued between the receive handler and the forward process, and second between the forward process and send handler. For the first queuing, the internal priority queues are shared by all frames. For the second queuing, separate internal priority queues for each network interface exist.

The queue management function operates inside the queue process module. The main task of the function is to provide the most suitable packet for the forward process and the send handler. The suitability is calculated according to packet internal priority that is first assigned by the receive handler and later possibly updated by forward process.

Queue management can also select packets for discarding, if the load of the AP driver grows too large and the queues approach overflow. Also, each packet is associated with the time it was input to the queuing system. If the queuing time exceeds a threshold for ageing, the packet is discarded. In order to avoid the exhausting of lower priority traffic, the queue management function can promote the internal priority of packets. The promotion is done according to the input time.

The management of priority values of IEEE 802 compatible LANs, service type classes of TUTWLAN, and DiffServ values of IP has been designed to operate dynamically. Therefore, the actual numerical values depend on the local management configuration and on the network wide DiffServ and LAN domain configurations.

For the internal priorities, the larger number of queues results into higher system loading. The number of internal priority levels can be dynamically changed by the management application of AP. In Figure 5, one alternative for mapping between the IEEE 802.1p, internal, and TUTWLAN priorities is presented.

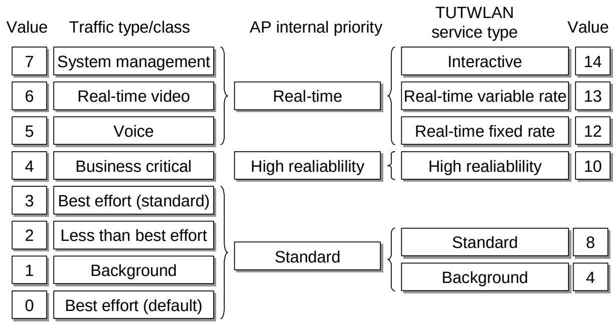


Figure 5. IEEE 802.1p and TUTWLAN priority mapping.

D. Send Handler

The sending of frames to a bound network adapter is the function of the send process. The suitable frame is received from the queue process. Due to the nature of NDIS functions performing the operation, the send operation is completed later, while the function call and execution can return to other modules. After completion the internal packet reserved for the frame is freed for future use [12].

The number of threads performing the send process functionality is currently undefined, and it can vary from one to the number of connected adapters with a separate thread devoted for each adapter. The operation performance must be tested in real environment. The number of queues between the forward process and a send handler is always the number of priority levels supported by the driver.

V. AP DRIVER IMPLEMENTATION IN WINDOWS NT

Because the AP is implemented as an NDIS protocol driver, certain challenges that are not visible in lower layer Access Point implementations are encountered. Here the implementation related issues are discussed, beginning from the characteristics of the Windows NT environment and continuing to the implementation of the AP driver functions.

A. Operating System Characteristics

The communication between device drivers and the AP driver is done using NDIS functions. The interface is fixed as prototypes and the operations of these functions are accurately specified [12]. This limits the communication between device drivers and the AP driver. In addition, the required interface functions must be implemented in all cases.

Kernel mode drivers in Windows NT execute in a single or in multiple threads, thus the threads must be prioritised in order to schedule the processor time. Interrupt ReQuest Levels (IRQLs) are designed for this purpose. The priority level of each thread can be defined in two ways. First, a function that the thread is executing can have a predefined priority, or second, the priority level can be set by the predefined priority of a function that initiated the executing function. Standard interface functions specified in NDIS have a priority level assigned to them. On the other hand, the priority of a thread executing a driver function depends on the interface function that called the function. Therefore, the IRQL of the threads initiated by the AP driver must be set to the same level as the priority of the receive functions specified by NDIS. This prevents the starving of threads at lower IRQL in the system.

In order to avoid processor time being wasted in loops while waiting for an I/O completion, asynchronous completion of I/O operations is supported by Windows NT in communication between layers. Thus, the operation is initiated in a protocol driver routine, and after the completion of e.g. a hardware operation on the NIC, the specified protocol driver completion routine is called by applying an NDIS function [12]. Therefore, the completion routines for the asynchronous I/O operations must be implemented in the AP driver.

B. Device Binding and Initialisation

In order to send and receive packets, the AP protocol driver must be bound to all underlying NIC drivers that need to be connected. Static binding for NDIS protocol drivers can be utilised straightforwardly for the AP driver. Also, dynamic binding can be quite easily implemented if such extensibility is required. In static binding the connected adapters are specified in the source code level, while in dynamic binding the connection is indicated by a device driver to a protocol driver and the binding is made using the indication information. During the binding operation, a handle to the bound device driver is returned by NDIS. A separate handle must be saved for all underlying device drivers. The handle is used in subsequent function calls related to this device driver.

By default, NICs filter the traffic targeted to destination MAC address other than the host itself. Because the AP driver is operating above NIC device drivers, the underlying NICs must be configured to accept all frames regardless of the destination address. The frame filtering can be controlled by a Windows NT object identifier called OID_GEN_CURRENT_PACKET_FILTER. It must be set to the value of NDIS_PACKET_TYPE_PROMISCUOUS [12]. However, also all frames sent are immediately looped back in this mode. Moreover, these loop-back frames cannot be recognised and filtered by the NIC drivers and must therefore be identified in the AP driver. A direct solution for this is provided neither by NDIS nor in other Windows NT kernel mode libraries. However, the looping of frames can be prevented by comparing the bridging table information to the MAC source address of a received frame. Only if the adapter associated to the MAC address and the adapter from which the frame was received are the same, the frame is accepted.

C. Frame Receiving

Device drivers operating under AP driver can indicate incoming data by two NDIS functions. First, the function *NdisMIndicateReceivePacket* is used if the whole data is located in an array of complete frames. Second, only the header data of an incoming frame can be relayed to the upper drivers by the function *NdisMEthIndicateReceive*. Protocol drivers generally implement corresponding receiving functions for both of these cases. A separate transferring of the frame data from the device driver to the AP driver is required in the latter case, but only if the frame is selected for processing [12].

Several device drivers can indicate incoming frames almost simultaneously. In order to avoid unnecessary reservation of the other device drivers' resources, the receive handler of the AP driver should execute as fast as possible. Because of this, data indicated as whole packets is also copied to the memory area of the AP driver, and the ownership to the device driver packet is released.

D. Memory and Internal Packet Management

Memory management for the internal packets in the AP driver's memory space must be implemented. Memory allocation and freeing functions are provided by the NDIS interface. The memory used by the driver is non-paged unshared memory. Memory for a static number of internal packets is reserved during the initialisation phase of the driver, and it is not freed until in the unloading phase. Before a new frame is copied into the driver memory, an internal packet is allocated in the receive handler. After sending, the internal packet is released in the send handler immediately after the frame data stored into the packet is sent. During the queuing process, only pointers are relayed and no actual copying of data is required.

E. Bridging Table Handling

The network topology learning is based on the bridging table, into which the MAC-addresses and the associated adapter indexes are stored. Additional information, such as related IP-addresses, can also be associated to an entry. In addition to the dynamic learning, static entries can be added from the management application. Therefore, a type field indicating a static entry is required.

Dynamic memory allocation and dynamic data structures are not allowed in the Windows NT kernel mode programming. Therefore, a static amount of memory must be allocated for the bridging table. As the size of the table increases dynamically during operation, a remove algorithm must be applied. The algorithm used for the AP driver removes the entries that have not been used for a certain period. Thus, each time an existing entry in the table is accessed, it is marked. When a marked entry is processed by the remove algorithm, it is unmarked.

To enable the operation of an AP in a changing environment, the size of the reserved bridging table memory can be altered through the management application.

F. Management Control Handling

The communication between management application and AP driver is implemented by a custom API. The AP driver must implement an upper interface for the communication with the API. The API uses I/O Request Packets (IRPs) to communicate with the protocol driver [12]. The IRPs are processed by different functions of the protocol driver, depending on the I/O control code of the IRP. Separate functions are dedicated for the creating and closing of handles to protocol driver, and for device control messages.

Create and close control codes are sent to the protocol driver when a file handle representing the driver is created or closed in the API, respectively. Normal communication between API and protocol driver is performed using device control messages. The contents of the messages can be chosen freely as the communication is not restricted. After an IRP is received in the AP driver, it is either completed immediately with appropriate status message, or marked as pending. In the latter case, it is later completed asynchronously with the final status message of the operation.

Device control messages are applied to control and monitoring purposes. Static bridging table entries and bridging table size alternations are examples of such control messages.

Wide range of management possibilities is achieved through the implementation of the AP as a protocol driver. The communication between the AP driver and the management application can be easily implemented and therefore even heavy monitoring functions can be performed. Furthermore, the configuration of the AP in different environments can be easily done.

VI. CONCLUSIONS AND FUTURE WORK

The implementation of the AP as a high layer protocol driver provides extension possibilities. Thus, different kinds of network interfaces from WLAN and LAN to Bluetooth adapters can be connected to the system [13]. Only the implementation of the service adaptation for maintaining QoS and for adapting the different frame structures is required.

The design of the driver was started with a background study to the operating environment. The driver functionality was first specified in high level, and gradually implementation specific design was performed. At the moment, the first version is being implemented. Functional errors in the design have not been encountered. The AP driver initialisation and the functionality of the forwarding operation are implemented and are currently being tested. The implementation of the agent operation and the bridged network topology construction are postponed to the next version.

The final size of the AP driver is estimated to be about 4500 lines of C code. In addition to the driver implementation, the performance of the AP depends on the capacity of the computer it is executing. In addition, a constant traffic flow from Ethernet to WLANs experience frame discards due the difference in network capacities. It is left to the flow control of TCP and applications to recover from overflow frames.

The current operating environment, Windows NT 4, is currently widely used. The new Windows 2000 operating system is based on the NT technology and Windows NT drivers are partly compatible with Windows 2000. Thus, it is expected that no major alternations are required for porting the drivers to the new system. Furthermore, Windows 2000 contains additional functionality that can be utilised. For example, the immediate loop-back in frame sending can be directly prevented by Windows 2000 NDIS.

REFERENCES

[1] H. Custer, *Inside Windows NT*, Microsoft Press, 1992.

[2] M. Hännikäinen et al. "Design and Implementation of a Wireless LAN Interface Card Driver in Windows NT", International Conference on Telecommunications (ICT'99), Cheju, Korea, June 15-18, 1999, pp. 347-351.

[3] M. Kuorilehto et al. "Bridging Network Traffic between Wireless and Wired LANs in Windows NT", European Signal Processing Conference (EUSIPCO'2000), Tampere, Finland, Sep. 5 - 8, 2000, pp. 267-270.

[4] K. Tikkanen et al. "Advanced Prototype Platform for a Wireless Multimedia Local Area Network", European Signal Processing Conference (EUSIPCO'2000), Tampere, Finland, Sep. 5 - 8, 2000, pp. 2309-2312.

[5] Hännikäinen et al. "TUTMAC: A Medium Access Control Protocol for A New Multimedia Wireless Local Area Network", International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC'98), Boston, USA, Sep. 8–11, 1998, pp. 592-596.

[6] IEEE P802.1Q D11, IEEE Standard for local and metropolitan area networks: Virtual Bridged Local Area Networks, IEEE Standards Department, June 30, 1998

[7] IEEE P802.11, IEEE Standard for local and metropolitan area networks: Wireless LAN Medium Access Control (MAC) and Physical Layer Layer (PHY) specifications, IEEE Standards Department, Aug. 20, 1999

[8] IEEE P802.3, 2000 ed. IEEE Standards for local and metropolitan area networks: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specification, IEEE Standards Department, 2000

[9] Homepage of Nokia WLAN, http://www.nokia.com/corporate/wlan/, Feb. 10, 2001.

[10] K. Kilkki, *Differentiated Services for the Internet*, Macmillan Technology Series, USA, 1999

[11] Homepage of Internet Engineering Task Force (IETF), http://www.ietf.org/, Feb. 6, 2001.

[12] Microsoft, *Windows NT Device Driver Kit* (DDK), Microsoft Corporation, 1997

[13] Specification of the Bluetooth System, v1.1, Feb. 22, 2001, available at http://www.bluetooth.com/

Publication [P11]

Hännikäinen M., Lehtoranta O., Kuorilehto M., Suhonen J., Niemi M., Hämäläinen T., "Architecture of a Wireless Video Transfer Demonstrator", International Zurich Seminar on Broadband Communications (IZS 2002), pp. 53-1 - 53-6, Zurich, Switzerland, February 19-21, 2002.

2002 International Zurich Seminar on Broadband Communications
Access - Transmission - Networking
February 19-21, ETH Zurich, Switzerland

Architecture of a Wireless Video Transfer Demonstrator

Marko Hännikäinen[1], Olli Lehtoranta[1], Mauri Kuorilehto[1], Jukka Suhonen[1], Markku Niemi[2], Timo Hämäläinen[1]

[1] Tampere University of Technology
Institute of Digital and Computer Systems
Hermiankatu 12 C, FIN-33720 Tampere, Finland
Tel. +358 3 365 2111, Fax +358 3 365 4575
marko.hannikainen@tut.fi, olli.lehtoranta@tut.fi

[2] Nokia Mobile Phones
Sinitaival 5, FIN-33721 Tampere, Finland
Tel. +358 50 511 7341, Fax +358 3 318 3690
markku.t.niemi@nokia.com

Abstract-- This paper presents the architecture of a wireless video transfer demonstrator. The demonstrator has been implemented for developing control protocols and QoS support for real-time video streaming services. The demonstrator contains modules for video capture, encoding, stream protection, and transfer for wireless link or network, and the decoding and displaying at the receiver. H.263 encoding is performed in real-time using a dedicated hardware. A video control protocol has been designed and implemented for managing the stream transfer, and for collecting measurement information. The current implementation operates over Wireless LAN, GSM data, Bluetooth and proprietary wireless LAN called TUTWLAN. In addition, a special module has been implemented for simulating different wireless links or networks locally.

Index Terms-- Wireless video, H.263, QoS, Video Protocol

I. INTRODUCTION

Wireless connectivity is emerging as a ubiquitous service. Continuous network coverage is achieved by utilising several hierarchical and parallel network technologies, ranging from personal area networks (PAN) to wide area (WAN). Thus, the diversity of systems to be supported is increasing. In order to achieve a seamless service coverage, interoperability, and lower communications costs, a multi-homed terminal containing several network interfaces can be a practical alternative. Both service and network adaptation and monitoring are required for enabling users to roam between several access technologies [1].

This paper presents a wireless video demonstrator that is utilised for developing control protocols and Quality of Service (QoS) support functions for real-time services over different network technologies. Video has been selected to be the target application because of its demanding nature, including real-time characteristics. The demonstrator combines the functions of lower layer network protocols and higher layer adaptation for enabling an uninterrupted service. In addition, video encoding and decoding are connected to the video control model.

This paper is organised as follows. In Section 2, the reference architecture of the wireless video demonstrator is presented. Section 3 describes the video encoding and decoding layer of the system. In section 4, the adaptation of the encoded video stream for wireless transfer is presented. Different link types are discussed in Section 5, followed by preliminary tests. Finally, conclusions are given and future work projected.

II. DEMONSTRATOR ARCHITECTURE

The three-layer reference architecture of the video demonstrator is depicted in Figure 1. The reference model has been introduced as the actual system implementation alternates according to operational environments and used hardware. There are a number of different networking technologies combined for video delivery, each technology containing its own characteristics and configurable parameters, such as network type specific QoS. The purpose is to manage heterogeneous networks in order to provide an end-to-end service.

Interoperability is another reason for the three-layer model. The control of the system is centralised in the middle adaptation layer, while the complexity of demonstrator implementation can vary. In the full implementation, the higher video encoder/decoder layer and the underlying network/data link layer are application-aware, thus providing support for video transfer. In the middle implementation, the two layers provide a signalling interface, through which the parameters of the video encoding and decoding and the network QoS can be configured. However, dynamic adaptation for different application types is not required. In the low implementation, the adaptation layer operates between legacy video application layer and legacy link/networks, thus supporting video without requiring configurability from the other layers.

The basic architecture of the demonstrator is a client-server system, in which the connection between the entities can be realised using different technologies. The client is referred to as a *video receiver* and the server as *video sender*. Video receiver is a terminal with varying processing capabilities and network connection parameters. The video sender is referred to being a high capacity server.

Figure 1 introduces the main functions for protecting the video for wireless transfer and presents the main parameters used to verify and compare the operability of applied functions. For the video *encoder/decoder* layer, the basic functions are to adapt a stream at the sender and to conceal occurred errors at the receiver end. The quality of the decoded video is measured by frame loss and by computing Peak-Signal-to-Noise-Ratio (PSNR) against the original stream. The encoded stream is prepared for the wireless transfer by the *video stream adaptation* layer that introduces robustness against wireless link errors. The layer supports

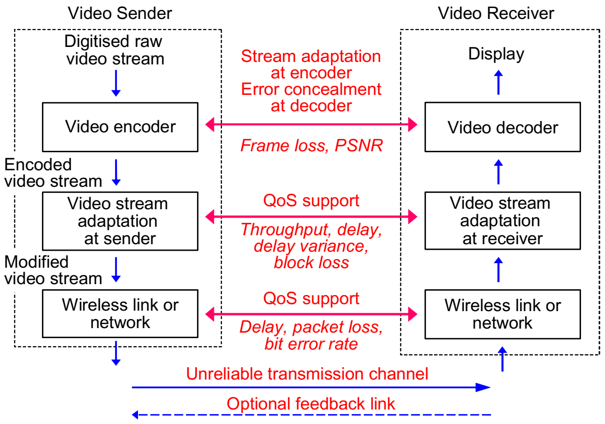


Figure 1. Reference model of the demonstrator system.

stream transfer by several QoS means. The adaptation layer can also exploit the dynamic QoS of network layer, if this is available. The quality of the video stream transfer is monitored by various QoS parameters.

The third layer of the demonstrator reference model is the wireless *network/link* layer. A wireless link can be directly available between a video sender and receiver, for example through a Wireless Local Area Network (WLAN) or Bluetooth [12]. On the other hand, a wireless WAN connection is created by concatenating different technologies (thus the term network describes the connection better). The tools for video protection depend on the network/link technology. Adequate capacity and configurable network QoS support are the most important.

The architecture of the current demonstrator is presented in Figure 2. The sender and receiver operate in Windows NT computers. Network Interface Cards (NIC) are attached to a PC through device drivers that operate under the legacy TCP/IP protocol stack. Over network drivers, the Winsock2 Application Programming Interface (API) is applied for applications [8]. The *Video Sender* and *Video Receiver* modules that implement most of the video adaptation can also connect to NIC device drivers through tailored drivers, without any specific protocol stack.

The basic architecture of the system contains video input, video encoding, and video sending at the sender host, with video receiving, video decoding, and displaying at the receiving host. The central connective control module is the *EncoderIO* at the sender and *DecoderIO* at the receiver. These modules connect the different components together. In addition, a *Link Simulation* module has been implemented for simulating the video transfer locally. There are User Interfaces (UI) available for the different modules for configuration and the collecting of measurement data.

III. Video Encoder/Decoder Layer

The video input for the demonstrator is acquired by either real-time video capture or by using test sequences stored on the local disk. The stored video can be either pre-encoded H.263 stream or raw video. The source video is digitised using Matrox Meteor II video capture card for CIF or QCIF formats. The video encoder is a parallel implementation of ITU-T H.263 v1 [3] on a Hunt Engineering Hepc8 [2] platform that includes four TMS320C6201 DSPs. Frame rates exceeding 30 fps can be provided also for CIF.

The real-time video decoder is an error robust and proprietary implementation of a H.263 decoder. For displaying the video decoderIO module creates a separate scaleable window. After the decoderIO has received an incoming image from video receiver, the image is decoded, converted to RGB colour format, and drawn onto the video window using `DrawDib` functions of Microsoft Windows Platform SDK [9].

A. Metrics for Video Quality

In the wireless video demonstrator two main factors affect the quality of decoded video. First, since the encoder is using lossy compression techniques, the image quality degrades as the compression ratio increases. Secondly, the bit errors and packet losses can cause severe error artefacts to the decoded output. To measure these effects as well as to evaluate the performance of different video protection tools, PSNR is calculated for output according to Equation 1.

$$PSNR = 10\log_{10}\left(\frac{255^2}{\text{noise signal energy}}\right) \qquad (1)$$

On the other hand, it is important to show how many video frames are lost during the transmission. The packet loss (P_{loss}) is characterised with Equation 2.

$$P_{loss} = \frac{\text{number of lost frames}}{\text{number of frames in original sequence}} * 100\% \qquad (2)$$

B. Means for Protecting the Video

In error-prone wireless environments, techniques for protecting video are necessary [1]. Current video coding standards, such as MPEG and H.263, typically employ variable length codes (VLCs) to achieve a high compression ratio. However, VLCs are sensitive to errors since a single bit error may lead to loss of synchronisation at the decoder side, which usually leads to the loss of several following VLCs. In addition, the use of a motion compensation process creates a situation where errors begin to propagate in spatial and temporal directions in the decoded video [6]. In the following, various error resiliency tools that are studied using the demonstrator are described.

FEC is a coding technique where redundant parity bits are added to the original data for detecting and correcting of bit errors. FEC is generally capable of recovering from single or few bit errors, but long burst errors are difficult to correct. Thus, the video demonstrator currently utilises repeated transmissions for the most critical components. Also, Reed-Solomon block code (255, 223) is the currently used block based coding [13]. *Unequal error protection* of video elements classifies different bit stream elements according to

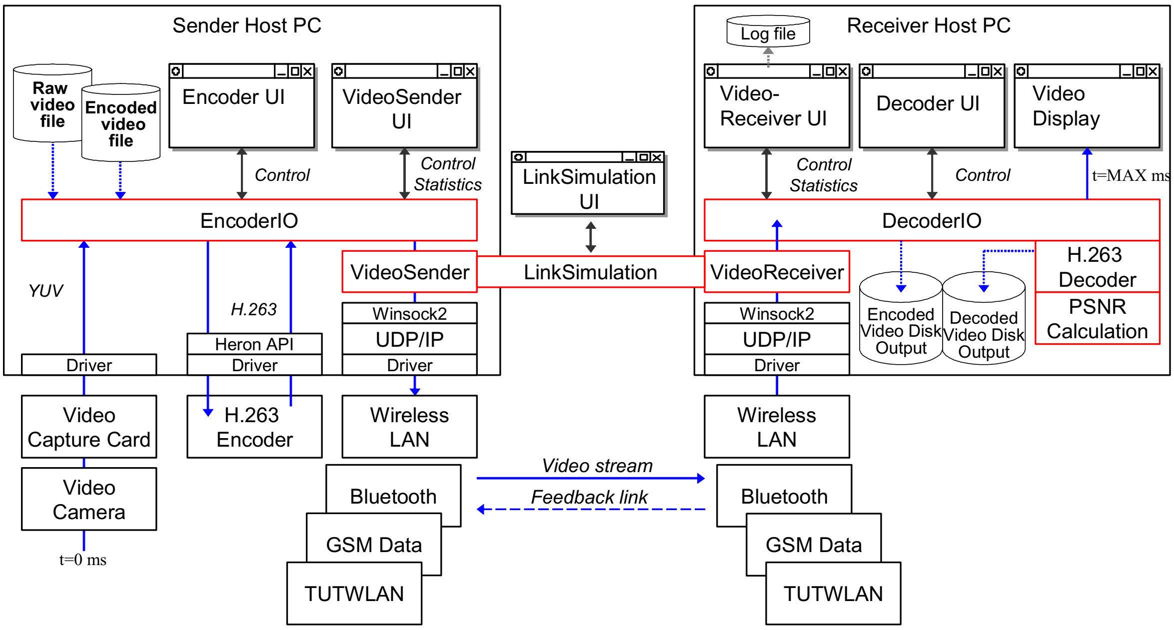


Figure 2. Current demonstrator implementation architecture.

their importance. After classification and element separation, the different elements are protected by different strength FEC.

Intra updating means an encoding strategy where once a while a macroblock is intra coded (i.e. not predicted from the previous frame). Intra macroblocks repair propagating errors from the decoded output. If a feedback channel is unavailable, the encoder does not know which macroblocks to update and a *random* or *pre-defined* intra updating *pattern* has to be used. *Intra updating based on feedback information* implies that the coding strategy of macroblocks is modified according to error information feedback provided by the decoder to stop the error propagation.

Synchronisation markers are unique, fixed length VLCs, which the decoder is able to recognise without knowing the following structure of the bit stream. When the decoder detects an error, it seeks the following synchronisation marker. After that the knowledge of bit stream position is re-established.

Network-aware video packetisation means that a video communication system will attempt to start a network data packet with a synchronisation marker. This allows an independent decoding of the packets. The size of a coded video segment can be adapted to a network packet size by using slices.

Reversible variable length codes (RVLC) are VLCs that have prefix property in the forward and reverse directions. This property allows a decoder to look the bit stream to both directions. When the decoder detects an error, it seeks the following synchronisation marker and then begins to decode to the reverse direction until an error is encountered again.

Currently, the wireless video demonstrator supports error resiliency by implementing synchronisation markers and intra updating. Moreover, each image slice begins with synchronisation marker, i.e. with a H.263 Group-Of-Blocks (GOB) header.

C. Means for Recovering from Errors

Despite of the use of error resiliency tools, any damage done to the video bit stream will likely lead to visual distortion at the decoder. This distortion can range from momentary quality degradation to a completely unusable image. Therefore, it is necessary to perform error concealment by post-processing to minimise distortion.

Basically, all error concealment techniques attempt to recover lost information by estimation and interpolation, exploiting the temporal and spatial smoothness property of the video signal. Spatial domain error concealment includes techniques of *spatial domain interpolation*, *maximally smooth recovery* and *projection onto convex sets* (POCS). The temporal domain error concealment uses methods of *motion compensated prediction/interpolation*, *spatial-temporal smoothing*, *temporal estimation of blocks with missing motion vectors* and *motion field interpolation*, respectively [6],[7]. In the first phase, the H.263+ decoder error concealment model (TCON model) is implemented for the demonstrator [5]. The idea with TCON model is to detect serious errors and prevent this part of the image from being used. The advantages of the TCON model are that it is simple and gives good results if motion vectors of damaged blocks are available. However, unequal error protection will be used to make sure that the decoder receives motion vectors undamaged.

D. *Encoder and Decoder UIs*

The encoder UI presented in Figure 5 controls the encoding process and the encoderIO module. It enables a user to select the video source, destination link/network, and set the properties of the transmitted video stream. Currently, frame rate and bit rate can be configured. A network test procedure can be used for estimating the network/link capacity. The encoder UI collects and presents the statistics of the encoded stream. The video decoder UI in Figure 5 displays statistics for a received stream. Collected data contains the number of received frames, current bit and frame rates, and number of lost picture frames. The stream source can be selected between network feed and local disk. Also, the output can be decoded to the local display or the encoded video received from network can be directed to the local disk. The encoded video is decoded later for measuring the PSNR metrics.

IV. VIDEO STREAM ADAPTATION LAYER

The basic tools for video adaptation for the wireless transfer between the sender and receiver contain the following functions: FEC coding, retransmissions, fragmentation, packet scheduling and queue management algorithms, the use of QoS parameters of the underlying link or network, and the use of the configurable properties of the video encoder.

Most operations of the video adaptation layer are implemented in a *Video Control Protocol* (VCP). VCP contains the video adaptation tools and manages connections between the video sender and receivers. VCP is implemented as the video sender and video receiver modules. Its control architecture and functional tools are illustrated in Figure 3. The central control module of VCP is the traffic class definitions and control. A traffic class defines what functional layers are applied and controls their operations. Also, the current implementation of VCP contains the collecting of statistics, the exchanging of VCP control messages for link establishment and link measurement, and the retransmission control and queue scheduling. The VCP tools available at the moment are the parsing and assembling of H.263 frames, time stamping, VCP packet assembly, QoS classification, retransmissions, and traffic class based queuing.

The video sender and video receiver exchange video data and control messages using VCP frames. The general frame structure and the different fields are explained in Figure 4. VCP exchanges control messages (frames without the data field) between video sender and receiver for connection establishment, disconnection, synchronisation, acknowledgements, and for exchanging traffic class definitions. Each frame carries its send timestamp, traffic class, data content, and fragmentation status fields.

A. *Parsing and Reassembling the Video Stream*

To support unequal error protection by FEC and retransmissions, separate H.263 frame parser and assembly functions are implemented. At the parser module, a standard stream is first partially decoded. The bit stream is re-encoded to more error robust form taking into account the desired options for wireless transfer. These options include:

- Addition of new synchronisation markers
- Replacing of original VLCs with RVLCs
- Separation of different bit stream elements
- Preparing video into suitable network packets

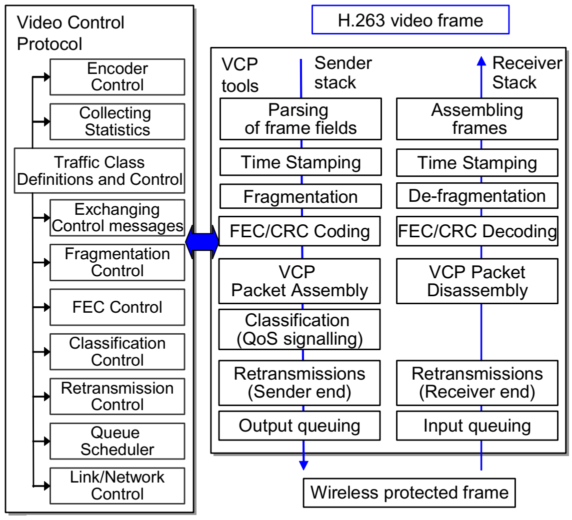
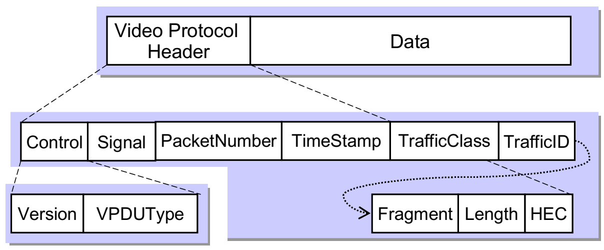


Figure 3. Video Control Protocol.

The VCP video parser module works on VLC level and performs similar operations as a video transcoder. For unequal error protection, the following five bit stream elements are separated, re-grouped, and buffered separately.

1. *H.263 picture headers* –starts an encoded picture
2. *H.263 GOB header* – starts an encoded horizontal slice
3. *Macroblock header* – starts each macroblock
4. *Motion vectors* –in the macroblock header
5. *Macroblock data* – DCT coefficients

Each buffer is assigned a traffic class and given to the lower layer. On the video receiver host, the video frame assembly module receives different components and merges them



Control: Version number and the type of the unit
Signal: Used for lower level protocol specific signaling
PacketNumber: Running number of video protocol packets
TimeStamp: Time stamp of the frame set by VideoSender
TrafficClass: Traffic class of payload data
TrafficID: Identification of the type of payload data
Fragmentation: Inform if fragmentation has been used, contains also the fragment number
Length: Length of payload data
HEC: Header Error Control, detects errors from the header
Data: 0-Maximum data unit of bytes

Figure 4. Video Protocol Data Unit (VPDU) and unit fields.

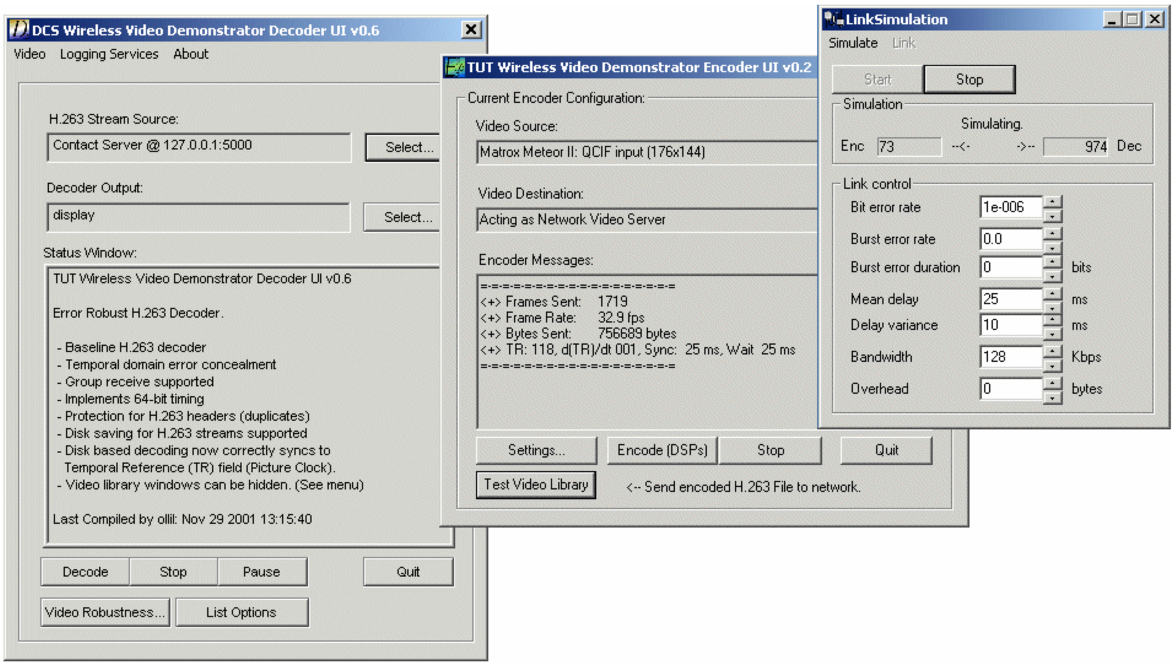


Figure 5. Demonstrator UIs: Decoder, Encoder, and Link simulation.

together to reconstruct the original video stream. If one of the three buffers is missing or there are bit errors, the video assembly module will perform error concealment according to H.263+ TCON model.

B. Traffic Class Construction

Traffic classes are constructed according to the performed /network measurements or they can be pre-assigned. The parameters of a class depend on the available functional layers of VCP and configurability of data link/network layers. An example of a class definition is presented in Figure 6. A user can set the ageing time for video data to become obsolete. Retransmissions with controllable sliding window can be applied. Errors are normally detected from the VCP headers, while the error detection can be extended to cover also the content payload by additional CRC calculation. The CRC field is added after the payload data. An immediate acknowledgement and retransmission scheme, mainly targeted for control frames, is available as the ARQ selection. The urgency in the traffic class definition UI refers to user assigned priority.

C. Queue Scheduling

The different picture elements are queued by VCP according to their traffic class. The queue scheduler decides the next packet to be taken from the queues. The scheduler selects a packet according to the sum of the queue urgency, queue priority, and the priority of the frame. Packet with the highest value is selected. In order to prevent starvation, the queue priorities and priorities of queued frames are updated always when scheduling is performed.

D. Video Sender and Receiver UIs

The video sender UI allows a user to select the link/network to be used. The VCP operates over UDP/IP for achieving interoperability, but also other protocol alternatives are available. The connection establishment synchronises the sender and receiver clocks by repeated frame exchange and transfer delay measurements in order to measure transfer delays. The video sender UI displays the created connections, the queue sizes and the number of sent frames.

The video receiver module collects and displays detailed statistical information of the transfer characteristics of a stream. The measurement data can also be directed into a log file for further analysing. The collected information contains the number of packets, throughput, mean delay, and delay variance that are calculated using the timestamp information. The throughput is separated for the overhead and video data portions. Received frames are separated according to successful transmissions, with and without error corrections. Missing frames are also indicated.

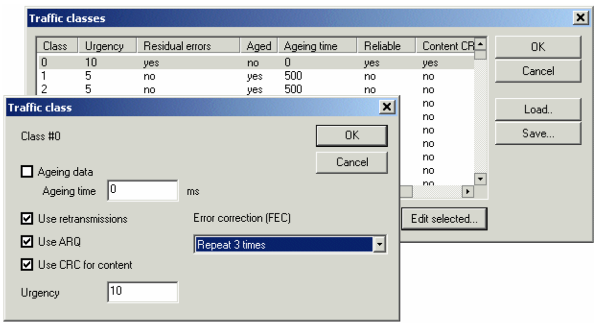


Figure 6. Traffic class construction UI.

V. LINK/NETWORK LAYER

Current alternatives for the demonstrator link/network technologies are the generic UDP operating over commercial WLAN, GSM data, General Packet Radio Service (GPRS) of GSM, and any other compatible NIC. A link simulation module has been implemented for enabling the simulation of different link characteristics. A Bluetooth link is implemented differentially compared to the UDP/IP links, as the VCP operates over Bluetooth specific RFCOMM and L2CAP protocols [12]. The Ericsson Bluetooth Starter Kits are utilised for testing the demonstrator [10]. In addition, TUTWLAN [11] allows a highly configurable link testing for the demonstrator. TUTWLAN is a proprietary WLAN system that enables the developing and testing of tailored link protocols and co-operation between the three layers of the demonstrator.

The link simulation module enables the complete demonstrator system to operate in a single host PC. The module can be configured to simulate different network characteristics, as presented in its UI in Figure 5. The available parameters are the average bit error rate, burst error probability with the average duration of the burst, and variables for delays, bandwidth, and overhead. The overhead simulates the effect of lower layer protocol headers of the link/network. If an error is generated in the simulated lower level overhead, the packet can be considered lost as in real networks. It is also possible to serialise multiple simulated links. Parameters for each link will be set separately., modelling a real situation with several network access technologies between video sender and receiver.

VI. PRELIMINARY TESTING

In the preliminary testing the communication delay and video bandwidth of various GSM links, WLAN, and link simulation were investigated. For the measurements, several versions of a pre-encoded test sequence called *salesman.qcif* were transmitted. The transfer delays measured by VCP time stamping are plotted on Figure 7. The legend presents the network type by giving the link name, the maximum measured video bandwidth, and the theoretical link bandwidth (e.g. GSM/7.8 kbps/9.6 kbps). The *BER 10e-5* was produced by the link simulation with the bit error rate (BER) of 10^{-5}. The maximum video bandwidth is the highest bit rate that did not cause frame losses. The results for WLAN are provided for comparison, since its capacity is well beyond the requirements of the tested video stream. (The WLAN average delay is 22 ms.)

The results in Figure 7 show that the first frames of a stream cause a delay spike. This is because the first intra frame is substantially larger than of the following inter frames. Excluding the 9.6 kbps GSM link, the average delay decreases between 592 and 776 ms after 50 seconds (100 frames). The different delay characteristics demonstrate that for improving of the performance, the video encoder should be aware of available bandwidth as well as the communication delay [1].

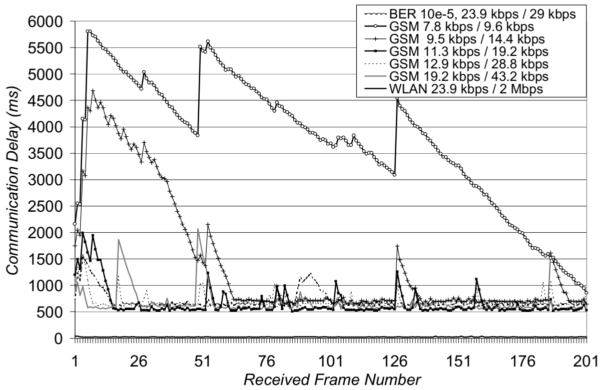


Figure 7. Preliminary video transfer delay results.

VII. CONCLUSIONS AND ONGOING WORK

A demonstrator has been designed and implemented for developing end-to-end real-time video services for wireless access technologies. The demonstrator research covers protocol layers from data link to video encoding. The work for completing and enhancing of the VCP functions and the developing of more accurate link/network models for the link simulation are currently in progress. The system has been preliminarily tested with WLAN, Bluetooth, and GSM data, and more testing has been done using the link simulation. Next, a new video compression method H.26L [4] will also be implemented. H.26L is especially suited for very low bit rate, real-time mobile applications. Dynamic adaptation of the encoder and VCP according to feedback information will be developed to further enhance the video stream transfer.

REFERENCES

[1] Dapeng Wu, et. al., *Scalable Video Coding and Transport over Broadband Wireless Networks*, Proceedings of the IEEE, Vol. 89, No. 3, January 2001.

[2] Hunt Engineering, *Hepc8 Full Length PCI Heron Module carrier USER MANUAL,* www.hunteng.co.uk/, 2001.

[3] ITU-T Recommendation H.263, *Video coding for low bitrate communication*, January 1998.

[4] ITU-T Telecommunications Standardization Sector, Gisle Bjontegaard, *H.26L Test Model Long Term Number 8 (TML-8) draft0*, 2001.

[5] ITU-T Telecommunications Standardization Sector, *Video Codec Test Model Near Term*, Version 7 (TMN7), February 1997.

[6] Ming-Ting Sun, Amy R. Reibman, *Compressed Video over Networks*, Marcel Dekker Inc., 2001.

[7] Yao Wang, Qin-Fan Zhu, *Error Control and Concealment for Video Communication: A Review*, Proceedings of the IEEE, Vol. 86, No. 5, May 1998.

[8] Bob Quinn, David Shute, *Windows Sockets Network Programming*, Addison Wesley, 2000.

[9] Microsoft, *Win32 Software Development Kit (SDK)*, Microsoft Corporation, 2000.

[10] Bluetooth Starter Kit product information, http://www.ericsson.com/microe/, 2001.

[11] K. Tikkanen, et. al. *Advanced Prototype Platform for a Wireless Multimedia Local Area Network*, EUSIPCO'2000, September 5 -8, 2000, Tampere, Finland, pp. 2309-2312.

[12] *Specification of the Bluetooth System*, v1.1, February 22, 2001, available at http://www.bluetooth.com/.

[13] F. J. MacWilliams, N. J. A. Sloane, *The theory of error-correcting codes*, North-Holland, Amsterdam, 1988.

Publication [P12]

Suhonen J., Hännikäinen M., Lehtoranta O., Kuorilehto M., Niemi M., Hämäläinen T., "Video Transfer Control Protocol for a Wireless Video Demonstrator", IEEE International Conference on Information Technology: Coding and Computing (ITCC 2002), pp. 462-467, Las Vegas, USA, April 8-10, 2002.

Video Transfer Control Protocol for a Wireless Video Demonstrator

Jukka Suhonen[1], Marko Hännikäinen[1], Olli Lehtoranta[1], Mauri Kuorilehto[1], Markku Niemi[2], and Timo Hämäläinen[1]

[1] Tampere University of Technology
Institute of Digital and Computer Systems
Korkeakoulunkatu 1, FIN-33720 Tampere, Finland
Tel. +358 3 3115 2111, Fax +358 3 3115 4575
jukka.suhonen@tut.fi, marko.hannikainen@tut.fi

[2] Nokia Mobile Phones
Sinitaival 5, FIN-33721 Tampere, Finland
Tel. +358 50 511 7320, Fax +358 10 505 7606
markku.t.niemi@nokia.com

Abstract

Real-time streaming video is expected to emerge as a key service in different telecommunications systems, including wireless networks. This paper presents the functionality and implementation of a wireless video control protocol (VCP). The protocol has been implemented for developing the functionality for real-time video stream transmission over heterogeneous wireless network technologies. VCP is embedded into a wireless video demonstrator. The demonstrator consists of Windows NT hosts containing a real-time H.263 encoder, video stream parsing functionality, and several network connections, such as Wireless LAN, Bluetooth, and GSM data. The protocol contains functionality for protecting the video stream transfer and adapting different network technologies together.

1. Introduction

Wireless connectivity is emerging as a ubiquitous networking technology. There are networking technologies already available for legacy data users, including mobile cellular networks, Wireless Local Area Networks (WLAN), Wireless Personal Area Networks (WPAN). Multimedia communication used in point-to-point or point-to-multipoint conversations can be seen as the next step of wireless communication services.

To enable an end-to-end multimedia conversation or real time streaming, the quality of service (QoS) support of a network is required. QoS functionality is emerging into wireless networks, based on the advancing development work on telecommunication standardisations. The support for QoS over separate links is first required, while the adaptation of the different links together for establishing and end-to-end service is the next critical element.

The seamless service implementation on wireless access technologies has been researched in the Institute of Digital and Computer Systems at Tampere University of Technology. A wireless video demonstrator has been implemented for developing control protocols and QoS support functions for real-time services [14]. The control of video transfer, its adaptation over heterogeneous wireless links and networks, and the overall control for an end-to-end service are managed by a video control protocol (VCP) that is a central entity of the demonstrator. The protocol is used for developing, implementing, and testing required functionality for improving the quality of the real-time video transfer. VCP contains its internal data processing functionality, and it also controls lower layer protocols and higher layer video encoding for enabling an uninterrupted service.

This paper is organised as follows. In Section 2, the architecture of the wireless video demonstrator is introduced. In Section 3, VCP is presented, consisting its functional layers, packet structures, connection procedures, configuration, and implementation. Conclusions and future work are given in the final section.

2. Video Demonstrator Architecture

The functional reference model of the video demonstrator is presented in Figure 1. The architecture consists of three main layers: video encoding/decoding, video adaptation with VCP, and link/network layers. The reference model has been introduced, as the actual implementation of the demonstrator alternates according to the operational environment and used network technologies. The middle layer has been divided into two sub-layers for differentiating the video contents related functions (video stream adaptation) and video transfer functions (VCP).

VCP centralises the control on a heterogeneous delivery system. The video encoder/decoder layer is independent of other layers, thus enabling the use of existing standard encoders and decoders. Similarly, standard link or network technologies may be directly utilised for video transfer. VCP functionality adapts the different layers together.

The basic architecture of the demonstrator is a client-server system, in which the connection between the

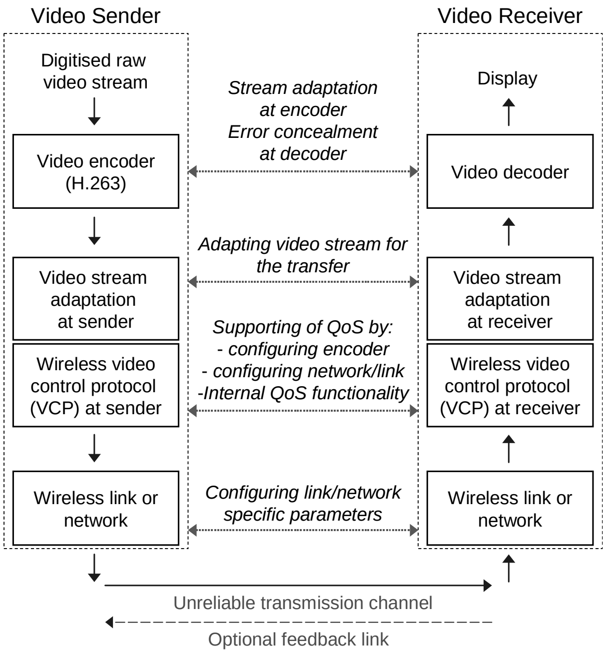


Figure 1. Reference architecture of the demonstrator.

entities can be realised using different technologies. The client is referred to as *video receiver* and the server as *video sender*. Video receiver is a terminal with varying processing capabilities and network connection types. The video sender is referred to being a high capacity server.

The video encoder produces an encoded video stream from a raw digitised video input. Real-time video capture for encoding can be used, and has been implemented in the current demonstrator [5]. For video encoder/decoder layer, the basic functions are to adapt a stream at the sender and conceal occurred errors at the receiver end. The quality of the decoded video is measured by frame loss and by computing the Peak-Signal-to-Noise-Ratio (PSNR) against the original video material. Different video adaptation tools will be developed and tested with the demonstrator [12]. The encoded video stream is prepared for the wireless transfer by the video stream adaptation and VCP layers. This layer processes the video stream for robustness against wireless link errors [11].

The video sender and receiver operate in Windows NT host computers [3]. Network Interface Cards (NIC) are attached to the Windows PC through a device driver that operates under the legacy UDP/IP protocol stack. The sender and receiver modules can also connect to NIC device drivers directly without a legacy protocol stack. In this case, the connection is made through tailored components.

The basic architecture of the system contains video input, video encoding, and video sending at the sender host, and video receiving, video decoding, and displaying at the receiving host. The video stream adaptation module is implemented in video encoding/decoding modules. The VCP is implemented in video sending and receiving modules, which are responsible for transferring the video packets from sender to receiver using an underlying link or network. In addition, a link simulation module has been implemented for simulating video transfer locally. Different demonstrator system modules provide User Interfaces (UI) for the configuration and collecting of measurement data.

3. Video Control Protocol (VCP)

VCP contains different functional layers for the protecting of video packet transfer and for managing connections between the video sender and receivers. The functional architecture is presented in Figure 3. The protocol has been divided into two planes for control and data processing functions. The central control module of VCP is the traffic class definition module. The class defines what functional layers of the data processing are applied and controls their operations. VCP collects statistical information of its operation. The information is delivered to UI for further processing, displaying, and storing [10].

When a video packet is given to VCP, the packet is time stamped and put into a queue. After the queue scheduler takes the packet from the queue, it is prepared for sending by adding redundancy Forward Error Correction (FEC) coding, Cyclic Redundancy Check (CRC) fields, and by fragmenting the packet to smaller segments, if these are required by the used traffic class or by the underlying link/network. VCP uses the traffic class definitions to control both the packet sending and receiving procedures implemented in the link/network access module. The queuing has been placed relatively high on the functional stack on VCP. This eases the queue management, as packets can be easily added, removed, and reorganised in the queue. Also, it is assumed that the video sender contains enough processing power for performing the required data processing functions without significant delays.

3.1. Video Stream Adaptation

To support unequal error protection by FEC and retransmissions, the video stream adaptation layer contains H.263 frame parser and assembly functions. At the module, a standard stream is first partially decoded. The bit stream is re-encoded to more error robust form by taking into account the desired QoS and network capabilities. The video stream adaptation layer acquires this information from VCP. The adaptation layer also separates H.263 encoded headers, motion vectors, and the macroblock data [6]. The separated picture components are placed into separate packets, a traffic class for each

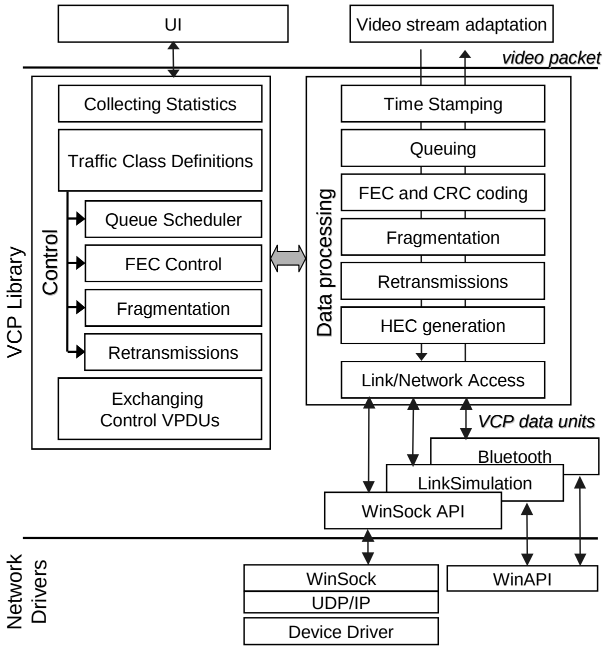


Figure 3. Functional arhictecture of VCP.

packet is assigned, and the packets are given to VCP for transfer.

3.2. VCP Data Unit

The video sender and video receiver exchange video traffic and control messages by using VCP data units. The structure of the unit packet is presented in Figure 2 and the different fields are explained in detail in Table 1.

VCP exchanges control messages that are packets without the payload video data between video sender and receiver for connection establishment, disconnection, synchronisation, acknowledgements, and for the exchanging of the traffic class definitions. Each packet carries timing information of its send time, traffic class, data content, and fragmentation status.

3.3. Traffic Classes

The protection of video is based on traffic differentiation by assigned the video packets with different traffic types. A class currently has the following parameters, while new parameters and classes are developed as the research continues:

- Ageing time
- Use of user configured re-transmissions
- Use Automatic Repeat Request (ARQ)
- Use of CRC for payload content
- Use of FEC for content
- Urgency

Ageing time gives the time-to-live for a video packet, and is set to be shorter for real-time applications. For

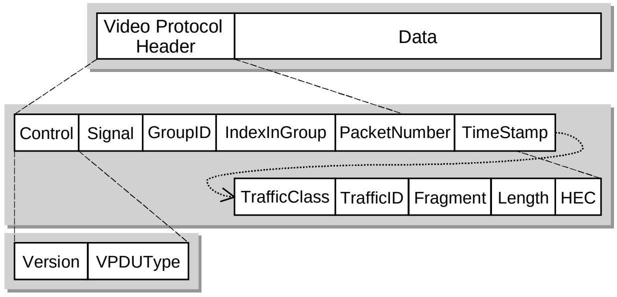


Figure 2. VCP data unit structure.

retransmissions, a sliding window and the sending of negative acknowledges for missing packets are used. ARQ is mainly targeted for control frames, and traffic classes where real-time requirements are not so strict. Also, the retransmission policy can be configured by a user. For a video stream that does not require a strict defined transfer delay, but tolerate buffering for maximising the video quality, this is expected to be suitable parameter.

Transmission errors are always checked from VCP data unit header part. In addition, error detection can be extended to cover whole content by defining this parameter for the traffic class. In this case, a separate error detection field is added after the payload of the data unit. The error detection field is calculated over the payload using a 32-bit CRC. The utilised CRC generator polynomial is the same as used in the IEEE 802.11 WLAN [2]. The video data with transfer errors can still be given to the user, if this is defined by the traffic class.

Convolutional coding and block coding are the two major forms of FEC coding. The current FEC implementation includes two alternatives: Reed-Solomon coding and byte repetition coding. Reed-Solomon codes are linear block-based error correcting codes [13]. The protocol uses (255,223) over GF(256) parameters for the Reed Solomon code. The byte repetition coding is a simple technique in which same data is repeated three times. Because errors occur often in bursts, instead of repeating a bit three times, whole content is written once, after which it is repeated.

At the moment, the user assigns the identification number of the algorithm when building a traffic class. The urgency parameter is a user assigned urgency value for the traffic class. It is an integer value used by the queue scheduling algorithm of VCP. A higher value indicates higher urgency. The UI for traffic class construction is presented in Figure 4.

3.4. Connection Establishment

When establishing a connection between video sender and receiver, VCP requires that the sender is acting as server and listening to incoming connection requests. The connection establishment is thus initialised by the receiver client. After the connection has been created, both parts

Field	Length/B	Description
Control:	1	Consists of Version and VCP data unit type (VCPDUType)
Version VPDUType		Version number of the used protocol. Used to distinct user data and protocol messages (synchronisation, acknowledgement, and connection establishment related messages)
Signal	2	Used for the underlying protocol specific signalling during the connection establishment and tier-down procedures. When connection is established, the high order byte is used for VCP signalling and the low order byte is reserved for lower level signalling. VCP signalling e.g. informs the receiver to acknowledge received packets, when the sender needs to free buffer space.
GroupID	1	Specifies packets that belong to the same group
IndexInGroup	1	The order of packet in its group. 0 is reserved for the group content packet. This is needed to deliver packets of a single group in right order.
PacketNumber	2	Running number of VCP packets. Each traffic class has own numbering, which enables efficient retransmission scheme.
TimeStamp	4	Time stamp of the packet set by the sender. The field contains the clock of the VCP sender at the stamping time. When the receiver sends messages to the sender, the receiver converts its local time to correspond the server clock.
TrafficClass	1	Traffic class of payload data. The sender and receiver exchange the same traffic class definitions and are able to determine the needed error recovery procedures and packet handling.
TrafficID	1	Identification of the type of payload data. This is used to help Video Protocol user to distinct packets between same traffic classes and is transparently transferred by VCP.
Fragmentation	2	Inform if fragmentation has been used, contains also the fragment number
Length	2	Length of payload data
HEC	4	Header Error Control (HEC), detects errors from the header fields
Data	0-Max	0-Maximum data unit of bytes. The maximum unit size depends on the underlying link/network connection, and on the traffic class definitions.

Table 1. VPC data unit fields.

can send and receive simultaneously and either the server or client can terminate the connection.

When the video sender acknowledges a connection request, the receiver initiates a synchronisation procedure. The synchronisation is performed for determining the time difference between the internal clocks of the sender and receiver.

The clock difference is acquired by using the Cristian's algorithm [1]. Algorithm calculates round-trip-time, which is used in queuing and retransmissions. Synchronisation message exchange contains two messages: synchronisation requests send by the VCP receiver and synchronisation confirms send by the VCP sender. Although packet timestamps usually contain the VCP sender time, synchronisation messages carry the local time of both hosts. Several synchronisation messages are exchanged to get more reliable value for the round trip time.

After the synchronisation, the video sender transmits the traffic class definitions to the client. The receiver responds by sending a connect confirm message.

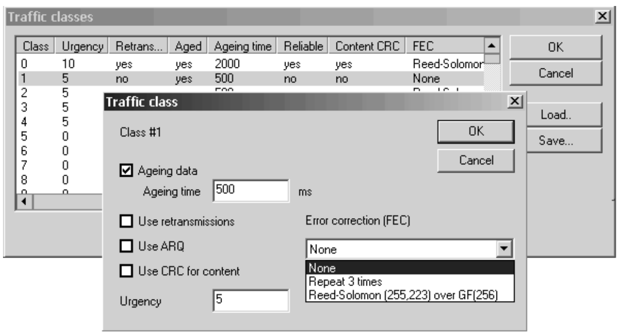


Figure 4. Traffic class construction UI.

Synchronisation procedure can also be initialised, if there has not been any incoming traffic for a predetermined interval. The purpose is to detect disconnected sessions and to adapt in possible changes in link/network layer. Repeated synchronisation also corrects the clock drift caused by clocks running at different speeds on receiver and sender host PCs.

VCP performs a link/network measurement procedure at connection establishment and also periodically during operation for determining an estimate for the supported and maximum throughputs for the connection. This information is used by the video encoder and video stream adaptation functions for scaling the bit rate and the protection of a transmitted stream.

3.5. Sending User Data in Groups

As a single H.263 video frame (or other type of encoded frame) is separated to several packets, it is required to group the packets containing information of the same video frame together and to transfer them as a single group. VCP allows a user to group packets, while each packet can have its own traffic class definition. Groups are given to the user (video adaptation layer) of the VCP receiver in the same order in which they were issued to the VCP sender and the internal order of packets within the group is preserved. Depending on the traffic class definitions, a missing packet can result into the discarding of the whole group it belongs to. This is practical in situations where no retransmissions can be applied because of missing feedback link or strict timing requirements of the traffic class. As the whole group can turn to be unusable because of a single missing packet, discarding the remaining packets saves bandwidth.

The VCP receiver needs to know the size of the group and the traffic classes for packets belonging to the group in order to efficiently determinate whether the whole group can be delivered to user. A separate content packet for each group is sent for this purpose.

3.6. Queue Management and Scheduling

In the internal queues of VCP, packets having the same traffic class are located in a same queue. The VCP queue scheduler is used for deciding the next packet to be taken from the queues.

When the ageing time is set, a packet is to be delivered within the given time. Its internal priority is increased as the waiting time in output queues increases. When the ageing time approaches, the packet is discarded as obsolete.

Although a packet with the highest urgency of each class is most likely to be selected, the scheduler prevents starvation of the lower urgency packets. Each queue and each packet has an internal priority value. Packet priority is set to the value zero, when a packet is put into a queue. Queue priority is also set to zero, when a connection is established. Queue urgency has the same value as defined for the corresponding traffic class. The queue structure with different priority/urgency values is presented in Figure 5.

The scheduler selects a packet in the following algorithm. The queue urgency and priority are summed with packets priority. The packet having the highest value is taken. Because the internal packet priorities in a queue are in order, only one calculation per each queue must be performed. After the packet selection, the priority of the queue from which the packet was taken is set to zero. For other queues the priority is increased by one. Finally, the priority of all packets is increased by one.

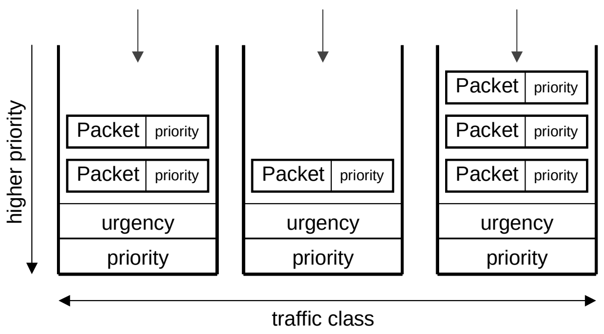


Figure 5: VCP queue architecture.

3.7. Link/Network Access

When using the basic ARQ, every received packet is acknowledged immediately after receiving the packet. For non-ARQ packets only negative acknowledge is sent, when VCP receiver detects missing packet and retransmission is specified for the traffic class. However, after receiving a fixed amount of non-ARQ packets, an acknowledge message is sent. Because the VCP sender does not know what packets have been successfully received, it must keep all sent packets in a buffer. By sending an acknowledge packet for non-ARQ packets, buffer space can be released.

When configurable retransmissions are applied, the ageing time is used by the VCP receiver to determine, if packet can be retransmitted within the required timing limits. A retransmission is asked, if the sum of current round-trip-time and packets age is smaller than ageing time. The retransmission decision for a traffic class that does not use negative acknowledgements is made by the sender in a similar manner. In this case, the round-trip-time is calculated as half of its value.

Several acknowledgements can be carried with one packet by forming a list of packet number and traffic class pairs. Negative acknowledgements (NAK) are constructed likewise, but the fragment field is also included. The negative acknowledgement packet contains a list of all missing packets, but it can also be used to acknowledge received packets, as the correctly received packets can be concluded from the list of missing packets.

Header error control (HEC) field is constructed always just before sending the packet. This is because VCP uses the header fields for transferring protocol signalling information that is regularly updated. HEC is calculated over the header using the same 32-bit CRC as used for content.

Currently supported protocols for link/network access include standard UDP/IP stack, Bluetooth, and the link simulation. UDP/IP is used over variety of wireless link/network technologies. The Bluetooth link connects the VCP to the Bluetooth protocol host stack of the local PC, which is why a separate link functions have been implemented [7]. The link simulation implemented for the demonstrator simulates a link/network between VCP sender and receiver. The simulation module can simulate a connection consisting of several serialised links with different properties including link delay, bandwidth and bit and burst error rates.

3.8. Collecting Statistics

VCP collects statistics from every received and sent packet. The statistics is viewed in a separate UI, or can be saved to a file for later processing. The collected information contains the number of packets, throughput, mean delay, and delay variance that are calculated using the timestamp information. The throughput is separated for the overhead and video data portions. Received frames are separated according to successful transmissions, with and without error corrections. Missing frames are also indicated.

3.9. Implementation in Windows NT

The protocol was specified using the Specification and Description Language (SDL) [8]. SDL is a formal language that enables to convert the specification into an executable simulation. By simulating, the functional correctness of the protocol was verified.

VCP is implemented as a single 32-bit dynamic link library (DLL) of Windows NT. The library is implemented using the C++ language and networking is implemented by using Windows Sockets 2 API (WinSock2) [9]. A video library user can interact with the library by the defined video transfer interface or by a separate management user interface. Several lower level protocols can be used without major changes in the existing library. If a protocol does not support two-way connections, two separate connections can be established.

In the link simulation module, packets are exchanged by using shared memory between the VCP sender and receiver. Shared memory is managed with Windows API memory management functions by using virtual memory. The allowing of reading and writing to the shared memory is managed by using global mutexes provided by WinAPI [4]. The support for Bluetooth link is implemented in a similar manner. An externally implemented driver is used for the actual sending and receiving data to Bluetooth.

4. Conclusions and Future Work

The functionality and implementation of a wireless video control protocol (VCP) has bee presented. The protocol has been implemented for developing the functionality for protecting real-time video stream transmission over heterogeneous wireless network technologies. Real-world wireless links, such as WLAN, Bluetooth, and GSM data have been targeted, and general link or network properties can be simulated. VCP performance has been found adequate in the first tests. The delays caused by the protocol processing on the video stream are not significant concerning the overall end-to-end delay of the network connections.

VCP is used for testing what kind of video support and protection functionality is required over different link/network layers. It is also used as a central control entity to adapt different networking technologies and their QoS together, and to configure the video encoder to perform at best quality on the current network environment. As such, the VCP (its control plane) operates as a dynamic wireless video agent for improving user services.

References

[1] Coulouris, Dollimore, Kindberg, *Distributed Systems – Concept and Design* Second Edition, Addison-Wesley, 1998.

[2] IEEE P802.11, IEEE Standard for local and metropolitan area networks: Wireless LAN Medium Access Control (MAC) and Physical Layer Layer (PHY) specifications, IEEE Standards Department, August 20, 1999.

[3] H. Custer, *Inside Windows NT*, Microsoft Press, 1992.

[4] Microsoft, *Win32 Software Development Kit (SDK)*, Microsoft Corporation, 2000.

[5] Olli Lehtoranta, Timo Hämäläinen, Jukka Saarinen, *Real-time Implementation of H.263 Video Encoder on TMS320C6201 Fixed Point DSP*, European Signal Processing Conference (EUSIPCO2000), Tampere, Finland, vol. 3., pp. 1457-1460.

[6] Draft ITU-T Recommendation H.263, *Video coding for low bitrate communication*, 2. May, 1996.

[7] Specification of the Bluetooth System, v1.1, February 22, 2001, available at http://www.bluetooth.com/.

[8] Homepage of the SDL-Forum, http://www.sdl-forum.org/, 19.10.2001.

[9] Bob Quinn, David Shute, Windows Sockets Network Programming, Addison Wesley, 2000.

[10] Marko Hännikäinen et. al, *TUTMAC: A Medium Access Control Protocol for A New Multimedia Wireless Local Area Network*, IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC'98), Boston, USA, September 8-11, 1998, pp. 592-596.

[11] Ming-Ting Sun, Amy R. Reibman, Compressed Video over Networks, Marcel Dekker Inc., 2001.

[12] Yao Wang, Qin-Fan Zhu, Error Control and Concealment for Video Communication: A Review, Proceedings of the IEEE, VOL. 86, NO. 5, May 1998.

[13] F. J. MacWilliams, N. J. A. Sloane, The theory of error-correcting codes, North-Holland, Amsterdam, 1988.

[14] Marko Hännikäinen et. al, *Architecture of a Wireless Video Transfer Demonstrator*, International Zurich Seminar on Broadband Communications, ETH Zurich, Switzerland, February 19-21, 2002, accepted.

Tampereen teknillinen korkeakoulu
PL 527
33101 Tampere

Tampere University of Technology
P. O. B. 527
FIN-33101 Tampere Finland