TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY
Julkaisu 858 • Publication 858

Sakari Junnila

Modern Digital Interfaces for Personal Health Monitoring Devices

Tampere 2009

Tampereen teknillinen yliopisto. Julkaisu 858
Tampere University of Technology. Publication 858

Sakari Junnila

Modern Digital Interfaces for Personal Health Monitoring Devices

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB222, at Tampere University of Technology, on the 27th of November 2009, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2009

ISBN 978-952-15-2277-2 (printed)
ISBN 978-952-15-2281-9 (PDF)
ISSN 1459-2045

ars longa,
vita brevis

Abstract

The objectives of this work were to find out what interfacing technologies and emerging standards can be adopted from the personal computer market to medical devices targeted for personal and home use, and to gain understanding of technical and regulatory limitations regarding their use in medical applications through implementing prototypes of these interfaces. The growing demand for home health care services, attributed to the increase in population of elderly people in the society, requires development of new remote and home health care systems and services which can utilize cost-effective PC technologies and devices already present at homes.

The Thesis studies modern digital interface technologies, emerging standards and their use in medical devices. The interface technologies of interest are computer system peripheral interfaces and wireless personal area networking (PAN) technologies. The work aims at gaining understanding of technical and regulatory limitations and benefits regarding their use in medical applications, especially in personal health monitoring applications at home. Several steps are taken and prototypes built to assess the feasibility of these technologies and to obtain results usable in practical design cases. The Thesis also presents the current state of medical device regulation and standardization, of which the latter especially has been under rapid development over the recent years. Although the safety aspects of the developed implementations are addressed, the Thesis does not cover the full scope of risk analysis of networked medical devices or medical device software.

The introductory part of the Thesis begins with the presentation of currently used digital interfaces and their design. Then, the medical device is defined, and safety, regulation, standardization, security, and privacy issues related to medical devices are presented, including current state of medical device interface standardization. Also, the use of PC and consumer electronics technology in medical devices and how medical device networking is changing the medical device design are discussed. The concept of personal health monitoring and its applications at home are presented, followed by a presentation of health monitoring needs and applications in various healthcare facilities. Finally, the direct contributions to the field of personal health monitoring system and device design contained in this work and derived conclusions are presented.

A method for interfacing medical sensory devices of a commercial patient monitoring system to a standard PC was developed. This work shows how medical devices can be connected

using a standard cable based (USB) interface, and presents different hardware implementation strategies and software related issues. Cable connected devices are more vulnerable to electrical hazards than wireless devices. This can be alleviated by electrical isolation of the cable interface. Different isolation strategies were studied and a method to isolate USB data signals was developed.

A medical monitoring chair for ballistocardiogram (BCG) recording with a novel electromechanical film sensor based method was implemented both as a traditional analog amplifier design and as a wireless digital measurement system. The analog system was used in a clinical trial to gain experience of the medical device approval process. The digital system is shown to provide similar signal quality as the analog system with less costly equipment and with increased device mobility. The work also provides insight into the different data transfer needs of various human originated signals.

Methods and technologies for wireless medical data-acquisition systems were studied and demonstrated. The application areas for wireless PAN devices in hospitals were surveyed and technologies assessed. The IEEE 802.15.4 wireless interface technology was selected for a wireless medical monitoring system (BCG chair) and further sensor network implementations. Networking of personal healthcare devices using the selected technology was studied by implementing a wireless sensor network. The interface technology was expanded to support networking by adding a Zigbee network communication protocol layer. Personal health monitoring devices were attached to the sensor network and fitted to a real-life home apartment.

Current medical devices rely much on proprietary interfaces and even more on proprietary data presentation models making if often impossible to use devices from different manufactures together. The main claim of this work is that standardized interfaces should be used in medical devices to obtain reliable, safe, and more interoperable devices cost-effectively. A higher level standard to specify common nomenclature for physiological variables and to enable communication across different interface technologies is also needed, and attempts to develop one exist.

Acknowledgments

The research work presented in this Thesis has been accomplished in the Department of Signal Processing and of Department of Computer Systems at Tampere University of Technology (TUT), Tampere, Finland.

I wish to express my gratitude to my supervisors Adjunct Prof. Alpo Värri and Adjunct Prof. Jarkko Niittylahti for their support and guidance, and for making it possible to carry out the research presented in this Thesis. It was Adjunct Prof. Niittylahti how induced and motivated me to an academic career as a third year masters student, and supervised my work in the Department of Computer Systems until the end of his professorship. It is greatly thanks to Adjunct Prof. Värri, who took me into his research group in the Department of Signal Processing, that I could carry out my thesis work to this end.

I would like to thank the reviewers of my Thesis, Malcolm Clarke, Ph.D., and Prof. Raimo Sepponen for their time and efforts and especially for their in-depth and constructive comments on the manuscript. I am also thankful for Prof. Irek Defee and Prof. Jari Hyttinen with whom I have had the privilege to work with.

I would also like to thank my co-authors, friends and colleagues of the past years at Tampere University of Technology, especially Alireza Akhbardeh, Dr. Tech., Juha Alakarhu, Dr. Tech., Jarmo Alametsä, M.Sc., Magnus Armholt, M.Sc., Laurentiu Barna, Dr. Tech., Florean Curticapean, Dr. Tech. Harri Kailanto, M.Sc, Mikko Koivuluoma, M.Sc, Jouni Paulus, M.Sc., Jarkko Ruoho, Tero Sihvo, M.Sc., Jarno Tanskanen, Dr. Tech., Riku Uusikartano, M.Sc., Antti-Matti Vainio, M.Sc., Antti Vehkaoja, M.Sc., and Mari Zakrzewski, M.Sc.

During the past years, I have had the pleasure to work in collaboration with GE Healthcare Finland (formerly Datex-Ohmeda), Nokia, VLSI Solution, Tampere University Hospital, Arcticare Technologies, Kotosalla-säätiö, Lempäälän sosiaali- ja terveystoimi (Lempäälä Social Welfare and Health Services), STT Condigi (formerly Pikosystems), Suomen ensiapupalvelu, TeliaSonera, Tieto (formerly TietoEnator), Turun Orthofysio, UPM-Kymmene Corporation, VTT Technical Research Centre of Finland, YH Länsi, University of California, Aarhus University, Alexandra Institute and the TUT departments of Automation Science and Engineering, Biomedical Engineering, Electronics, and Mechanics and Design. I would especially like to thank Väinö Turjanmaa, MD, Ph.D., Dos. Tiit Kööbi, MD, Teemu Koivistoinen, M.Sc., MD.,

and Marjaana Sipilä, BM, RN, and Pirjo Järventausta, RN, Tampere University Hospital, Prof. Jukka Lekkala, Lab. Mgr. Jarmo Poutala, Sr. Lab. Tech. Seppo Mikkola, Lasse Kaila, M.Sc., Jarmo Verho, M.Sc., and Timo Vuorela, M.Sc from Tampere University of Technology, Juho Merilahti, M.Sc, from VTT Technical Research Centre of Finland, Simon B. Larsen, Ph.D., from the Alexandra Institute, and Aki Backman, M.Sc., MBA, and Markku Roiha, M.Sc. from the former Datex-Ohmeda, for their contributions, including comments, guidance, performed measurements, and technical assistance.

The financial assistance from the following funding organizations and programmes is hereby greatly acknowledged: The Finnish Funding Agency for Technology and Innovation (Tekes) - Electronics for the Information Society Technology Programme (ETX) (1997-2001), The Academy of Finland - The Finnish Centre of Excellence Programme (2000-2005) and The Proactive Information Technology Programme (2002-2005), Tampere Graduate School in Information Science and Engineering (TISE), Industrial Research Fund at the Tampere University of Technology - Pekka Ahonen foundation, and Finnish Cultural Foundation - Artturi and Aina Helenius Foundation.

I wish to express my deepest gratitude to my parents, Kirsti and Jyrki Junnila, and to all my relatives and friends for their support and encouragement through my studies. Most of all, I thank you Heidi, for your loving support.

Tampere, November 2009
Sakari Junnila

Abbreviations

AIMD	Active Implantable Medical Devices
A/D	Analog to Digital
A/V	Audio/video
API	Application Program Interface
ASIC	Application Specific Integrated Circuit
BAN	Body Area Network
BCG	Ballistocardiogram
CEN	Comité Européen de Normalisation, European Committee for Standardization
CENELEC	Comité Européen de Normalisation Electrotechnique, European Committee for Electrotechnical Standardization
CHA	Continua Health Alliance
CircMon	Circulation Monitor
CPU	Central Processing Unit
CSMA/CD	Carrier Sense Multiple Access With Collision Detection
ECG/EKG	Electrocardiogram
EMFi	Electromechanical Film. EMFi is a registered trademark of Emfit Ltd, Vaajakoski, Finland.
EMI	Electromagnetic Interference
ERC	European Radiocommunications Committee
FDA	Food and Drug Administration

FPGA	Field-Programmable Gate Array
GHTF	Global Harmonization Task Force
HCI	Human-Computer Interface
HL	Higher level
HW	Hardware
Hz	Hertz
IC	Integrated Circuit
ICG	Impedance Cardiogram
ICU	Intensive care unit
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IHE	Integrating the Healthcare Enterprise
I/O	Input/output
IP	Internet Protocol, Intellectual Property
ISM	Industrial, Scientific and Medical
ISO	International Organization for Standardization
ITU	International Telecommunication Union
ITU-R	ITU Radiocommunication Sector
ITU-T	ITU Telecommunication Standardization Sector
IVDD	In Vitro Devices Directive
LAN	Local Area Network
LL	Lower level
MAC	Medium Access Control
MDA	Medical Device Agency
MDD	Medical Device Directive

NB	Notified Bodies
OS	Operating System
OSI	Open Systems Interconnection
PAN	Personal Area Network
PC	Personal Computer
PSPICE	Personal Computer Simulation Program with Integrated Circuit Emphasis
RF	Radio Frequency
RS-232	Recommended Standard 232
SPI	Serial Peripheral Interface
SRD	Short Range Device
SW	Software
UART	Universal Asynchronous Receiver/Transmitter
UI	User Interface
UML	Unified Modeling Language
USB	Universal Serial Bus
USB-OTG	Universal Serial Bus On-The-Go
UWB	Ultra-Wideband
WG	Working Group (ISO), Work Group (CEN)
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network
XML	eXtensible Markup Language

Contents

List of Figures

List of Publications

This Thesis consists of an introductory section and the following publications. In the text, these publications will be referred to as [P1], [P2],..., [P8]. The publications are reproduced here with kind permissions from the publishers.

[P1] S. Junnila and J. Niittylahti, "Implementing USB Function Devices," In *Proceedings of the 18th IASTED International Conference on Applied Informatics (AI 2000)*, Innsbruck, Austria, Feb 14-17, 2000, pp. 579-582.

[P2] S. Junnila and J. Niittylahti, "A Patient Monitoring System Based on Standard PC Platform," In *Proceedings of the 15th International EURASIP EuroConference BIOSIGNAL 2000*, Brno, Czech Republic, June 21-23, 2000, pp. 348-350.

[P3] S. Junnila, J. Ruoho, and J. Niittylahti, "Medical Isolation of Universal Serial Bus Data Signals," In *Proceedings of the 9th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2002)*, Dubrovnik, Croatia, Sep 15-18, 2002, pp. 1215-1218.

[P4] S. Junnila and J. Niittylahti, "Use of Bluetooth in Medical Systems," In *Proceedings of the 19th IASTED International Conference on Applied Informatics (AI 2001)*, Vol. 1, Innsbruck, Austria, Feb 19-22, 2001, pp. 488-494.

[P5] S. Junnila and J. Niittylahti, "Wireless Technologies for Data Acquisition Systems," In *Proceedings of the International Symposium on Information and Communication Technologies (ISICT 2003)*, Dublin, Ireland, Sep 24-26, 2003, pp. 132-137.

[P6] S. Junnila, A. Akhbardeh, and A. Värri, "An Electromechanical Film Sensor based Wireless Ballistocardiographic Chair: Implementation and Performance," *Journal of Signal Processing Systems*, Vol. 57, Issue 3, 2009, pp. 305-320.

[P7] M. Armholt, S. Junnila, and I. Defee, "A Non-beaconing ZigBee Network Implementation and Performance Study," In *Proceedings of the IEEE International Conference on Communications (ICC 2007)*, Glasgow, Scotland, UK, Jun. 24-28, 2007.

[P8] S. Junnila, M. Zakrzewski, A-M. Vainio, J. Vanhala, and I. Defee, "UUTE Home Network for Wireless Health Monitoring," In *Proceedings of the International Conference on Biocomputation, Bioinformatics, and Biomedical Technologies (BIOTECHNO 2008)*, Bucharest, Romania, June 29-July 5, 2008, pp. 125-130.

Chapter 1

Introduction

Computer-based systems have grown in performance and in complexity. It is often no longer feasible for a manufacturer to build proprietary computer systems with dedicated hardware and custom software. Instead, multipurpose hardware (HW) platforms are used for smaller systems, and PC technology for systems requiring more computing power. Omitting the very simplest systems, the devices and systems use some operating system (OS). If PC technology is used, either Windows or Linux is usually used as the OS. The increase of computing power has also changed the structure of computer interfaces from multipin connectors with simple communication protocols to simple connectors with complex communication protocols and multilayer protocol stacks. The interoperability related issues have become more important as the number of different physical interfaces has been reduced, and the interfaces are now able to support a wider range of applications. Because of these developments, standards have become complex and their development takes years.

In general, medical devices tend to develop with a longer delay than the consumer electronics devices. This is due to the safety and reliability requirements set to medical devices. The technology has to be tried and tested before it is adopted. It has been clear for some time that most future medical systems requiring computing power will be based on PC technology. However, the systems and devices still often use proprietary interfaces for several reasons, some technical and security related, but largely to protect obtained market shares. As technical requirements for interfaces increase, it no longer makes sense to invest in the development of proprietary solutions when tested and proven standard solutions exist. Also the system buyers have started to demand more interoperable solutions. The trend therefore is to use more standard interfaces in medical devices.

Personal health monitoring and health monitoring at home is an area in which huge growth is expected. When bringing devices to home, the manufacturers are confronted with an existing home computer and consumer electronics environment. The personal home computer is typically the centerpiece of the home monitoring system, often providing the gateway to internet

and remote services. Health monitoring devices designed for home should therefore be able to interact with the PC, e.g., to be able to connect to a interface provided by the PC. Consumers are often demanding and price aware, and may easily buy health monitoring devices from different manufacturers depending on the features, personal taste and pricing. The demand for interoperable personal health monitoring devices is clear.

Standardization is a key part of medical devices and systems. It is required to guarantee patient safety and the reliability of the systems. This Thesis gives an overview of different standards and ongoing standardization efforts closely related to digital interface implementation. Some relevant topics, like medical software standardization and risk analysis of networked medical devices are out of the scope of this Thesis, but are presented in brief.

This Thesis studies the use of modern consumer electronics technology in medical systems and especially in health monitoring applications targeted for personal and home use. It is clear that the physical bottleneck in the use of consumer electronics technology, and especially PC technology, is the interface between the devices. The main work in this Thesis has been on the use of modern digital interfaces in different health monitoring environments; how to use them and what problems will arise with their use. Cable based and wireless solutions have been studied, prototypes built, and devices tested with real subjects both in clinical and home trials. Implementations of these prototypes and related clinical trials have not only provided new information on interfaces but have also produced vast amounts of clinical data for others to use. The analysis of the recorded data has produced several international publications and two signal processing theses and more are expected. In this regard, the true value of the work of this Thesis will be defined by the future research of others.

1.1 Objective and scope of research

The objective of the research presented in this Thesis is to find out what interfacing technologies and emerging standards can be adopted from the personal computer market to medical devices targeted for personal and home use, and to gain understanding of technical and regulatory limitations regarding their use in medical applications through implementing prototypes of these interfaces. The following approaches are taken to reach this objective.

- Research methods for interfacing medical monitoring sensory devices from a commercial patient monitoring system. The objective is to show at hardware and software level how medical devices can be interfaced to a standard PC platform using cable based interface. Special emphasis is given to USB and its use in medical devices. Different USB device implementation strategies will be evaluated.

- Study the feasibility of technologies for wireless medical data-acquisition systems and survey usage areas for wireless personal area networking (PAN) technologies in hospitals.

Select suitable technology/technologies for implementing a wireless bio-signal monitor capable of interfacing to a PC.

- Implement a medical monitoring device (a ballistocardiographic chair) using a traditional analog wired based strategy and by using a modern digital wireless approach and compare the approaches. The work includes evaluating different data transfer needs of various human originated signals, including electrocardiogram (ECG), ballistocardiogram (BCG) and heart rate.

- Implement a wireless home health monitoring system. Research how to expand the use of wireless digital interfaces into wireless networks of devices and how to add commercial wired devices into the system.

1.2 Thesis outline

This Thesis is comprised of two parts: the introductory part of Chapters 1-8 and the bibliography followed by the eight publications carrying the main research results. The introductory part is organized as follows: Chapter 2 introduces the concept and definition of digital interface used in this Thesis, presents common interface standards relevant to this Thesis, and presents different approaches for implementing modern digital interfaces. Chapter 3 defines medical devices as used in this Thesis, and explains various regulatory and design issues special to medical devices. The latter part of the Chapter discusses some currently very relevant design issues of medical devices; use of PC technology, ongoing standardization work, and medical device networking. The main application area of this Thesis, the personal health monitoring, is presented in Chapter 4. This includes a brief presentation of signals and variables used for personal health monitoring at home, motivation for it, and practical implementation issues. The Chapter 5 looks at health monitoring in a wider perspective in different healthcare facilities, and looks at the use of computers and interfaces in healthcare facilities. It also briefly describes the history and special characteristics of critical care health monitoring. Chapter 6 summarizes the research results. The discussion and self-evaluation of the results are in Chapter 7, including discussion on future trends. Chapter 8 concludes the Thesis.

1.3 Summary of publication contents

Publication [P1] presents software and hardware implementation schemes for a custom USB peripheral. It presents the then new USB interface and considers its suitability for data acquisition applications. Different hardware architectures for USB device design are presented and evaluated. A test design case using a Lucent USS-820 interface controller and Hitachi H8 microcontroller is presented. Host software and device driver development issues are discussed,

and problems and limitations observed in the operating system USB support are presented. The paper concludes that from the hardware development side the USB peripheral development is relatively easy, but the software and driver development required are more complex than with the existing interfaces. It was also observed that the Windows environment does not suite applications requiring hard real-time requirements or low-latency, and that the Windows 98 USB support had major limitations.

Publication [P2] presents the use of a standard PC-platform in patient monitoring application, by means of connecting commercially available measurement instruments to a standard PC. The paper builds on the USB device development work presented in [P1]. The paper argues that PC technology is well suited to replace the custom proprietary platforms used in medical systems, and that out of the available interfaces USB would be the best choice for the presented monitoring system application. The developed prototype system is presented. Issues and limitations in OS USB stack, which partly contradict the USB specification, are discussed. The paper also argues that it would make sense to implement the USB support directly in the sensory measurement device, but notes that this would require us to solve some isolation and electrical safety related issues in certain applications. The USB isolation issues were later addressed in the work of [P3]. The paper concludes that the current PC systems and their USB support are not yet reliable enough to be used in critical care applications, but that the system could be used in less critical applications. USB bus itself was found to be robust and well suited for medical applications, but the bottleneck in performance and reliability is the device driver and operating system.

Publication [P3] evaluates different methods for USB data signal isolation and presents a prototype implementation of USB data signal isolation. Power signal isolation was not implemented in this paper, as there were existing solutions for performing it. The paper argues that USB will likely replace RS-232 and RS-485 in measurement applications and that isolation of the bus would be required to enable wide use in measurement systems requiring high sensitivity or safety. The paper presents the requirements of medical isolation and how they can be met. Specific issues related to optical, transformer, and capacitive isolation are discussed. The paper then presents the possible choices for USB data signal isolation. The isolation is implemented with the USB cable isolation method by using transformers and some optoisolators. The isolation implementation was tested with the prototype device implemented in [P1] and results compared against the USB specification. The isolation prototype was found to work with a 3 m USB cable, although some time delay constrains were not met. The implementation was found unreliable with a 5 m cable. The paper concludes that the developed isolation method is useful and functional with limitations, but it can't be used as such in open USB systems, as it causes too much upstream delay.

Publication [P4] is a technology study on the Bluetooth technology and a case study on its usability in medical systems at a time when first Bluetooth sample devices were just coming to

market. It was also used as an internal report in Datex-Ohmeda (now part of GE Healthcare) to give information on the potential of Bluetooth. The paper presents the throughput, performance, and interference features of the Bluetooth technology standard. Specified performance is compared to the results form Bluetooth performance studies. Communication setup and delays involved are discussed. The modern hospital environment is presented and the impact of new wireless technologies to it is discussed. Possible Bluetooth application areas in wireless medical systems are presented and evaluated.

Publication [P5] is a literature and technology study on the available wireless technologies and their suitability for a six-channel medical BCG/ECG data-acquisition system. Based on the study, technology choices were made for the development of the wireless BCG chair presented in [P6]. The paper introduces a term "mid-speed" to describe data transfer needs of well below the 1 Mbps range but over 1 kbps, and focuses on finding a suitable technology for 60 kbps transfer needs. The paper first presents the CEPT/ERC regulations on the use of the SRD radio bands from 400 - 6000 MHz. It then presents the available technologies, the open worldwide standards, the available closed systems, and the plain radio transceivers which can be used to implement proprietary protocols. Also UWB and its regulatory state are presented. The paper concludes that Bluetooth and Zigbee are viable solutions for the targeted application, but that Zigbee device and software support is not yet good enough for it to be selected. The paper also concludes that Zigbee may replace Bluetooth in new mid-speed applications when completed. Although Bluetooth was selected in the paper, later events postponed the start of the design of the wireless BCG chair. The wired BCG bio-amplifier presented in [P6] was built first, and when it was finished, the Zigbee hardware and software support had reached an adequate level.

Publication [P6] is a summary journal article which comprises the hardware development work related to the ballistocardiographic chair and its wireless version during three years of the ProHeMon project [Koi04c] and results of the tests done after the end of the project. The article presents the ballistocardiogram signal, its origin, history, and measurement methods. Next, the electromechanical sensor film (EMFi), its operation and features are presented. It then describes the two implementations made for BCG measurement using the EMFi sensor, a wired and a wireless system. A short summary of signal analysis methods work is given. The functionality (linearity and frequency response) of both the developed systems is evaluated thoroughly with several tests. The operation of the sensor is tested in various ways, including calibration tests with a mechanical vibrator, and an extensive test on the amplifiers is conducted. The systems are found to produce reliable ballistocardiogram signals for medical recordings.

Publication [P7] presents a partial Zigbee network layer implementation build on top of the IEEE 802.15.4 MAC and its measured performance. The stack was implemented as a separate layer on top of a 802.15.4 MAC which did not support beacon networks. The test showed that adding the network layer did not significantly affect the system performance or link throughput. It was also observed that with knowledge of the network structure, adding a waiting period

between packets sent can reduce the probability of a channel access failure without a decrease in throughput. The wireless BCG chair presented in [P6] was used as a test and demonstration platform for the developed network software. Later the Zigbee network was used to build home sensor network presented in [P8].

Publication [P8] presents a home sensor network developed to assist in home living of elderly people by means of wireless health monitoring. A proprietary sensor network was built on top of the Zigbee network layer developed in [P7]. A common sensor software and hardware interface was developed to enable joint use of sensor technology in three different projects. Custom radio-boards were built and interfaced to commercial and self-made sensors. A set-up consisting of four sensors was developed and tested in the test apartment in real home environment. The architectural overview of the system and main technical design choices are presented.

1.3.1 Author's contribution

This Thesis includes eight publications. To the author's knowledge, none of them have been previously been used for another person's academic dissertation. For publications [P1], [P2], [P4], and [P5] the writing, ideas, and implementations have been the work of the author.

Publication [P3] is co-written by Jarkko Ruoho, M.Sc. student. The author supervised and assisted in the research work and made the tests and recordings presented in the publication together with Ruoho.

The [P6] is written by the author except for the short section on Signal Analysis Methods, which was written by Alireza Akhbardeh, Dr. Tech. The author designed and made the wired bio-amplifier, designed and made the wireless electronics of the chair, including the bio-amplifiers, software for the wireless link, mounted the electronics and the electrodes on the chair, tested the system and the wireless link, and designed and performed the EMFi sensor calibration tests.

For publication [P7] Magnus Armholt, M.Sc., implemented the Zigbee protocol stack under the authors technical supervision and guidance. The author assisted in providing ideas and solutions for technical implementation issues, did part of the debugging and testing, and co-wrote the publication with Armholt. The author was also responsible for setting up and maintaining the hardware used.

For publication [P8], the author made the sensor network implementation, co-designed the sensor node software architecture, made the sample implementations of the sensor node softwares, wrote serial and SPI device drivers, wrote sensor drivers for scale and bed sensor, co-wrote ECG and blood pressure sensor drivers, programmed all the sensor nodes, wrote the sensor network related parts of the publication and compiled the final paper.

Chapter 2

Digital interfaces

Digital interface is a common interconnection between systems and devices in which information is exchanged using discrete numerical digits. In this Chapter the definition of digital interface is given and distinction to analog interfaces is made. A basic implementation of a digital interface consists of three identifiable parts, the physical interface, the application programming interface, and the communication protocol in between them. Their features and role in the interface design are presented. An interface which uses a non-shared medium and is capable of interconnecting with only one device is called a single device interface. Otherwise, if several devices can connect to the same interface, it is called multiple device interface. An interface can be cable based or wireless. Cable based and wireless interfaces have some distinct features which are presented along with some of the currently most used commercial digital interface standards. Data modeling is often used in designing the software data structures for the digital interface. Regardless of the nature of the interface medium certain basic architectures for digital interface implementation can be identified and are presented at the end of the Chapter.

Websters's Encyclopedic Unabridged Dictionary defines interface as "1. a surface regarded as the common boundary between two bodies or spaces", "3. a common boundary or interconnection between systems, equipment, concepts, or human beings", and specially for computer technology "4. a. equipment or programs designed to communicate information from one system of computing devices or programs to another" "b. any arrangement for such communication" [Web94]. Digital is defined (in computer technology) as "involving or using numerical digits expressed in a scale of notation to represent discretely all variables occurring in a problem" [Web94]. In this Thesis, we use the term *digital interface* to describe a common interconnection between systems and devices in which information is exchanged using discrete numerical digits. A more rigorous definition would have been *digital I/O (input/output) interface* could have also been used. The terms digital I/O and I/O interfaces are commonly used in computer architecture literature to describe peripheral interfaces for computer systems [Heu97]. With the development of microprocessors and increasingly intelligent peripheral devices, the border

between peripheral interfaces, computer communications and communication networks has become more vague. Terms such as digital interface bus or digital bus limit themselves to bus based solutions. The more generic term digital interfaces is used to better cover all possible forms interface technologies, wired and wireless.

Other commonly used interfaces in field of computer technology are the User interface (UI) and software application programming interfaces (API). User interface is the boundary between the computer system and the user, the means by which the users interact with the system, also known as the Human Computer Interface (HCI) or the Man-Machine Interface (MMI). Application programming interfaces, or software interfaces in general, are boundaries between blocks of software or software and hardware blocks within the computer system.

Devices can be attached to computer systems using digital or analog interfaces. Their differences are discussed in Section 2.1. The definition of a digital interface defines both the *physical interface* and the *communication protocol* used to communicate the data between the devices. Thirdly, the *application programming interface* of the digital interface, e.g., how the digital interface can be used and accessed from within the computer system, is often relevant information for the computer system designer implementing the support for an digital interface. The structure of a digital interface is presented in more detail in Section 2.2.

The physical interface has traditionally been regarded as the physical connector. This applies for digital cable interfaces or so-called wired interfaces presented in Section 2.3. However, we expand the definition of digital interfaces to include also wireless interfaces, presented in Section 2.4. For wireless digital interfaces the physical interface is the wireless radio communication. As shown in Section 2.5, the basic block architecture alternatives of a digital interface implementation are the same for wired and wireless digital interfaces, and object-oriented data modeling approach can be used in the high-level design of the data structures regardless of the underlying transport technology.

2.1 Digital vs. analog interfaces

The simple but accurate description of the difference between digital and analog interfaces is that for digital interfaces the data exchanged among the interconnected apparatus is digital (as distinct from analog). Analog signal is continuous in both time and amplitude, while digital signal exists only at discrete points in time and at each point can have one of 2^B value (discrete time discrete-value signal) [Ife93]. Most signals in nature are in analog form, while computer systems process data in digital form.

In modern computer systems signals and data are transmitted in digital form. Digital signal transmission is more robust, it makes the receiver design simpler, and the usage of the communication channel bandwidth is more flexible [Pet92]. The signal quality of an analog signal degrades in transmission, and it is usually most beneficial to perform analog-to-digital conver-

sion nearest to the signal source. Historically, a high quality analog-to-digital (A/D) conversion required a computer system with an A/D conversion unit. This required relaying the analog signal from its origin to the analog interface of the computer systems A/D conversion unit. Advances in computer and microcontroller technology have made it possible to perform most everyday A/D conversions near to the signal source. Thus, there is no-longer necessity for transmission of analog signals and analog interfaces in computer systems are becoming obsolete. Only the traditional microphone, speaker, and video interfaces remain, as the analog audio interface is still commonly used in audio and video (A/V) systems and the number of legacy monitors with only analog video inputs still in use necessitates analog video support. It is likely, that the future PC systems will exclude these analog interfaces, and require an adapter to perform the conversion to/from a digital interface. Completely digital audio and video solutions already exist, as do audio and video adapters for conversions between analog and digital interfaces.

2.1.1 Need for interface standards

The need for standardization of interfaces has become more important with digital interfaces. For most analog interfaces, the signal is quite simply a continuous analog signal, for which the variables are the signal amplitude limits and possibly different coding strategies. For early digital interfaces, like the RS-232 presented in Section 2.3.2, one needed to know the bitrate, byte order, and some other parameters, after which a byte stream could be received. If and when each byte represents a value or a letter, like in the case of terminal communication, this was enough. However, modern digital interfaces, like the USB presented in Section 2.3.3, use complex headers structures in which the data is encapsulated. To receive the data one must know the language and the rules of the communication. Hence the communication has to be standardized for devices from different manufacturers to be able to communicate.

A modern interface standard defines the physical interface (connector or radio subsystem), communication protocol, and often also the application programming interface and related data structures. These are addressed in the next Section 2.2.

2.2 Structure of a digital interface

In 1977 the International Organization for Standardization (ISO) started to develop its Open Systems Interconnection architecture [Zim80]. It divides network architecture into seven layers as shown in the left hand side of Figure 2.1. The OSI model is a generic model of a networking system, in which each layer interacts directly with only the layer beneath it, and provides services for the layer above it. It should be noted that many practical implementations do not follow the OSI model strictly. The functionality of layers is often combined to optimize the performance. Although developed for networking systems, the OSI communication model represents well also

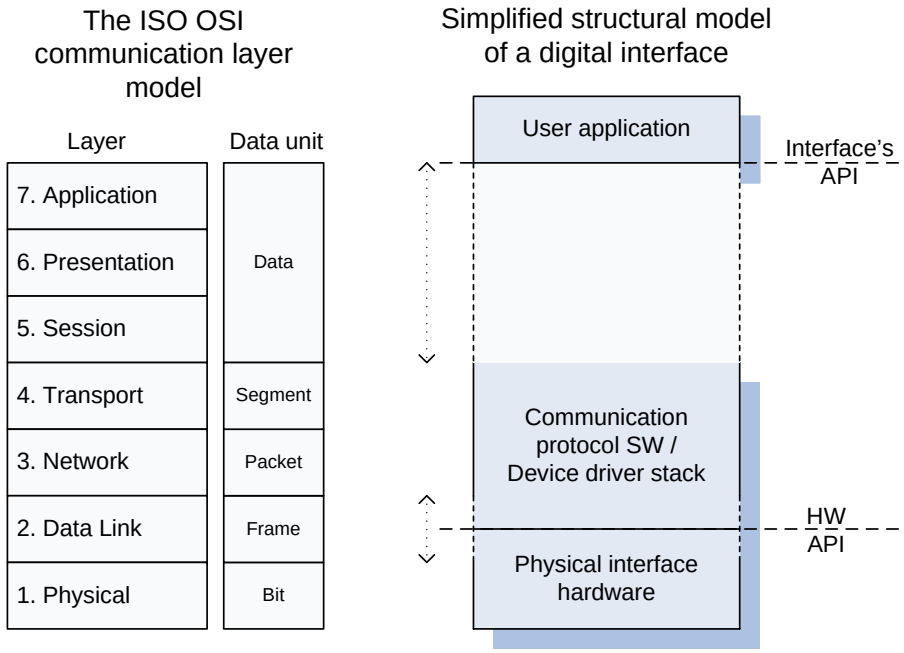


Figure 2.1: The seven layers of ISO OSI reference model and a simplified model of the digital interface structure. The communication protocol is often implemented as a part of the device driver SW stack. It may not always implement all of the OSI models higher layer functionality. The interface electronics often include HW support for some of the lower layers.

the properties of complex modern digital interfaces.

In this Thesis a division of the digital interface into three distinct parts is used as depicted in the right hand side of Figure 2.1 next to the OSI model. This simplified model or view of the digital interface does not replace the OSI communication model, but instead offers a more developer oriented structural model of the digital interface. The model tries to identify the key structural elements of the digital interface which require different specialties and are usually developed by different designers. The physical interface is of interest to the mechanical and electrical designer of a system, and the application programming interface is of interest to the user application software designer. The complexity of the interface is hidden in the communication protocol stack, and should be of interest only to a person developing the interface, not an application (software) or a device (electronics) engineer using it. As will be shown later in this Thesis, a practical interface implementation can be more complex than the structure represented in this model. The electronics implementing the physical interface often include IC's which implement some communication protocol functionality on hardware. Further, the communication protocol may not be a single software component as depicted. Instead, it can include several layered or parallel software components which may even be executed on different

processors. Internal API's can be used between these communication protocol components. In PC systems, it is not uncommon to use one interface technology for attaching another, take a USB attached Bluetooth adapter for example. However, this simplified model serves as a good starting point in understanding digital interface implementations. In the next sections, we describe the roles of these three distinct parts in detail.

2.2.1 Physical interface

The physical interface defines the physical connection as defined by the OSI models physical layer, e.g., how to connect, transmit and receive, using the physical medium. For a cable based interface, this covers the mechanical and electrical (or optical) properties of the interface, e.g., the cable connector and its electrical pins or optical fiber links. Other defined variables are the signal levels and their interpretation, or as stated in [Zim80] "protocols for establishing, controlling, and releasing switched data circuits". For wireless systems, which have no mechanical or electrical connection the physical interface is the radio link, and the devices needed to implement the radio link, e.g., an antenna and the radio.

In case of digital interface standards, the physical interface may or may not be unique to the standard. For cable based interfaces the trend is to have unique connector/connectors for each standard. This reduces the possibilities of erroneous connections. For wireless interfaces, it is more common to use same radio circuits for both standardized and proprietary implementations. A practical implementation of a physical interface of a modern digital interface, such as a USB or Zigbee, includes an IC which controls the transmission and reception of the signals to and from the physical connector [P1][P7]. This IC will usually include at least some data link layer (Figure 2.1) functionality. The rest of the communication protocol may reside in the same device or in a separate device, as discussed in Section 2.5.2.

2.2.2 Communication protocol and device driver SW stack

A *device driver* is a software component allowing higher-level computer programs to interact with a hardware device. It can consist of several layered parts, and is then referred to as the driver stack. An interface's device driver stack usually implements the bulk of the communication protocol, omitting the lower level protocol functionality which is often implemented in hardware.

The communication protocol defines the rules determining the format and transmission of data [Pet92, Heu97]. The communication protocol can be simple, like in the case of RS-232, or complex multi-layer protocol stack close to the OSI model, like in the case of Bluetooth [Haa00, Blu07]. At the simplest level, the communication protocol defines how the signal in the interface is interpreted into bits, and how these bits form bytes. It may also include some error detection. In practice, this kind of low-level protocol functionality is often implemented by the physical interface and related electronics. If the physical interface medium is shared,

as is the case with bus type and wireless interfaces, the communication protocol defines how the medium usage is allocated. The more complex protocols handle the data as packets, and may offer different delivery methods for these packets with variable error correction schemes, throughput and latency limits. A fully implemented communication protocol following the OSI model also manages the connections between the devices, ensures compatibility between systems by providing independence from differences in data representation, including possible data encryption services, and manages the communication resources and identities of the communicating partners. However, not all interface standards implement all the features of the OSI model or follow its layered structure strictly.

2.2.3 Application programming interfaces

In the simplified model of Figure 2.1, application programming interface (API) defines how the user sees the interface, or in the case of system development, the view of the software programmer using the interface. This includes a set of routines, data structures, and protocols for communicating with the interface.

Figure 2.1 depicts the ideal case, in which the interface's end user API is provided by the fully implemented driver stack. In practice, also other API's can be identified, and for a developing interface, the end user API definition is not as clear as the above. The interfaces are implemented with ICs, which implement lower level protocol functions, but often leave the higher layer protocol parts to be implemented in software, e.g., in the device driver stack (Figure 2.1). The interface IC has a HW API, which is the interface used to connect it to the rest of the system (Figure 2.2). The device driver then accesses this HW API to interact with the interface. The device driver stack offers the final API to the end user application. We can therefore define a driver level API and an end-user application API. The driver level API is often manufacturer depended and not defined in the interface standard. In the special case that no device driver is used only one API exists. Moreover, it is not uncommon for the device driver to be of layered structure with APIs between the layers.

Based on experience gained from the work of this Thesis, the interfaces tend to develop from bottom upwards. For a new interface standard, the designer usually gets hardware support for the lower communication layers and a register read-write API interface to access data and change communication settings on an interface IC. The top layers are often implemented in software so that changes in the still living specification can be easily corrected. Some parts of the top layers may be missing from the early implementations, and device profiles and classes may not yet have evolved, so the end user API may have to be developed by the designer himself and it can be a mix of lower level function calls and proprietary data structures. When using a more mature interface, the developer usually gets a more integrated solution, and a fully developed standard which includes device profiles/classes and a higher lever end-user API.

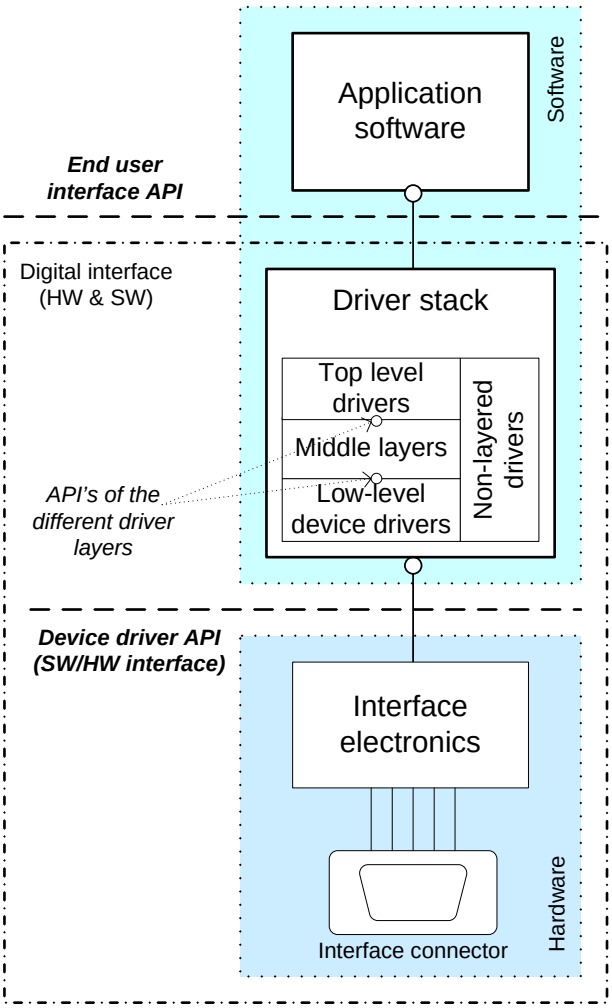


Figure 2.2: The HW/SW structure of a practical interface and the different application programming interfaces associated to it. The end user interface API is the interface that the application programmer sees. The HW/SW interface is the API used by the low-level programmer when implementing the device driver. The device driver can also have a layered or otherwise separated structure. These layers or driver components can have API's between them.

The higher integration is more visible in the embedded microcontroller systems, which often have integrated interface peripheral blocks for the more mature technologies, and no additional interface HW IC is necessarily needed.

To conclude, the interface API is the API available for the application software developer. This API provides access to the devices attached to the interface and to the interface settings for the user application. The device driver does not necessarily implement all of the ISO OSI models functionality, e.g., some functionality may be implemented in the user application.

2.2.4 Single device interfaces

A single device interface is a direct connection between two devices on a media to which no other devices have access (Figure 2.3). The communicating devices can operate either in equal or in unequal manner. If the devices are equal, then there must be either separate data paths for transmitting and receiving, or some other signals used to reserve and indicate the state of the data lines. A more complex way is to detect the state of the transmission medium before transmitting and also detect if a collision happens, although these kinds of techniques usually lead to a shared bus based architecture, presented in the next subsection.

If the devices operate in an unequal manner, then one takes the role of an master device and the another works as a slave device. The master controls the communication. If only one data path is used, the slave can only transmit after the master asks it to do so. The main benefit is that only one data path is required, as same data path can be used for transmitting and receiving. Downsides are that the slave device can not communicate freely and needs to be polled (shared data path), which requires constant activity from the master and adds latency for the slave to master communication.

2.2.5 Multiple device interfaces

Multiple device interfaces, shared buses or networks, have a shared transfer medium to which multiple devices can connect (Figure 2.3). Wireless devices are by nature shared, as any device within the transmission range is able to transmit and receive to the medium. For wired devices each device requires a physical connection to the bus. In the simplest form this is done by having multiple connectors to a cable or backplane based bus. For high speed buses this multi-connector topology is problematic as control of the bus topology is easily lost and given to the end user, and as open connectors should be terminated. It is easier and requires cheaper cabling to implement fast data links using point-to-point cabling. Modern digital cable interfaces have developed into the direction in which each cable has only two end-connectors, which plug into two different devices. By using different end connectors, the designer has greater control over the bus topology and is able to prevent erroneous connections. Bus type of architecture can still be used by chaining the devices, i.e., each device have an in and out port, implemented either by

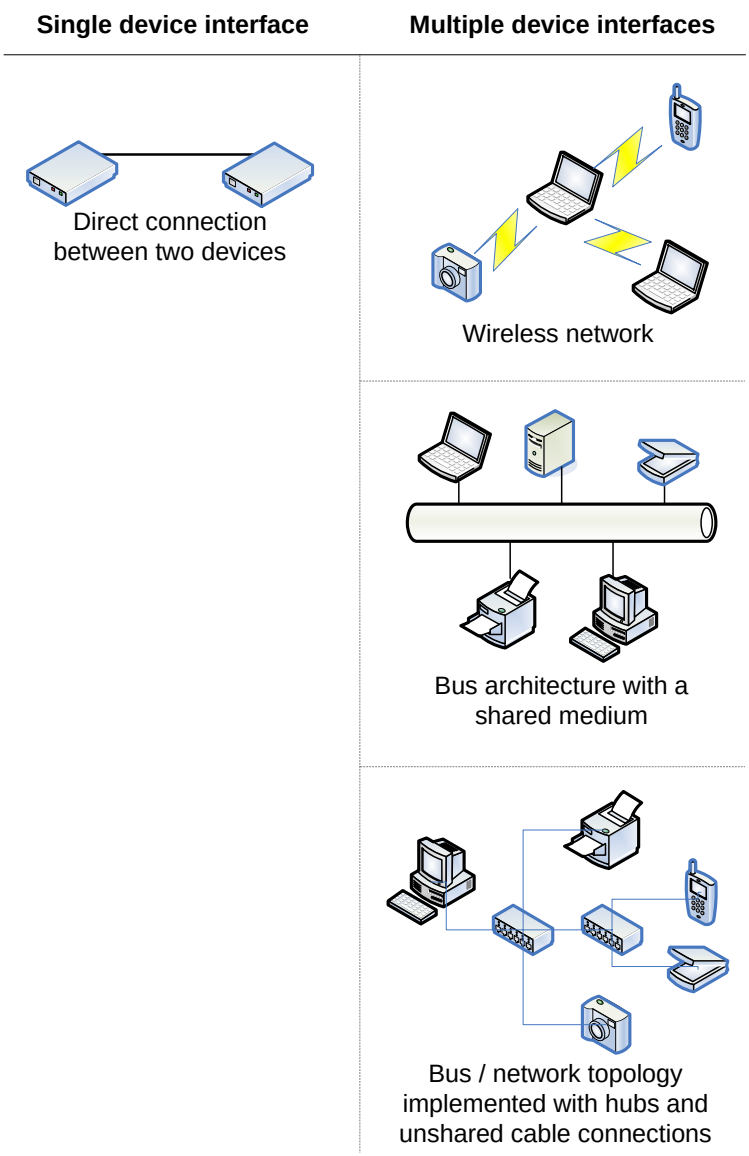


Figure 2.3: Different interface topologies.

direct connection or by using one or two-way repeater. More commonly, some tree or star type of topology is used, and hubs or switches are used to extend the bus communication. For wireless networks, same principles are applied, and router devices are used to extend the network using various network topologies. In addition, wireless mesh networks can also be formed, in which each device is capable of forwarding information to the next one. Chaining of wired devices can be regarded as a special case of a mesh network.

A direct data link between two devices on a shared medium is the simplest case of communication. Peterson and Davie [Pet07] define five basic issues which have to be addressed before a direct link can be formed between two devices. These are encoding, framing, error detection, reliable delivery and media access control. The encoding defines how the bits are described on the transmission medium. Then, the sequence of bits has to be delineated into frames to form complete messages that are delivered to the receiver. The process is called framing. Transferred messages are sometimes corrupted during the transmission, and this has to be detected by means of error detection. However, the link should appear reliable to the user in spite of errors. Finally, if the link (the access medium) is shared, it is necessary to mediate access to the media.

Whenever a shared bus or network is used, media access control has to be resolved. As explained in the previous subsection, two common approaches exist. The master-slave(s) approach, in which one devices controls the bus/network access, and equal devices approach, in which devices must sense the state of the shared medium before use and/or detect collisions when using the medium. Other approaches include using a common synchronized clock and allocated/negotiated time slots for transmission, but these are more common to larger networking systems than digital interface technologies.

2.2.6 Digital interface data rates

Digital interfaces operate at a certain fixed clock frequency which defines the bit rate. This bit rate is often used misleadingly to market the interface and its speed. The maximum data rate or throughput for actual payload data that can be obtained from a digital interface is always less than the bit rate or the bandwidth of the communication media (Figure 2.4). Firstly, the payload data on a digital bus or in a wireless transmission is always framed [Pet92]. The frame headers consume a percentage of the total bandwidth. In addition, the communication protocol often uses control packets to confirm received packets, synchronize media communication, relay information regarding media configuration/topology changes and routing information on wireless systems [Pet07]. Even when the media is only used by two devices and no control information is sent, the maximum theoretical throughput calculated from packet payload sizes might not be obtained. This is the case with the newer interfaces having over 100 Mbps clock rates and complex protocols. The driver stack and data processing for it can consume so much CPU time that packets can not be sent or received as fast as possible. This is easy to understand

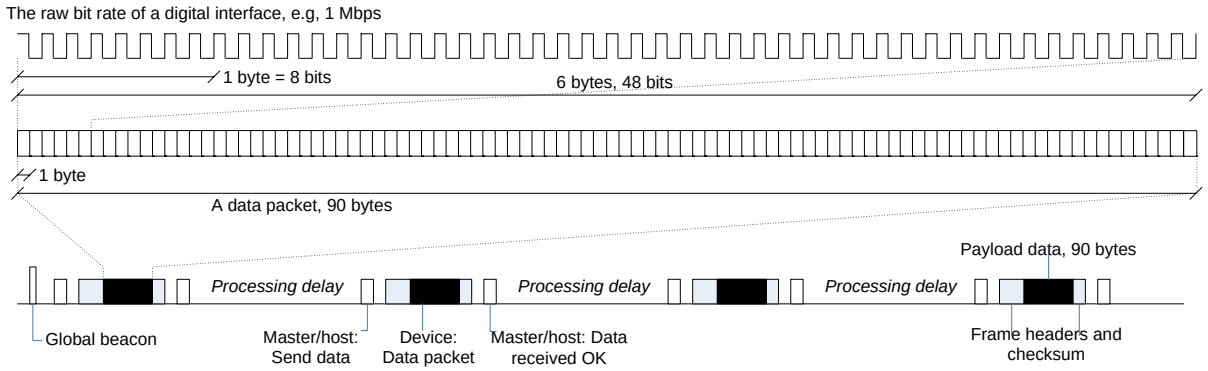


Figure 2.4: Digital interface bit rate vs. real data rate. The raw bit rate of an interface is based on the operation frequency of the transceiver. The actual obtainable data rate is limited by several factors including protocol overhead. The amount of payload data per time interval defines the true data rate or throughput of the interface. This can also depend of the transmission type, other connected devices, and environment variables.

in low-power embedded applications running at few MHz clock rates, but this also applies to desktop PC's. For example, the USB host driver stack is implemented in a way that each transfer requires generating a special data structure and when the transfer is done, the results have to be checked. The USB protocol and the host controller generate significant overhead on each data packet transfer, and the speed of the RAM and the processor bus may become a bottleneck of the system [Hu08]. Especially, if the computer is performing other tasks simultaneously, as it is usually the case.

Data transfer latency is also an important design issue is real-time systems. Practical implementations operate in a fashion in which the data to be sent is buffered, and the buffer is then given forward to the transmitter to send. To obtain maximum data throughput the size of the buffer should be as large as possible, so that the percentage of bit rate consumed by the packet header is made smaller. However, a larger buffer means waiting longer for the data, and thus increases the total latency of the system [Pet07]. If the interface medium is unreliable, i.e., packet errors occur frequently, as can be the case with wireless interfaces, then the maximum data packet size is also a compromise between packet retransmission times, maximum throughput, and latency.

2.3 Digital cable interfaces

In this Thesis, the interfaces are divided into two subclasses, the *cable based* and *wireless*. This Section addresses the *cable interfaces*, also called as the *wired interfaces* or *fixed interfaces*. Their unique characteristic is the physical connector used to connect two or more devices together.

The cable based interfaces usually use electrical signaling, although optical fiber communication is also used. Major differences to wireless interfaces are:

- Power transfer. The interface can supply power to the device.
- Increased security. Eavesdropping requires physical contact with the interface cable while wireless transmission can be picked up from the air.
- Connection topology easier to see visually. The cable shows the connection between two devices. This applies as long as there are not too many cables.
- Cable length defines maximum distance between the devices. Movement of the devices does not effect the communication as long as the cables remain connected.
- Cables restrict device movement.

Digital data is generally processed as bytes or multiples of bytes. Historically, digital cable interfaces could be divided into two sub groups: parallel and serial interfaces. Parallel interfaces had multiple parallel signal pins and were able to transfer complete bytes (or N * bytes) at once. Serial interfaces could only transfer one bit per time. Parallel interfaces thus offered higher data speeds. With the development of high-speed digital electronics, the clock rates of the interfaces increased. Using higher frequencies requires better cable shielding to reduce outside interference and crosstalk from the other signals carried withing the cable. In practice, it became cost effective to use differential signaling to carry only a single bit at higher clock rate, than several bits at somewhat lower rate. Thus all modern digital interfaces are serial by nature, some having full-duplex communication, and others half-duplex with two differential signal pairs for separate transmit and receive operation. Parallel interfaces are still used within computer systems, on motherboards, where transfer distances are very short.

Another major development has been the move towards bus based cable interfaces. Instead of a point-to-point connection and direct communication between two interconnected devices the interfaces use shared bus architecture implemented with hubs/repeaters or by device chaining, which allows several devices to connect to a shared bus. From the communication protocol viewpoint, using a shared serial bus based medium is very close to using a wireless medium, such as a radio band. The only major difference is that the cable medium is more reliable and need for data retransmission is minute. From the perspective of a systems designer using a modern digital interface, the major difference these days is to select between a connector and an antenna, as shown in Section 2.5.

2.3.1 Common properties of cable interfaces

For cable based interface standards, the physical connector either determines or gives a strong indication of the interface type. As stated previously, modern cable interface standards define

unique connectors. Of course, this does not prevent the use of these connectors for proprietary non-standard implementations, but this is typically a small issue related to prototypes and non-commercial devices. For cable interfaces, establishing the connection is very clearly indicated as plugging in the connector makes the connection.

Modern cable interfaces are of serial type, having either one two-way (full-duplex) data bus or two one-way (half-duplex) physical data links. The full-duplex links are limited by the signal propagation delay of the copper cable, which produces a trade-off between the maximum cable length and optimal bandwidth usage. Extra wait periods need to be inserted when signal direction is changed, and the length of the wait increases with the maximum cable length. Half-duplex data links can thus utilize the bandwidth more optimally. The advantages of an half-duplex data link are greatly reduced if the data link is extended to a bus to which multiple devices can connect. The signal propagation speed also sets limits for the maximal signaling rate in a copper cable. The cheaper twisted pair copper cable based link is capable to seeds up to 1 Gb/s [Im02]. Using more expensive coaxial cables even 10 Gb/s speeds can be obtained [Sch04]. It is often not the cable technology, but the IC technology, that limits the maximum speed [Sch04]. To go even faster, a glass fiber optical link can be used. They can reach speeds over 1 Tb/s. Optical links are also commonly used to implement Gb/s links instead of coaxial electrical cables.

Electrical signal can be relayed in two ways, by single-ended of differential signaling. In the single ended method, the signal level is determined by comparing it against a common ground level. The ground reference is common to all the signals, and thus the total number of signals needed is the number of signals $N + 1$. In differential signaling, the signal level is the difference between two signals, and no common ground exists. The number of signals needed is $N \times 2$. Single-ended signaling works with lower speeds and shorter interconnects, but is giving way to differential signaling. Single-ended signaling was beneficial because of its lower signal pin usage with parallel interfaces and interfaces having several signal pins, but cable interfaces are developing towards physically simpler serial interfaces. Devices using the single-ended signaling are also prone to ground loop issues, leakage currents, and cross-talk. Twisted differential cables produce less electromagnetic interference (EMI) and are less susceptible to interference, and are cheaper than shielded cables for single-ended communication.

2.3.2 RS-232 (Serial port)

The *RS-232* is often referred as the serial port in PCs, or as the serial interface or the UART (Universal Asynchronous Receiver Transmitter) in electronic devices. The name RS-232 is an abbreviation of Recommended Standard 232 [RS-85]. The most well known version is C released in 1969 [RS-85], e.g., RS-232C. The current revision is the "Telecommunication Industry Association TIA-232-F Interface Between Data Terminal Equipment and Data Circuit-Terminating

Equipment Employing Serial Binary Data Interchange" issued in 1997. RS-232 is somewhat of a nonstandard standard, as it leaves freedom for interpretation and implementation, which has lead to many slightly different protocols and cables using the same connector but being incompatible with each other. The full RS-232 standard utilizes 22 pins of the DB-25 connector, but as most devices use only a few of the specified signals, a 9-pin connector (DE-9) is often used and is also the most common in devices using RS-232.

The RS-232 should have become obsolete in PCs after USB, and it was deprecated from standard PC specification by Microsoft already in 1997 [Mic97]. It has in fact been replaced by USB in consumer electronics, mainly because the volumes of data transferred have become so huge. RS-232 is still very used in embedded systems and microcontroller devices due to its simplicity and wide availability. Almost every larger microcontroller has an UART included, so adding a serial port to a microcontroller based device amounts to just adding a RS-232 connector to the device and software routines to service the UART. The RS-232 interface is also often used for debugging purposes during device development, even though it might not be used in the final production device. As new PCs don't have serial ports, virtual ones can be added using USB adapters. These adapters do not always operate as well as real serial ports, especially when higher data rates with short packet intervals are required.

The RS-232 is a standard for serial binary data transfers, i.e., it relays bytes of data in serial form. The standard does not define character encoding, framing, protocols for error detection or data compression, which leads to the above-mentioned incompatibility issues. Even at the lowest level the communication link will fail if the user does not manually select and set (or predefine) the same communication parameters (bit rate, number of data and stop bits, and parity) to devices he want's to connect to. The user API of the interface is simple, and in general the application is responsible for most of the communication protocol functionality desired.

2.3.3 Universal serial bus

Development of the *Universal serial bus* (USB) was started in 1994 and the first complete revision 1.0 was released in January 1996 [USB96]. The year 1998 was a breakthrough year with devices really starting to come to market, USB 1.1 specification released, Apple shipping iMac with USB ports only, and Windows 98 being the first Windows version having fully functioning USB support. USB was designed to be the new common peripheral interface for personal computers, replacing all the existing and difficult to set-up implementations like serial and parallel port and PS/2. One aim was to reduce the number of connectors on the backplane of a PC. USB had a slow start, partly due to the complexity of the bus vs. previous implementations, and partly due to poor and unreliable operating system support in the beginning. Later versions of Windows 95 received an update giving some USB support, but in all it was quite useless, and Windows NT, the Windows version most common in enterprise use at that time did not get USB support.

Windows 98 was the first Windows version to really support USB, but the USB support and the whole OS was quite unstable at that time, and not really an option for heavy-duty use. The problems with USB and Windows 98 were usually caused by faulty device driver code, which because of an open OS structure, could easily crash the system. Linux support for USB emerged slowly, and most device manufacturers were quite reluctant to offer any support for people trying to debug the devices and develop USB drivers for them. The manufacturers themselves had no or little interest to support Linux at that time.

With the development of media applications and the Internet, the total data transfer capacity of USB was noticed to be too slow already at the time the first devices came to market. Web-cameras and growing data transfer needs filled quickly the bus operating at 12 Mbps, the speed which was thought to be more than enough for keyboards, mice and modems. So an updated 2.0 version of the standard was developed and released in 2000, which increased the maximum data rates to up to 480 Mbps (the bus clock speed), while retaining backward compatibility using complex micro-frame structures [Com00]. At the same time Windows 2000 was released, which offered stable and reliable USB support and really made USB a mainstream bus technology. USB memory sticks quickly became popular and made office users aware of the new interface. Recently, as external hard drives are gaining popularity and LANs are becoming faster and faster, even the USB 2.0 speeds are becoming a bottleneck. Basically, the problems or the growing needs are caused by high quality video and related applications. As the limits of single full-duplex differential copper cable are starting to limit the speed, a new 3.0 version of USB has been developed and released (2008) using asynchronous half-duplex communication with two differential cable pairs similar to IEEE 1394 [Hew08]. Optical communication has been proposed, but not yet included in the current specification. A wireless version of USB called Wireless USB [Lea07] is under development and adopted as a part of the USB family. It uses UWB technology and should not be mixed with WirelessUSB technology by Cypress Semiconductor, which is a trademarked technology using the 2.4 GHz radio band. Table 2.1 summarizes USB versions and their features.

The physical USB 1.0-2.0 interface is simple; power, ground and a differential data bus. A small niche point is that the power and ground pins of the connector are designed to connect slightly before the data bus pins, so that the device receives power to set up the data bus terminations just before it is connected. The USB 3.0 introduces four new signals, consisting of two differential pairs for transmit and receive. The existing full-duplex pair is left for backward compatibility. There are several official USB connector types: USB A-type, B-type, micro-A, micro-B, micro-AB, and mini-B. The USB A-type is the most common. B-type connector is designed to be used in upstream ports and A-type at downstream ones. Mini and micro models have been developed for portable devices (cell phones, GPS units, PDA's and digital cameras) which do not have space for the standard size connector. There are also some unofficial and widely used mini connectors that can be found on some digital cameras brands (at least Kodak

	USB 1.0/1.1	**USB 2.0**	**USB 3.0**	**Wireless USB**
Bus speed (Mbps)	1.5 / 12	480	4800	480 (3 meters), 110 (10 m) [Lea07]
Mode	Low-speed / Full-speed	Hi-speed	Super-speed	Wireless
Cable	4 pin	4 pin	8 pin	Wireless (UWB)
Maximum cable length	3 m	5 m	Not specified (approx. 3 m)	Radio range 10 m (approx.)
Other	Full-duplex operation	Full-duplex operation	Increased maximum bus power, half-duplex operation	Does not support hubs (repeaters)

Table 2.1: Different USB versions and their features. The real maximum obtainable speed is reduced by the protocol overhead, and also depends on the host computer speed and driver stack performance, the speed at which the end device can service the interface, transfer type, and other peripherals attached to the bus.

and Fuji).

USB (1.0-2.0) is a shared bus based medium, with exception of the OTG (on-the-go) devices included in the addendum to the 2.0 specification [USB06]. The topology is tiered star, where one device, the PC, is the host (master). The bus is extended via hubs. A hub is a USB device itself. So the physical connection is always point-to-point between two USB devices and their respective ports. The USB complexity lies in the communication protocol. Communication is controlled by the host controller, and the data is structured using complex frame structure. A device on the bus is only able to transmit when the host asks or gives a time slot for it to do so.

What is notable is that a normal USB bus itself does not support device to device communication. A specific application running in the host is needed to implement this. To solve this issue, an USB OTG group was formed, and USB OTG specification was included in the USB 2.0 release. An USB OTG device can communicate with any other USB OTG device using a single cable between the devices.

The USB application interface at the host end is provided by the operating system and USB driver stack. The device driver interface (HW/SW interface) is complex and implemented by the OS. At device end the interface parameters are usually set using register values, and the USB communication is handled by a specific hardware block or device. The application typically uses the bus via data buffers.

USB has a device class definition for Personal Healthcare Devices (PHDC), released in 2007 [USB07]. It was designed to enable seamless interoperability between healthcare devices and USB hosts, and to diminish the number of proprietary implementations of USB healthcare devices. The USB PHDC was implemented to support the ISO/IEEE 11073 standard (Section 3.2.3), and is a transport method used by the ISO/IEEE 11073 [Cla07, Sch07].

2.3.4 IEEE 1394 (Firewire)

The original *IEEE 1394* standard [Ins96] was approved in 1995, roughly at the same time as USB 1.0. The interface is often referred as Firewire, which is a trademark name owned by Apple. Lesser known marketing names, i.Link and Lynx, are used by Sony and Texas Instruments, respectively. The latest version of the standard was released in 2008 [Ins08a].

The IEEE 1394 was targeted for media applications requiring large bandwidth, and has gained most ground in those applications. Most common applications include digital video cameras and external hard drives. IEEE 1394 has not been able to break into PC world as a standard peripheral port like the USB, although it can be easily added to any PC system with an add-on card and it is prevalent on Apple's computers. IEEE 1394 lost more ground after USB 2.0 was announced, and although the newest version [Ins08a] is again noticeably faster than USB 2.0, it hasn't been able to conquer ground from the USB. IEEE 1394 is more costly to implement than USB, but it is faster and generally more stable, both partly due to the fact

that it does not require a centralized controller. IEEE 1394 devices communicate directly with each other without a computer in-between them.

The physical IEEE 1394 interface is interesting compared to the original USB one (USB versions 1.0-2.0). The fundamental design is different. Both data directions have their own twisted pair cables carrying balanced differential data signals. There are three types of connectors: 4-pin, 6-pin and 9-pin. The 4-pin connector has only the data signals and no power, so it can only be used in devices having their own power supply. The 6-pin version adds power and ground pins to the 4-pin connector, so that it can supply or receive power from the bus. The connector is also designed to be more rigid to prevent accidental disconnection and loss-of-power due caused by it. The new 9-pin connector came with newer IEEE-1394b (Firewire 800) standard.

2.3.5 Ethernet

The name *Ethernet* refers to the family of local-area-network (LAN) products covered by the IEEE 802.3 standard that defines what is commonly known as the carrier sense multiple access/collision detection (CSMA/CD) protocol [Cis08, Chapter 7]. The original Ethernet was developed in 1970s by Xerox Corporation. The first official 10 Mbps version 1.0 of the standard (ANSI/IEEE Std. 802.3-1985) was published in 1985.

Ethernet is not a peripheral interface as such, but LAN-networks have become so common, that most PCs nowadays have it integrated on the motherboard or available via low-cost expansion cards. In some sense it can be considered as a interface bus, although it really is a gateway to a local or even global network. Ethernet's fast speed, reliability and popularity make it tempting for applications requiring stable high speed data transfers. Internet protocol (IP) stack is mainly used to communicate via Ethernet interface, leading to a complex protocol suite, as IP stack needs to be implemented at both ends, and services run on top of it. As IP technology is so common nowadays, software and hardware solutions are widely available. The protocols required are well tested, optimized and often familiar to the software developers, which lead to the fact that the total cost of using Ethernet as an interface is not a problem anymore, even though it leads to seemingly more complex designs than with traditional peripheral interfaces.

2.3.6 Others

As previously mentioned, the USB is slowly making the RS-232 interface obsolete in PCs, and if not for its popularity in embedded systems and number of legacy devices still used, the RS-232 would probably be extinct already. This is the case with PS/2 and Parallel port.

The *PS/2* is an interface designed for mice and keyboards and it was introduced in IBM Personal Systems/2 (PS/2) series microcomputers in 1987. The PS/2 interface is not used for any other applications and is becoming obsolete as new mice and keyboards use USB.

The *Parallel port* is often know as the printer port or Centronics port. It was designed in 1970 for printer use at Wang Laboratories, and became a de facto standard. In 1994 a bidirectional version of the parallel port was standardized as the IEEE 1284. The parallel port was used for devices requiring higher transfer speeds, like printers, scanners, cameras, sound cards, joysticks, dongles, and external drives. The USB interface, and in some cases the Ethernet, have made the parallel port obsolete in consumer devices.

The *Personal Computer Memory Card International Association* (PCMCIA) is an expansion interface widely used in laptops. The PCMCIA-devices are nowadays called PC Cards. The PCMCIA interface is basically a hot swappable version of the ISA extension bus. After USB 2.0 introduction, the PCMCIA/PC Card started to lose ground to it, and also somewhat to Ethernet. The PCMCIA association developed a new PCExpressCard standard introduced in 2003 which uses PCI Express or USB 2.0 (now also 3.0) to interface to the CPU.

Internal expansion slots like the PCI and AGP are used in desktop computers for very fast and low-latency devices. They require opening of the computer case, and do not support hot swapping. In peripheral use the expansion slots have lost ground to USB. The need for the expansion slots is becoming smaller as audio/video/network interface controllers are more and more frequently integrated into the motherboards.

Internal bus interfaces (SATA, IDE) are used for high speed disk drives and CD/DVD drives. They have been highly optimized for this purpose only, and not designed to be used outside the PC casing. Similarly the external *digital video and audio interfaces* are highly optimized for transferring digital audio/video (A/V) signals. The *Sony/Philips Digital Interconnect Format* (S/PDIF) is the main digital audio interface which multiple connectors and in implemented either via coaxial (RCA, BNC connectors) or optical cables (TOSLINK connector). *Digital Visual Interface* (DVI) supports also analog video and has dedicated connectors. *High-Definition Multimedia Interface* (HDMI, www.hdmi.org) and Displayport (www.displayport.org) support both video and audio and have dedicated connectors. The HDMI is electronically compatible with the DVI, so no signal conversion is needed in implementing HDMI-DVI adapters. However, it is recommended not to use these kinds of dedicated interfaces outside their application area.

2.4 Wireless interfaces

This section addresses the second digital interface subclass, the wireless interfaces. Their unique characteristic is that no physical connection is needed to connect two or more devices together. Wireless communication is based on electromagnetic waves. Acoustic energy can be used to implement wireless communication, but its use is rare or non-existent in modern digital interfaces. The modern wireless interfaces usually use radio frequency (RF) communication or sometimes infrared (IR) light. IR communication requires line-of-sight and pointing of the device towards the receiver, which limits its usability as a general wireless communication media, but also gives

possibility to implement applications which benefit from these limitations.

Major differences of the wireless interfaces compared to wired ones are:

- Devices self powered, the interface can't supply power[1]. Leads to low-power system design and battery-powered or energy scavenging devices.
- Easy to eavesdrop unnoticeably if communication is not encrypted, and generally more vulnerable to attacks than cable based interfaces.
- User needs additional information to see which devices are communicating/connected.
- More unpredictable communication performance. Connection will brake if the device is moved too far. Maximum connection distance can not be defined exactly, depends on the environment. Movement of devices and people/other devices in the proximity may influence communication.
- Devices can be moved without restricting cables. Connection can be established, disconnected or changed without physical access to the devices, enabling remote control and automatic reconfiguration of connections.

2.4.1 Common properties of wireless interfaces

Most modern digital wireless interfaces use RF communication. Infrared light is used in some applications. A RF based interface requires an antenna and some analog front-end electronics before it can be attached to the digital system. An infrared based interface requires an IR transmitter/receiver (IR-led and detector) and some electronics to drive it. Often these are packaged in a single chip. The implementation architectures for wireless interfaces are thus same as for the wired interfaces, and is discussed in detail in Section 2.5.

By the nature of the communication, wireless interfaces are less reliable at the data link level than wired interfaces. Part of this is due to the fact that the communication medium is always shared. At best, it is shared with other devices of the same interface standard (like the case of wired interfaces using a shared bus), but more often it means that the communication medium is shared with other competing interface standards operating at the same radio band. Also, general household appliances and electrical equipment cause wideband RF interface which can interfere with the RF communication. Changes in the environment, moving objects and people, can also break established communication paths. Wireless devices are inherently more mobile, and can be purposely or accidentally moved beyond the range of the wireless connection. All of this has to be taken into account when designing the communication protocol. In general, this leads to wireless interfaces having more complex communication protocols than the wired ones. More complex protocols require more processing power and resources. More memory is

[1]It is possible to transfer some power wirelessly, but this is not a standard feature on wireless interfaces.

required by the protocol to buffer data in case retransmissions are needed, and to buffer data when short term breaks in the wireless connection occur. This is especially the case with most medical applications, for which zero data loss is often desired. To recover from short breaks in the data communication, a wireless interface needs to have sufficient extra capacity in the bandwidth to be able to send the buffered data when the link is active again and catch-up to normal operation. This is why the same application may be implemented with a slightly slower wired interface than the wireless one, e.g., the wireless interface needs to have a higher nominal data rate than the wired one. Another design aspect favoring faster transmission speed is, that it is often beneficial to minimize the radio transmission time, and to use short data bursts to transmit data. This reduces the air time and probability of transmission errors due to interference. A notable issue in wireless interface design is, that for RF communication, the receiver and the transmitter consume approximately equal amount of power. In fact, the receiver can consume more power than the transmitter. A common misconception is to think that only data transmission consumes a significant amount of power.

An RF based interface works at a certain frequency band. The radio band regulations set limits to transmission power, maximum bandwidth/data rate, and possibly regulate band usage time (duty cycle) [P5]. The frequency band determines some interface characteristics including the range and wave propagation/deflection properties (how well the signal goes through/around walls). The radio modulation and schemes for frequency spectrum use determine the physical characteristics of the radio communication. Narrow band systems operate on a narrow frequency band, for which the width of the band determines the maximum obtainable bandwidth. A frequency hopping scheme can be utilized instead of a fixed band, to obtain better interference resistance and to slightly improve security. A spread spectrum system uses wide band, noise-like signals. The information is coded with a "spreading" code, producing a pseudo random noise-like signal which has much larger band than the actual signal bandwidth. The wide-band signal has lower spectral power density, is more difficult to detect and harder to interfere than the narrow-band signal. Moreover, wide-band and narrow-band signals can occupy the same band with little or no interference. The ultra wide-band (UWB) signal is a special case of wide-band communication. The UWB signal does not fit into existing frequency bands but instead it is spread over hundreds of megahertz, even over gigahertz, band. The spectral power density is very low, in most cases unnoticeable from the normal background noise.

A wireless interface is invisible to the human eye. This is a major feature and a difference compared to wired interfaces for which a visible cable connects the devices communicating together. From the usability viewpoint, converting a device from wired to wireless requires more than upgrading the interface electronics and redesigning the communication protocol. This is especially true for medical devices used in hospitals and other environments where several devices co-exist. A clear indication of, i.e., which sensor and which monitor/display are connected is needed, and if the sensor is moved along with the subject it should be able to establish a

connection with an another monitor/display located in the new room/residence. These kinds of issues require rethinking device use and possibly redesigning the physical user interface.

2.4.2 Wireless networking

In this Thesis our emphasis is on interfaces, and wireless technologies are discussed from the interface viewpoint. However, a short discussion on different wireless networking topologies is included.

In general, a wireless communication link is of asymmetric point-to-multipoint type by nature, i.e., the signal transmitted by the transmitter can be received by multiple receivers. Exceptions to this rule are applications with very directionally oriented antennas and directionally quite sensitive infrared communication, which are more of symmetric point-to-point type. In the simplest case, when two devices are communicating together, a point-to-point connection protocol can be used. This is a typical ad-hoc way of joining two devices together, and can be implemented with simple communication protocol (a protocol implementing only some OSI link-layer functionality). To join more devices together, a network or piconet is typically used. In a *piconet*, one device, usually the one having the most computing and power resources, becomes a coordinator of the network to which other devices connect. This is similar to the bus based architecture of wired interfaces, where a master controls the bus activity. These two are the most common ways for joining wireless peripheral devices to a PC based system via a wireless interface. If the devices support routing and connections are allowed to extend over more than one radio hops, then we move more into the field of wireless networks, which is out of the scope of this Thesis.

It should be mentioned that most wireless technologies support some means of networking. The networking capabilities can be extended by adding protocol layers as defined by the OSI 7-layer model [Zim80]. Lot of research interest is on running IP-protocols on all wireless technologies to obtain uniform and interoperable networks. Extending the communication over wireless networks also brings new security issues, as a network is more vulnerable to attacks than a simple piconet or point-to-point link, because the information is routed via devices which security can be compromised.

2.4.3 Bluetooth

Bluetooth wireless technology [Haa00, Blu07] released in 1998 was originally developed for mobile phone headsets, but interest sprung during specification development, and several different usage areas were envisioned. The Bluetooth development followed USB footsteps in that a new faster specification was released quite soon after the original one, as new media applications required more bandwidth than was originally envisioned. The final Bluetooth specification became rather complex for simple low-power applications, which left room for other low-power

wireless technologies, and reduced interest in battery powered Bluetooth devices. In all, Bluetooth has become popular in media and data storage applications requiring medium to high transfer speeds, or occasional hundred kB to few MB file transfer capabilities. The networking properties and so-called scatternets have not really emerged, and Bluetooth has remained primarily as a point-to-point technology.

Recent development has been to introduce Bluetooth into battery powered devices by defining a new simpler technology which uses the same physical interface technology (antenna and radio subsystem) as the original Bluetooth. This technology, which started from Nokia's Bluetooth Low End Extension (BTlee) development work [Hon04], has now been included in to the Bluetooth standard [Rey08] as Bluetooth Low-Energy technology. The technology was also known as Wibree before inclusion to the Bluetooth specification. The specification defines simple low-power slave type devices which should operate even over a year with a small battery.

Bluetooth 3.0 specification was released in April 2009. The new release supports alternative medium access control (MAC) protocols. Currently only the 802.11 (the MAC used by WLAN) is supported. The Bluetooth radio is used to form the connection and configure it, and standard Bluetooth API/Profile structure is seen by the user. When high speed data transmission is required the system can switch to using the alternative MAC, i.e., currently only 802.11, instead of Bluetooth radio.

Bluetooth has a Health Device Profile defined by the Medical Devices Working Group of Bluetooth Special Interest Group (SIG) and released in 2008 [Blu08]. It is an application profile that defines the requirements for qualified Bluetooth Healthcare and Fitness device implementations. The profile provides strong support for the ISO/IEEE 11073 [Cla07, Sch07] standards, including the new Personal Health Profiles [Ins08b, Cla07], and is used by the ISO/IEEE 11073 (Section 3.2.3) as a transport method.

2.4.4 Wireless LAN (Wi-Fi)

Wireless LAN (WLAN) is the wireless extension of LAN, also called as *Wireless Fidelity* (Wi-Fi). Wi-Fi is a brand name licensed by the Wi-Fi Alliance for WLAN technology based on the 802.11 specifications [IEE07]. It is primarily designed to form fixed network infrastructures, but it can also be used as a high speed peripheral interface or small PAN in home use. Compared to Bluetooth [Fer05] and Zigbee, its main advantage is that it supports large network installations and routing of high speed data streams. On the other hand, it uses more power than these technologies [Zah04], especially for the newer and faster standard versions (802.11a, 802.11g). The IP based protocol is complicated, and not very lightweight for small devices, but feasible for large media transfers and storages. Its wide and common use lowers implementation cost as devices and chipsets are widely available.

2.4.5 Zigbee

Zigbee technology [Zig05, Bar07] is based on *IEEE 802.15.4 standard* [Ins03, Gut01], building on the link-layer functionality defined by the IEEE 802.15.4. The development of Zigbee started as an attempt to build self-organizing ad-hoc wireless networks. Zigbee is developed by ZigBee Alliance, an association of companies working together to develop standards for reliable, cost-effective, low-power wireless networking. Zigbee was aimed at home automation and similar appliances requiring low-power functionality and flexible networking. One goal was to replace proprietary solutions in this area and to introduce a common standard. Zigbee aimed at gaining ground in applications for which Bluetooth or WLAN were too complex. Zigbee has somewhat suffered from the fact that mobile phone manufacturers have not included it in their products and that WLAN has become so cost efficient. It is unclear what will be the role of Zigbee if Bluetooth Low-Energy devices become popular, and especially if IP technology is introduced to sensors. In March 2009 the Zigbee Alliance announced the health care public application profile, which is said to support the ISO/IEEE 11073 (Section 3.2.3) standard [Zig09a, Zig09b].

2.4.6 Others

Infrared communication was popular before ad-hoc RF networks emerged. IrDA standard [Wil00] was widely used in mobile phones and laptop computers basically as a wireless serial port as it offered a very low-power way to implement ad-hoc communication. Infrared communication is somewhat cumbersome to use, as the connection requires exact device orientation, and the connection is easily lost if a device is moved. It is the author's opinion that IrDA will be replaced in the future by RF technologies, as has already happened in many applications. Yet it is still useful in some special applications where it is advantageous to require pointing the device to certain direction/place to transmit data, or in application areas where RF interference is not tolerated.

Ultra-wideband (UWB) technology [Win98, Opp04] promised much in the early 2000's, but has then quietly fallen from the highlights to the background. There are still technological and regulatory issues to be solved for it to become a global success. Lot of time and resources on the UWB camp have been used on developing two competing standards which has slowed the total development of the technology. Official IEEE UWB physical layer standardization work was attempted to be carried out in the IEEE 802.15.3a group, but its progress was stalled by the two competing camps, and it was left to the markets to decide which technology will prevail. Wimedia Alliance, which promoted the UWB technology, has announced in March 2009 that it will move its technology development under the Bluetooth Special Interest Group (SIG), Wireless USB Promoter Group and the USB Implementers Forum. Currently it seems that UWB technology will not become a new interface standard, but rather an alternative physical transfer media to existing interface standards.

A notable feature of the UWB technology is that the obtained bandwidth depends strongly on the distance between the transmitter and the receiver. UWB is mainly targeted as a cable replacement technology for high data rates at short distances. The radio technology for UWB is simpler than for carrier based systems, and considering the data rates UWB can offer it is a very low cost solution. UWB technology is also very resistant to interference unlike the current carrier based technologies. UWB is specified to be used in the wireless version of the USB interface (Wireless USB) [Lea07], and plans to include it at some stage to Bluetooth 3.0 standard have been presented.

Current developments in wireless digital interfaces are focused on developing high speed interfaces which support digital video applications. New standards such as WHDI, WirelessHD and WiGig are all targeted for home consumer AV applications. *Wireless Home Digital Interface* (WHDI, www.whdi.org) uses the 5 GHz unlicensed band for uncompressed video and audio transmission. It supports data rates of up to 3 Gbps in a 40 MHz channel and up to 1.5 Gbps in a 20 MHz channel, of which only the latter is worldwide available. The range is defined to be beyond 30 meters. *WirelessHD* (www.wirelesshd.org) is an industry led effort to define a specification for the next generation wireless digital network interface specification for consumer electronics products with emphasis on high-definition audio, video and data streaming between source devices and high-definition displays. It is based on the unlicensed 60 GHz spectrum and promotes theoretical data rates as high as 25 Gbps supporting uncompressed transmission of full HD video and audio at 10 meter range. *Wireless Gigabit Alliance* (WiGig, www.wirelessgigabitalliance.org) enables multi-gigabit wireless communication using the unlicensed 60 GHz spectrum, offering wireless connectivity at data rates previously accomplished only with fiber optic cables. The typical range for WiGig devices is said to be 10 meters. WiGig is a slightly newer standard than WirelessHD and has some of the same founding members.

The *Digital Enhanced Cordless Telecommunications* (DECT, www.dect.ch) standard also includes support for digital data communication. DECT has its own reserved frequency band, which increases reliability due to minimal interference from other devices. DECT requires radio infrastructure (basestations), and as such is expensive for plain point-to-point applications. DECT telephones are used in some hospital facilities and for those institutions it may present a viable solution. Due to long range and high transmission power, the commercial DECT HW-modules consume noticeably more power than PAN technology based devices such as Bluetooth [P5].

2.5 Digital interface design and implementation

Use of digital interfaces in practical system designs involves both hardware and software design. Systems design is based on modeling. Models can be derived for both hardware and software, and from different viewpoints. The traditional engineering view, mostly used in this Thesis,

looks at the system as a black box with data inputs and outputs. The engineering systems model tries to describe what is needed in the black box to implement the desired functionality. In information sciences, systems are traditionally described as entities and their relationships, or with modern object oriented methodologies, as entities, their behavior, and their relationships. The information science model is more abstract; it does not describe well what is required at the lower level to implement the communication between the entities. Instead, it is better suited to describing the higher level data structures of complex systems. The engineering model is closer to implementation; it is good in describing the lower level architectures of systems with clear data paths, but it is too constrained to describe complex distributed systems and their behavior. The challenge in the digital interface design is to bridge the information between these two very different modeling viewpoints.

In this Thesis, the emphasis is on low-level implementation architectures and issues of digital interfaces. However, understanding of digital interface standards and higher level protocol functionality requires understanding of some of the data modeling principles associated to information sciences and information systems. Data modeling in interface design is presented in Section 2.5.1. Returning to a more engineering view of digital interfaces, four basic architectures for digital interface implementation in embedded systems have been found and are presented in Section 2.5.2.

2.5.1 Data modeling in digital interface design

Data modeling is a key part of design of databases and systems & software involving complex data structures. As such, it is also a key part of modern digital interface design. In fact, a device attached to an interface is often viewed as a data resource which is accessed or presented to the operating system similarly to a remote database or data storage unit. Structured data on the remote devices needs to be accessed effectively, and established data modeling techniques and practices developed for database systems can be applied in the design process of digital interfaces. Traditional data modeling only models data. Although usable, this approach has limitations when modeling devices with functionality or resources for more complex operations. Modern data modeling approaches use object-oriented modeling techniques, which add behavior modeling alongside data modeling. These object-oriented techniques can therefore be used to model advanced computerized devices with functionality in addition to data resources, which extends their usability in modern interface design.

What is data modeling?

A *data model* is the main tool for hiding details of data storage and providing data abstraction [Elm94]. Data model defines the format of the data. Most modern data models also include concepts to specify behavior. Consistent use of data models enables data compatibility. A

schema is a description of a data model, usually diagrammatic.

Three basic styles of data models can be identified [Elm94, Amb09]:

- Conceptual data models (domain models), explore domain concepts with project's stakeholders. A conceptual schema describes the semantics of a domain.
- Logical data models (LDMs), explore domain concepts and their relationships. They depict logical entity types, the data attributes describing those entities, and the relationships between the entities. A logical schema describes the semantics as represented by the particular technology
- Physical data models (PDMs), used to design the internal schema of a database, depicting the data tables, the data columns of those tables, and the relationships between the tables. A physical schema described the physical means by which data are stored.

The conceptual data models are high-level data models close to the way many users perceive the data; where as the physical data models are low-level models which describe details of the computer implementation. The logical data models are a class of representational or implementation models, which provide concepts that, can be understood by users but which can be implemented quite directly on computer systems.

There has been some confusion on the use of terms "data model" and "information model". The RFC 3444 [Pra03] proposes the following distinction. *Information model* models managed objects at a conceptual level, independent of any specific implementations or protocols used to transport data. Data models are defined at lower level of abstraction and include many details. They are intended for implementations and include protocol specific constructs. I.e., multiple data models can be derived from a single information model [Pra03]. Object-oriented techniques, such as UML [Obj09b, Obj09a], are commonly used to describe information models.

Data models define managed objects at a lower level of abstraction. They include implementation and protocol-specific details, e.g., rules that explain how to map managed objects onto lower-level protocol constructs [Pra03]. Languages such as ASN.1 [Int87] have been used to define data models.

Object-oriented data models

Object-oriented data models were proposed for database systems to meet the needs of emerging complex applications and growing data sizes. The object-oriented data modeling approach offers flexibility without being limited by the data types and query languages available in traditional database systems. A key feature of the object-oriented databases is the power they give the designer to specify both the structure of complex objects and the operations that can be applied to these objects. [Elm94]

The concept of encapsulation is one of the main characteristics of object-oriented languages and systems. It is also related to the concepts of abstract data types and of information hiding in programming languages. In traditional databases, it is customary to make the structure of database objects visible to users and external programs. In object-oriented approach, the main idea is to define the behavior of a type of object based on the operations that can be externally applied to objects of that type. The internal structure of the object is hidden, and the object is accessible through a number of predefined operations. The implementation of an operation can be specified in a general-purpose programming language that provides flexibility and power in defining the operations. The external users of the object are only made aware of the interface to the object, which defines the names and arguments of each operation. The implementation of the object is hidden from the external users. In object-oriented terminology the operation implementation in called a method. Typically, a method is invoked by sending a message to the object to execute the corresponding method. [Elm94]

As can be seen from above, the modeling approach is readily and directly applicable to communication interfaces, in which the data is exchanged using messages between the communicating entities. By using methods to update data of the objects instead of direct access to the data the data integrity constraints can be met.

Object-oriented software design

Modern digital interfaces rely heavily on software implemented protocol stacks. The interconnected devices and their data structures are often modeled using object-oriented techniques and tools. To understand modern digital interface design, one must also understand the basic concepts of object-oriented software design.

Object-oriented software design is part of an object-oriented development process, which includes:

1. Object-oriented analysis - develop an object-oriented model of the application domain

2. Object-oriented design - develop an object-oriented model of the software system to implement the requirements

3. Object-oriented programming - realize a software design using an object-oriented programming language. [Som00]

Two types of design models are normally used to describe an object oriented design [Som00]. Static models describe the static structure of the system in terms of the object classes and their relationships. Dynamic models describe the dynamic structure of the system and show the interactions between the system objects. The static models are sometimes referred as structural and dynamic as behavioral models, views or diagrams. For digital interfaces, the static models

describe how the data is structured in the devices, and the dynamic models describe the interactions between the devices over the interface media. To date, UML 2.2 [Obj09b, Obj09a] supports 14 types of diagrams [Obj09b]. Class, composite structure, component, deployment, object, package, and profile diagrams are used to describe the static structure. Dynamic operation is modeled using activity, interaction, use case, and state machine diagrams. The interaction diagrams have been further divided into sequence, communication, interaction overview, and timing diagrams, to support different application needs.

Data models in modern digital interface design

Data models are indispensable when designing and documenting complex interface standards. Modern object-oriented techniques and methodologies can be used as a tool when specifying a new standard, and thus providing a readily implementable specification. For a user of established standard interfaces, the data structures, information models and supporting device profiles provided by the standard can be used and users modeling needs are reduced. Even then, the data modeling paradigm still exits at higher level of systems development, which is out of the scope of this Thesis. An example of the object-oriented interface design is the ISO/IEEE 11073 standard introduced in Chapter 3, which is based on an object-oriented data (information) model.

Each digital interface has some low-level data schema, even if it lacks a defined data model. If the data structure and the communication needs of the device are very simple, then a model based design may not offer benefits that outweigh the additional work. In a very resource limited environment, such as a small battery-powered wireless device (including short data payload, infrequent transmissions, power consumed on each transmission), a more engineering view based design approach of looking at the system as a data path/pipe between two entities may result in more optimal performance. Indeed, this engineering approach is often used to produce efficient communication protocols for simple systems with very fixed data properties, usually using point-to-point communication. For example, a small sensory device may broadcast a message once in an hour. This gives the designer a limited number of bytes which he will then try to use as effectively as possible to convey as much information as possible. A packet data structure schema is developed without a true underlying data model. Although very optimal in performance, systems and communication protocols designed in this way are often inflexible and difficult to expand in functionality afterwards.

Benefits of data model based interface data structure design include avoidance of redundant information, flexible design, and that changes to structure are later possible. The drawback of model based design for an interface designer is that some control over resource usage optimization is lost and some extra headers are introduced compared to fixed (inflexible) implementations. Furthermore, object-oriented models are not as straightforward to implement in a resource limited embedded systems programmed using non-object-oriented languages such as C, than

in a computer system with object-oriented programming language. Embedded systems have limited RAM, and it is easy to overuse it with memory structures required for objects.

2.5.2 Digital interface implementation architectures

In the research performed for this Thesis, it has been observed that practical implementations of modern digital interfaces for embedded systems have certain basic implementation architectures. The same basic architectures can be applied for both wired and wireless interfaces.

A modern digital interface requires three parts: A physical media interface, interface logic or a driver circuit, and a protocol processing unit. The physical media interface for wired interfaces is the physical cable connector which is replaced by an antenna for wireless interfaces. Apart from the very simplest interfaces, some form of interface driver electronics are needed to convert the physical signaling into logic level signals. Low-level communication protocol functionality is usually implemented in hardware due to strict timing requirements. Lastly, at least for the more complex protocols and protocols which are still in development, the higher level communication protocol functionality is often implemented in software.

All systems require the connector/antenna, and various amount of electronics to join the connector/antenna to the rest of the system. For the rest of the system four basic implementation architectures presented in Figure 2.5 exist.

In the first approach, the interface is implemented completely by the system designer either in software using a processor core or by logic using an field-programmable gate array (FPGA) or an application specific integrated circuit (ASIC). The software approach is useful only for very simple interfaces in which the interface bit rate or duty cycle are low or if the main functionality of the system is to serve the interface, i.e., the application is implemented on a separate processor. Modern digital interfaces require low-latency responses and complex protocol processing which often make the software implementations impractical for one CPU systems. For ASIC and FPGA implementations it is often safest to rely on proven IP blocks instead of implementing the interface by hand. The advantage of this approach is that the designer can freely choose the technologies he wants to use, e.g., the microcontroller or FPGA type, based on other peripheral needs or existing systems specifications.

The second transceiver IC approach is very common. It is typically the first realistic alternative for a emerging modern digital interface standard for the smaller and medium size companies/institutions. In this approach, the transceiver handles the low-level (LL) protocol functionality which requires low-latency responses and accurate timing. Data is relayed in memory registers between the main processor and the transceiver. The functionality of the transceiver can be modified by changing control register values. The advantages of this approach compared to the previous software based one are lower main CPU load and possibility to utilize sleep modes (wake by interrupt), protocol complexity is hidden for the system designer and only a simple

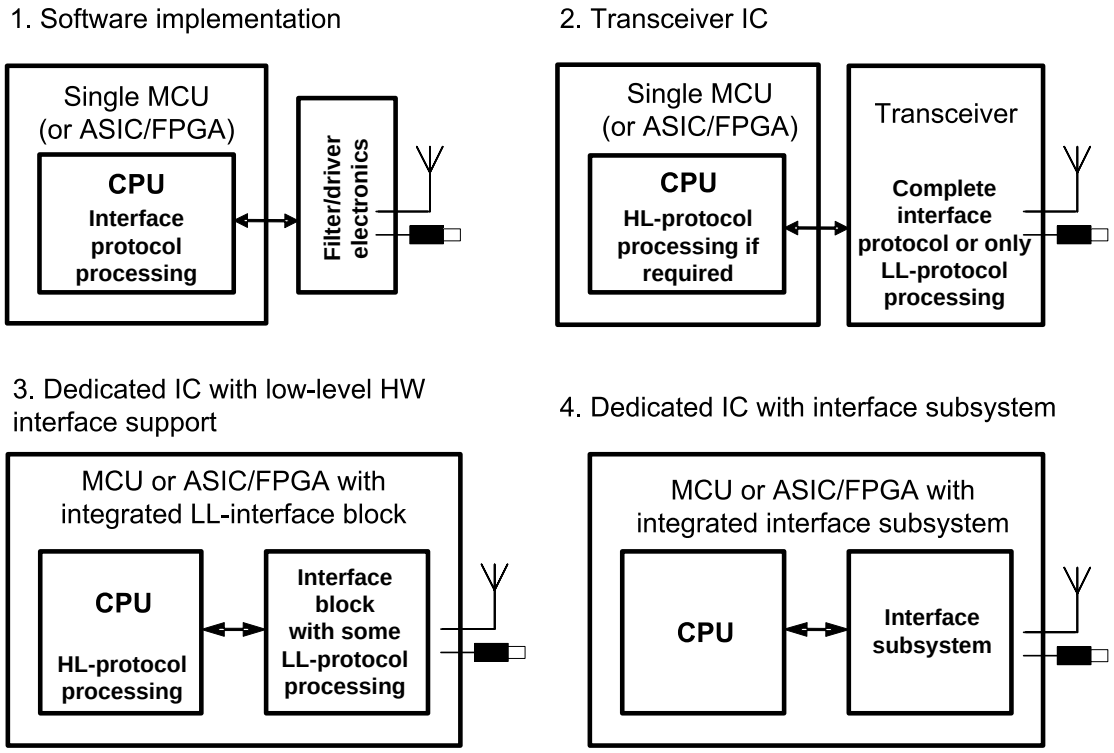


Figure 2.5: Four different interface implementation architectures for embedded systems.

register API is used to interface to the transceiver, the interface can be added to an existing embedded system without complete system redesign, and the processor core can be chosen quite freely. In some cases, generally regarding wireless technologies, the transceiver only implements the lower protocol layers. The transceiver manufacturer often provides software support for the higher layers (HL), but this support may be processor depended and/or design tool depended. These may limit the designers freedom for processor selection, and may also limit future system modifications.

The third approach uses a dedicated chip with support for the selected interface by having a dedicated HW interface support block. Typically this means a microcontroller unit with interface peripheral unit that implements some low-level functionality of the desired interface. Higher protocol layers are implemented in software using the main CPU. This approach enables small single-chip designs, and is often desirable for small mobile or battery-powered systems. These kinds of chips usually emerge as the interface standard matures and becomes popular, especially for the wireless interfaces, which are more demanding to implement in single-chip fashion. The disadvantages of this approach are that the processor type is fixed and alternative chip suppliers may not exists. Also, for the more complex protocols the higher protocol layers implemented in software reduce significantly the resources available for the application SW. The fixed processor type means that peripheral needs, such as the A/D converter (ADC) resolution, number of ADC channels, and peripheral units for other interfaces, can't be optimally filled and system clock rate is often fixed and thus can't be optimized (leading to increased power consumption). If the designer is adding interface support to an existing embedded system then using this approach he may need to change the complete technology platform, including hardware, software, and development tools.

The fourth approach is the ideal version of the third approach, in which the interface functionality is implemented completely by a separate interface subsystem with minimal load to the main CPU. This kind of implementation is possible with either simple interfaces or mature standards in high volume production. Compared to the third approach, this approach does not load main CPU, and thus gives the software designer better control over the CPU use. It is rare to find these kinds of products for emerging and developing interface standards. It is also sometimes difficult to classify a device between type three and four if the actual interface standard is simple, defining only the lower OSI-layer functionality, and requires additional non-standard protocol layers or device profiles to be implemented in SW.

In this Thesis all four approaches have been used in implementations. The first approach has been used to implement serial parallel interface (SPI) in SW to interface the measurement board to the radio board in the wireless BCG chair presented in Publication [P6] and in the networked sensors presented in Publication [P8]. The second approach has been the most used, as both the USB patient monitoring (Publications [P1], [P2]) and isolation work (Publication [P3]), and Zigbee/IEEE 802.15.4 radio work (Publications [P6], [P7], [P8]) have been done using interface

transceivers. The third approach has been used in the alternative USB-microcontroller based system implemented for the USB patient monitoring work (Publication [P1]) and also during the development of the Zigbee network the Chipcon CC2430 integrated CPU and RF IC was evaluated. The fourth approach has been used in several RS-232 implementations, for sending debug messages to serial terminal (Publications [P1], [P3], [P6], [P7]), for Zigbee radio platform to PC interface (Publications [P6], [P7], [P8]), and when interfacing measurement sensors to WSN radio boards (Publication [P8]). Also, the CC2420 RF transceiver was interfaced via HW SPI to the radio board's microcontroller (Publications [P6], [P7], [P8]).

Summary

The term digital interface was defined in this Chapter and the structure of a digital interface examined. Three main components in digital interface structure were identified, the physical interface, the application programming interface, and the communication protocol in between them, often implemented as a device driver SW stack. Digital interfaces can be single or multiple device interfaces of which the latter are more common. Object-oriented data modeling approach enables flexible development of complex data structures and message exchange mechanisms required for the higher protocol layers of digital interfaces. Four different basic architectures for digital interface implementation were identified, regardless of the nature of the interface medium. Cable based and wireless interfaces have some distinct features which were presented in detail along with the currently most used commercial digital interface standards.

The next Chapter focuses on medical devices, their definition and unique features, to understand the requirements set by them for digital interfaces.

Chapter 3

Medical devices

In general, a *medical device* can be anything from computerized medical equipment to simple wooden tongue depressors. The intended primary mode of action of a medical device on the human body is not metabolic, immunological, or pharmacological, as with the medicinal products [Che03]. This Chapter presents the regulations governing medical devices and the standards relevant to medical device and medical device interface design. Furthermore, important issues such as security and privacy, use of non-medical technology, and medical device networking are presented.

Medical devices are governed by regulation which aims at guaranteeing the patient and operator safety, and well as the effectiveness or performance of the device. Medical device standards are tools for the regulators and well as the implementers, which aid in achieving of safety, performance, and other technical goals. Security and privacy issues sometimes contradict the goals of safety and performance design. Despite, they have a key role and should also be taken into account in medical device design. Home healthcare systems will face non-medical devices and may be required to interoperate with non-medical systems. Also, for cost-effectiveness, the use of PC and consumer electronics technology should be expanded.

In this Thesis the European definition of medical device is used. The directive 2007/47/EC [EU.07], which amended the Council Directive 93/42/EEC [EU.93] concerning medical devices, defines a medical device as "any instrument, apparatus, appliance, software, material or other article, whether used alone or in combination, including the software intended by its manufacturer to be used specifically for diagnostic and/or therapeutic purposes and necessary for its proper application, intended by the manufacturer to be used for human beings. Devices are to be used for the purpose of:

- diagnosis, prevention, monitoring, treatment or alleviation of disease,
- diagnosis, monitoring, treatment, alleviation of or compensation for an injury or handicap,
- investigation, replacement or modification of the anatomy or of a physiological process,

- control of conception,

and which does not achieve its principal intended action in or on the human body by pharmacological, immunological or metabolic means, but which may be assisted in its function by such means." [EU.07,EU.93]. Other national/regional definitions exists, of which the U.S. Food and Drug Administrations (FDA) definition is the best known. In section 201(h) of the Federal Food Drug & Cosmetic (FD&C) Act it defines medical device as "an instrument, apparatus, implement, machine, contrivance, implant, in vitro reagent, or other similar or related article, including a component part, or accessory which is:

- recognized in the official National Formulary, or the United States Pharmacopoeia, or any supplement to them,
- intended for use in the diagnosis of disease or other conditions, or in the cure, mitigation, treatment, or prevention of disease, in man or other animals, or
- intended to affect the structure or any function of the body of man or other animals, and which does not achieve any of its primary intended purposes through chemical action within or on the body of man or other animals and which is not dependent upon being metabolized for the achievement of any of its primary intended purposes."

Medical devices range from cardiac pacemakers to syringes or from complex computer based systems to simple mechanical aids. Medical device industry can be subdivided roughly into four sectors: a) medical-electrical devices; b) non-electrical products; c) implantables; and d) diagnostic products [Alt03]. In this Thesis the use of modern digital interfaces in medical devices is studied, and the main focus is on monitoring devices implemented using embedded computer/microcomputer systems, e.g., in the *medical-electrical device* sector.

An accessory is not considered to be a medical device, except if it is intended to be used specifically with a parent medical device to enable the parent medical device to achieve its intended purpose [Che03]. In this case it should be subject to the same regulatory procedures as the medical device itself.

When talking about medical devices, the question of safety usually rises up. The definition of safety and approaches for achieving it are presented in Section 3.1. In general, all commercial electronic devices must conform to certain regulations and standards. Additional regulations must be met and standards taken into account when designing a medical device. As medical device field is so heterogeneous, the field of regulations governing it is also wide and complex. These issues are discussed in the Section 3.2, which presents briefly how EU and US regulate medical devices. Standards are closely related to regulations, as discussed in Section 3.3. The Section also introduces the key standards for medical devices design, and then focuses on medical device interface standards which have been introduced to improve medical electrical device interoperability. Section 3.4 addresses the security and patient privacy issues related to medical

electrical devices. Modern computerized medical devices often adopt technology from the PC and consumer electronics market. The motivation and reasoning, and problems related to it, are discussed in Section 3.4. Modern computer systems and applications employ communication networks in everyday use. Issues regarding medical device networking are briefly presented in Section 3.5.

3.1 Medical device regulation and safety

Medical devices are regulated by national, regional and world wide bodies to improve the health and safety of patients and medical staff by providing them with reliable and effective equipment. To achieve this, regulations have been set on device design, manufacturing, documentation, marketing, sale, use, and disposal. In this Section, the medical device safety is first defined, and key players involving it identified. It is then shown how the safety of medical devices is achieved with the current regulatory systems. This is followed by a presentation of current EU and US medical device (pre-market) regulations.

3.1.1 Medical device safety

A good summary of what medical device safety means in practice is given by the Global Harmonization Task Force (GHTF, www.ghtf.org) in the Essential principles of safety and performance of medical devices. The first requirement states that:

"Medical devices should be designed and manufactured in such a way that, when used under the conditions and for the purposes intended and, where applicable, by virtue of the technical knowledge, experience, education or training of intended users, they will not compromise the clinical condition or the safety of patients, or the safety and health of users or, where applicable, other persons, provided that any risks which may be associated with their use constitute acceptable risks when weighed against the benefits to the patient and are compatible with a high level of protection of health and safety." [The05]

Absolute safety of medical devices cannot be guaranteed, as all devices carry a certain degree of risk and can cause problems in specific circumstances. Instead, safety is a risk management issue which is closely related to device effectiveness/performance. Optimum safety and performance require cooperation among all who are involved in the life span of a medical device: the manufacturer, importer/vendor, government, user and public [Che03]. Performance of a medical device is normally considered together with safety, as poor performance may lead to serious clinical safety problems.

The current approach to medical device safety is to estimate the potential of a device to cause hazards which could lead to safety problems and harm. This is known as the risk assessment. A *hazard* is a potential for an *adverse event*. *Risk* is a measure of the combination of the hazard,

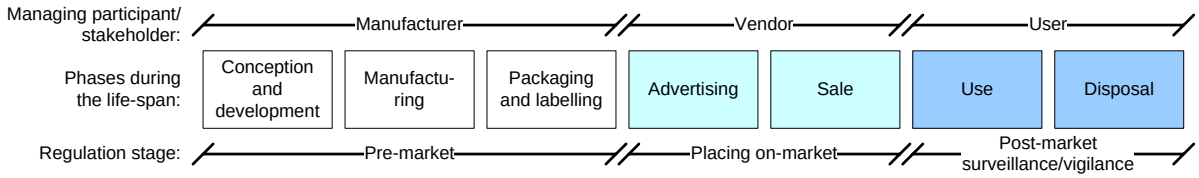


Figure 3.1: Major phases in the lifespan of a medical device and the managing participants and regulatory stages related to them. [Che03]

the likelihood of occurrence of the adverse event, and the severity or overall impact [Che03]. *Risk assessment* begins with identification of possible hazards, i.e., *risk analysis*. This is followed by *risk evaluation*, in which risk of each hazard is evaluated. The whole process is complex, based often on experience, evidence, computation, or even guesswork. Personal perception and other cultural, economic, and political factors may influence it.

The potential of hazard is affected by certain aspects of the medical device. These include the degree of invasiveness, duration of contact, the body system affected, and local versus systemic effects. The degree of regulation imposed on any device is proportional to its potential hazard. This approach is called *risk management*. The International Organization for Standardization (ISO) has produced a document (ISO 14971:2007) which provides a framework for risk analysis, risk evaluation, and risk control for risk management in medical device design, development, and manufacturing as well as for post-market surveillance.

The life span of a medical device can be split into seven phases Figure 3.1. All of these phases can affect the safety and performance of a medical device, but in this Thesis the focus is on the first phase, conception and development and related issues. The parties that are involved, and thus have their role in ensuring the safety of medical devices, are the manufacturer, the vendor, the user, the public (patient), and the government. In case of home-use devices, the user and patient may be the same. The ideal conditions that ensure the safety and performance of medical devices require shared responsibility by all five parties. From the device designer point-of-view it is worth to remember the results of Medical Device Agency (MDA) study reported by Grant [Gra98], in which 7000 reported medical device incidents in the UK during mid 1990's were investigated. Only about 300 incidents (just over 4%) were due to a genuine fault in the system, and 900 (13%) due to user error. The MDA, who conducted the study, also suspects that significant portion of the 2100 cases for which the reason could not be established were in fact due to user error. So, based on the MDA study, the risk for a user error is somewhere between 3 to 10 times greater that equipment failure. This is important to remember, as it shows the importance of the general device usability design, the documentation, and the training given to the device users. For home use devices, for which the user is usually the patient himself, the likelihood of user error is undoubtedly even greater.

3.1.2 Medical device regulation

Three stages of medical device regulation can be identified, the pre-market, placing on-market, and post-market stages 3.1. The placing on-market focuses on the sale of the product, and is of the responsibility of the product vendor. The post-market stage focuses on the use and after-sale of the product, and consists of surveillance/vigilance measures which are of the responsibility of the vendor and the user. The focus of this Thesis is mainly on the *pre-market stage*, which consists of control and monitoring of the product itself by the manufacturer. Furthermore, the emphasis is on the conception and development phase of the pre-market stage, mainly on the safety and performance regulations regarding the device attributes. The manufacturing (quality systems) and packaging and labeling issues are not discussed in detail.

The EU regulatory system for medical devices is quite young. The FDA regulatory system presented later in this section is 25 years older, as is (approximately) the EU regulation for pharmaceuticals. EU regulation of medical devices is embedded in the general policy on the single market (Articles 100 and 100a) and three device-specific directives:

- The council directive on active implantable medical devices (AIMD, 90/385/EEC), as amended by Directive 2007/47 EC;
- The council directive on medical devices (MDD, 93/42/EEC) [EU.93], as amended by Directive 2007/47 EC; and
- The in vitro diagnostic devices directive (IVDD, 98/79/EC) of the European Parliament and of the Council. (Adopted on December 8, 1998.) The IVDD amended sections in the AIMD and the MDD.

The medical device sector is also affected by various other directives, regarding public procurement, general product safety, electromagnetic compatibility, liability for defective products, and for computerized devices, and the information technology equipment. [Alt03] The current EU approach for legislation separates law and technical standards, i.e., the directives do not include technical standards and specifications. Instead, the directives define essential requirements and the use of voluntary standards. Finally, the EU directives themselves need to be transposed into national law in each member country, before coming effective. The EU post-market surveillance and medical device vigilance reporting systems, which are said to be the weakest link in EU regulation [Alt03], are not addressed in this Thesis. For control of directive conformity the EU systems uses so-called *Notified Bodies* (NB), which are impartial certification organizations appointed by the EU, which conduct formal audits of products and quality systems [Alt03]. Notified Bodies are primarily private organizations, they work on a fee-based contractual basis, and they have to be independent, impartial, and competent.

The EU medical device regulation uses a four-tiered system for device classification based on the degree of risk associated with device usage, the amount of time the device is in contact

with the human body, and the degree of of invasiveness of the device as shown in Table 3.1 [EU.93, EU.07, Cha00]. In the table, the types of devices of each class are given as an example. For *Class I*, the device manufacturer can declare conformity without use of a Notified Body, although they must maintain a predefined set of documentation in case of inspection. *Class IIa* devices have to be verified for conformity by a NB in the production stage. Directive 2007/47/EC [EU.07] introduced Class I sterile or measuring devices. *Class IIb and III* designs need to verified by a NB both at design and production stages. In addition, Class III devices need to be approved by a NB before they can be marketed. It should be noted, that no government authority reviews the decision of the NB, unlike in the US system, where the FDA makes the final decision. The exact guidelines on to which classes devices should be placed are in Annex IX of [EU.07]. Health monitoring devices implemented in this Thesis generally fall into the category of "Active devices intended for diagnosis", in which case they would be classified as Class IIa devices. If they had no diagnostic value, then they would be Class I, and if they would, e.g., be use for direct diagnosis, administration of potentially hazardous medicines, or monitoring of vital physiological processes, then they would go to Class IIb.

The EU system does not require demonstration of clinical effectiveness. It is sufficient that the device performs to the manufacturer's intended purpose. [Cha00] The CE marking on the medical device acts as a sign showing that the device meets the relevant regulatory requirements and when used as intended works properly and is acceptably safe. For more information on the EU regulations, please refer to the aforementioned directives and [Alt03, Alt07, Cha00, Hor95].

It should be noted, that in EU medical devices and pharmaceuticals have their own regulations. However, some medical devices may have components that are covered by pharmaceuticals regulation. This area is out of the scope of this Thesis, so the field of EU pharmaceutical regulation is not covered.

US legislation, governed by the FDA, is based on the idea that the degree of device regulation should correlate with the degree of risk posed by the device [Mai04]. The devices are divided into three classes listed in Table 3.2. Although similar, the classification is slightly different than the EU one. It should be noted, that the Global Harmonization Task Force is proposing a harmonized scheme for medical device classification in [The06]. Devices of all FDA classes must fill "general controls", including proper labeling and adherence to predefined "good manufacturing practices". This is sufficient regulation for Class I devices. Class II devices must meet or exceed certain predefined product performance standards. Class III devices must undergo thorough premarket evaluation and approval process. Before approval, new devices must demonstrate *safety* (device benefits outweigh the risks) and *effectiveness* (device reliably does what it is intended to do). The specific data required by the FDA to determine these depends on the type of the device, its use, and perceived risk to the patient. Moderate and higher risk devices usually enter the market in two ways, either by demonstrating "substantial equivalence" to a previously approved device, or by demonstrating its safety and effectiveness through a Premarket Approval

Class	Risk	Description	Examples
I	Low risk	Devices that enter the market with only the manufacturer's self declaration of conformity	Bandages, light sources
	Low risk (sterile)	Devices which wish to use the full quality assurance route to demonstrate compliance with sterile aspects of the device	Disposable surgical instruments
	Low risk (measuring)	Devices which wish to use the full quality assurance route to demonstrate compliance with measuring aspects of the device	Scales, digital thermometers
IIa	Medium risk	Devices that are subject to production quality system control registration by a third-party body	IV catheters, ultrasound devices
IIb	Elevated risk	Devices that are subject to quality system control for both production and design	Breast implants, ventilators
III	High risk	Devices that are subject to quality system control for both production and design. Conformity of the device's design must be considered separately before device is placed on the market.	Heart valves, reabsorbable implants

Table 3.1: EU medical device regulatory classes. [Cha00, Hor95, EU.07]

Class	Risk	Description	Examples
I	Low risk	Devices having minimal potential for harm	Stethoscopes, tongue blades
II	Moderate risk	Devices that need additional regulation to guarantee safety and effectiveness	CT scanners, endoscopes
III	Higher risk	Safety and effectiveness ensured by thorough premarket evaluation and approval process	Pacemakers, silicon breast implants

Table 3.2: FDA regulatory classes.

Application. "Substantial equivalence" requires demonstrating that the new device is similar to a legally marketed device, and applications regarding it are called "510(k) applications" on basis of the amendments that established them. For more details on the FDA procedures, please refer to [Mai04, Mon97], and for differences in EU an US regulation to [Cha00]. According to [Mai04] published in 2004, FDA receives annually approximately 4000 510(k) applications compared to fewer than 100 Premarket Approval Applications. For a manufacturer, the 510(k) application offers a generally faster and less expensive route to approval. The FDA post-market evaluation process, including reporting procedures and surveillance, is not addressed in this Thesis.

In general, unapproved medical devices should not be used by physicians. FDA has some special cases when it may be possible: Investigational use (strictly regulated and requires proper approvals), emergency use (certain criteria must be met [Mai04]), humanitarian use (for treatment of rare diseases, manufacturer has to only show that device has "probable" benefits that outweigh its risks). Unapproved use of an approved medical device is more of a gray area. In general, in US, it is left for the physicians to use their medical judgment on the matter.

For information on other regional regulatory systems, the reader is suggested to read the comprehensive book by Higson [Hig02] and the WHO's guidebook authored by Cheng [Che03].

A medical device targeted for the global market has to pass various national pre-market regulations. To reduce the work required for device approval WHO recommends that countries which do not yet have medical device standardization procedures should take advantage of existing approval systems and international standards. To further enhance medical device control and safe and effective use of medical devices, the WHO proposes that a global uniform certification format could be used, and that a global centre to coordinate and relay medical device problems, recalls, and alerts could be set up [Che03]. The *Global Harmonization Task Force* (GHTF, www.ghtf.org), founded in 1992, is a voluntary group with representatives from national medical device regulatory authorities and the regulated industry. Its purpose is "to encourage convergence in regulatory practices related to ensuring the safety, effectiveness/performance and quality of medical devices, promoting technological innovation and facilitating international trade. The primary way in which this purpose is accomplished is via the publication and dissemination of harmonized documents on basic regulatory practices. These documents, which

are developed by five different GHTF Study Groups, provide a model for the regulation of medical devices that can then be adopted/implemented by national regulatory authorities." (www.ghtf.org) The GHTF also serves as an information exchange forum for parties involved in medical device regulatory systems.

3.2 Medical device standards and interfaces

Standard is defined by the GHTF in [The08] as "document, established by consensus and approved by a recognized body, that provides, for common and repeated use, rules, guidelines or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context". Standards are closely related to the regulations, which were discussed in the previous Section. Regulations often state that a device has to or should conform to certain standards, and usually the approval for devices which conform to known standards is easier. A standard may be called a regulation when it becomes mandatory. This mandate may have legal basis, but this is not always the case.

Medical standards are related to medical devices. As noted in the beginning of the chapter, medical devices come in all shapes and sizes. This leads to a wide field of medical standards of which only a small subpart is often relevant to a single device. Besides medicals standards, a device may or can conform to various other technical and even non-technical standards. In this Thesis, our emphasis is on medical electrical devices, so the standards mainly related to them are discussed. In particular, the interface standards related to medical devices are of interest. Interface standards are standards which define common communication interfaces. Interface standards also improve safety and reliability in communication and promote device interoperability. *Interoperability* is defined by the IEEE as "the ability of two or more systems or components to exchange information and to use the information that has been exchanged" [Ins90]. In general, interoperability is usually used to describe the ability of two systems to work together. Two forms of interoperability can be identified: syntactic and semantic interoperability. Syntactic interoperability implies that two systems are able to exchange data, they have a common physical interface, communication protocol and data formats. Semantic interoperability requires syntactic interoperability, and extends the level of interoperability to include the meaning of exchanged information. Semantic interoperability ensures that both the sender and the receiver of the exchanged data have a common meaning of the requested services and data. It is based on agreements on, for example, algorithms for computing requested values, the expected side effect of a requested procedure, or the source or accuracy of requested data elements [Hei95]. In practice, all interface standards aim at providing syntactic interoperability by defining the hardware, software and protocols required for the data exchange, as was explained in the previous Chapter. Semantic interoperability is often provided by extending the interface standard with standardized device profiles or classes, or by implementing a common information exchange

model using other means such as additional communication layers. These may be part of the interface standard, non-standard proprietary definitions, or defined in another standard which builds on the underlying interface standard.

3.2.1 Standards

Standards are created and published by national or international standards organizations or by regulatory authorities. Examples of international standard bodies are International Electrotechnical Commission (IEC) and International Organization for Standardization (ISO), of regional bodies are European Committee for Standardization (CEN) and European Committee For Electrotechnical Standardization (CENELEC), and of national bodies the Deutsches Institut fuer Normung (DIN), the British Standards Institute (BSI) and the American National Standards Institute (ANSI). Most standards are voluntary, but in some countries standards are used as regulatory requirements rather than being voluntary. When a standard is mandated by, e.g., a government, it normally becomes legally obligatory based on regulations or a law established by the government. Standards can be classified in different ways. In [The08] three groups are used:

- Basic standard (also known as horizontal standards - Standards indicating fundamental concepts, principles and requirements with regard to general safety aspects applicable to all kinds or a wide range of products and/or processes (e.g., standards concerning risk management, clinical investigation and the quality management system for the manufacture of medical devices).

- Group standard (also known as semi-horizontal standards) - Standards indicating aspects applicable to families of similar products and/or processes making reference as fas as possible to basic standards (e.g., standards concerning sterile medical devices, electrically-powered medical devices, stability of IVD reagents).

- Product standard (also known as vertical standards) - standards indicating necessary safety and performance aspects of specific products and/or processes, making reference, as far as possible, to basic standards and group standards (e.g., standards for infusion pumps, for anesthetic machines or for blood glucose meters for self testing). [The08]

Standards can give specification for products, processes and services. In [Che03] four types of specifications are given:

1. Prescriptive specifications. Obligate product characteristics (dimensions, materials, test or calibration procedures) and definitions of terms and terminologies.

2. Design specifications. Specific design or technical characteristics of a product.

3. Performance specifications. Tests and requirements on performance.

4. Management specifications. Processes and procedures for companies (quality systems, environmental management systems etc).

Standards may contain a combination of these specifications.

Generic management system standards, which are becoming increasingly popular, can be applied to any organizations process management. The two most well known are the ISO 9000 series for managing quality systems and the ISO 14000 series for environmental management systems. The ISO 13485 and ISO 13488 are specific ISO quality systems standards for medical device manufacturing.

The benefits of using standards with a product, process or service are that they can [Che03]:

1. Provide reference criteria that should be met.

2. Provide information that enhances safety, reliability and performance.

3. Assure consumers about reliability or other characteristics of goods or services provided in the marketplace.

4. Give consumers more choice by allowing one company's products to be substituted for, or combined with those of another.

The four common industrial methods for assessing conformity to a standard are: testing, auditing (certification), registration, and accreditation [Che03].

The most well known national (regional) standardization bodies in Europe are CEN, CENELEC, and ETSI, and in the US the American National Standards Institute (ANSI). *CEN* (Comité Européen de Normalisation, or European Committee for Standardization in English, www.cen.eu) produces European EN-standards from all areas, except electronics and telecommunication, which are handled by CENELEC and ETSI respectively. *CENELEC* (Comité Européen de Normalisation Electrotechnique, or European Committee for Electrotechnical Standardization in English, www.cenelec.org) produces all European electrotechnical standards, of which many are based on international IEC-standards. *ETSI* (European Telecommunications Standards Institute, www.etsi.org) produces European telecommunication standards.

The three major international standardization bodies are ISO, IEC, and ITU, which mainly focus on three different areas. The *ITU* (International Telecommunication Union, www.itu.int) covers telecommunication. The *IEC* (International Electrotechnical Commission) covers electrical and electronic engineering. The *ISO* (International Organization for Standardization, www.iso.org) covers the rest. For some areas, like information technology, risk management, and quality systems, joint ISO/IEC technical committees exist. Other organizations, like the Institute of Electrical and Electronics Engineers (IEEE), also produce documents on international standardization. These documents are usually adopted by ISO/IEC/ITU as international

standards if they have been developed in proper manner. An example of this procedure is the ISO/IEEE 11073 medical device standards family.

The current trend in medical device regulations is to use voluntary standards established by a consensus from all interested parties. This eases the work of the regulatory authorities and frees their limited resources to other relevant issues, and it is also beneficial to the end users, who can get better, global, up-to-date standards developed by professionals of that area.

3.2.2 Medical device standards

Medical device standards are developed practically in all the major standardization organizations. The technical committee 215 on health informatics at ISO (ISO/TC215) works on standardization of health information and communications technology to allow compatibility and interoperability between independent systems. It consists of nine working groups (WG):

- WG1 Health records and modeling coordination
- WG2 Messaging and communication
- WG3 Health concept representation
- WG4 Security
- WG5 Health cards
- WG6 E-pharmacy and medicines business
- WG7 Devices
- WG8 Business requirements for Electronic Health Records
- WG9 SDO Harmonization

In Europe, health informatics standards are developed in CEN technical committee 251 (CEN/TC 251) in four workgroups:

- WG1 Information models
- WG2 Terminology and knowledge representation
- WG3 Security, safety and quality
- WG4 Technology for interoperability

In addition to these two committees, groups developing standards useful in medical devices and health information systems include:

- CDISC (Clinical Data Interchange Standards Consortium) - Develops and supports global, platform-independent data standards that enable information system interoperability to improve medical research and related areas of healthcare.
- DICOM (Digital Imaging and Communication in Medicine) - Medical imaging communication, world wide standards for radiology, ultrasound, endorsed by CEN and ISO.
- HL7 (Health Level Seven) - Works on health care messaging for clinical patient care and healthcare services. Co-operation with CEN and ISO.
- IHTSDO (International Health Terminology Standards Development Organisation) - Develops, maintains, promotes and enables the uptake and correct use of its terminology products in health systems, services and products, notably SNOMED CT (Systematized Nomenclature of Medicine-Clinical Terms), to support safe, accurate, and effective exchange of clinical and related health information. Focuses on enabling the implementation of semantically accurate health records that are interoperable.
- ISO/IEC JTC1 (Joint Technical Committee 1) - Development of information technology standards
- ITU-T/SG 16 - Works on multimedia communication, and also how multimedia system can support e-health applications, particularly in the area of telemedicine.
- IETF (Internet Engineering Task Force) - Development of the internet architecture.
- OMG (Object Management Group) - International not-for-profit computer industry consortium that develops enterprise integration standards for wide range of technologies. Some well known specification include UML and CORBA.
- W3C (World Wide Web Consortium) - Develops interoperable technologies related to the WWW.

As the medical device field is so heterogeneous, the field of applicable standards is huge. In Table 3.3 some of the most relevant standards for a medical-electrical device designer are presented. The proposed use and properties of the device designed define further standards which should be used.

Historically, the standardization of medical electrical devices has focused on electrical safety, which has be addressed by the IEC 60601-1 since 1977 [Sid06]. The 2005 release of the IEC 60601-1, third edition, adds risk management (ISO 14971 standards based) to its contents [Sid06]. By applying risk management type of procedures, the third edition is designed to keep pace with the technology developments, and can be applied to new emerging technologies. This change has meant that the IEC 60601-1 has evolved from a basic safety standard to also an performance

Generic device standards	
EN 980	Symbols For Use In The Labeling Of Medical Devices
EN 1041	Information supplied by the manufacturer of medical devices
Safety and quality standards	
EN ISO 13485	Medical Devices - Quality Management Systems - Requirements For Regulatory Purposes. (Particular requirements for the application of EN ISO 9001)
EN ISO 14971	Medical Devices - Application Of Risk Management To Medical Devices
IEC 80001	Application of risk management to information technology (IT) networks incorporating medical devices (Note: Scheduled for approval in 2010)
Medical electrical devices	
IEC 60601-1	Medical Electrical Equipment - Part 1: General Requirements For Basic Safety And Essential Performance
IEC 60601-1-2	Medical Electrical Equipment Part 1-2: General Requirements For Safety - Collateral Standard: Electromagnetic Compatibility - Requirements And Tests
IEC 60601-1-4	Medical electrical equipment Part 1-4: General requirements for safety - Collateral standard: Programmable electrical medical systems
IEC 60601-1-6	Medical electrical equipment Part 1-6: General requirements for safety - Collateral standard: Usability
Medical devices with software	
IEC 62304	Medical Device Software - Software Life Cycle Processes
IEC 61508-3	Functional safety of electrical/electronic/programmable safety related systems
Software in health systems other than medical devices	
ISO/PRF TS 29321	Health Informatics - Application of clinical risk management to the manufacture of health software (Note: Did not pass the vote in ISO)
ISO/IEC 12207	Information technology Software life cycle processes
Patient safety	
CEN CR 13694	CEN Report: Health Informatics - Safety and security related software quality standards for healthcare (SSQS)
CEN TS 15260	Health informatics - Categorization of risks from health informatics products
CEN TR 15299	Health informatics - Safety procedures for identification of patients and related objects
ISO DTS 25238	Health Informatics - Classification of safety risks from health informatics products
ISO/TR 21730	Health informatics - Use of mobile wireless communication and computing technology in healthcare facilities. Recommendations for the management of unintentional electromagnetic interference with medical devices

Table 3.3: Medical device standards.

standard, kind of which was lacking in the field. The combination of the IEC 60601-1 and the ISO 14971 standards is an unavoidable part any medical electrical equipment and system design [Sid06].

Medical software regulation and standardization is rather new area. The current EU directives define, that "Software that is intended to be used specifically for diagnostic or therapeutic purposes will be subject to the directive's requirements. Recital (6) of 2007/47/EC amplifies this by stating that software for general purposes when used in a healthcare setting is not a medical device." [EU.07] The Directive 2007/47/EC will be applied from 21.3.2010. The practical interpretation and scope of application of the directive has been under heavy discussion, and it seems that a more stricter interpretation will be applied, which will lead to widespread regulation of software. The practical tools and procedures for software MDD conformance are still under development, and are currently hastily being worked on.

An interesting developing standard regarding personal health monitoring at home, is the "EN 60601-1-11 General requirements for basic safety and essential performance. Collateral standard. Requirements for medical electrical equipment and medical electrical systems used in home care applications CH/62/1". It may have impact in the future design of home healthcare systems.

Also other non-medical standards and regulations have to be taken into account. For example, wireless devices must adhere to the RF regulations defined for the used radio frequency band and communicate as defined by the interface standard. The interface standards are discussed in the next Section.

3.2.3 Medical device interface standards

As presented and discussed in this Chapter, medical devices are governed by several regulations and must conform to various standards. From interface technology viewpoint, the general move has been towards more standardized interfaces. Also media storage formats utilize standardization, so that recorded data can be transferred between computer systems and software applications from different manufacturers. What has been missing is a universal so-called plug-and-play (PnP) system to connect medical electrical devices, which would not be limited to a single physical interface technology. Work in this area has been done for a quarter century, starting from the IEEE 1073 standard [Ken94], and ending to the efforts of Continua Health Alliance [Car07] and the ISO/IEEE 11073 standards [Sch07] which they promote, as presented later in this Section. Only in the very last years has this effort started to show, as the industry has recognized the benefits of standardized interfaces, and started to promote their use in widespread fashion.

Most of the new medical devices coming to market, especially in the home and personal health market use some known and widely used standard interface technology. Yet still, this

does not allow us to plug devices using the same interface together. They share the same physical interface, same communication protocol, but have no standard means to interact with (i.e., talk to) the application. Thus a device from vendor A cannot be plugged to system by vendor B and expected to work, even though they use the exact same digital interface. Indeed, they may even prevent each others from working, or reduce the total system performance, if used together. To a non-technical user this can be both frustrating and confusing. To address these issues, interface specifications have implemented device classes and profiles to define common characteristics and APIs for devices belonging to certain application groups. USB, Bluetooth, and Zigbee have all implemented health related device profiles and classes to harmonize their use in medical and health related devices [USB07, Blu08, Zig09b]. These are the USB Personal Healthcare Devices Class (PHDC), Bluetooth Health Device Profile (HDP), and Zigbee Health Care profile.

Medical device market has traditionally been quite closed. Vendors provide complete systems, and the system can only be extended by adding more devices from the same vendor to the system. This of course makes it simpler for the vendor to guarantee that new devices will work together with the old system, but it also guarantees long-term income from the installed systems, as the users are tied to the existing system. Also, it can be argued that a system or use case specific interface design can be made optimal for the design, compared to a common standard interface solution, by reducing communication overhead and optimizing resource usage. These three are the main reasons why the system providers have been quite passive in universal medical interface development and deployment. On the other hand, by using a standard interface, the device approval process will often be easier, as the safety and effectiveness of the interface has usually already been proven.

Clinical data modelling

At low-level, achieving technical interoperability between systems is enough to obtain syntactic interoperability. For semantic interoperability high-level ontologies and detailed clinical models are required to define standardized nomenclature on how to express data related to different values. Indeed, detailed clinical models are the basis for retaining computable meaning when data are exchanged between computer systems [Coy03]. The goal in use of clinical data models is not only to have the data available for human to read and to understand, but to have it structured and coded in a way that will allow computers to understand and use the information [Coy03]. For this there must be a formal way of stating the information model and for referencing standardized terminologies that are used for the data elements in the model. The challenge is that a unique and comprehensive ontology of the medical domain is not within sight [Len07]. Semantic integration of heterogeneous systems in healthcare will have to deal with volatile medical concepts. This issue can be tackled in standardization by limiting the scope of a standard to a certain use cases or usage areas, such as point-of-care or personal health devices in medical devices, and providing flexibility and means for extendibility in the specification.

ISO/IEEE 11073

The *ISO/IEEE 11073 standards* (also commonly referred as the x73) family development goes back to year 1984, when the IEEE standards committee was charged with writing the "Standard for Medical Device Communications", the IEEE P1073, to interface medical electronic device to host computer systems in a standard, interchangeable manner [Ken94]. The original standard was optimized for acute care environments such as ICUs, operating rooms, and emergency rooms. During the development, four key requirements were identified: 1. Frequent reconfiguration of the network; 2. Allow "plug-and-play" operation by users; 3. Associate devices with a specific bed and patient; 4. Support a wide range of hospital computer system topologies. From the beginning, the standard was designed using the OSI seven layer model [Zim80] and using existing ISO standards in the layer implementations, so that it could come day be made an international standard. The P1073 was approved by the IEEE as the standard 1073 in 1994 and approved as an ANSI standard in 1995. At that time the interface was called the Medical Information Bus (MIB). As physical interface, the IEEE 1073 relied on the Serial (RS-232) and IrDA ports, for which physical interface specifications were written. This was probably one of the reasons why the IEEE 1073 somewhat failed, and did not become as popular as was hoped. At same time when the IEEE 1073 was finally approved, new wired and wireless interface standards emerged, and made the IEEE 1073 seem somewhat outdated. Meanwhile, in Europe, CEN had developed Point-of-Care Medical Device Communication (PoC-MDC) standards, which were approved in 1999. In 2000, work on creating a single ISO standard was started, and in 2004 the first series of ISO/IEEE 11073 Health informatics - Point-of-care medical device communication standards was published. The standard adopted the lower-layers of the IEEE 1073 standards and absorbed ENV13734 (VITAL) and ENV13735 (INTERMED) for the upper and middle layers, respectively. The IEEE is still responsible for the development and maintenance of these standards with the participation and input from the ISO member bodies.

The ISO/IEEE 11073 *Point-of-Care (PoC)* standards family consists of three main series, the 11073-1x, 11073-2x, and 11073-3x series. The 11073-1x series defines the Medical Device Data Language (MDDL), which defines the semantics needed to communicate and properties of specific device specializations, i.e., what properties all Heart rate monitors (P11073-10406) or Pulse oximeters (P11073-10404) have. In other words, the 11073-1x series defines the OSI layer 7 (application). The 11073-2x series, Medical Device Application Profiles (MDAP) has some layer 7 functions, but mainly it defines the OSI layer 5 and 6 (presentation and session) properties. OSI layers 1 to 4 (physical, data link, network, and transport) functions are defined by 11073-3x series, Transport & Physical Profiles (TPP) [Gal06]. In addition, the ISO/IEEE 11073 standard series consist of additional standards related to its use and special applications. Figure 3.2, left hand side, depicts how the three base standard series build the PoC protocol model. The ISO/IEEE 11073 is rather complex and sets heavy requirements to software developers, which has also reduced interest to adopt it. However, it can be implemented on less computationally

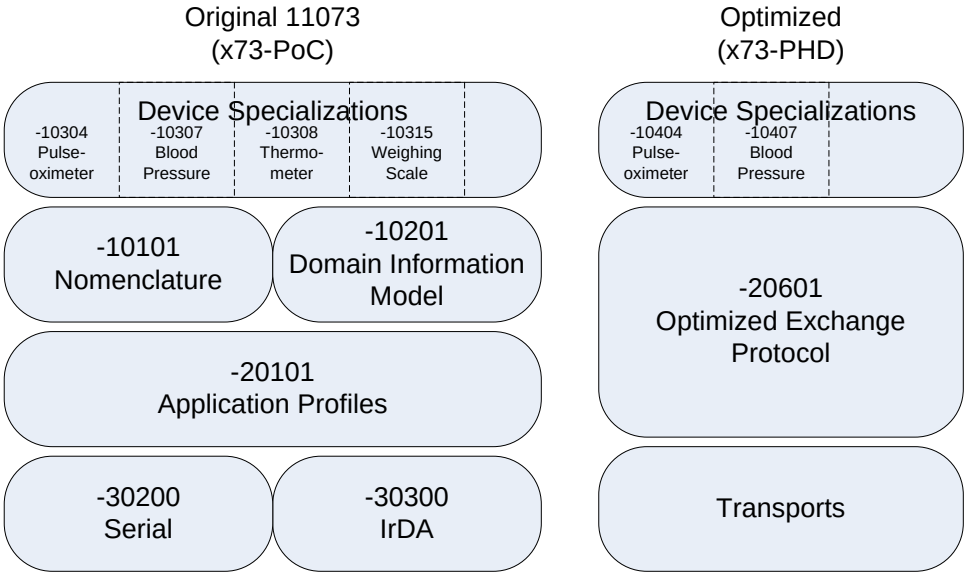


Figure 3.2: The two ISO/IEEE 11073 standard series and their protocol models.

powerful systems by hardcoding some fixed parts of the messages and by using a pattern library, as discussed in [ME08].

Recently, a new optimized exchange protocol (ISO/IEEE 11073-20601) for *Personal Health Device (PHD)* has been designed [Cla07, Ins08b]. The new protocol has been designed to be portable on different new transports (including Bluetooth and USB), and it is specially targeted for personal health in home and mobile environments and the portable devices used in those contexts. The 11073-20601 includes functionality from -20101 and -10201. Also, functionality from 11073-10101 has been split to -20601 and the new Device specialization (-104xx standards), to obtain the simplified structure shown in Figure 3.2, right hand side. The Continua Health Alliance, presented in the next subsection, has been a major driving force behind this new optimized standard. The optimized protocol transmits all static information in one configuration phase, and then only sends the dynamic information, meaning the actual measured values without parameters describing the type of measurement or unit of the value. This reduces message overhead and shortens time spent in transmission [Sch07]. The transport specifications are out of the scope of the 11073-PHD standards, and are developed as medical profiles in special interest groups of various standards (Bluetooth, USB).

Continua Health Alliance

The *Continua Health Alliance* (CHA, www.continuaalliance.org) was formed in 2006 by healthcare and technology companies to improve the quality of personal healthcare by developing interoperability guidelines for the emerging personal telehealth ecosystem. CHA develops its

guidelines using mainly existing industry standards, which are selected by thorough evaluation and comparison against the requirements in the healthcare field. The selected standards are promoted by the Continua Health Alliance, and interoperability guidelines made to define profiles over standards and guide in product certification. To ensure compatibility, CHA also defines a certification and testing program to verify compliance, which when passed allow the product to carry a CHA interoperability logo. [Car07]

The Continua Health Alliance focuses on three use areas:

- Disease management: managing a chronic disease outside of a clinical setting.
- Aging independently: using technology and services to live in the home longer.
- Health and fitness: expanding personal health and wellness to everyday life.

CHA uses a so-called End-to-End (E2E) Reference Architecture, which defines five device types (PAN device, LAN device, Application-hosting device, WAN device, and Health record device) and four interfaces to connect them (PAN interface, LAN interface, WAN interface, and xHRN interface). The architecture covers the full data path of a measurement made by a sensor as to a stored value in the patient health records. The application-hosting device is a central device to which data is sent by the PAN and LAN device using the PAN-IF and LAN-IF. It then communicates with the Health record device via the WAN device. The PAN-IF and LAN-IF both have lower- and upper-layer component. The upper-layer consists of OSI layers 5-7, and lower-layer from OSI layers 1-4. Indeed, the upper-layer for PAN-IF and LAN-IF is the ISO/IEEE 11073-20601, and the lower-layers are the transports supported by the ISO/IEEE 11073-20601, including the currently worked on USB and Bluetooth for PAN-IF, and proposed WLAN and Ethernet in the future for LAN-IF. Then WAN-IF will also be compatible with the 11073 data model, and it will support various IP-centric communication technologies. The health record network interface (xHRN-IF), will utilize standards developed by the HL7 group.

The Continua Health Alliance has been accepted well by the industry, and has grown quickly. First products have come to market, and it is likely that CHA will make the ISO/IEEE 11073 standard a major player in the home health monitoring devices of the near future, and also enhance its use in other medical devices.

IHE Patient Care Device

Integrating the Healthcare Enterprise (IHE, www.ihe.net), is an initiative by healthcare professionals and industry to improve the way computer systems in healthcare share information. IHE works by defining IHE Integration Profiles which describe clinical information management use cases and specify how to use existing standards to address them. IHE itself does not define new integration standards, but rather support for existing standards [Len07]. Thus IHE is an implementation framework, not a standard. A system that implements an IHE profile interoperates

with another device implementing the same profile. IHE also provides Integration statements for the customers which describe the IHE profiles that a specific product implements. Furthermore, IHE provides Technical Framework documentation for the device designers which describe the technical implementation of an IHE profile and the associated systems and transactions. IHE also provides annual "Connectathon" events where the interoperability of products can be tested with other vendors. [Sie01, Car03]

IHE is organized into domains which address different areas of healthcare. The IHE profiles focus on the interoperability problems at information systems level. However, there is one IHE domain which profiles are closely related to medical device interfaces, the IHE Patient Care Device (PCD) domain. In the technical framework, IHE-PCD focuses on promoting the use of ISO/IEEE 11073 and HL7 standards, and specifies how information is exchanged between these standards. The current technical framework documents for IHE-PCD are at "Out for Trial Implementation" state, final releases do not exist yet.

The work of IHE PCD is similar to Continua Health Alliance, and they both base their work on the ISO/IEEE 11073. CHA focuses on personal well being and consumer markets, while IHE PCD is focused towards hospital and healthcare sector, and there is co-operation between these organizations.

Medical device Plug-and-Play (MD PnP)

The mainly US-based M*edical Device Plug-and-Play* (MD PnP) program, previously known as the Operating Room of the Future Plug-and-Play (ORF PnP) [Gol05], started as a forum for bringing together diverse groups of stakeholders for achieving a standard for interoperability of medical devices in the operating room environment. The current MD PnP program has five goals stated in [Gol05]:

1. Lead the adoption of open standards and related technology to support medical device interoperability and system solutions.
2. Define a regulatory pathway in partnership with the FDA and other regulatory agencies.
3. Elicit clinical requirements for the proposed interoperable solutions to maintain focus on patient safety.
4. Use vendor-neutral laboratory to: evaluate interoperability standards and solutions, model clinical use cases (in simulation environment), and serve as a resource for medical device interoperability.
5. Investigate safety of proposed engineering solutions.

The MD PnP group is currently developing an Integrated Clinical Environment (ICE) standard for integrating medical devices. The ICE relies on existing interface technologies to provide data transport. It then builds its own data models on top of these technologies. Based

on the draft release [AST09] the approach is somewhat similar to that of ISO/IEEE 11073. For more information on MD PnP and the ICE development, please refer to their web-site, www.mdpnp.org.

3.3 Security and privacy issues in medical devices

Besides regulations and standards which aim for safety and effectiveness of medical devices, also *security* and *privacy* issues should be considered in medical device design. When a breach of security compromises the safety of the system, the security also becomes a safety issue, and is governed by safety regulations. More than often, however, security and privacy goals have to be balanced against safety and utility goals, as improvements in overall security often effect the usability of the device. Strong security mechanisms, such as public-key cryptography, can be expensive in terms of both computational time and energy consumption, which is an example of security and usability trade-off. Patient data security and privacy is a major factor in design of hospital information systems and electronics health records, and their applications. For medical electrical devices the security issues are often related to the use of the digital interface of the device. Security issues are also more relevant in devices having wireless interfaces, which are easier to listen and manipulate undetected. Security and privacy issues are related to regulatory issues discussed previously in this chapter, as there are ethical rules, and regulations which define patients' rights on privacy. The role of security and privacy in medical device design is growing, as medical devices are no longer isolated from networks and start to interoperate. Also, established methods for providing safety don't prevent intentional failures or other security and privacy problems [Hal08].

[Hal08] presents how standards principles of computer security, including confidentiality, integrity, and availability, extend to implantable medical devices. The following goals apply well also for non-implantable personal health medical devices, which are the main scope of this Thesis:

- Authorization and authentication. Who can access the device, and are we talking to the correct device.
- Availability. Prevent denial-of-service type of attacks, drainage of devices battery, internal memory overflow, or jamming of the communication channel.
- Device software and settings. Only authorized users should be able to modify software and settings.
- Device-existence privacy. An unauthorized party should not be able to detect that a person is using a medical device.

- Device-type privacy. If device reveals its existence, its type should only be disclosed to authorized entities. This is more related to implantable devices and other devices related to a treatment or monitoring of a specific illness, of which nature would be revealed if device type is known.
- Specific-device ID privacy. A public device ID on a personal device would allow an adversary to track a person, and compromise individual's location privacy.
- Measurement and log privacy. An unauthorized party should not have access to private information about the measurements.
- Bearer privacy. A personal medical device, which may contain information regarding patients name, medical history and diagnoses, should not reveal this identity information.
- Data integrity. An adversary should not be able to tamper with past or future data stored in the device, including modification, deletion, and addition of data. [Hal08]

The solutions used currently to tackle security issues include [Mei06]:

- Role-based access control: Different user groups are assigned rights based on their role in the care of the person.
- Encryption: Ensure data security and prevent eavesdropping and skimming.
- Authentication mechanisms: Used to ensure that the data is coming from the person/entity that it is claiming to be from. Methods include passwords, digital signatures, and challenge-response protocols.

As noted before, the digital interface of a medical device and its use, communication over the interface, is a key player in medical device security. Security and privacy requirements depend on the application. For a main-stream personal health device, the heart rate monitor, a long operating time and small size are more important than privacy issues. But for a remote controlled infusion pump security is extremely important. For a sensory device, which does not relay life-critical data, the most sensitive data is usually the one containing patient identification information; patient name and other personal information. This data is rather static, and depending on the application it may not be needed to transmit very often, but when it is, it should be encrypted well.

Medical devices using wired interfaces can in most cases be considered as closed systems, i.e., for an outsider to be able to eavesdrop on the data communication or have access to the device he would have to break into a physical environment (a room etc) and make a physical connection to the system. Thus these kinds of systems can be considered rather secure if the access to the premises and to the devices is controlled. Of course this does not prevent from

an insider attack, where the adversary comes from inside the organization. But as networking of medical devices, wired and wireless, increases so do the opportunities of an outside attack. These issues are briefly discussed in Section 3.5.

Wireless interfaces are less secure by their nature than wired devices. They use an open medium, the radio waves, for communication, and this communication can be listened if one is within the range of the devices. Thick and specially constructed walls limit the radio signal propagation very effectively, but relying on it is insecure. An attacker can use a more sensitive and powerful radio transceiver than specified in the standard to gain access beyond the normal range. As long as open radio standards are used for the physical radio communication one has to assume that anybody can listen to what is broadcasted to the radio waves, be the radio system complex frequency hopping one or not. It is also poor design to assume that using a proprietary non-standard radio link would somehow guarantee security, though it does make the system more difficult to listen than a standard based one. Indeed, in current communication systems design, open security standards are considered the most secure solution, as they are the most stringently tested. It is often the poor configuration of wireless systems that leaves the door open for attack, rather than poor design of the communication protocol, that is to blame.

A notable issue for a device in personal health use is that it can be easily stolen, and thus the device can be opened and components accessed directly. This should be taken into account when devices that store sensitive data are designed. If the device uses standard non-volatile storage solutions, then the data should be stored in encrypted fashion. A stolen wireless monitoring device can also be used to send false recordings to the system, if no security means are implemented to prevent it.

Lastly, privacy issues regarding clinical data and patient information are also problematic in medical device development and approval processes. The regulators require documentation in support of submissions for market approval, and evidence that products are safe and effective based on clinical evaluation data. [Alt07]. The issues become even more complex when effectiveness is defined in terms of improved clinical outcomes for a new device over an existing product in the market.

3.4 Use of PC and consumer electronics technology in medical devices

The first computer-based medical instrument was a computerized axial tomography scanner, which revolutionized clinical approaches to noninvasive diagnostic imaging procedures [Bro95]. Since then, computer based systems have evolved among the general development of microcomputers to the powerful multi-core PCs with animated graphical user interfaces and integrated networking services. The eight basic functions that may be provided by medical computer

systems are identified in [Wie06]:

1. Data-acquisition and presentation
2. Record keeping and access
3. Communication and integration of information
4. Surveillance
5. Information storage and retrieval
6. Data analysis
7. Decision support
8. Education

Nowadays, desktop computer systems are so complex that if one would be able to design and build a working computer system from scratch, including hardware and software, testing it and its interoperability with other systems would take ages. Therefore, if a powerful multi-purpose computer system is needed, it is usually based on standard PC components. This has lead to wide use of PC technology, including interfaces and other standards, also in medical devices.

Even for the smaller medical electrical devices, implemented with microcontrollers, the use of interfaces and communications standards derived from the PC world is important. Firstly, many of the devices are connected to PCs, so a common interface is required. And if they are not directly connected to a PC, then to a standardized networked interface, which also leads to using communication protocols popular in the PC world, like IP stacks. It can be argued, that PC world and consumer electronics market (mobile phones, digital cameras, media players etc) with their huge volumes of manufactured devices drive the development of interface technologies. The high production volumes drive the component prices down, and the manufacturers are required to implement and provide software support for their devices and the related interface. The medical device manufacturers thus face a choice of choosing a cheap to implement and well supported & tested interface with at least adequate performance or implementing their own proprietary interface. In addition, modern medical systems are increasingly more often networked. Networks pose such security issues that proving the systems secure and fixing the occurring security faults in the communication protocols would require huge resources. Using a standard interface with verified security protocols is often the safest solution, and the only viable alternative for a smaller manufacturer.

During the last decade, handheld mobile computers, like PDA's and smartphones, have become hugely popular in medical use [Gar06b, Tor03]. Equipped with a wireless interface, usually WLAN and/or Bluetooth, they paved the way for open wireless standards use with

medical devices, and even in hospitals, which for long time were reluctant to allow use of personal wireless devices.

It is not only medical devices that are affected by the PC/consumer electronics driven technology. Also more generic data acquisition systems are moving towards using standard interfaces like USB and WLAN [Gre09]. Previously, Gilsinn reported in [Gil01] that generic wireless interface technologies are considered for adoption in the IEEE 1451 sensor interface standard. Similar trends can be seen in all areas of industry, as shown by [Cou03] in which proprietary systems in retail industry were replaced with modern PDA technology and wireless interfaces.

Medical devices adopt new technologies slower than consumer electronics, as the technology used has to be reliable, i.e., tried and tested [Bra05]. The home healthcare market is an area of medical devices which is closest to the consumer electronics market, and thus often the first to adopt new technologies from its surroundings. Currently, Bluetooth, Ethernet, USB, and Wi-Fi (WLAN) are technologies which are becoming or have become mature enough to be used in medical devices. Ethernet and Wi-Fi especially have introduced IP-technology to medical devices, and it is likely that services and standards built on-top of IP stack will have impact on future medical devices. Technologies promoted by DLNA (Digital Living Network Alliance), UPnP (Universal Plug-and-Play) forum, W3C (World Wide Web consortium, XML document development), OSGi Alliance (interoperability of applications and services based on component integration platform) will surely have some impact on home healthcare systems of the future. Especially UPnP efforts have similar device interoperability goals to Continua Health Alliance, and it is interesting to see how the interoperability standards for home and personal devices develop, and what it will mean for personal and home health care device designers. It is clear that in the long run home healthcare devices should by some means interoperate with other home electronics devices, as it will allow the devices to give more and better services to the user.

The use of common standardized interfaces in medical devices has many benefits, which have been discussed previously. However, there are also some disadvantages and issues which should be considered and noted in design. Devices using common standardized wireless interfaces should note that the non-medical devices operating in the same radio band may downgrade medical system performance and at worst case use all of the radio bandwidth. This is especially the case with wireless systems using the 2.4 GHz radio band, which is very popular and shared among several competing radio standards. Even if the technology used would have quality-of-service measures to guarantee certain bandwidth, a competing technology may still interfere and downgrade the radio link performance. For wired systems, using the physical connector defined by the interface standard enables plugging in to the system any non-medical device using the same connector. If this hasn't been taken into consideration, it may lead to problems. For bus based wired systems the worst case scenario is that a non-medical device takes up all or most of the bus transfer capacity, and disables the operation of the medical devices on the bus or any new device connected to the bus later on. An example of this is the case where

a web camera or a similar high bandwidth device is connected to a USB bus of a PC, and a later connected medical device requiring guaranteed bandwidth can not be added to the bus, as there is not enough guaranteed bandwidth available. Later and faster versions of USB have more bandwidth, so this is now less of an issue than previously, but it still serves as a good example. Also, a medical device having a standard interface connector can be misused and connected to a non-medical device such as standard PC, which does not have the necessary software to communicate with the device. This may cause confusion and ambiguities, especially for a non-technical user who expects the device to work just by plugging it in, and should be considered in the design of the system. In addition, as a general design note, not all components manufactured for consumer electronics may not fill the quality requirements or expectations set for medical devices. An example of this, which happened during the work done for this Thesis, is the case in which a poor-quality 9 V battery-connector failed on a BCG amplifier as the nurses made a just-in-case battery change before a start of a long patient trial.

3.5 Medical device networking

Several networking solutions have been adopted for use in medical systems, which range from hospital wide (and even country wide) networks to the communication needs of implantable devices. Wide area networks (WAN) are used to connect hospital and other health care facilities together. Local area networks (LAN) are in use in hospital wide systems and in smaller scale applications. Personal area network (PAN) technologies are be used in local applications, where networks are often formed ad-doc with no predefined lifetime and configuration. This is often the case with mobile devices. These traditional network definitions do not fully suffice the needs of medical practitioners. The term *Body area network* (BAN) has been developed to describe a small network which connects vital sings and other biosignal sensors placed on a person together. Also terms Body sensor networks (BSN), and Wireless body sensor networks (WBSN) have been used in the literature. Where PAN refers to devices at near proximity the a person, the BAN refers to devices on (implanted) or worn (wearable) by the person, although currently most wireless BAN implementations are based on standards PAN technologies and some researchers do not make distinction between BAN and PAN. IEEE 802.15 standard defines BAN as "a communication standard optimized for low power devices and operation on, in or around the human body (but not limited to humans) to serve a variety of applications including medical, consumer electronics / personal entertainment and other".

The networking of medical devices opens new application areas, created by using data fusion from various sources. It can also improve the quality of care by faster data transfer between systems and by reducing human errors in information relay. However, the networks also pose risks and issues familiar from the computer networks. A networked device is more vulnerable to outside attacks, which is both a safety and security issue. Some systems may use the Ethernet

interface which should form a closed intranet between the devices, or use a protected hospital intranet. If these devices are connected to standard internet, they may be very vulnerable. Interoperability of networked devices is critical, using standardized data and message formats alleviates changes of error, but packet loss or a duplicate packet may cause an application to deadlock. New forms of device errors and system level malfunctions may emerge when existing devices are networked. This is an area which is currently looked in IEC, where a standards "IEC 80001 - Application of risk management to IT networks incorporating medical devices" is currently being worked on. The standard is expected to finalize in 2010.

Networked medical device safety and security is highly related to medical device software development, as the network protocol stacks are mostly implemented in software, and in embedded systems make up for a large portion of the total code space. This is why ongoing efforts and discussions on medical device software standardization effect also the future designs of networked medical devices. As noted by Lee et al. in 2006 [Lee06], medical device software development has reached the point where testing as the primary way to gain confidence in a system is impractical or ineffective. As devices grow more complex and rely more on embedded software to achieve critical functionality, existing certification processes are being stressed. This leads to higher development costs, longer time to market, and increased chances of device failure. Component and model based development and design of software, including interface and networking components, an adoption of it in medical device software certification is presented as a solution in [Lee06]. Standardized interfaces and other communication standards play a key role in this development.

3.5.1 Wireless medical devices

Wireless interface technologies and networks are often desired in medical devices, as they increase mobility. Not only device mobility, but also patient mobility, if the patient under monitoring is not physically forced to stay in his bed because of measurement cables. This mobility can reduce demands on the health care professionals, as the patient can perform some daily activities himself [Cha07]. Wireless networks may also reduce costs by reducing the amount of wiring needed for communication networks [Che05].

Wireless medical devices have some design requirements, with some key points to consider:

1. Low battery power requirement. To achieve this, the system design sometimes needs to be rethinked. An active wireless (radio) transmitter consumes much power. Often the total power consumption is reduced if data processing is moved from the receiver end to the sensor/transmitter end, i.e., instead of sending continuous raw sensory data only generated alarms or calculated values are sent over the wireless link.

2. Safety and reliability. No radio link is 100 % reliable. Wireless technologies should be avoided in those medical applications where guaranteed immediate communication is re-

quired to avoid injury or even death. On body worn devices the Specific Absorption Rate (SAR) limits, which define the rate at which RF energy is absorbed by the body when exposed to RF electromagnetic field, may need to be considered to guarantee physical safety of the user.

3. Data rate. The real required data rate should be calculated by rigorous system analysis, which is not usually the case. Increases in the data rate are costly and lead to increased power consumption or compromises in other systems design parameters.

4. Data latency. Should be considered in design. The latency needs of the application on hand should be evaluated and prioritized, as some application are more critical than others. [Cha07]

Only a small number of globally available radio bands free for medical use exist. Not surprisingly, most wireless medical devices operate on these bands, of which the 2.4 GHz band is currently the most used. For implantable devices, having their own issues regarding body loss and inefficient antennas, the 400 MHz region provides a good compromise [Cha07].

Summary

Medical devices are governed by regulation which aim at guaranteeing the patient and operator safety, and well as the effectiveness or performance of the device. No global unified regulations exist. Medical device standards are tools for the regulators and well as the implementers, which aid in achieving of safety, performance, and other technical goals. The EU legislation separates law and technical standards. Risk evaluation based methods are increasingly more employed in the safety standardization, also in the evaluation of electrical safety. Security and privacy issues sometimes contradict the goals of safety and performance design. Nonetheless, they have a key role and should also be taken into account in medical device design. Home healthcare systems will face non-medical devices and may be required to interoperate with non-medical systems [P8] [Zak09]. Also, for cost-effectiveness, the use of PC and consumer electronics technology should be expanded [P2][P4][P8][P1]. Networking of medical devices will play a key role in future medical device design [P4][P8][P7].

The next Chapter looks at medical devices' use in personal health monitoring, the primary application area of this Thesis.

Chapter 4

Personal health monitoring

Personal health monitoring at home is the main application area of this Thesis. This Chapter introduces the terminology and different research areas related to personal health monitoring. Motivation and reasoning for implementing personal health monitoring systems is given and different forms of measurement used for personal health monitoring are presented.

The earliest foundations of physiological monitoring date back to as far as 1625, when Santorio published his methods for measuring body temperature with the spirit thermometer and for timing the pulse (heart) rate with a pendulum [Gar06a]. To monitor, is defined as "to observe, record, or detect an operation or condition with instruments that have no effect upon the operation or condition" [Web94]. In general, *health monitoring* means measurement and recording of vital signs, physiological signals and variables related to the human health, and comparing them to guideline values and following their development trends to detect a condition. *Personal health monitoring* focuses on the individual person and his health. Compared to traditional health monitoring, in personal health monitoring the person being monitored is usually partly operating and administering or otherwise somehow involved in the monitoring process.

Many terms are related to personal health monitoring. *Home health monitoring* [Oga03, Oga97], *health monitoring at home*, or *home health care* and similar terms are used to describe health monitoring activities at home. *Remote patient monitoring* [Bra05] is used to describe monitoring of vital signs in a setting other than a hospital. *Telehealth* [Koc06], *telecare*, *telemonitoring* (use of information technology to monitor patients at a distance) [Mey05], *mHealth* [Ist04] (mobile health), are newer terms related to the health services implemented using telecommunications technologies when distance separates the participants, including interactions with automated systems or information resources [Fag06], and in case of mHealth using wireless mobile devices. *Telemedicine* [Bre06, Lin99, BL02, Koc06] is an older term with narrower scope, connoting communication between two persons separated by distance using telecommunication technologies and often associated with video-conferencing between patient and the provider [Bre06]. The term *eHealth* (electronic health) [Eys01] also has a wider mean-

ing, it includes all forms of healthcare practices supported by electronic process and communication. *Wellness* [Dis04], and wellness technology, as a term associated with personal health devices is rather new, and it is used to describe devices and systems designed to improve general well-being and quality of life, often also supporting personal healthcare goals, but which also includes non-medical devices and devices not associated with healthcare directly. *Pervasive health care* [Var03, Var07] is defined by Varshney as "healthcare to anyone, anytime, and anywhere by removing locational, time and other restraints while increasing both the coverage and the quality of healthcare" [Var07]. The terms ubiquitous and proactive are often used in the same context as pervasive, but ubiquitous health care is a lesser used term. *Proactive health care* is sometimes used to describe intelligent systems and services targeted for automated health status monitoring.

The field of *informatics*, information systems and information processing, is related closely to health monitoring. Informatics systems are used to display, process and communicate the data obtained with the monitoring systems and other sources. *Biomedical informatics* emphasizes methods for handling the information, irrespective of the distance between the patient and the provider [Bre06]. *Consumer health informatics* [Eys00, Bre06] discipline analyzes consumers' need for information and produces applications which are designed to interact directly with the consumer.

To summarize the terminology, the word "tele" is used to signal the use of telecommunications technology to allow remote operations. "Mobile" or "m" when added, signals that the system allows mobility by using wireless telecommunication technologies. "Electronic" or "e" usually means applying information and communication technologies. Pervasive, ubiquitous, and proactive are used to describe systems which work mostly unnoticed on the background and add value to systems and services. "Medicine" is used when the systems are used in clinical context or health services between two persons. "Health", "care", and "healthcare" generally have a wider meaning than "medical"; they also include health promotion, disease prevention, and the use of automated systems and information resources. "Monitoring" is a more technical term, which describes that the system is able to measure and monitor activities and parameters, but does not by itself define what the collected data is used for.

In this Thesis, the focus is on personal health monitoring. Health monitoring is used instead of healthcare, as the main focus is on systems and sensors for measurement and monitoring of biosignals. The main application area is the home environment, although research on PC based patient monitoring has been performed, and some of the techniques are usable also in clinical environments. Motivation and reasoning behind the employment of personal health monitoring techniques at home is given in Section 4.1. The physiological signals, parameters and other techniques commonly used in personal health monitoring are presented in Section 4.2. Finally, applications for health monitoring at home are presented in Section 4.3.

4.1 Motivation for personal health monitoring at home

Historically, before 20th century and before the modern healthcare system evolved into the contemporary centralized hospitals with specialized doctors and nurses, the home was the place where a person was treated. Many services provided by the physician could also be obtained from experienced amateurs in the community. Rapid developments in applied sciences in the turn of the century lead to giant strides in technological development for diagnosis and treatment of diseases. New devices and instruments to assist doctors and nurses became available. For economic reasons, the centralization of healthcare services became essential. [Bro95]

These days, the healthcare industry in developed countries is confronting a number of challenges, including rapidly rising costs, a growing incidence of medical errors, inadequate staffing, and the lack of coverage in rural and underserved urban areas. Healthcare workers are under increased pressure to provide better services to more people using limited financial an human resources, i.e., to increase the efficiency of the services. These pressures, coupled with ready availability of personal hand-held devices, wireless connectivity, mobile systems, and related medical applications, are driving very rapid deployment of pervasive computing solutions in healthcare [Var03, Sta02]. The increased demand of healthcare has partly been due to an increased number of elderly and lifestyle changes leading to an increase in chronic diseases [Koc06, Bra06]. All this has lead to a demand for increased accessibility of care outside hospitals, i.e., moving health services into the patient's own homes [Koc06, Bra05]. In [Lee06] it is also suspected, that as generations of technology-savvy healthcare consumers enter retirement, they will ask for and even demand for sophisticated home healthcare and integrated health information databases.

Personal health monitoring at home is beneficial from two main viewpoints. Firstly, health monitoring at home frees space and resources in hospitals, as follow-up patients can be sent to home earlier and regular check-up visit can be performed less frequently. It is also a well established fact that long term patients in home care have a better change of recovery than long term hospital patients [Jon01]. A patient, and especially elderly people, can continue having active lives at their homes, which improves recovery and helps in maintaining the mental and physical condition. It is believed that this kind of monitoring is more cost-effective than traditional institutional care [Bra06]. Secondly, as is well known, the first hour following a trauma is of crucial importance in trauma care [Jon01]. Active health monitoring systems can assist in the detection of the trauma events so that help can be alerted faster. Moreover, a reliable long term log of monitoring information from the patient prior to a trauma helps the doctors to make a more reliable diagnosis, which improves the accuracy of treatment. The collected monitoring data can also be accessed by the doctors while the patient is being transported to the hospital, which will lead to quicker diagnosis and faster treatment, all of which leads to improved changes of recovery. Delayed or inaccurate treatment can lead to longer recovery times, more expensive medication and treatments, reduced working capabilities, and rehabilitation needs, all

of which is costly for the society in whole.

Personal health monitoring at home has also additional benefits for its user. As many basic measurements can be performed at home, less traveling to the healthcare providers is needed. For some handicapped patients or people living at remote rural locations this is a notable benefit. Travel has direct costs in form of gasoline or transportation tickets and also indirect costs in terms of travel time, delayed treatment, and lost productivity. In fact, travel has accounted for a significant proportion of the total cost in healthcare [Bre06]. However, it should also be noted that some patients may actually prefer frequent visits to the doctors because of social and related factors, and reduced travel would not be a benefit for them. Health monitoring at home may also, in some cases, yield more reliable results. One reason for this is the so-called "white coat syndrome" [Cel04], i.e., the patient feels stressed when the doctor/nurse is performing measurements in a clinic and can obtain more reliable results when performing the measurement relaxed at home. This benefit is arguable, as many recordings can also produce false results if performed in the wrong manner without the strict guidance of a medical professional. Nonetheless, by performing the measurements at home they can be performed more frequently, generating a log of long-term follow-up data, kind of which is difficult to obtain otherwise. The user can also perform a recording at the instant he feels ill, thus giving very precise data right at the instant of an seizure or an attack, which is very helpful for the diagnosis purposes.

4.2 Measurements used for personal health monitoring at home

Various signal and values can be used in personal health monitoring measurement at home. They can be categorized into two main groups: Physiological and related measurements, and ambient activity, location, and condition measurements. *Physiological and related measurements* are signals and parameters measured from the person or values calculated based on the measurement. *Ambient measurements* are signals and events which can be measured from the home environment and used to calculate indirect indicators of person's physical condition by means of activity monitoring and behavioral patters [Bar05]. Ambient measurements are becoming interesting as home appliances and devices become increasingly more computerized and networked which makes this kind of data readily available. Alternatively, home healthcare devices can be categorized in two by the way they are installed: wearable and environmental, as presented in [Kor03]. In this Thesis the division to physiological and ambient measurements is used.

Other means than measurements can also be applied in personal health monitoring at home. Person's health can be followed with health diaries, traditional type or electronic ones, where the user himself logs events and activities. The diaries can be analyzed by a supervisor and

for electronic diaries automated data analysis and mining tools can be applied to generate automated alerts and notifications based on the data. The traditional telephone interview performed by a medical practitioner also remains an effective surveillance tool. Video-call and teleconferencing techniques enable visual interaction which enhances the possibilities of this manual surveillance [BL02]. Advances in IP-telephony and instant messaging technologies have made inexpensive video-calls available for all home computer users having wideband internet access.

4.2.1 Physiological measurements

Physiological measurements are the main source of information for personal health monitoring. From the engineering or sensor designer viewpoint two types of physiological measurements can be identified: Distinct values and variables, and the recordings of biosignals. One can correctly argue that the second group is only raw data for which the calculation of values and parameters is yet to be done. In most cases this is true, the signal analysis being so complex that it is not possible or wise to perform in the measurement device. But it is also possible that we do not know beforehand what analysis method to use if the disease/condition to be identified is not known, or there isn't yet an automated analysis method for the signal. In the latter case the analysis is usually done by the doctor visually, for which the graph generated from the raw data is needed. It is also useful to have raw signals in long-time monitoring when we do not know right from the start what to look for.

Distinct values and variables usually tell us immediately something about the person. The weight and blood pressure are easy examples of values that if out of predefined range give cause for immediate attention. Their long-term monitoring gives us trends which indicate the progress of person's health. Other values and variables, like the amount of sleep per night or heart rate (or heart rate variability) are values which tell us the current state, but which may have large daily variations and no diagnosis can be given based on one measurement. A longer monitoring over a few weeks gives us enough data to make some diagnosis of the person's health and well being.

For the recorded physiological signals a similar division exits. There are events and diseases for which we have known patterns of known changes in the signals. If we detect these known abnormalities in the recorded signal, then we can make immediate evaluation of the users health or diagnosis in case of the doctor. Also, if we have prior recorded "normal", i.e., healthy, signal of the person, we can detect abnormal events with increased accuracy. In long-term monitoring we can observe how the state of the patient develops. If we record some less known biosignal, like the ballistocardiogram, we may not be able to evaluate the person's health based on one recording, but in long-term monitoring we can detect abnormal changes and use them to generate alerts or evaluate the development of his condition. Table 4.1 lists the physiological parameters

Signal/Technique	**Source**	**Range of parameter**	**Frequency range (Hz)**	**Primary sensor types**
Ballistocardiography (BCG)	Heart (mechanical)	0-7 mg	DC-40	Force, displacement
Blood pressure (BP), non-invasive	Arm (Blood vessel)	25-400 mm Hg	DC-60	Cuff, auscultation
Electrocardiography (ECG)	Heart (electrical)	0.5-4 mV	0.01-250	Skin electrodes
Electroencephalography (EEG)	Brain	5-5000 μV	DC-150	Surface electrodes
Electromyography (EMG)	Muscle	0.1-5 mV	DC-10000	Surface electrodes
Eye potentials (EOG, ERG)	Eye dipole field, eye retina	0-3500 μV	DC-50	Contact electrodes
Pulse oximetry (SpO_2), noninvasive	Blood	30-100 %	DC-2	Infrared
Respiratory rate	Lungs	2-50 breaths per minute	0.1-10	Strain gage on chest, impedance, BCG
Temperature of body	Body	32-40 °C	DC-0.1	Thermistor, thermocouple

Table 4.1: Physiological signals used in personal health monitoring, adapted from [JWC98, Bro95].

traditionally used in personal health monitoring applications at home.

In addition to physiological signals, other closely related signals and parameters can also be measured. Weighing scales are used to monitor the body weight, balance boards can be used to detect abnormal balance, and accelerometers and other wearable sensors [Axi05] to detect abnormal gait, agitation, motor activity, and special incidents like falling. Some sensors, like a bed mounted BCG sensor, can be used to measure biosignals (ballistocardiogram and respiratory & movement activity signals derived from it), average heart rate or respiration values, to recognize sleep/awakeness and calculate amount of daily sleeps, or used as proximity/location sensors to detect that a person is in the bed. This shows how a sensory device cannot always be classified just by its method of measurement, and also how a same sensory unit can have very different data communication needs based on how and to what purpose it is applied.

Sensor type	Detect or measure
Infra red sensor	Presence, mobility
Infra red camera	Surface temperature and temperature differences of objects, use of heat sources such as stove/oven
Light sensor	Night/day and switching on/off of lights
Temperature sensor	Ambient temperature and detection of use of shower, water flow, use of electrical appliances
Mechanical switches	Alarm buttons, mobility measurements by detecting door openings and closings, identify use of major appliances as the refrigerator
Central power sensor	Daily power consumption, detect daily routines
Appliance power sensor	Power consumption, measured from the power cord directly or indirectly, measure use of electrical appliance
Pressure sensor	Presence, mobility, ballistocardiogram
Sound sensor	Presence, activities, based on automatic recognition
Capacitive sensors	Presence, mobility
Microwave radar	Presence, movement

Table 4.2: Sensors for ambient measurements used for personal health monitoring, adapted from [Cel95][P8].

4.2.2 Ambient measurements

Non-physiological signals and values can be monitored to support personal health monitoring and health status assessments. Changes in the health status change the patterns of daily living and use of household resources [Cel95, Cel96, Kor03, Bar05]. These "activities of daily living" (ADL) signals give us indirect information about the users living habits and activities, functional health status, and possible functional deterioration [Cel95]. Table 4.2 presents sensors useful in ambient monitoring.

In addition, modern computerized and networked devices may provide state/usage information which can be used in activity monitoring. Home security & automation systems also provide ambient activity data useful in healthcare monitoring, and interaction with these systems is beneficial.

Daily activity information can be enhanced by using personal wearable or portable devices, which may or may not also provide physiological and related measurements. Accelerometers are used to implement pedometers to measure amount of daily movement. Personal mobile phones and GPS sensors can be used to track the person and to monitor the daily activity patters outside the home. [Axi05] [P8]

4.3 Personal health monitoring at home

Personal health monitoring is done in a heterogeneous environment. Traditional medical devices are used in healthcare facilities, described in the next Chapter, but personal health monitoring is at most time performed at home. Healthcare facilities have strict control over used devices, but this level of control and supervision is difficult to impose at home. For home care devices three areas should be considered when equipment is used outside hospitals [Bro95, Chapter 89]: (1) Device must provide a positive clinical outcome, (2) device must be safe and easy to use, and (3) device must be user-friendly enough so that it will be used. [Bro95, Chapter 89] points out, that home care devices should be even safer and more reliable than devices used in hospitals, due to the absence of trained medical staff which normally keep an eye on the patients and equipment monitoring them.

As presented in [Bra05], three distinct types of remote health monitoring system can be identified:

- Chronic and other disease management systems: Low to medium portability, periodic measurements
- Acute monitoring: Medium portability, continuous measurements
- Ambulatory monitoring: Mobile/wearable devices, continuous measurements

These systems have different requirements. *Chronic and other disease management* is the simplest form of monitoring [Bra05]. It is based on periodic measurements of parameters such as blood pressure, weight, respiration, which can often be measured by the patients themselves [Bra05,Cla05]. These measurements are then monitored for predefined change indicatives. The purpose of chronic monitoring is prevention of hospitalizations, and early responses such as admission to hospital or changes in medication [Bra05].

Acute monitoring is used to monitor a patient in situations where there might be a rapid deterioration in the condition which must receive prompt medical intervention [Bra05]. Data transmission in acute monitoring situations needs to be near continuous. Acute monitoring serves the purpose of providing hospital-style monitoring at home, with the associated prompt emergency intervention [Bra05].

Ambulatory monitoring is used to identify a critical and sudden change in the vital sign in an otherwise healthy person [Bra05]. It requires continuous monitoring and algorithms for detection of events. Ambulatory monitoring should also allow mobility for the patient and is likely to be based on wireless communication [Bra05]. Ambulatory monitoring is used to detect and respond to an emergency situation quickly. Acute and ambulatory monitoring requires a 24-hour response center to be effective [Bra06].

Currently, the primary use of health monitoring systems is for management rather than diagnosis. Remote monitoring systems are typically used with patients who have already been

diagnosed with a chronic disease or condition. Another important feature of most remote monitoring systems is that the measurement of the parameter and the transmission of the data are typically separate events [Bre06]. As has been observed in [Cla05], patients in chronic disease management may deteriorate rapidly and require acute monitoring. A system supporting both types of monitoring offers possibilities for improvements in the management of patients. [Cla05] also shows than an ISO/IEEE 11073 standard based system can be used for both chronic disease management and acute monitoring.

As presented by Korhonen et al. [Kor03], two main models of health status monitoring for out-of-hospital conditions exist: *Wellness and disease management* and *Independent living and remote monitoring*. In wellness and disease management the user at home has an active role in the treatment, and the caregiver often has a secondary role. The systems and devices are usually owned and managed by the user who also monitors the measurement results, and minimal linking to an electronic patient record is needed. A single measurement value is not very critical and measurement data do not usually have to be transferred to a server in real time. The caregiver is often the primary user of the measured data in the independent living and remote monitoring concept, and the user may not have any interest in the data. The devices should be technically very easy to operate, as the users may have limited technical skills and may be reluctant to accept devices in their homes. If the independent living depends strongly on the monitoring systems, the user may be highly motivated to accept even inconvenient devices. Yet, the devices should be designed as unobtrusive as possible to enable the user to continue normal living. As these systems are often used to report health hazards by means of alarms, the reliability of the systems and data communication is important. [Kor03]

The research in personal health monitoring has focused much on the sensory techniques and systems, rather than services, as also in the work of this Thesis. Ogawa, Tamura, and Togawa et al. have done research in how to measure biosignals with ordinary home furniture and equipment [Tog89, Tam95, Oga97, Tam98, Kaw99, Oga00, Oga01]. Celler et al. attempted to evaluate health status by using indirect ambient measurements [Cel95, Cel96]. Clarke and Bratan et al. have studied remote patient monitoring services and systems, and use of ISO/IEEE 11073 for these purposes [Cla04, Bra05, Cla05, Bra06, Cla07]. Dishman has presented Intel's efforts to support the aging population [Dis04]. Korhonen et al. have presented how a simple inexpensive wrist watch type of device can be used to implement a service for elderly home monitoring [Kor03]. Varshney has studied how pervasive computing will change healthcare systems and what is the current state in pervasive health care [Var03, Var07]. Milenkovic, Otto, and Jovanov have studied wireless sensor networks for home health monitoring [Mil06, Ott06]. Aziz et al. predict that body sensor networks (BSN) will be included in future pervasive health care systems, and in implantable and wearable systems. Yao and Warren have applied the ISO/IEEE 11073 standard to wearable home health monitoring systems [Yao05, War05]. Mobile health issues have been presented in [Ist04].

Summary

This Chapter introduced the concept of personal health monitoring, the main application area of this Thesis and all the publications [P1]-[P8]. Several terms are used to describe personal health monitoring systems used at home and related research. Although most of them have slightly different emphasis, considerable overlapping in the terms exists. Personal health monitoring at home can be performed with physiological and related measurement or by monitoring the ambient activities and daily behavior. Three distinct types of monitoring can be identified, chronic disease management [P2][P6][P8], acute monitoring [P2][P6][P8], and ambulatory monitoring [P4]. Moreover, two distinct models of home health monitoring services can be identified: the wellness and disease management and the independent living and remote monitoring model.

In the next Chapter, the more generic monitoring needs and uses of computer-based systems in healthcare facilities are presented.

Chapter 5

Monitoring in healthcare facilities

Healthcare facilities are the prime users of medical devices. Their monitoring systems have evolved from systems used for critical care monitoring. This Chapter introduces the use of computers and interfaces in healthcare facilities and the critical care monitoring environment.

Healthcare facilities include hospitals, clinics, dental offices, out-patient surgery centers, birthing centers, and nursing homes, as listed by US Occupational Safety & Health Administration on their web-pages (www.osha.gov). Monitoring in healthcare facilities is the traditional form of medical monitoring which evolved during the 20th century from the use of simple mechanical aids to current use of complex electrical medical instruments and devices [Bro95]. A new discipline of engineering - clinical engineering - has evolved to support development and maintenance of clinical equipment. Hospitals and other larger healthcare facilities have separate clinical engineering departments, which manage technologies and related risks, assess new technologies, assist in facility design and project management, and organize training related to medical equipment [Bro95, Chapter 165].

Use of devices for patient care in healthcare facilities is strictly regulated and controlled. Devices used must fill requirements set for medical devices and misuse of unapproved devices can lead to legal and financial liabilities. The devices are operated by trained personnel. Each performed operation/measurement and its results are usually stored in patient health records.

Healthcare facilities have larger and more complex measurement systems requiring trained staff for operation and performing the analysis of the results, including several different medical imaging systems, and biomedical measurements requiring off-line laboratory analysis for example. The role of informatics has grown in the last 30 years, as the volume of data produced by the measurements has grown and computerized analysis methods have developed. This has lead to the growth of medical informatics, a discipline closely related to clinical monitoring [Bro95, Section XVII]. Medical informatics includes hospital information systems, computer-based patient records, clinical imaging, health care computer networks, and decision support and monitoring systems. Computers and telecommunications are an inseparable part of medical informatics

systems [Bro95]. Thus medical informatics systems are a key discipline when new computer interface and communication technologies are introduced to healthcare facilities. Especially hospital information systems (HIS), which integrate all patient related information systems in the hospital, widely use standard PC technologies. The tasks of a HIS include patient databases, acquisition of clinical data, patient admission, transfer, and discharge functions, patient evaluation applications, and patient management. [Bro95, Chapter 176] These systems are computer based and use common technologies and digital interfaces. However, the focus in this Thesis is on monitoring and measurement devices, i.e., devices for acquisition of clinical and health related data, which is the focus of the next two Sections.

In the next Section 5.1, the use of computers and digital interfaces in healthcare facilities such as general practices and health care centers/clinics is briefly presented. Computerized health monitoring has evolved from the needs of the critical care. The history, applications, demands and special issues related to critical care monitoring are discussed in Section 5.2.

5.1 Computers and interfaces in healthcare facilities

PC technology is nowadays common at healthcare facilities due to wide spread use of medical informatics and handheld computers [Bro95, Tor03]. Medical practitioners use IP-based (internet/intranet) information databases and clinical guidelines for finding additional and decision support information about diseases, store data in electronic patient records, and manage patient appointments and billing electronically. They use standalone medical devices, like the blood pressure monitor, to perform measurements on the patient. Medical imaging applications, including computed tomography (CT), magnetic resonance imaging (MRI), and digital subtraction angiography, require calculations which are computationally intensive and can only be performed with computers [Wie06]. In [Bro95, Chapter 179] Sengupta identifies four different general application areas for computers in medical centers:

- Ancillary instruments and systems, for example in laboratory, pharmacy, radiology, pathology
- Clinical and scholarly purposes including electronic medical records, imaging, searching for medical references
- Administrative contexts like billing, patient management, transport, payroll
- Use in basic research functions like modeling and genetics

As can be seen, the computer systems have become an essential part of the healthcare delivery. A PC is invariably present when measurements are being performed in medical centers as the results of the measurement are stored in electronic patient records using them. As stated

in [Bro95, Chapter 176], automated acquisition of data is preferable to manual entry, as it is more accurate, reliable and resource intensive. It is this which promotes the use of common standard digital interfaces in medical monitoring and measurement devices. Less time is lost in everyday use and better accuracy and reliability obtained, if the devices can transfer the measured data automatically to the electronic records. The automated acquisition also enables automatic storage of measurement time, measurement device type, and additional context information. This kind of information can be useful in cases where a faulty device is detected and affected measurements would be traced.

Medical information systems, electronics medical records, and clinical imaging systems producing vast amounts of data have made computer networks common in the medical centers of the developed countries [Bro95, Chapter 179]. Ethernet based LAN has been the dominant standard with IP-based services. Wired LAN and related services have been easily extended to wireless using WLAN techniques when mobility has been desired [Häm08]. The fact that Ethernet LAN-connection or WLAN is available in most places has reduced the need for other local area networking techniques. Devices and systems which are capable of communicating with the HIS directly usually do so by interfacing to the hospital LAN or WLAN. Direct access to the HIS requires certain safety measures and capability to communicate with the HIS using its protocol. This is why most instrument type medical devices can not connect to HIS directly. Instead, they rely on personal area networking or cable based interface technologies to connect to a near-by computerized system which functions as a gateway to the HIS.

A bedside patient monitor is a commonly used tool in hospitals to collect patient related measurement information from surrounding measurement devices. A *patient monitoring system* [P2] is a system which measures parameters from the patient, uses different algorithms to calculate some physical values based on the measurements, and finally displays the values to the monitor user. A monitor can work as a stand-alone device or as part of a network of monitors where measurements from one monitor can be viewed on other monitor or in a centralized monitoring room [P4]. Optimal interface selection for any device requires good knowledge of the data transfer needs, as was described previously in this Thesis. In addition, when designing a general purpose patient monitoring systems and choosing interface technologies for it, the requirements to take into account include [Var06]:

- Monitoring and transmission of both routine and emergency vital signs (data transfer needs in emergency situations may be considerably more higher and more frequent)
- Reliability of message delivery
- Message delivery in reasonable time
- Power conservation
- Coverage for both fixed and mobile patients

- Support for diverse and battery-limited devices on and around the patient
- Scalability
- Manageable cognitive load for health care professionals
- Confidentiality and privacy

Some of these requirements relate more to critical care monitoring discussed in the next Section. However, it is sometimes difficult to distinguish between non-critical and critical patient monitoring systems, as non-critical care systems may also occasionally face sudden changes of health and emergency situations.

5.2 Critical care monitoring

Monitoring and measurements in healthcare facilities can be categorized into two main classes: (1) critical care monitoring and (2) other forms of monitoring and measurements. In this Section the focus is on critical care monitoring. The MedTerms Dictionary at MedicineNet.com defines critical care (intensive care) as "The specialized care of patients whose conditions are life-threatening and who require comprehensive care and constant monitoring, usually in intensive care units."

Critical care patient monitoring involves continuous measurement of patient parameters such as heart rate and rhythm, respiratory rate, blood pressure, blood-oxygen saturation, and many other parameters [Gar06a]. Care of a critically ill patient requires prompt and accurate decision making so that life-protecting and life-saving therapy can be appropriately applied. This has lead to widespread use of intensive care units (ICUs). These units use computers almost universally for the following purposes [Gar06a]:

- Acquisition of physiological data frequently or continuously
- Communicating information from data-producing systems to remote locations
- Storing, organizing, and reporting data
- Integration and correlation of data from multiple sources
- Providing clinical alerts and advisories based on multiple data sources
- Decision making tool for health professionals to assist in planning the care
- Measure the severity of illness for patient classification purposes
- Analyzing the clinical and cost effectiveness of the care

First use of computers in critical care units were described in the mid 1960s [Cle04, Gar06a]. Computers where introduced to critical care monitoring for five reasons: (1) to increase the availability and accuracy of data, (2) to compute derived variables that could not be measured directly, (3) to increase patient-care efficacy, (4) to allow display of the time trend of patient data, and (5) to assist in computer-aided decision-making [Gar06a]. As the microchips and processors became available in the 1970s, computerized devices replaced old mechanical devices with more sophisticated and smaller electronics ones [Cle04], and also made it possible to implement devices using digital signal processing instead of simple analog signal processing [Gar06a]. These days, there are virtually no bedside monitors or ventilators marketed which do not use at least one microcomputer. In recent years, the bed side monitor has become the focal point for data entry and presentation. Each of these monitors has its own proprietary communications protocol and data acquisition scheme. The bedside monitors can acquire data from other devices and clinical laboratories. This has lead to bedside monitors that function like mini patient data management systems. Specialized ICU information systems have been developed by the bedside monitor manufacturers to manage patients in the critical care, and these systems in turn may be interfaced with the hospitals information systems. [Gar06a] This development has not promoted the use of standard interfaces for the ICU devices; instead it has enabled the existence of closed, market limiting, proprietary interface technologies.

Although microcomputer based systems are widely used in critical care monitoring, it is still the area where the application of common PC systems and components, especially software components, is limited. The use of common standard interfaces has been quite rare, and proprietary interfaces are widely used, as noted by Gardner and Shabot [Gar06a]. However, standard interfaces are coming and increasingly more required for the ICUs also. Hospital information systems require standard interfaces between systems to exchange information. Automated acquisition of data is preferable to manual entry, as it is more accurate, reliable and resource intensive, which is important in the ICU where time is scarce [Bro95, Chapter 176]. It is currently common for a nurse or a therapist to read a computer display from one device and then enter data through a workstation to another device [Gar06a]. Furthermore, having systems from different manufacturers which do not interoperate leads to having overlapping infrastructure, displays and monitors, in the ICU, and valuable information may be left unnoticed due to cognitive information overflow. Automated data entry and information system interoperability in the ICUs is important because decisions have to be made in very timely manner, and delays in data entry may force decision making without all possible data [Bro95, Chapter 176]. The ISO/IEEE 11073 Point-of-Care is currently the only viable standard for medical devices in the ICU. Matrinez et al. have demonstrated how it can be used to provide end-to-end standard based patient monitoring solution applicable in the ICU [Mar08].

Even though critical care monitoring is not in the scope of this Thesis, monitoring in the ICU is interesting because techniques that were used only in the ICU just a few years ago are now

routinely used on general hospital units and in some situations by patients at home [Gar06a]. New trends like virtual medical care have appeared which introduce innovative telemedicine systems such as the use of cameras to assist in real time remote monitoring [Cle04]. These kinds of innovative systems may introduce rather standards PC technologies to even critical care surroundings.

Summary

A brief introduction and summary on the use of computers and interfaces in healthcare facilities was given. PC technology is widely used in modern healthcare systems. Various PC-based systems are used not only for instruments and monitoring [P2][P5][P4], but also for administration, research, and health informatics systems. Bedside monitoring [P2] and critical care monitoring are the main monitoring uses in healthcare facilities. For these purposes, medical isolation of cable interfaces [P3] is usually required and for some applications wireless technologies can be applied [P4][P5][P6]. Continuing growth in medical device networking and wireless systems as described in [P4] expands the needs for interoperability standards also in healthcare facilities.

The following Chapter summarizes the results obtained in this Thesis.

Chapter 6

Research results

The results of this Thesis are mainly presented in the included publications [P1] - [P8] and are summarized in this Chapter. In addition, novel ideas on digital interface implementation architectures have been presented in Chapter 2 and some new results were obtained from the literature study of Chapter 3 concerning medical device standardization. These are included in the summaries. The results are divided in four categories. Section 6.1 summarizes the results of digital interface implementation work. Section 6.2 summarizes the interface technology evaluation studies. Section 6.3 presents the results obtained from the prototype implementations. Lastly, Section 6.4 presents the key results of the literature study of Chapter 3.

6.1 Digital interface implementation

Digital interfaces have evolved in protocol complexity. Interface standards define more functionality than previously, when an interface was merely for bit and byte exchange. The actual payload data transfer process is straightforward but the complexity is related to the interface configuration [P1][P7]. More tasks are performed in software, i.e., in interface drivers or protocol stacks, than previously. In a single processor embedded device, the software components of the interface execute in the same processor and memory space than the application, and use shared resources. The interface can affect the application functionality and vice versa, and the application programmer has to be aware of these possible dependencies and conflicts [P7]. Certain basic hardware architectural design alternatives for implementing standardized digital interfaces in computer and embedded systems have been found to exist, as presented in Section 2.5.2 and [P1]. These architectures can be applied to both wired and wireless interfaces.

Regardless of the interface technology, the maximum obtainable data rate is never the advertised signaling rate (nominal bit rate) of the physical interface. Protocol overhead is the main reason, but not the only one. Larger packet sizes reduce protocol overhead but increase latency. Maximum packet size in interfaces in which retransmissions are likely (such as the wireless inter-

faces) is always a trade-off because larger (longer in time) packets are more likely to be disturbed and retransmitted. Because of this data link unreliability, a wireless interface always requires a higher nominal bit rate than a cable based one used for the same application. An interesting result noted in the Bluetooth work of [P4] was that even detailed simulations of operation may fail in the estimation of practical data throughput and produce completely opposite results to real world measurements in line-of-sight conditions (when estimating performance of different data transfer methods). Moreover, the performance of Bluetooth wireless interface was noted to be very unpredictable in non line-of-sight cases. More generally, the design of a wireless interface technology is always a trade-off between different parameters, including speed, range, latency, robustness, and energy consumption. The radio protocol is the key component when truly low power design is desired. Low-power radio technologies require low radio use duty-cycles to achieve truly low power consumption. Most current radio interface technologies, including Bluetooth and Zigbee, consume approximately as much power in reception as in transmission, so the radios have to be fully switched off most of the time to reduce power consumption. As shown in [P4], the use of very small duty-cycles, i.e., long sleep intervals, in turn lead to very slow communication setup and long response times between the devices.

Implementing hard real-time systems or systems requiring very low-latency on standard Windows PCs is difficult because the operating system produces unpredictable delays [P1][P2]. It was also observed in trials with Windows 98 OS, that the complex and layered Windows driver model (WDM) causes notable processor overhead when performing frequent operations [P1]. The processor overhead can be reduced by performing larger data transfers at the API level, which increases latency [P1]. The WDM is still used in the current Windows versions, although a new Windows Driver Foundation was introduced with Windows Vista to simplify the device driver API seen by the driver developer. The complexity of the WDM architecture makes testing of the device driver code difficult [Bal06] and device driver errors are the main cause of modern Windows system crashes [Gan06]. In general, the PC was found well suited for data acquisition systems with soft real-time requirements, including health monitoring at home [P2]. It has interfaces and network connectivity which cover wide range of applications, and it offers an excellent platform for complex signal processing, automated analysis, and graphical user interface purposes due to its considerable computing power [P2][P5]. Huge advances in the computing power of the PC have also given new life to abandoned measurement methods like the BCG, which is again becoming interesting for the researchers [P5].

It was observed, that for new technologies, the first devices and implementations do not often fully fill the standard specification. This was observed both in USB host driver implementations [P1][P2], and in the IEEE 802.15.4/Zigbee work [P7]. In both cases the original system design had to be altered, and the performance promised in the standard specification could not be obtained.

6.2 Interface technology feasibility studies

Table 6.1 summarizes the differences of cable based (electrical) and wireless (RF) interfaces based on the work performed in this Thesis. Table 6.2 summarizes the key differences of the three main interface technologies of interest to this Thesis, Bluetooth, USB, and Zigbee. Other, more detailed results, are presented next.

USB was found to be robust and well suitable for medical applications [P1][P2], and it will likely become popular also in other measurement and data acquisition applications [P3] [Gre09]. The complexity of USB is hidden in the host device, and peripheral devices require lighter implementation [P1]. In [P3] it is shown that USB 1.1 can be electrically isolated, although the implemented solution is not universally applicable as such. Paper [P1] deems the capacitive isolation as unusable. Although this is the case for USB data signal isolation, it still has potential for medical isolation as shown by [Pii07] (using RF modulation over capacitor) and TI (integrated high-speed digital isolator products ISO721/722). By using extremely small capacitances, of order of pF, the capacitance over the isolation barrier can be reduced to the level of traditional isolation circuitry [Pii07]. However, only very high frequencies, of the order of GHz and above, can pass through these small capacitances. Lower signal frequencies require the use of modulation technologies. Alternatively, very high frequency digital signals can be isolated using a series capacitor and latch circuit, as presented in [Kug09]. Both of these methods are unidirectional by nature, e.g., full-duplex operation requires separate circuits for both directions and circuit for detection of transmission direction. The use of these kinds of circuits is similar to optoisolators, but due to the faster operation of capacitive isolators, they are an attractive alternative in high-speed applications. In medical devices, the voltage and clearance regulations, which are easier to meet with transformers, limit their use. In case of USB, the 1.0 to 2.0 standards (low, full, and high-speed communication) are all based on 12 MHz bus clock speed, with the high-speed USB using faster microframes within the basic frames. USB 3.0 has half-duplex data signals operating at 4.8 GHz bus speed that could be isolated using capacitive isolation. However, USB 3.0 connector also includes signals for the USB 1.0 to 2.0 versions to retain backward compatibility, and therefore the same limitations apply for an overall solution as do for the older USB versions. To isolate USB data signals using capacitive isolation a fast, extremely low latency modulator with extremely fast detection of transmission direction, would be required. As has been shown by the recently released Analog Devices ADuM 4160, this kind of modulation and direction detection can be implemented with transformers currently at only low and full speed (USB 1.0-1.1). USB has been chosen as a transport method for the ISO/IEEE 11073 standard, which thanks to promotion by the Continua Alliance and the IHE, is expected to become widely adopted.

Wireless technologies can assist and improve care in hospitals [P4]. Bluetooth and other PAN technologies may not be the best technologies for hospital wide networks, but they have

	Cable-based (electrical)	**Wireless (RF)**
Power	The interface can supply power.	Requires battery or external power source.
Maximum range	Defined by the cable length.	Can not be defined exactly, the performance will deteriorate with distance.
Security	Physical access or very complicated techniques required to eavesdrop.	Everyone within the transmission range can pick up the signal.
Electrical safety	Electrical connection outside the device casing requires electrical isolation and protection from over-currents.	No physical connector requires, the device casing can be made completely non-electrical (requires batteries).
EMI caused	Very small when using proper cables.	Depends on transmission power.
Susceptibility to interference	Cables can pick up interference if not properly shielded and isolated. Data communication not easily effected.	Interference decreases communication performance. Small and sensitive portable sensory device (analog inputs) may pick up interference from its own transmissions.
Device mobility	Cable restricts movement. The communication is not effected by movement.	No movement restrictions, the connection will deteriorate as range grows. Movement alters connection quality unpredictably and in some cases may change the underlying network topology which can effect also other devices in the premises.
Quality of service	Can be implemented in a way that bandwidth is guaranteed by allocating the device a fixed portion of the bandwidth.	Unpredictable medium, data can be prioritized, and time reserved for retransmissions, but bandwidth can not be guaranteed.
Bandwidth utilization	Good, only packet headers decrease efficiency. Increased data payload size improves bandwidth utilization but increases latency.	Packet headers reduce efficiency, and smaller packet sizes/payload are preferable to reduce possibility of interference and retransmissions. Space has to be allocated for retransmission and higher bandwidth utilization reduces reliability.
Usability	Number of attached devices increases the number of cables and degrades usability.	Number of attached devices does not directly effect usability.
Topology management	Cables reflect the connections between the devices.	Visual identification and management of connections requires additional components or systems.
Global use	Not restricted (electrical safety must be approved).	Use of RF bands is regulated. Only a few globally available radio bands exist.

Table 6.1: Comparison of the use of cable based and wireless technologies in medical devices from the interface and data communication viewpoint.

	Bluetooth	**Zibgee**	**USB**
Type	Wireless, 2.4 GHz	Wireless, 2.4 GHz and 868/915 MHz	Cable based (wireless version in development)
Data rates	Moderate to high	Low	High to very high
Signaling rate (nominal)	Up to 3 Mbps using BT radio, higher using alternative MACs (Chapter 2)	Up to 0.25 Mbps (2.4 GHz)	Up to 4800 Mbps (Chapter 2)
Data link robustness	Good, frequency hopping	Moderate, narrow band	Very good, shielded cable
Networking	Small scale piconets, point-to-multipoint	Supports forming bigger networks and different topologies	Tiered star topology, can be extended with hubs, but all communication is between host and a end-device
Power consumption	Moderate to low	Low to very low	Power can be supplied from the bus

Table 6.2: Comparison of Bluetooth, Zigbee and USB interfaces.

several uses in local monitoring applications, including both measurement signal communication and monitoring user interfaces [P4]. Bluetooth and Zigbee are the most dominant wireless PAN technologies, and operate in the radio band which allows virtually world wide coverage [P5]. Zigbee offers better networking functionality than Bluetooth and slightly lower power consumption due to Bluetooth's frequency hopping. Frequency hopping used by Bluetooth makes it more robust and less susceptible to interference [P4]. New low-power Bluetooth version reduces Zigbee's benefits in low-power applications. Bluetooth has also been chosen as a transport method for the ISO/IEEE 11073 standard. Zigbee has Health care application profile which supports ISO/IEEE 11073 device specializations, and it is likely that it too will be included as an ISO/IEEE 11073 transport method in the future.

[P5] shows that in 433 MHz to 5.9 GHz range only the 2.4 and 5.8 GHz ISM bands offer sufficient resources for a medium data rate radio link. Medium data rate in this case was defined to be in the order of 60 kbps. Ultra wide band technologies are problematic for regulators and may cause EMI when transmitting due to the large energy related to the transmission. In [P5] we identified six issues which should be considered when selecting a wireless technology:

- Target environment - Is the use of RF bands limited, are some bands more preferable because of regulatory issues.
- The amount of integration - RF module requiring only external antenna vs. plain chip which requires board design and additional electronics.
- Physical chip/module parameters - Power consumption, operating voltage, interface to the rest of the system.
- How much does the chip/module handle the protocol - Implementing the protocol (e.g., Bluetooth higher layers) needs resources.
- Software support from the manufacturer.
- Price, availability, alternative components if the technology is discontinued.

In [P4] it was found, that PAN technologies such as Bluetooth have a clear usage area even in hospitals equipped with LAN and WLAN. The main application area was noted to be the point-of-care data communication, transfer of measurement data from the measurement instruments and other bedside devices to a bed side coordinator device, i.e., patient monitor or similar device. Bluetooth based wireless user interfaces were also proposed. Short-range wireless interfaces can also be used as a wireless patient identifier tag, and for staff location discovery. Specific RFID technologies have later been developed for these kinds of "tagging" purposes.

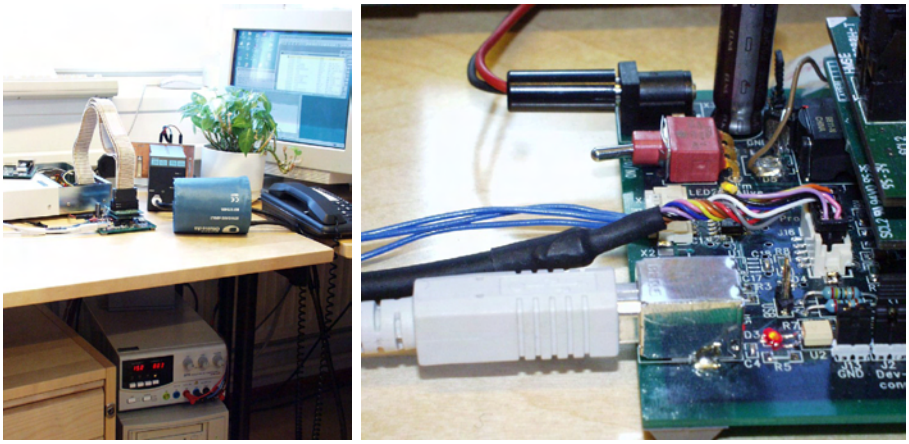


Figure 6.1: Prototype device used to interface medical measurement modules via USB to a standard PC [P1][P2].

6.3 Implemented monitoring prototypes

Three prototypes which use the PC have been implemented. The first system [P2][P1] used USB to interface medical measurement devices (modules of a clinical patient monitor) to a standard PC. The second used an IEEE 802.15.4 interface (later extended to Zigbee) to interface a medical biosignal monitoring device, a ballistocardiographic chair, to the PC [P6]. Its performance was compared to the previous bio-monitor based implementation presented in [Jun04,Koi04b,Jun05]. The third implementation was a Zigbee based wireless home sensor network, which used a PC primarily as a gateway and storage unit. In addition, a wired analog amplifier implementation of the ballistocardiographic chair was constructed, which was connected to a commercial bio-monitor [P6] [Jun04, Koi04b, Jun05].

The structure of a modern patient monitor used in hospitals resembles a PC-platform [P2]. USB interface was found to be the most suitable cable based standard PC interface for data-acquisition and patient monitoring purposes [P2]. In [P2] it is shown that measurement modules from a commercial patient monitor can be attached to a standard PC platform using the USB. The implemented USB interface device (Figure 6.1) would have enabled commercial development of a PC based health monitoring system for non-critical monitoring using the existing commercial measurement module hardware.

The ballistocardiographic chair [P6], and especially its first wired version (Figure 6.2-a) originally presented and evaluated in [Jun04, Koi04a, Koi04b], has been accepted for hospital use in Tampere University Hospital, and has been used to record over 150 subjects in a two year clinical trial. The analysis of recorded data has produced several international publications including [Koi04a, Koi04b] and 12 papers by Dr. Akhbardeh including three journal articles [Akh07d, Akh07b, Akh07a], and two Theses [Akh07c, Bar08]. More publications and a Thesis are

(a) (b) (c)

Figure 6.2: Prototype BCG chair. (a) The wired setup used in the Clinical trials. (b) The wireless BCG chair with additional armrest electrodes. The electronics are hidden under the chair. (c) The bio-amplifier unit of the wireless BCG chair. [P6]

expected. The successful recordings and the vast amount of data collected with the device and measurement system designed and implemented as a part of this Thesis are a notable result of which true value will be defined by the future research of others.

It has been shown that ballistocardiogram (BCG) can be recorded using an unobtrusive and even unnoticeable method with rather inexpensive and simple chair type of device suitable for everyday home health monitoring (Figure 6.2-b) [P6]. A normal PC can be used to record, present, and even analyze the data. The BCG signal recorded with the device is visually close to ultra-low frequency acceleration BCG. Armrest electrodes mounted on chairs armrests can be used to record very clear ECG signal [P6]. We have shown that a large multi-device medical monitoring setup using analog cables and interfaces can be replaced with a small wireless low-power unit (Figure 6.2-c), and attached wirelessly to a PC, without compromising the signal quality [P6] [Koi04b].

In [P8] the sensor and radio implementations of a wireless sensor device were separated by defining a common interface. The common interface defined both the physical (HW) interface and the API (SW interface). The communication protocol (interface drivers) were implemented and provided to the sensor developer by the implementer of the radio/microcontroller platform. This shows how it is possible to separate the sensor application and the interface running in the same processor platform by providing the sensor application with a predetermined environment which provides the necessary APIs for obtaining the data and communicating it onwards.

Medical device software will become under regulation soon and measures enabling independently approved modifications to the so-called application software are needed, i.e., so that only the modified application code needs to be approved and the software framework including the communication and device drivers can be excluded from the re-approval process.

[P8] shows a method of collecting health related data from commercial devices by adding a small radio unit which interfaces the sensor device to a wireless sensor network. The developed system required that the data measured by the device can be extracted from the device in analog or digital form. A simple and lightweight sensor network based on the Zigbee network-layer has been implemented for this purpose, and initial test results are presented in the paper. The system has later been used in real user trials [Zak09]. Preliminary trials in the test apartment revealed a problem which occurred when the network coordinator was rebooted. This issue was solved and the network configuration routines rewritten, so that the sensor network could recover from any device's reboot. The revised version of the WSN has been extremely reliable and robust in the successive trials [Zak09]. In [P8] was also noted, that the original ISO/IEEE didn't suite light-weight wireless technologies such as Zigbee. In Chapter 3 we observe that the ISO/IEEE 11073 has responded by releasing the new personal healthcare version of the ISO/IEEE 11073 which addresses these issues.

Literature points out that personal health monitoring and health care needs at home are growing. The benefits for both the health care system and the user are shown to exist. The technologies are shown to exist for implementing health monitoring at home [P8][P5]. Standard PC and consumer electronics components can be used to implement the systems and infrastructure at a reasonable cost [P2][P8]. New innovative sensor technologies have arisen to measure even more biosignals unobtrusively [P6]. Daily living habits and activities can be monitored to assist in the evaluation of health status and its progress [P8].

6.4 Standardization of medical electrical devices

Designing a medical device is a complex process, which does not yet have harmonized universal guidelines. Good detailed documentation of all the design phases is an essential universal requirement. Medical devices should be safe to their users. Absolute safety can not be obtained, but the benefits of using a medical device should outweigh its risks. The US regulations also require the devices to be clinically effective, while in the EU it is sufficient that the device performs to the manufacturer's intended purpose.

Standards are tools for the regulators and the implementers. Standards can be obligatory or voluntary. The current regulatory trend is towards regulations which use voluntary international standards to define detailed implementations. For medical electrical devices, the electrical safety and related properties are well standardized. The third edition of the IEC 60601-1 released in 2005 adds risk management (ISO 14971) to its contents [Sid06]. By applying risk management

type of procedures, the third edition should keep pace with the technology developments, and can be applied to new emerging technologies. The IEC 60601-1 has evolved from a basic safety standard to also an performance standard. The IEC 60601-1 and the ISO 14971 are essential standards for any medical electrical device. Software standardization is very much in the early development phase. Standardization on medical device networking is also under development. Standards for medical device and network interfaces exist, but their adaptation has been slow.

Medical device interoperability and the lack of it is a recognized problem. Unique and comprehensive ontology of the complete medical domain does not yet exist, and standards need to focus on a limited area of it. The ISO/IEEE 11073 is a proposed solution for point-of-care and personal health devices, defining the interface and also the nomenclature on how medical data for these types of device is presented. It has adopted modern digital interface techniques, USB and Bluetooth, as its transport methods, and has now strong support in the industry. Further guidelines and use cases, such as the ones developed by the Continua Health Alliance and the IHE initiative, describe how these interface standards should be deployed and used in practice, enhancing the interoperability and use of practical medical systems.

The main results of this Thesis have been summarized in this Chapter. The discussion of the results follows in the next Chapter.

Chapter 7

Discussion

This Chapter discusses the results obtained in this Thesis. First, in Section 7.1, the results are evaluated against the goals set for this Thesis. This is followed by a more general discussion on the relevant issues of this Thesis. Finally, future trends and work are discussed.

7.1 Achieving the goals of research

The objectives of this work, as defined in Chapter 1, were to find out what interfacing technologies and emerging standards can be adopted from the personal computer market to medical devices targeted for personal and home use, and to gain understanding of technical and regulatory limitations regarding their use in medical applications through implementing prototypes of these interfaces.

The work presented in this Thesis has shown early on that USB, Bluetooth, and Zigbee are standards suitable for personal and home health monitoring applications. They have been widely used in recent scientific research, and USB and Bluetooth were the first new transport methods chosen for the ISO/IEEE 11073, which previously (IEEE 1073) relied only on serial port and IrDA. Zigbee is being considered for ISO/IEEE 11073, and already offers ISO/IEEE 11073 compliant health care profile.

Feasibility of each technology has been studied both by using the literature (including the documents defining the standard) and in practice by prototype implementations, which has produced low level expertise and findings, of which some are reported in the included publications. Medical device regulation and standardization has been presented in Chapter 3 and the technical expertise obtained through the implementations have been used to make conclusions on the limitations and requirements set by regulations for implementing medical devices with chosen interfaces.

Through research and implementations, special requirements of patient monitoring, biosignal acquisition, and home health monitoring, including specific requirements for various groups like

the elderly and aging population, and heart and lung patients have been learned. The obtained expertise has been used in assessing the suitability of interfaces for different health monitoring needs.

Through implementations, it has also been discovered that some basic implementation architectures exist regardless of the interface technology. These findings have been summarized in the last section of Chapter 2.

Four approaches were taken to reach the main objective. The results of these approaches are discussed in the following subsections. In most approaches, the final aim has been an implementation of a concrete system. The successful implementation as such shows that the primary objective of the approach has been reached.

The publication included in this Thesis presented new scientific results and novel ideas at the time of their publication. The timeline of the Thesis has been long, and conversely the technological development in the field of the Thesis has been swift. Because of this, some of the results obtained in the earlier publications have lost their novelty and become widespread knowledge. Newer versions of technologies have replaced older ones and address some of the issues and problems found in the earlier versions. Digital interfaces develop quickly, and their implementations require both HW, low-level (embedded, device driver) SW, and high-level (PC application) expertise. Building complete, practical and working, systems takes considerably more time than just conceptualizing and simulating systems, and experts from all these areas. The lack of larger supporting research group in the area of the Thesis and limited implementation resources required the author to perform most of the practical implementations. This, together with fulfilling the different technical goals of the funding projects, delayed achieving the goals set for the Thesis and greatly contributed to the extended timeline of it.

7.1.1 PC based health monitoring using cable connected sensors

The first approach was to research methods for interfacing medical monitoring devices from a commercial patient monitoring system using standard interfaces. The objective was to show at hardware and software level how medical devices can be interfaced to a standard PC platform using cable based interface.

This study was part of a Finnish Funding Agency for Technology and Innovation (Tekes) project with partial funding from industry. As such, its general aim was to develop new business and products for Finnish companies.

The studies related to this work were included in publication [P1][P2][P3]. Technical objectives were reached. The first USB implementation had performance problems because the Windows 98 USB driver stack did not function as specified, i.e., it could only perform one transaction to an endpoint per frame. The fundamental communication protocol design had to be altered, which reduced the amount and frequency of data transmission, but slightly in-

creased latency. Later versions of Windows have improved the USB support, but the effect of the improvements has not been evaluated in this research. Also the stability of the OS, which was problematic during the studies performed with Windows 98, has been improved after the Windows NT based kernel was adopted for all versions of Windows starting from Windows 2000. Overall, the implemented device functioned well, and to the author's knowledge, the prototype device was used in further studies by some of the funding companies.

As noted in [P2], medical isolation of USB would allow directly interfacing measurement devices to USB without having to design each measurement device with internal isolation. In [P3] isolation of USB data signals was implemented. Although functional, it did not fully fill the USB specification requirements concerning delays in data transmission. The timing requirements of USB are strict, because the same data lines are used in both transmitting and receiving the data (USB versions 1.0-2.0), and increasing the transmission turn-around time reduces the total bus performance. The implementation of [P3] could have probably been used by incorporating a USB hub into the device, i.e., with hub-isolation-device type of internal construction. This idea, however, was never implemented in practice. To date, at least three scientific publications have referred to the work done in [P2]. Currently there are commercial alternatives available for USB isolation. For example, USB interface controllers using the SPI interface allow quite easy isolation, which has been described in the Maxim AN3891 application note. For data signal isolation, the new ADuM 4160 from Analog Device's offers an attractive single chip solution for isolating the USB 1.0-1.1 low-speed and full-speed data signals. USB 3.0 with its half-duplex communication may be technically easier to isolate, although the extremely high bus clock rate poses new challenges, and maintaining backward compatibility would require also isolation of the full-duplex data signals. The isolation work and the publication [P3] have been cited in at least three international publications.

The work of [P1][P2] was done at the time when USB was still a rather new interface, and had restricted and somewhat poor software support. USB has since proven itself as a viable alternative for demanding data-acquisition applications [Gre09] and even medical applications, for which the approval for the ISO/IEEE 11073 transport method serves as a proof. Looking back, the choice of USB was the correct one. IEEE 1394 (Firewire) failed to reach to popularity of USB and RS-232 has all but disappeared from PCs. The OS USB support has greatly improved in stability. The chosen isolation strategy was challenging. Indeed, the commercial alternatives which have surfaced have used different approaches.

7.1.2 New technologies for wireless medical data-acquisition systems

The second approach consisted of evaluating technologies for wireless medical data-acquisition systems and surveying usage areas for wireless personal area networking (PAN) technologies in

hospitals. The evaluations were to be used for selecting suitable technology/technologies for implementing a wireless bio-signal monitor capable of interfacing to a PC.

The research for this topic was mainly done in publication [P4] and [P5]. In [P4] we evaluated suitability of Bluetooth for medical devices and possible applications, and concluded that several applications exist. Medical device market has since shown this to be correct, and similar ideas have been presented in for example [Häm08]. Real-time material tracking, and operation phase-information type of services [Häm08], more related to RFID technologies, were not envisioned at the time. The evaluations of [P5] were used to select wireless technologies used in the later publications [P6]-[P8], which have proved that the selected technology fills the requirements. The slow regulatory advance of UWB was correctly foreseen. The fact that Bluetooth has been selected as the ISO/IEEE 11073 transport method and Zigbee is being considered further emphasizes the findings of [P4] and [P5]. The publication [P5] has so far been cited in four scientific international publications and was used as seminar course material in Technical University of Dortmund in 2005/2006.

One lack of the Thesis is that WLAN has not been included in the trials. High volume production has meant that there is extensive support for WLAN and the implementation cost is competitive. IP based technologies have increased in popularity and WLAN enable easy integration of WLAN equipped devices into the rest of the IP-based hospital information system, as has been shown in the Wilho-project for example [Ris06, Häm07, Häm08]. Indeed, many modern hospitals now have WLAN installed [Häm08]. The popularity of WLAN has most likely impacted Zigbee, which has not become as popular as expected in low-power application requiring networking support. The recent developments of Bluetooth, the addition of a simpler low-energy specification, will also compete with Zigbee in low-power devices.

7.1.3 Medical monitoring device implementation

The third approach consisted of implementing a medical monitoring device using a traditional analog wired based strategy and by using a modern digital wireless approach and comparing the approaches.

The work of this approach is contained in publication [P6], which is a summary journal article on the development work of the BCG chair [Jun04,Jun05,Jun06]. This work was part of Academy of Finland Proactive health monitoring (ProHeMon) project, which aimed on developing new proactive health monitoring methods, including devices and signal processing methods. The devices implemented in this Thesis were used in a clinical study of more than 150 subjects, and these recordings have been used by signal processing researchers and clinical researchers to publish new analysis methods and findings.

The design goals of the two implementations were met. The first device was designed so that a clinical trial could be initiated. The measurement system was evaluated in a pre-trial by clinical

experts and it was concluded to produce signal from which the ballistocardiogram waveforms described in the literature could be recognized [Koi04b, Koi04a]. The device was approved for medical use in Tampere University Hospital. The second wireless version was finished near the end of the ProHeMon project, and its performance hasn't to date been clinically evaluated, which is a slight drawback. The sensory chair and the sensor films for the both versions are the same, and the small differences in the frequency response are presented and discussed in [P6]. The goal of implementing a wireless unobtrusive ballistocardiographic chair was reached. The prior articles on the implementation [Jun05, Jun06] have so far been cited in six international publications excluding publications of our own group.

The ballistocardiographic chair developed [P6] and the applied signal analysis methods are similar to works of Xinsheng Yu made in mid and late-90's, including publication [Yu96, Yu95]. Yu's publications focus on the BCG signal processing and hardware implementations of the proposed algorithms. The exact implementation of the chair used by Yu is not well known, but it was based on two ultra-low frequency accelerometers. The recordings were made with Cambridge University School of Clinical Medicine and the ECG was similarly recorded from the armrest electrodes. The EMFi-sensor measures force applied to the film. The signal it records represents the change of force, ΔF, applied on the film. Theoretically, as $F = m \times a$, an accelerometer based chair should produce similar BCG to the EMFi-sensor based chair, although the actual measurement device is somewhat different. The EMFi chair requires a rigid platform, on top of which the force sensor sits. An accelerometer based chair requires a seat, or surface of the seat, which can move as unrestrictedly as possible so that it and the attached accelerometers can follow the subtle movements of the body accurately. In practice, a rigid platform construction is mechanically simpler and easier to build than a floating seat required by the accelerometer, although the rigid structure may be more prone to structural resonance [Rit02]. Respectively, Yu's accelerometer based implementation is electrically simpler; it requires simpler analog amplifier electronics.

Other recent work on ballistocardiographic chairs has been performed in Helsinki University of Technology [Rit02, Ket01], though the group has published internationally very little on their findings. Their approach presented in [Rit02] has been based on four load-cells placed between the chairs legs and the floor. The BCG signal has been calculated as a sum of the four signals. The load cells measure both the weight of the subject and the BCG signal, which requires a wider dynamic range from the AD converters compared to our implementation. The chair also has to be adjusted precisely so that the load is placed evenly on all the four cells. Otherwise the obtained signal quality is reduces. Respectively, their implementation tries to minimize the effect of body position, size and movements by using adjustable foot support, using restraining belts, and by providing head support. They have also modelled and studied the mechanical properties of their chairs.

It should be pointed out that the BCG chair's mechanical properties have a small influence

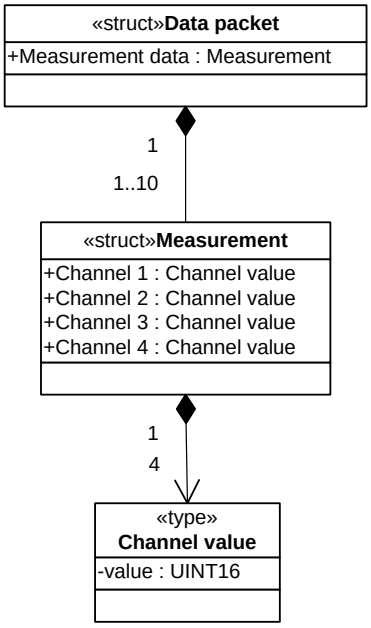


Figure 7.1: The data structure model for the wireless ballistocardiograph.

on the BCG signal, and that this area hasn't yet been extensively studied by our group at Tampere University of Technology. Also the positioning and posture of the subject are known to affect the recorded BCG signal [Rit02]. These factors, which are not in the scope of this Thesis, are probably more significant in determining the clinical usefulness of the system than the technical implementation details.

The development of the wireless ballistocardiograph was an extensive project. The technical implementation was done mainly the author. The mechanical and electrical implementation of the wireless ballistocardiograph finished so late in the project that the messaging implementation had to be designed in a simple fashion. At the time, only an 802.15.4 MAC implementation was available for the RF-interface, which provided a data link level connection. A more engineering type of approach was used, and the data communication was modeled as a data pipe between the two devices. The data packet model used is presented in Figure 7.1. In it, the ballistocardiograph is seen as four distinct measurement channels, each of which produces a single 16 bit unsigned value. When connected to a device, the ballistocardiograph sends continuously the recorded data in RF packets which contain 1 to 10 sets of measurements. This data structure tries to optimize the RF packet payload usage, so that only 25 data packets need to be sent per second when sampling at 250 Hz.

An 11073 messaging implementation, or at least the 11073 packet format and headers, were later considered to be used for data messaging. However, the packet sizes of the original 11073 standard, the only standard available at that time, exceeded the small packet payload size

provided by Zigbee. We investigated methods on implementing transport protocol over Zigbee-NWK layer, which would have enabled larger packets by way of segmentation. A WAP transport protocol (WTP, wireless transport protocol) implementation over Zigbee-NWK layer based on a free source code of a WAP protocol stack was attempted. This three month project ran out of funding and did not produce a usable system in that time, partly because the original source code was designed for a multitasking environment and did not fit well into our interrupt driven network stack.

7.1.4 Wireless home health monitoring system

The fourth and final approach was an implementation of a wireless home health monitoring system. This included research on the expansion of wireless digital interfaces into wireless networks of devices, and the addition of commercial wired devices into the sensor network.

In publication [P7] a Zigbee network layer was implemented, which expanded the IEEE 802.15.4 interface into a network. Based on measurement, adding the Zigbee layer did not notably effect the total radio system performance. In publication [P8] a lightweight sensor network was implemented on top of it to form a wireless health monitoring sensor network. Commercial measurement devices, including a weighing scale and a blood pressure monitor, were successfully debugged and interfaced to the sensor network. These devices provided a RS-232 interface via which measurement data could be read.

In [P8] a so-called common sensor interface was also implemented, which allowed the used of developed sensor using different radio platforms, including completely different radio technologies.

The Zigbee and WSN development project was a good example on what working with developing techniques can mean in practice. The 802.15.4 radio had MAC-support from the manufacturer which was targeted for a certain microcontroller and toolset. This MAC lacked features which were in the specification, which reduced its performance. A newer IC with integrated radio and microcontroller was introduced, but was impossible to buy for half a year. Because of the poor availability, we had to design and build sensor network node devices with the older IC, which was a considerable effort with our small resources. Then the Zigbee specification was updated, and the new version was incompatible with the older one. Meanwhile, the RF IC manufacturer was bought by a bigger company, and the support and reference architecture for the older IC were changed to use another microcontroller and toolset. Support for the newer Zigbee version was only provided for them.

The wireless home health monitoring network implementation did not manage to achieve all the low-power requirements set. One reason was the lacking support for beacon enabled networks, which would have been necessary for low-power two-directional communication. This was originally considered only as a MAC issue, but even the updated software versions have not

solved this issue. However, even after switching to a one-way "broadcast" messaging with the sensor values, the sleep mode power consumption remained high. The cause for this, probably HW related issue, remains unclear.

The expansion of an interface into a network in this case was quite straightforward and simple. One reason was the clear layered model on which the Zigbee-NWK is based. However, networking increases the overall complexity of the system and the application using the interface. These issues are not in the scope of the Thesis.

7.2 General discussion

New techniques applied to health monitoring, or health care in general, should always answer to a recognized need. This process is often reversed, i.e., the development of new technologies may motivate system developers and marketers to apply innovative technology in a medical context. However, development driven by the technology often fails to deal with clinical realities [Wie06].

As engineers or experts in medical device field, we must always remember the needs and requirements of the users to whom we are designing the systems. Health monitoring devices targeted for home use face a diverse market represented by all groups of society and of all ages. The aging population presents its own design challenges as devices for the hearing-impaired and persons with reduced vision or memory disorders place special demands on the usability and user interface of the device. User acceptance for the devices is critical in most cases. Devices should not intimidate the users. An image recognition based activity monitor might not be accepted if the user detects the presence of a camera like component. For some, a blinking led or an abnormal chirping sound from the device may be enough to deter from using a device. A medical professional, on the other hand, may be unwilling to change from a working system to another, if he does not see a clear reason for the change. Any new system requires time to learn and adjust to a new routine and time is something which is often lacking. Studying existing practices and needs, and combining this information with well thought usability design can lead to improved acceptance [Bra05].

The personal health device market differs from the general medical device market in that personal health devices are mostly paid by the users, i.e., patients, themselves. Traditional medical devices are bought and paid for by the hospitals and clinics. To market a personal health monitoring device, one must impress the user of its benefits. On the other hand, to market a medical device for health care institutions, one must impress the health care professionals and then convince the hospital financial administration of the financial benefits. This combined with the strict, costly and time consuming medical device approval process makes the step from personal wellness device market to medical devices market a large one. Also, the related standards are mostly closed commercial documents. It is usually not possible to evaluate them and their potential without purchasing them first. This may be enough to deter from using

them and from medical device markets.

As was observed in the beginning of Chapter 4, the terminology related to personal health monitoring devices and services is not yet uniform. Different terms are used to describe the same systems, slightly different systems used in the same context, or same systems used in a somewhat different context. It would be beneficial to harmonize these terms and their use. This would promote interaction of all fields of research as related publications could be found more easily than currently. Furthermore, having several terms with very similar basic meaning but a small difference in emphasis is problematic for national translations. It is sometimes challenging enough to find a translation to a term. Making a distinction between two related terms can be impossible. There is currently ongoing ISO/TC215/WG3 efforts on harmonization of the standards terminology under the name "Joint Initiative for Global Standards Harmonization, Health Informatics Document Registry and Glossary, Standards Knowledge Management Tool" (www.cred.ca/skmt_glossary). This work may influence the future use of medical device terminology.

7.2.1 Digital interfaces

Digital interface development has been fast in the last ten years. The standard PC interfaces have been redesigned since the mid 1990s. The focus was first on the cable based interfaces and has then shifted to wireless interfaces. Instead of developing new cable based interfaces, the trend in this decade has been to push up the data rates of existing standards. Similar trends can also be seen in wireless interfaces, but in these the radio band limitations often limit the maximum performance and new technologies for higher radio bands are proposed and developed to obtain faster data rates. In general, the current development of digital interfaces has focused on application specific high speed media interfaces, driven by the digital video applications.

The popularity of the RS-232 in the embedded systems shows that there is still demand for simpler interfaces, and that end user API of USB was too complex in the beginning. Embedded systems often require lightweight point-to-point connections, and this was something that the standard USB versions did not support. Instead, the standard USB always requires a host, as all communication is between the host and the device. The implementation of a host is often seen too complex for small embedded devices. The USB OTG standard [USB06] makes way for USB's use in these devices. Similar trends can be seen also in wireless interfaces. The industry still lacks a commonly accepted low-power low data rate solution. Bluetooth tried to satisfy too many needs and became overly complex for simple systems. In the 2.4 GHz band the IEEE 802.15.4 standard has received much support, and is a base for many implementations requiring light-weight wireless interface. It could emerge as a widely adopted radio platform. However, it will face competition from the Bluetooth Low-energy standard, especially if the users of IEEE 802.15.4 remain divided into multiple competing top-layer standards.

Implementation architectures

There are some basic implementation architectures for digital interface implementation, as have been shown in this Thesis. For an embedded system designer, the interface API depends heavily on the interface implementation. The basic architecture may be the same, but components (ICs) from different manufacturers can provide very different APIs. Functionally similar components from different manufacturers usually have different chip pinouts and physical packages. Thirdly, the components may provide support code for a certain commercial tool (programming environment) and microcontroller/processor type or family. The microcontroller type or series has a major effect on the total system performance. The microcontroller peripherals, like timers and A/D converters, and lack/inadequacy of them define how many additional components are required for the system. So, not only should the interface itself be chosen carefully to match the applications requirements, but also the implementation architecture should be carefully evaluated before making final component choices, as changing the primary components of the interface later will lead to major redesign of the whole system.

As has been noted in this Thesis, the modern digital interfaces, wired and wireless, can basically be considered as single data-path systems in which the complexity is in the processing of data frames. In a sense, the interfaces are or could be implemented mainly in software. This opens possibilities towards software defined multi-purpose interfaces, such as software defined radios (SDRs) [Häm08].

Designing digital interfaces for interoperability of systems and components

Interoperability of systems would benefit from having less diverse field of digital interface standards and proprietary solutions. This is easier to achieve in the wired world, where the cable can supply the power, and there are more possibilities to use the same interface with both very high data rate devices and low-power devices. This is especially true in PC based systems, where the PC can be used to supply power to the low-power device connected to it. Direct communication between two low-power devices is a challenge in this respect. The situation with wireless interfaces is more complex. High data rate, low latency, security, quality of service, transfer distance, power consumption, regulatory limitations of the used radio spectrum and other parameters all affect each other and an interface is always a compromise between them. This is why so many diverse wireless interfaces exist and will remain, at least on the physical or transport level. There is, however, some hope, as recent trends have been to use an existing standard as an umbrella under which multiple transport methods are included. This is familiar from the IP world, where the IP protocol has for long time been used in both wired and wireless networks. Indeed, multiple WLAN technologies exist, which are not interoperable at the physical level. More recently, Bluetooth has started to take similar steps, adding support for multiple data link-level technologies. At the same time, Bluetooth itself has been selected to be

used as a transport of the ISO/IEEE 11073 standard. It also has support for the IP protocol.

The trend presented will likely continue. Protocols are complex, their development takes time, and the specifications are refined based on practical experience, resolving operational and security related issues. It is more practical to utilize existing standards when applicable. The OSI 7-layer model gives an example of how the interface could be separated into seven distinct components. It can be argued, that this level of separation is too fine. The physical and data-link layer together provide a basic point-to-point interface. Together they define the most basic properties of the interface and its maximum performance limits. Currently, it is common for an interface IC to include at least a part of the data-link layer functionally. Network and transport layer build on these data-links, by providing a virtual data path or "pipe". These layers can be used to fine tune the performance of the interface within the limits given by the lower layers. These two layers are closely related to the physical and data-link layer, i.e., for performance reasons they are often implemented close to the interface IC. The three higher layers form a third distinct entity, focusing on the data representation, models, and interface to the users application. They can generate large memory structures and hence their implementation often resides on the same processing platform as the main application.

This kind of low, middle and high superlayer protocol component structuring matches the way the practical protocol implementation is often separated onto a HW interface IC, an embedded co-processor running protocol SW and the main processor. If the system is implemented with fewer components, then two (or even all) superlayers can be implemented on the same component. This three superlayer structure can also co-exist with the OSI 7-layer model, i.e., the superlayers can preserve the OSI layers internally if so desired.

Whenever an application uses a digital interface a HW/SW interface must reside somewhere. Having multiple processing units in the digital interface implementation increases the number of HW/SW interfaces, and presents problems for maintaining the strict layered protocol structure. One way to tackle the HW/SW problem, while maintaining layer interfaces, would be placing the HW/SW interface within a layer. This way, a software component offers the standard layer interface upwards, and performs the communication with the hardware using an API provided by it. While preserving the other layers, the internal structure of the layer containing the HW/SW interface would become complex, divided and nonstandard. An alternative approach is presented in Figure 7.2. Using separate interface layers; it preserves the original layered structure by providing a standard interface for upper and lower layer. The actual interface layer implementation is platform dependent/system specific but the communication layers can be made more generic. At its simplest, when both layers connected to the interface layer are residing on the same processor & memory space, it only passes data between functions. When the layer implementation is divided onto two different processing units, the interface layer seamlessly connects these two layers via a HW interface (PCB bus or other digital interface). The interface layer can use the standard protocol interfaces of the upper and lower layers it connects. In this

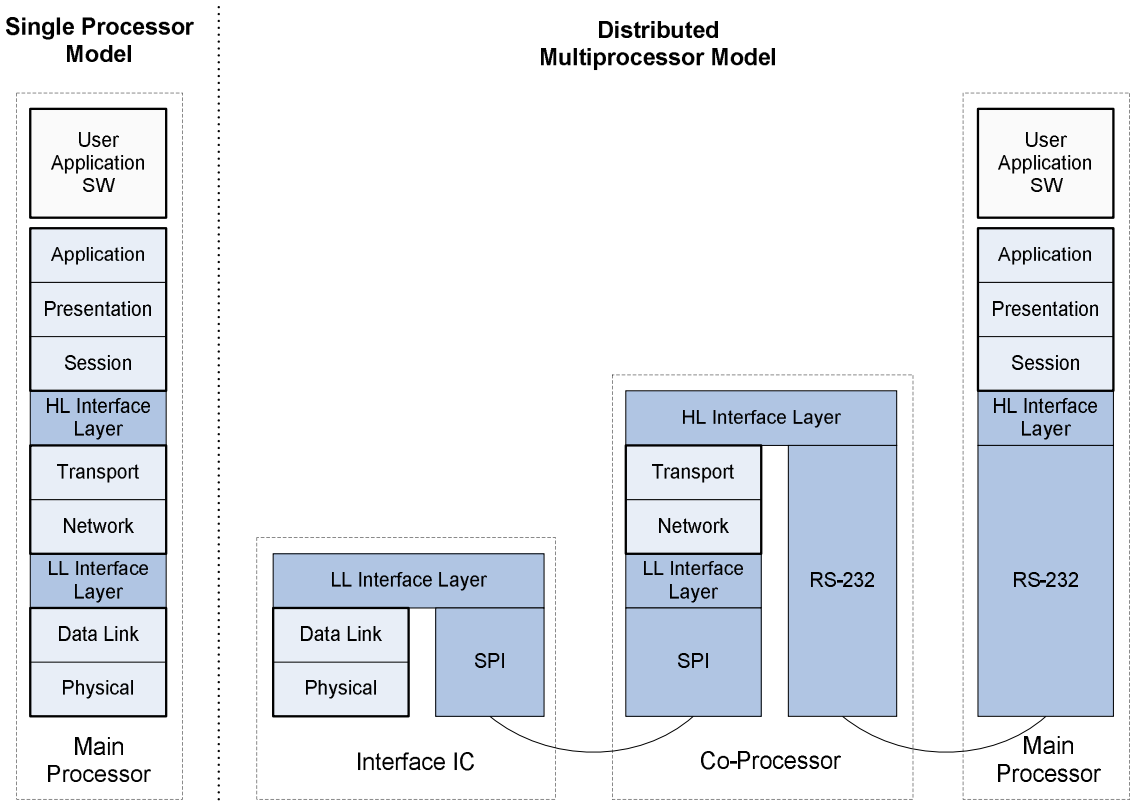


Figure 7.2: An example of using separate interface-layers to distribute communication protocol to separate processing units. In the example, the low-level (LL) interface layer connects the lower and middle layer components via SPI bus, and the high-level (HL) interface layer uses the RS-232 to connect the middle and higher layers.

case no new protocol interfaces need to be specified, the interface layer emulates the operation of the layer it replaces. However, as cross-layer design techniques are applied increasingly more, it can make sense to implement a separate and more versatile interface layer API which can provide support for these features.

The OSI 7-layer model has served well for wired networks, but it has become evident that in wireless networks cross-layer design can provide significant benefits. Cross-layer design is defined by Srivastava and Motani as "protocol design done by actively exploiting the dependences between protocol layers to obtain performance gains" [Sri05]. Most performance benefits focus on the four lowest layers and their interactions. Especially the networking layer can benefit from cross-layer design with the data-link layer [Sri05, and references therein]. As has been learned in this research, the HW/SW boundary often exists below the networking layer. So, even if using the three superlayer structure discussed earlier, the interface between the lower and middle layer should be more than the mere data-pipe traditionally used between the layers.

As so, challenges still remain in defining the common interfaces to these layers, so that these layers could be designed as interchangeable components.

The second major challenge on interfaces is to design and agree on how interfaces should be partitioned, i.e., where to draw the lines for the possible HW/SW interface and to place possible interface layers, and to specify these interfaces also in HW level. Again, the challenge in specifying these layer interfaces would be in predicting the needs of the future, so that these interfaces would not become the bottleneck in performance or limitation in expandability while still keeping them practical in the implementation sense. Were all this achieved, we could truly build interfaces from blocks, selecting the blocks required to match the performance or features needed, while still operating under a more common higher layer standard providing application level interoperability with other systems.

7.2.2 Medical device interfaces

The performance development of digital interfaces and computer systems in general has changed some of the main issues associated to the use of digital interfaces for data acquisition. For traditional biosignal measurements the question no longer is "how does the interface limit the sampling rate and measurement accuracy" or "what compromises have to be done due to limitations in data transfer". For wired systems, the question really is just "how much data do we need to gather", e.g., where is the limit when you just end up recording more and more noise instead of the signal. In today's world it is possible to measure so much data from various sources that we end up drowning in it, i.e., the storage and analysis of the data becomes an issue. All this has somewhat changed the design paradigm of a medical device with regards to the amount of data captured.

Traditionally, we needed to know the performance requirements for the system to select an interface used so that it could match them. Now, especially for wireless interfaces and mobile devices, the questions are "what are the data transfer requirements", "what interface will sufficiently fulfill them", and "how does the selected interface downgrade the system performance in terms of mobility, lifetime, size and portability". Indeed, this often becomes an iterative process, in which the original requirements for the amount of data transferred can be altered in preference of other parameters which improve the devices' usability.

Medical device interface standardization

The global standardization of medical device interfaces would have tremendous effect in the whole. It would increase patient safety, reduce time delays in healthcare, and cut down on errors due to mistakes in data entry. Currently, for example, a doctor in a general practice fills the measurement values by hand to the electronic patient records after performing the measurements. This takes time and typing errors may cause false values to be entered to the

database. Technically, an improved solution would be to have the devices used in the practices and clinics connected directly to a desktop PC using a common digital interface. To fully automate the process, application level software support for the direct transfer of the recorded results into the patient records database would also be needed. This could be implemented by the patient record software manufactures if the medical devices would have a common interface standard, like the ISO/IEEE 11073. This would also enable automatic storage of measurement time, measurement device type, and additional context information. This kind of additional information would be useful in cases where a faulty device is detected and affected measurements would be traced.

At medical systems level, which usually means the hospital and healthcare information systems and related electronic health records, the standardized digital interfaces of medical devices are tools for achieving interoperability, but they themselves do not guarantee it. Interoperability design for larger systems starts always at the business level, which describes business processes, real-world structures, functions, and conditions [Blo02]. When designing medical and healthcare systems, we have to always define the business case first, and then design the system to fulfill the real-world needs. This is why we cannot and need not to solve "universal interoperability of everything" at digital interface level using a single universal standard. For larger healthcare systems we need information systems on top of our interfaces to implement interoperability at higher level and to provide services customized to the users needs. Alternatively, middleware components can be used between the application and the interfaces, to provide independence from the interface technology for the application [Hei04]. As an example, if a home monitoring system requires video surveillance along medical measurements, a non-medical interface standard for the video-camera can be used parallel to the measurement device interface standard (such as ISO/IEEE 11073), and information from both systems can be combined in either middleware layer or in an information system to provide a interoperable application level service.

Information systems use XML and similar languages to form their messages and data descriptions to improve syntactical compatibility [Len07, Coy03]. While these are versatile and powerful for describing complex data structures in an unambiguous manner, they are not very effective in data representation in that a description of a very simple data entity often becomes long and complex. This is one of the reasons why medical information systems are not extended to medical device level, but instead other standards are used to define medical device communication required for interoperability.

Medical device and interface interoperability through standardization can also introduce new problems. A medical device has an intended purpose for which its safety and effectiveness/performance has been conformed. An interoperable device can mistakenly or purposefully be used in combination of other devices for a purpose which is was not designed for. This is again a challenge for the usability and user interface design, topics which are not addresses by the technical interface standards like the ISO/IEEE 11073. The ISO/IEEE 11073 may be

technically adequate and sound, but it can still fail because of issues not directly related to the interface specification, such as poor usability due to poor, complex, or non-harmonized user interfaces, or failure to impress the device manufactures.

There exists some overlap in the development of medical device standards and organizations defining the deployment of these standards. Different organizations focus on the area of interest to them. While this has benefits; parallel competing efforts may produce more new ideas and division into smaller groups focusing on smaller application areas can yield more effective solutions due to tighter focus, this also wastes some of the limited and often voluntary human resources associated with standards development. Furthermore, it would be beneficial for the medical device standardization field as a whole if some of these overlaps could be reduced. This would lead to faster development and deployment and more importantly a more uniform medical device interface standards field. This would reduce the temptation of using emerging interoperability standards from the consumer electronics field.

In home use, and especially in wellness types of applications, medical interfaces and standards may have to give way to more general interoperability standards such as UPnP if true cooperation between all home devices is desired. Otherwise, a medical device coordinator/gateway is required, but this can lead to overlapping infrastructure like separate wireless sensor networks for medical devices and other systems. This topic is addressed in the Section 7.3 describing future trends and work.

Wireless interface technologies

Critical care monitoring would benefit greatly from the wireless interfaces. Having devices connected by cables is a necessity that many would like to see go away [Pak05]. The cables limit free movement around the patient, the movement of patient to and from the operating theatre, and are just generally on the way of the doctors and nurses. However, the critical care monitoring is an area in which relative simplicity in use and stable operation is also needed. The practitioners need to be able to identify quickly how or what devices are connected. Not having a wire between two devices can pose difficulties in understanding the device connections from the operator point-of-view, especially if and when devices are moved with the patient to another location. For example, a wireless device measuring patient A can be logically connected to patient monitoring system of patient B. These are challenges that require both development of the technology and new thinking in usability design. The battery life of a wireless device, and increased size and weight caused by the battery reduce the benefits of the cable free operation. An additional problem is that the lifetime of complex and expensive medical systems is long, and the older systems still in use were designed at a time when wireless digital interfaces were unheard of. Some of these older systems may experience problems (EMI) if used together with wireless devices.

Reducing the amount of data exchanged is the key point in obtaining low-power consump-

tion required for battery powered wireless systems. Data transmission of a sensory device can be reduced by moving intelligence, i.e., signal processing, to the sensory device. In this way, calculated parameters or alert notifications need only to be sent instead of raw biosignals. The key design dilemma here is which consumes more power: the signal processing or the transmission of the raw data.

A special characteristic of medical devices used for health monitoring purposes is the twofold character of operation these devices. In normal conditions, when the patient is well, the role of the device is basically to detect any life critical changes in the subject's condition. This kind of operation fits well to a low-power wireless technology, which optimizes power consumption over throughput and latency. In the critical monitoring state, the requirements for the interface change dramatically. Reliable real-time signal transmission becomes more important than low-power operation. This is a major design dilemma in wireless interface selection for these kinds of mobile battery-powered devices; a single interface is always a compromise. Bluetooth, for example, is tilted towards reliability and performance over low-power. As a basic of-the-self mobile phone model these days already contains multiple-radios, it would seem likely that also wireless medical devices can and will contain multiple-radios in the future. They could not only provide optimized low-power and high-throughput alternatives, but also enhance reliability by providing alternative equivalent wireless technology. The design challenge here is to group these alternative wireless technologies seamlessly under the same application level interface standard.

Reliability of interfaces

An interesting question is in which areas of health monitoring should the operation of the communication interface be most reliable and where could unreliability be tolerated. In critical care use the health of the monitored subject is in critical state, and small device errors may have serious consequences. On the other hand, the medical staff is close by and device errors caused by interfaces should be noticed quickly as they usually lead to "no-signal" types of errors. Failure of a home monitoring interface may remain undetected longer, as no medical staff is usually present to notice the malfunction and the possible deterioration of the subject's health. Furthermore, because of the traveling required, the costs of checking and fixing a malfunction caused by an interface at home is more expensive than in the healthcare facilities. The unreliability of home healthcare systems caused by the interfaces may also be frustrating to the often nontechnical users which may lead to the devices not being used. In this respect, it is safe to say that the reliable and easy operation of interfaces is important to all forms of health monitoring devices, from home to critical care.

7.2.3 Health monitoring systems for home

The home is a mixed environment of consumer entertainment devices, personal computers, home appliances and health monitoring applications, which often use shared resources in terms of power supply, communication, and user interfaces. Devices for personal health monitoring at home should be able to function in this diverse environment, and when applicable, to take advantage of the available resources.

Sports and physical training have used health monitoring techniques for years to maximize training effects and individual's physical performance. In recent year, health, wellness, and lifestyle issues have become popular among the health conscious consumers. Services and devices exist which aim to improve life quality of a "normal", healthy person, to motivate in physical exercise, and to perform self-monitoring of health. Services have also been developed for the elderly to assist them in home living. These kinds of wellness services and their devices are generally not considered as medical devices, and do not undergo the regulatory process required for medical devices.

Devices and technologies used in healthcare facilities can be easily applied for health monitoring at home. The relevant issues often are the device usability and end user documentation which may have to modified to support device's use at home. A secondary issue is often the data communication from the device to the possible remote caregiver which may require adding interfaces capable of connecting to home computer networks or the use of telecommunication technologies.

A non-medical device designed for home health & wellness service purposes can not be applied as such in healthcare facilities, as it lacks the medical device approval. As was shown in Chapter 3, obtaining medical device approval is easier if the possibility of the approval process is taken into account right from the start of the device design process, and relevant documentation of device design is made and stored. If a health monitoring device for wellness purposes is planned and there is potential for the device to also be applied to clinical purposes, one should consider carefully could the device be designed with a possibility for medical device approval. Besides the additional costs of medical device approval, it may also prove challenging to implement all the desired state-of-the-art features common in the home device market, including support for the latest interfaces and protocols, and obtaining medical approval for them in reasonable time.

Health monitoring at home can provide significant advantages to the health care system and the patients. However, the systems still require the supporting services from the health care provides [Bra05, Bra06]. An immediate detection of an attack or seizure is useless if there is no one to notice or respond to it. Widespread use of health monitoring systems at homes produces large amounts of data which requires technical and human resources from the caregivers and intelligent signal processing methods for automated analysis and alarms. Moreover, delays and waiting times in the emergency care services may reduce some of the advantages obtained with personal health monitoring devices used privately to detect sudden changes of the health.

7.2.4 Concurrent research developments

Especially wirelessness has been a hot topic during the recent years, both in general technological development of networks, protocols, circuits and systems, and in the development of applications and services using the possibilities offered by these technologies. The medical and healthcare sector has not been left out of this development. For example, the Finnish national FinnWell programme (2004-2009) alone included 381 projects, of which 115 were related to 24 larger aggregate projects such as Wilho which is presented later in this section. Recently, the EU has launched the 700 M€ Ambient Assisted Living programme (2008-2013, www.aal-europe.eu), which focuses on improving the life quality and daily living of the elderly. Clearly, much concurrent research has been done. In the following, some of the more well known academic projects and groups are presented to give examples of the topics and emphasises set by others.

The CodeBlue project (http://fiji.eecs.harvard.edu/CodeBlue) at Harward Sensor Networks Lab explores applications of wireless sensor network technology to a range of medical applications, including pre-hospital and in-hospital emergency care, disaster response, and stroke patient rehabilitation [Mal04]. The research focuses on the integration of medical sensors with low-power networks, wireless ad-hoc routing protocols for critical care, HW architectures for ultra-low-power sensors, interoperability with hospital information systems, 3D location tracking, and adaptive resource management in wireless networks.

Aware Home Research Initiative (http://awarehome.imtc.gatech.edu) at Georgia Institute of Technology is a multidisciplinary exploration of emerging technologies and services for home which was started in 1998 [Kid99]. The three research areas are chronic care management in the home, future tools for the home, and digital entertainment and media.

The Wilho project (http://www.wilho.net) [Ris06, Häm07, Häm08] focused on healthcare process management supported by wireless technology. The aim of the project was to create a plan for a wireless hospital aimed at boosting the care of patients for domestic as well as export markets. The wireless communication was based on wireless hospital area network (WHAN) with open interfaces both to/from the hospital information system and to/from applications in hospital processes.

The Center for Future Health (http://www.futurehealth.rochester.edu) at University of Rochester was founded in 1998 to create and validate personal health technology for proactive health systems. Their research focuses on automated health assessment, personal health assistant, and integrative research. Especially of interest is the middleware research, which aims to develop a framework to encapsulate various existing networking protocols and to provide a uniform high-level interface [Hei04].

The UbiMon project (http://www.doc.ic.ac.uk/vip/ubimon) at London Imperial College focused on mobile wearable and implantable sensors [Lae04]. It was followed by the SAPHE (Smart and Aware Pervasive Healthcare Environments, 2006-2009, vip.doc.ic.ac.uk/saphe) project which developed telecare networks with miniaturized wearable wireless sensors.

The House_n Research Group (http://architecture.mit.edu/house_n/) at Massachusetts Institute of Technology has various healthcare related projects run by Tech. Director Stephen Intille which focus on ubiquitous computing, intelligent homes, mobile systems using artificial intelligence and computational sensing [Int02, Int05].

The work of this Thesis, namely publications [P7] and [P8] done under the UUTE project, have been part of the Information Technology and Health Care (ITALH) project. The ITALH was a joint project of Tampere University of Technology, Aarhus University (Denmark) and EECS UC Berkeley (USA). The ITALH project focused on the use of IT for health in home, wearable sensors and fixed place monitors. It aimed to identify functions and devices as part of the development process. Robustness, reliability and low-power consumption were the key parameters for the developed systems.

Large companies, such as Intel, Philips, and Texas Instruments to name but a few have also been active in the field. Intel (www.intel.com/healthcare) has been active in standardization bodies and in 2008 released the Intel Health Guide personal health system. Philips Medical has a wide range of products for telehealth, and a line of remote in-home patient monitoring devices. During the recent years, Texas Instruments has started to promote its semiconductor products to healthcare systems. TI has a very strong offering in low-power wireless interface products, with attractive complete solutions including tools and SW-components. It has been interesting to see how traditional semiconductor companies, such as Intel and TI, have started to promote medical and healthcare services and related technologies.

From the above, it is clear that similar concurrent research to that of the Thesis has been done worldwide, and that similar findings and results have been obtained. In many cases, more advanced networks have been built and larger networking trials done, for example in the CodeBlue and Wilho-projects. Others have tried the ISO/IEEE 11073 standards in practice [Yao05, Gal07, ME08], a topic for which our group did not have the resources. The implementations done in this Thesis have studied the feasibility of technologies and arrived to similar conclusions to that of others. In general, the past decade has been very much a decade of emerging technologies, and new possible applications and services. Many of the projects have been demonstrative in their nature, as have been the works of this Thesis. These demonstrative works are important, as they bring forth issues and challenges, technological and non-technical, which can be addressed in the future systems as the miniaturization of the devices and the battery techniques advance. They also help in testing new service ideas and concepts. The challenge for the next decade is to build on these prototypes and small scale proprietary products, to mass produce them into interoperable systems and services.

7.3 Future trends and work

The development of computers and microprocessors will inevitably continue leading to even faster, smaller, and cheaper devices. However, sufficient computing power already exists for most applications. The size and cost have reached the level in which it is possible to include a microprocessor and memory to most pieces of medical equipment [Fag06]. The inclusion of microprocessors into medical and healthcare devices makes way for new pervasive healthcare techniques and services, which not only effect health monitoring, but also other forms of healthcare as presented by Varshney in [Var03]:

- Mobile telemedicine: Patient's health history can be accessed in remote locations where quick decisions are needed. Also remote consulting is possible, and the hospital can receive critical information about the patient's health while he is being transported to the hospital. If healthcare facilities are equipped with WLANs, doctors and staff can review and update patient's medical records from any location using hand-held devices. In addition, physicians can generate and transmit prescriptions wirelessly to a pharmacy which saves time and increases accuracy.

- Patient monitoring: Remote patient monitoring allows continuous monitoring virtually anywhere. Patient's can be moved to less costly care facilities or even to their homes earlier after operations.

- Location based-services: Patient tracking useful especially in elderly care and with patients having mental diseases. These can also be used to locate remotely monitored patients in case of emergency and users can be directed to services.

- Intelligent emergency response and management: Could be used to filter emergency calls by matching different reports of the same event, and avoid dispatching multiple vehicles to the same emergency.

- Pervasive access to medical data: Automated access and updates to clinical records.

- Health-aware mobile devices: Hand-held wireless devices could detect certain parameters from users touch and behavior. This information could be used to generate alerts for healthcare providers.

- Lifestyle incentive management: Good habits, exercise and healthy meals for example, could be rewarded financially using mobile payments.

Some of these pervasive services already exist, at least in prototypes, but many are still in development and require support from standards to be effective. For example, the electronic

prescription does not yet have a uniform standard which is needed before it can be adopted as a integral part of the services.

Media content is increasingly more often transferred in digital form. Digital television, digital PVR (personal video recorder) devices and set-top boxes, and multimedia computer systems have become common in homes. For a long time the media signals were transferred in analog form, but this is now changing as new digital interface technologies have been introduced for these purposes. IP-technology is also being incorporated into increasingly more devices, including home media applications. These media devices can offer much information and also resources (user interfaces, communication networks) which could be used in personal health monitoring applications. Medical device manufacturers and related organizations, such as the Continua Health Alliance, would like to see the ISO/IEEE 11073 adopted as widely as possible, including personal wellness applications, to enable a uniform and interoperable medical & related devices field. Manufacturers of home entertainment and multimedia applications may not have interest in supporting these medical standards, and they may incorporate wellness services into their own products using standards that are more common in home consumer devices.

It is clear that the market for health monitoring at home will grow, and media services, such as video call technologies, which allow the patient and the relatives & caregivers to keep in touch, will have a key role in the process. Future work should focus on ensuring systems interoperability of medical interfaces and the new media interfaces.

It is also likely that home automation and home security systems will increase in popularity, delivering more intelligent and networked devices into our everyday lives. These devices should be interfaced into health monitoring systems at home to obtain data on ambient activities and daily behavior which can provide additional information for the health assessment. In addition, some sensor technologies related to activity monitoring for health purposes could also be used in non-medical context if the sensor information could be exported to the home automation and home security systems. It would seem more practical to implement these kinds of general purpose home sensor networks using more general purpose (non-medical specific) technologies, and use a gateway device to implement the interface to the more application specific domains, such as the medical devices using the ISO/IEEE 11073.

Chapter 8

Conclusions

In this Thesis, the application of three modern digital interfaces to health monitoring purposes has been presented. USB, Bluetooth, and Zigbee have all been shown to suit particular requirements of specific personal health care applications. Prototype implementations of real biosignal monitoring, medical measurement, and home health monitoring systems were implemented to demonstrate the effectiveness and performance of the USB and Zigbee interfaces in practice. Two of the three implemented systems where used in clinical trials involving medical professionals and real test subjects with positive results.

Through the included publications, this Thesis presents a wide range of topics related to the medical monitoring devices interface implementation, starting from the actual measurement of the analog biosignal and ending in the user interface of the PC system used to monitor and record the measured data. In the Thesis, the current state of medical device regulation and standardization has been surveyed and reflected on the interface implementation.

The research reported in this Thesis has been to a great extent applied technical research rather than basic research. The developed USB patient monitoring prototype devices and software were used in further commercial trials, and the USB isolation studies have been referred by other researchers. The BCG chair implemented as a part of this Thesis has been used to obtain new information on the human cardiovascular and respiratory system. The Thesis covers the technical implementation of the BCG chairs sensors' measurement electronics, data transfer, and the interface to the PC and its GUI application, which have all been tested and validated. The wireless sensor network developed in this work has enabled the implementation of a portable health status monitoring system for home use. The Zigbee networking stack and the overlaying sensor network, and the sensor node HW/SW architecture have been implemented and tested in this Thesis. The BCG chair and the wireless sensor network have given a foundation and provided measurement data for applied and basic research of others. This data could not have been obtained without the correct and reliable operation of these systems. The importance of this data has been shown by the number of publications and the two theses already produced,

some of which are referred in the Thesis.

8.1 Main contribution of the thesis

The objective of the Thesis was to find out what interfacing technologies and emerging standards could be adopted from the personal computer market to medical devices targeted for personal and home use. Additionally, the Thesis sought out to gain understanding of technical and regulatory limitations regarding such use through prototype implementations.

The Thesis presented thoroughly the interface alternatives available in the personal computer market for small scale medical devices, such as sensory systems and actuators, practical findings on implementing such interfaces, and the trends in the development and attempts on the standardization of such interfaces. The hypothesis set at the start of the research was that new PC and consumer electronics' digital interface standards could be widely used in medical device applications, especially in personal health monitoring. This hypothesis has been proven by the publications included in this Thesis and recently also by the development of the commercial medical device market. This Thesis has studied and presented the implementation of digital interfaces from various distinct viewpoints, and brought up observed features and shown that there are problems and issues which are not evident or clear to see just by reading the interface specification.

The main contributions of this Thesis are included in the attached publications. In addition, the introductory part of the Thesis presents new information which has not been presented in the publications. Chapter 2 shows that four basic architectures for the implementation of digital interfaces in embedded systems can be found, and also presents the structure and components of a modern digital interface. Chapter 3 presents an up-to-date summary of the rapidly developing field of medical device and interface standardization and regulation. The overlapping terminology related to personal health monitoring is surveyed and categorized by the author in Chapter 4. From the introductory part of the Thesis, it can be concluded that interoperability of devices, such as medical electrical devices, is a multifaceted and layered issue. Standardized digital interfaces presented in this Thesis give the foundation to device interoperability. Common domain specific semantics are required so that the devices can exchange meaningful information. In the end, the way and why devices should interoperate are questions that are not answered by plain technical interface standards. These are questions related to the business domain and related processes and need to be defined separately by other standards and documents.

The publications included in the Thesis provided new scientific information at the time of their publication. Due to the rapid development of the technology and the timeline of the Thesis some of these achieved results have become commonly known or even outdated as new technologies have replaced their predecessors. The main findings and results of the included publications were: [P1] found out the different USB device implementation alternatives, showed

that there were lacks in the Windows 98 USB support, and brought up challenges associated with the device drivers of complex modern digital interfaces. [P2] presented a novel idea of a PC-based patient monitor with USB based instruments. This idea is now becoming reality with the inclusion of USB into the ISO/IEEE 11073 standard. [P3] presented a novel method for the electrical isolation of USB. [P4] analyzed the functionality of Bluetooth and its usage areas in medical devices. It also presented some findings on Bluetooth's limitations, which were not well understood at the time. The estimations on Bluetooth usage areas in medical devices presented in the paper have proven to be quite accurate. [P5] presented new and up-to-date information on short-range wireless technologies, which were used to select the interface technology used in later research. [P6] presented a new method for BCG measurement, and two practical implementations and their performance. In [P7], Zigbee communication stack was implemented and its performance and features presented. [P8] presented a self-implemented sensor network, an innovative "common sensor interface" architecture for the design of RF interface independent sensors, and preliminary studies performed with the system in a real home environment.

Bibliography

[Akh07a] A. Akhbardeh, S. Junnila, T. Koivistoinen, and A. Värri, "An intelligent ballistocardiographic chair using a novel SF-ART neural network and biorthogonal wavelets," *Journal of Medical Systems*, pp. 69–77, 2007.

[Akh07b] A. Akhbardeh, S. Junnila, M. Koivuluoma, T. Koivistoinen, and A. Värri, "Applying novel time-frequency moments singular value decomposition method and artificial neural networks for ballistocardiography," *Eurasip Journal on Advances in Signal Processing*, p. 9p, 2007.

[Akh07c] A. Akhbardeh, "Signal classification using novel pattern recognition methods and wavelet transforms," Ph.D. dissertation, Tampere University of Technology, 2007, publication 652.

[Akh07d] A. Akhbardeh, S. Junnila, M. Koivuluoma, T. Koivistoinen, V. Turjanmaa, T. Kööbi, and A. Värri, "Towards a heart disease diagnosing system based on force sensitive chair's measurement, biorthogonal wavelets and neural networks," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 4, pp. 493 – 502, 2007.

[Alt03] C. Altenstetter, "EU and member state medical devices regulation," *International journal of technology assessment in health care*, vol. 19, pp. 228–248, 2003.

[Alt07] C. Altenstetter and G. Permanand, "EU regulation of medical devices and pharmaceuticals in comparative perspective," *Review of Policy Research*, vol. 24, pp. 385–405, 2007.

[Amb09] S. W. Ambler. (2009) Data modeling 101. Ambysoft. [Online]. Available: http://www.agiledata.org/essays/dataModeling101.html

[AST09] *draft ASTM TC F29.21 Medical Devices and Medical Systems - Essential safety requirements for equipment comprising the patient-centric integrated clinical environment (ICE) Part 1: General requirements and conceptual model*, ASTM Std. Working Draft ASTM final F-2761, February 2009.

[Axi05] F. Axisa, P. M. Schmitt, C. Gehin, G. Delhomme, E. McAdams, and A. Dittmar, "Flexible technologies and smart clothing for citizen medicine, home healthcare, and disease prevention," *IEEE Transactions on Information Technology in Biomedicine*, vol. 9, no. 3, pp. 325–336, Sept. 2005.

[Bal06] T. Ball, E. Bounimova, B. Cook, V. Levin, J. Lichtenberg, C. McGarvey, B. Ondrusek, S. K. Rajamani, and A. Ustuner, "Thorough static analysis of device drivers," pp. 73–85, 2006.

[Bar05] T. Barger, D. Brown, and M. Alwan, "Health-status monitoring through analysis of behavioral patterns," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 35, no. 1, pp. 22–27, Jan. 2005.

[Bar07] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards," *Computer Communications*, vol. 30, no. 7, pp. 1655 – 1695, 2007.

[Bar08] L. C. Barna, "Acquisition systems and analysis methods for human originated signals," Ph.D. dissertation, Tampere University of Technology, 2008, publication 720.

[BL02] O. Boric-Lubecke and V. M. Lubecke, "Wireless house calls: using communications technology for health care and monitoring," *IEEE Microwave Magazine*, vol. 3, no. 3, pp. 43–48, Sept. 2002.

[Blo02] B. Blobel, *Analysis, Design and Implementation for Secure and Interoperable Distributed Health Information Systems*. IOS Press, 2002.

[Blu07] *Bluetooth Specification Version 2.1 + EDR*, Bluetooth SIG Std., July 2007.

[Blu08] *Health Device Profile*, Bluetooth SIG, Medical Devices Working Group Std., June 2008.

[Bra05] T. Bratan and M. Clarke, "Towards the design of a generic systems architecture for remote patient monitoring," in *27th Annual International Conference of the Engineering in Medicine and Biology Society IEEE-EMBS 2005*, Jan. 2005, pp. 106–109.

[Bra06] T. Bratan and M. Clarke, "Optimum design of remote patient monitoring systems," in *28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society EMBS '06*, Aug-Sept. 2006, pp. 6465–6468.

[Bre06] P. F. Brennan and J. B. Starren, *Biomedical Informatics, Computer Applications in Health Care and Biomedicine*, 3rd ed. Springer New York, 2006, vol. Unit II, ch. Consumer Health Informatics and Telehealth, pp. 511–536.

[Bro95] J. D. Bronzino, Ed., *The Biomedical Engineering Handbook*. CRC Press, Inc., 1995.

[Car03] C. D. Carr and S. M. Moore, "Ihe: a model for driving adoption of standards," *Computerized Medical Imaging and Graphics*, vol. 27, no. 2-3, pp. 137 – 146, 2003.

[Car07] R. Carroll, R. Cnossen, M. Schnell, and D. Simons, "Continua: An interoperable personal healthcare ecosystem," *Pervasive Computing, IEEE*, vol. 6, no. 4, pp. 90–94, Oct.-Dec. 2007.

[Cel95] B. G. Celler, W. Earnshaw, E. D. Ilsar, L. Betbeder-Matibet, M. F. Harris, R. Clark, T. Hesketh, and N. H. Lovell, "Remote monitoring of health status of the elderly at home. a multidisciplinary project on aging at the university of new south wales," *International Journal of Bio-Medical Computing*, vol. 40, no. 2, pp. 147 – 155, 1995.

[Cel96] B. Celler, E. Ilsar, and W. Earnshaw, "Preliminary results of a pilot project on remote

monitoring of functional health status in the home," in *Proceedings of the 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 1996. Bridging Disciplines for Biomedicine.*, vol. 1, Oct-3 Nov 1996, pp. 63–64 vol.1.

[Cel04] H. Celis and R. H. Fagard, "White-coat hypertension: a clinical review," *European Journal of Internal Medicine*, vol. 15, no. 6, pp. 348 – 357, 2004.

[Cha00] J. Y. Chai, "Medical device regulation in the united states and the european union: a comparative study," *Food and Drug Law Journal*, vol. 55(1), pp. 57–80, 2000.

[Cha07] P. Chadwick, "Regulations and standards for wireless applications in eHealth," in *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS) 2007*, Aug. 2007, pp. 6170–6173.

[Che03] M. Cheng, *Medical Device Regulations, Global overview and guiding principles.* Department of Blood Safety and Clinical Technology, World Health Organization, Geneva, Switzerland, 2003, iSBN 92 4 154618 2.

[Che05] N. Chevrollier and N. Golmie, "On the use of wireless network technologies in healthcare environments," in *Proceedings of the fifth IEEE workshop on Applications and Services in Wireless Networks (ASWN 2005)*, Paris, France, June 2005, pp. 147–152.

[Cis08] Cisco Systems, Inc., *Internetworking Technologies Handbook*, internet edition (2008) ed. Cisco Press, 2008. [Online]. Available: http://www.cisco.com/en/US/docs/internetworking/technology/handbook/ito_doc.html

[Cla04] M. Clarke, R. Jones, T. Bratan, and A. Larkworthy, "Providing remote patient monitoring services in residential care homes," in *Proc. Healthcare Computing 2004*, March 2004, pp. 114–122.

[Cla05] M. Clarke and R. Jones, "A telemonitoring architecture to support chronic disease management and acute episode monitoring," in *27th Annual International Conference of the Engineering in Medicine and Biology Society IEEE-EMBS 2005.*, Jan. 2005, pp. 3711–3713.

[Cla07] M. Clarke, D. Bogia, K. Hassing, L. Steubesand, T. Chan, and D. Ayyagari, "Developing a standard for personal health devices based on 11073," in *Proceedings of the 29th Annual International Conference of the IEEE EMBS*, August 2007, pp. 6174–6176.

[Cle04] T. P. Clemmer, "Computers in the ICU: Where we started and where we are now," *Journal of Critical Care*, vol. 19, no. 4, pp. 201 – 207, 2004.

[Com00] *Universal Serial Bus Specification, Revision 2.0*, www.usb.org, Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips Std., Rev. 2.0, April 2000.

[Cou03] S. Coughlan and J. Breslin, "The application of modern PDA technology for effective handheld solutions in the retail industry," in *Proceedings of IEEE International Conference on Industrial Technology, 2003*, vol. 1, Maribor, Slovenia, Dec 2003, pp. 411–415.

[Coy03] J. F. Coyle, A. R. Mori, and S. M. Huff, "Standards for detailed clinical models as

the basis for medical data exchange and decision support." *International Journal of Medical Informatics*, vol. 69, no. 2-3, pp. 157–174, March 2003.

[Dis04] E. Dishman, "Inventing wellness systems for aging in place," *Computer*, vol. 37, no. 5, pp. 34–41, May 2004.

[Elm94] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 2nd ed., ser. World Student Series Edition. 390 Bridge Parkway, Redwood City, CA 94065: The Benjamin/Cummings Publishing Company, Inc., 1994.

[EU.93] "Council directive 93/42/EEC of 14 june 1993 concerning medical devices," Official Journal L 169, 12/07/1993 P. 0001 - 0043, July 1993.

[EU.07] "Directive 2007/47/EC of the european parliament and of the council of 5 september 2007," Official Journal of the European Union, September 2007.

[Eys00] G. Eysenbach, "Consumer health informatics," *Brittish Medical Journal*, vol. 320, pp. 1713–1716, June 2000.

[Eys01] G. Eysenbach, "What is e-health?" *Journal of Medical Internet Research*, vol. 3(2):e20, April–June 2001.

[Fag06] L. M. Fagan and E. H. Shortliffe, *Biomedical Informatics, Computer Applications in Health Care and Biomedicine*, 3rd ed. Springer New York, 2006, vol. Unit II, ch. The Future of Computer Applications in Biomedicine, pp. 829–846.

[Fer05] E. Ferro and F. Potorti, "Bluetooth and Wi-Fi wireless protocols: a survey and a comparison," *Wireless Communications, IEEE*, vol. 12, no. 1, pp. 12–26, Feb. 2005.

[Gal06] M. Galarraga, L. Serrano, I. M. Ruiz, and P. de Toledo, *Review of the ISO/IEEE 11073 - PoCMDC standard for medical device interoperability and its applicability in home and ambulatory telemonitoring scenarios.* UPNA Press, 2006.

[Gal07] M. Galarraga, L. Serrano, I. Martinez, P. de Toledo, and M. Reynolds, "Telemonitoring systems interoperability challenge: An updated review of the applicability of ISO/IEEE 11073 standards for interoperability in telemonitoring," Aug. 2007, pp. 6161–6165.

[Gan06] A. Ganapathi, V. Ganapathi, and D. Patterson, "Windows xp kernel crash analysis," in *LISA '06: Proceedings of the 20th conference on Large Installation System Administration.* Berkeley, CA, USA: USENIX Association, 2006, pp. 12–12.

[Gar06a] R. M. Gardner and M. M. Shabot, *Biomedical Informatics, Computer Applications in Health Care and Biomedicine*, 3rd ed. Springer New York, 2006, vol. Unit II, ch. Patient-Monitoring Systems, pp. 585–625.

[Gar06b] C. Garritty and K. El Emam, "Who's using PDAs? estimates of PDA use by health care providers: A systematic review of surveys," *Journal of Medical Internet Res*, vol. 8, no. 2, p. e7, May 2006.

[Gil01] J. D. Gilsinn and K. Lee, "Wireless interfaces for IEEE 1451 sensor networks," in *Proc. First ISA/IEEE Conference Sensor for Industry*, 5–7 Nov. 2001, pp. 45–50.

[Gol05] J. M. Goldman, R. A. Schrenker, J. L. Jackson, and S. F. Whitehead, "Plug-and-play

in the operating room of the future," *Biomedical Instrumentation & Technology*, vol. 39(3), pp. 194–199, 2005.

[Gra98] L. J. Grant, "Regulations and safety in medical equipment design," *Anaesthesia*, vol. 53, no. 1, pp. 1–3, 1998.

[Gre09] G. Green, "Understanding next generation data acquisition," *Electronics World*, pp. 18–21, February 2009.

[Gut01] J. A. Gutierrez, M. Naeve, E. Callaway, M. Bourgeois, V. Mitter, and B. Heile, "IEEE 802.15.4: a developing standard for low-power low-cost wireless personal area networks," *IEEE Network*, vol. 15, no. 5, pp. 12–19, Sept.–Oct. 2001.

[Haa00] J. C. Haartsen, "The Bluetooth radio system," *IEEE Personal Communications*, vol. 7, no. 1, pp. 28–36, Feb. 2000.

[Hal08] D. Halperin, T. S. Heydt-Benjamin, K. Fu, T. Kohno, and W. H. Maisel, "Security and privacy for implantable medical devices," *IEEE Pervasive Computing*, vol. 7, no. 1, pp. 30–39, 2008.

[Häm07] M. Hämäläinen, P. Pirinen, and Z. Shelby, "Advanced wireless ict healthcare research," in *Mobile and Wireless Communications Summit, 2007. 16th IST*, July 2007, pp. 1–5.

[Häm08] M. Hämäläinen, P. Pirinen, Z. Shelby, and J. Iinatti, *Advances in Mobile and Wireless Communications*, ser. Lecture Notes in Electrical Engineering. Springer Berlin Heidelberg, 2008, vol. 16, ch. Wireless Applications in Healthcare and Welfare, pp. 351–364.

[Hei95] S. Heiler, "Semantic interoperability," *ACM Computing Surveys*, vol. 27, no. 2, pp. 271–273, June 1995.

[Hei04] W. Heinzelman, A. Murphy, H. Carvalho, and M. Perillo, "Middleware to support sensor network applications," *Network, IEEE*, vol. 18, no. 1, pp. 6–14, Jan/Feb 2004.

[Heu97] V. P. Heuring and H. F. Jordan, *Computer Systems Design and Architecture*. Addison-Wesley Longman, Inc., 1997.

[Hew08] *Universal Serial Bus 3.0 Specification*, Hewlett-Packard Company, Intel Corporation, Microsoft Corporation, NEC Corporation, ST-NXP Wireless, Texas Instruments Std., Rev. 1.0, November 2008.

[Hig02] G. R. Higson, *Medical device safety - The Regulation of Medical Devices for Public Health and Safety*. Institute of Physics Publishing, Bristol and Philadelphia, 2002.

[Hon04] M. Honkanen, A. Lappetelainen, and K. Kivekas, "Low end extension for bluetooth," in *2004 IEEE Radio and Wireless Conference*, Sept. 2004, pp. 199–202.

[Hor95] L. R. Horton, "Medical device regulation in the european union," *Food Drug Law Journal*, vol. 50(3), pp. 461–476, 1995.

[Hu08] Y. Hu, "USB embedded host performance," Micro Digital, Inc, White Paper, 11 2008.

[IEE07] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std 802.11-2007, IEEE Computer Society Std. ISBN 0-7381-5656-6, June

2007.

[Ife93] E. C. Ifeachor and B. W. Jervis, *Digital Signal Processing - A Practical Approach*. Harlow, England: Addison-Wesley, 1993.

[Im02] G.-H. Im, K.-M. Kang, and C.-J. Park, "FEXT cancellation for twisted-pair transmission," *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 5, pp. 959–973, Jun 2002.

[Ins90] *IEEE standard glossary of software engineering terminology*, Institute of Electrical and Electronics Engineers Std., Dec 1990.

[Ins96] *1394-1995 IEEE standard for a high performance serial bus*, Institute of Electrical and Electronics Engineers Std., Aug 1996.

[Ins03] *IEEE Std. 802.15.4-2003 Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Press, Institute of Electrical and Electronics Engineers Std., October 2003.

[Ins08a] *1394-2008 IEEE Standard for a High-Performance Serial Bus*, IEEE Std 1394-2008, Institute of Electrical and Electronics Engineers Std., October 2008.

[Ins08b] *Health informatics-Personal health device communication Part 20601: Application profile- Optimized Exchange Protocol*, IEEE Std 11073-20601-2008, Institute of Electrical and Electronics Engineers, Inc. Std., Dec. 2008.

[Int87] *Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1)*, International Organization for Standardization Std. International Standard 8824, December 1987.

[Int02] S. S. Intille, "Designing a home of the future," *IEEE Pervasive Computing*, vol. 1, no. 2, pp. 76–82, 2002.

[Int05] S. S. Intille, K. Larson, J. S. Beaudin, J. Nawyn, E. M. Tapia, and P. Kaushik, "A living laboratory for the design and evaluation of ubiquitous computing technologies," in *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2005, pp. 1941–1944.

[Ist04] R. S. H. Istepanian, E. Jovanov, and Y. T. Zhang, "Guest editorial introduction to the special section on M-Health: Beyond seamless mobility and global wireless health-care connectivity," *IEEE Transactions on Information Technology in Biomedicine*, vol. 8, no. 4, pp. 405–414, Dec. 2004.

[Jon01] V. Jones, R. Bults, D. Konstantas, and P. A. M. Vierhout, "Healthcare PANs: Personal area networks for trauma care and home care," in *Proceedings of the Fourth International Symbosium on Wireless Personal Multimedia Communications (WPMC)*, Aalborg, Denmark, September 2001, pp. 1369–1374.

[Jun04] S. Junnila, T. Koivistoinen, T. Kööbi, J. Niittylahti, and A. Värri, "A simple method for measuring and recording ballistocardiogram," in *17th int. EURASIP conference BIOSIGNAL*, Brno, Czech Republic, June 2004, pp. 232–234.

[Jun05] S. Junnila, A. Akhbardeh, A. Värri, and T. Koivistoinen, "An EMFi-film sensor based ballistocardiographic chair: performance and cycle extraction method," in *Proc. IEEE Workshop on Signal Processing Systems, Design and Implementation (SIPS 2005)*, Athens, Greece, 2–4 Nov. 2005, pp. 373–377.

[Jun06] S. Junnila, A. Akhbardeh, L. C. Barna, I. Defee, and A. Värri, "A wireless ballistocardiographic chair," in *Proc. 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society EMBS '06*, Aug. 30 2006–Sept. 3 2006, pp. 5932–5935.

[JWC98] J. John W. Clark, M. R. Neuman, W. H. Olson, R. A. Peura, J. Frank P. Primiano, M. P. Siedband, J. G. Webster, and L. A. Wheeler, *Medical Instrumentation: Application and Design*, 3rd ed., J. G. Webster, Ed. John Wiley & Sons, Inc., New York, 1998.

[Kaw99] A. Kawarada, A. Tsakada, K. Sasaki, M. Ishijima, T. Tamura, T. Togawa, and K. Yamakoshi, "Automated monitoring system for home health care," in *Proc. First Joint BMES/EMBS Conference*, vol. 2, Atlanta, GA, USA, 13–16 Oct. 1999, p. 694.

[Ken94] R. Kennelly and J. Wittenber, "New IEEE standard enables data collection for medical applications," in *Proceedings of Annual Symposium on Computer Applications in Medical Care*, 1994, pp. 531–535.

[Ket01] R. Kettunen, J. Heikkinen, R. Sepponen, P. Sipola, T. Anttonen, A. Kilpelinen, S. Koskinen, J. Ritola, R. von Hertzen, and E. Jrvinen, "Ballistocardiography in physical training preliminary study," in *International Journal of Bioelectromagnetism*, vol. 3, no. 2, 2001.

[Kid99] C. D. Kidd, R. Orr, G. D. Abowd, C. G. Atkeson, I. A. Essa, B. MacIntyre, E. D. Mynatt, T. Starner, and W. Newstetter, "The aware home: A living laboratory for ubiquitous computing research," in *CoBuild '99: Proceedings of the Second International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture*. London, UK: Springer-Verlag, 1999, pp. 191–198.

[Koc06] S. Koch, "Home telehealth–current state and future trends," *International Journal of Medical Informatics*, vol. 75, no. 8, pp. 565 – 576, 2006.

[Koi04a] T. Koivistoinen, S. Junnila, A. Värri, and T. Kööbi, "A new method for measuring the ballistocardiogram using EMFi sensors in a normal chair," in *Proc. 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '04)*, vol. 1, 1–5 Sept. 2004, pp. 2026–2029.

[Koi04b] T. Koivistoinen, L. Barna, J. Alametsä, S. Junnila, M. Koivuluoma, T. Kööbi, V. Turjanmaa, and A. Värri, "Ballistocardiograms from young adults in sitting position using EMFi sensors," in *Proc. of the 31st Annual Meeting of Computers in Cardiology*, Chicago, Illinois, USA, September 2004.

[Koi04c] M. Koivuluoma, L. Barna, J. Alametsä, and A. Värri, "Signal processing in ProHe-

Mon project: objectives and first results," in *Proceedings of the Proactive Computing Workshop, Linden, G. (ed.), PROW 2004*, Helsinki, Finland, 25–26 November 2004, pp. 55–58.

[Kor03] I. Korhonen, J. Parkka, and M. Van Gils, "Health monitoring in the home of the future," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 22, no. 3, pp. 66–73, May-June 2003.

[Kug09] T. Kugelstadt, "Designing with digital isolators," *Analog Applications Journal*, vol. 2Q, pp. 21–32, 2009.

[Lae04] K. V. Laerhoven, B. P. L. Lo, J. W. P. Ng, S. Thiemjarus, R. King, S. Kwan, H. werner Gellersen, M. Sloman, O. Wells, P. Needham, N. Peters, A. Darzi, C. Toumazou, and G. zhong Yang, "Medical healthcare monitoring with wearable and implantable sensors," in *Proc. of the 3rd International Workshop on Ubiquitous Computing for Healthcare Applications*, 2004.

[Lea07] N. Leavitt, "For wireless USB, the future starts now," *Computer*, vol. 40, no. 7, pp. 14–16, July 2007.

[Lee06] J. S. Lee, "Performance evaluation of IEEE 802.15.4 for low-rate wireless personal area networks," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 3, pp. 742–749, Aug. 2006.

[Len07] R. Lenz, M. Beyer, and K. A. Kuhn, "Semantic integration in healthcare networks," *International Journal of Medical Informatics*, vol. 76, no. 2-3, pp. 201 – 207, 2007, connecting Medical Informatics and Bio-Informatics - MIE 2005.

[Lin99] J. Lin, "Applying telecommunication technology to health-care delivery," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 18, no. 4, pp. 28–31, July-Aug. 1999.

[Mai04] W. H. Maisel, "Medical Device Regulation: An Introduction for the Practicing Physician," *Ann Intern Med*, vol. 140, no. 4, pp. 296–302, 2004.

[Mal04] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, "Codeblue: An ad hoc sensor network infrastructure for emergency medical care," in *Proceedings of the MobiSys 2004 Workshop on Applications of Mobile Embedded Systems (WAMES 2004)*, April 2004, pp. 12–14.

[Mar08] I. Martinez, J. Fernandez, M. Galarraga, L. Serrano, P. de Toledo, S. Jimenez-Fernandez, S. Led, M. Martfnez-Espronceda, and J. Garcia, "Implementation of an end-to-end standard-based patient monitoring solution," *Communications, IET*, vol. 2, no. 2, pp. 181–191, February 2008.

[ME08] M. Martinez-Espronceda, L. Serrano, I. Martinez, J. Escayola, S. Led, J. Trigo, and J. Garcia, "Implementing ISO/IEEE 11073: Proposal of two different strategic approaches," in *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS) 2008*, Aug. 2008, pp. 1805–1808.

[Mei06] M. Meingast, T. Roosta, and S. Sastry, "Security and privacy issues with health care

information technology," in *28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS 06)*, 30 2006-Sept. 3 2006, pp. 5453–5458.

[Mey05] S. Meystre, "The current state of telemonitoring: A comment on the literature," *Telemedicine and e-Health*, vol. 11(1), pp. 63–69, 2005.

[Mic97] Microsoft, *PC 97 Hardware Design Guide*, ser. ISBN 1-57231-381-1. Microsoft Press, Redmond, Washington, USA, 1997.

[Mil06] A. Milenkovic, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring : Issues and an implementation," *Computer communications*, vol. 29, no. 13-14, pp. 2521–2533, 2006.

[Mon97] L. Monsein, "Primer on medical device regulation. Part II. Regulation of medical devices by the U.S. Food and Drug Administration," *Radiology*, vol. 205, no. 1, pp. 10–18, 1997.

[Obj09a] *OMG Unified Modeling Language (OMG UML), Infrastructure, Version 2.2*, http://www.omg.org/spec/UML/2.2/Infrastructure, Object Management Group, Inc. Std. OMG Document Number: formal/2009-02-04, February 2009.

[Obj09b] *OMG Unified Modeling Language (OMG UML), Superstructure, Version 2.2*, http://www.omg.org/spec/UML/2.2/Superstructure, Object Management Group, Inc. Std. OMG Document Number: formal/2009-02-02, February 2009.

[Oga97] M. Ogawa, T. Tamura, M. Yoda, and T. Togawa, "Fully automated biosignal acquisition system for home health monitoring," in *Proc. 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 6, 30 Oct.–2 Nov. 1997, pp. 2403–2405.

[Oga00] R. Ogawa and T. Togawa, "Attempts at monitoring health status in the home," in *Proc. Conference On Microtechnologies in Medicine and Biology, 1st Annual International 2000*, 12–14 Oct. 2000, pp. 552–556.

[Oga01] M. Ogawa, S. Ochia, K. Otsuka, and T. Togawa, "Remote monitoring of daily activities and behaviors at home," in *Proc. 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 4, 25–28 Oct. 2001, pp. 3973–3976.

[Oga03] M. Ogawa and T. Togawa, "The concept of the home health monitoring," in *Proc. 5th International Workshop on Enterprise Networking and Computing in Healthcare Industry Healthcom 2003*, 6–7 June 2003, pp. 71–73.

[Opp04] I. Oppermann, M. Hämäläinen, and J. Iinatti, *UWB Theory and Applications*, ser. ISBN 0-470-86917-8. Chichester, West Sussex PO 19 8SQ, England: John Wiley & Sons, Ltd, 2004.

[Ott06] C. A. Otto, E. Jovanov, and A. Milenkovic, "A WBAN-based system for health monitoring at home," in *Proc. 3rd IEEE/EMBS International Summer School on Medical Devices and Biosensors*, 4–6 Sept. 2006, pp. 20–23.

[Pak05] M. Paksuniemi, H. Sorvoja, E. Alasaarela, and R. Myllyla, "Wireless sensor and data transmission needs and technologies for patient monitoring in the operating room and intensive care unit," pp. 5182–5185, Jan. 2005.

[Pet92] D. Petersen, *Audio, Video and Data Telecommunications*. McGraw-Hill Companies, 1992.

[Pet07] L. L. Peterson and B. S. Davie, *Computer Networks: A Systems Approach*, 4th ed. Morgan Kaufmann, 2007.

[Pii07] K. V. T. Piipponen, R. Sepponen, and P. Eskelinen, "A biosignal instrumentation system using capacitive coupling for power and signal isolation," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 10, pp. 1822–1828, Oct. 2007.

[Pra03] A. Pras and J. Schoenwaelder, "Rfc 3444 - on the difference between information models and data models," Network Working Group, Tech. Rep., January 2003.

[Rey08] F. Reynolds, "Whither bluetooth?" *Pervasive Computing, IEEE*, vol. 7, no. 3, pp. 6–8, July-Sept. 2008.

[Ris06] H. Rissanen, K. Isokanniainen, P. Pirinen, and E. Alasaarela, "Wilho - a new concept of wireless management of healthcare processes," in *6th Nordic Conference on eHealth and Telemedicine (NCeHT)*, Helsinki, Finland, Aug. 2006, pp. 229–230.

[Rit02] J. Ritola, "Evaluation of a ballistocardiographic measurement chair," Licentiate's Thesis, Helsinki University of Technology, Department of Engineering Physics and Mathematics, P.O.Box 2200, FIN 02015 HUT, Finland, September 2002.

[RS-85] *EIA Standard RS-232-C Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial Data Interchange*, August 1969 ed., ser. reprinted in Telebyte Technology Data Communication Library. Greenlawn NY, 1985.

[Sch04] J. H. R. Schrader, E. A. M. Klumperink, J. L. Visschers, and B. Nauta, "Data communication in read-out systems: how fast can we go over copper wires?" *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 531, no. 1-2, pp. 221 – 227, 2004, proceedings of the 5th International Workshop on Radiation Imaging Detectors.

[Sch07] L. Schmitt, T. Falck, F. Wartena, and D. Simons, "Novel ISO/IEEE 11073 standards for personal telehealth systems interoperability," in *Proc. HCMDSS-MDPnP High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability Joint Workshop on*, 25–27 June 2007, pp. 146–148.

[Sid06] C. Sidebottom, H. Rudolph, M. Schmidt, and L. Eisner, "IEC 60601-1 - the third edition," *Journal of Medical Device Regulation*, pp. 8–17, May 2006.

[Sie01] E. L. Siegel and D. S. Channin, "Integrating the healthcare enterprise: A primer - part 1. introduction," *RadioGraphics*, vol. 21, pp. 1339–1341, 2001.

[Som00] I. Sommerville, *Software Engineering*, 6th ed. Addison-Wesley, August 2000.

[Sri05] V. Srivastava and M. Motani, "Cross-layer design: a survey and the road ahead,"

Communications Magazine, IEEE, vol. 43, no. 12, pp. 112–119, Dec. 2005.

[Sta02] V. Stanford, "Pervasive health care applications face tough security challenges," *Pervasive Computing, IEEE*, vol. 1, no. 2, pp. 8–12, Apr-Jun 2002.

[Tam95] T. Tamura, T. Fujimoto, H. Muramoto, J. Huang, H. Sakaki, and T. Togawa, "The design of an ambulatory physical activity monitor and it application to the daily activity of the elderly," in *Proc. IEEE 17th Annual Conference Engineering in Medicine and Biology Society*, vol. 2, 20–23 Sept. 1995, pp. 1591–1592.

[Tam98] T. Tamura, T. Togawa, M. Ogawa, and M. Yoda, "Fully automated health monitoring system in the home," *Medical Engineering & Physics*, vol. 20, no. 8, pp. 573 – 579, 1998.

[The05] The Global Harmonization Task Force/SG1, *Essential Principles of Safety and Performance of Medical Devices*, May 2005.

[The06] The Global Harmonization Task Force/SG1, *Principles of Medical Devices Classification*, June 2006.

[The08] The Global Harmonization Task Force/SG1, *Role of Standards in the Assesment of Medical Devices*, March 2008.

[Tog89] T. Togawa, T. Tamura, J. Zhou, H. Mizukami, and M. Ishijima, "Physiological monitoring systems attached to the bed and sanitary equipments," in *Proc. Annual International Conference of the IEEE Engineering in Engineering in Medicine and Biology Society Images of the Twenty-First Century*, 9–12 Nov. 1989, pp. 1461–1463.

[Tor03] D. M. Torre and S. M. Wright, "Clinical and educational uses of handheld computers," *Southern Medical Journal*, vol. 96(10), pp. 996–999, 2003.

[USB96] *Universal Serial Bus Specification*, USB Implementers Forum, Inc. Std., Rev. 1.0, January 1996.

[USB06] *On-The-Go Supplement to the USB 2.0 Specification*, USB Implementers Forum, Inc Std., Rev. 1.3, December 2006.

[USB07] *Universal Serial Bus Device Class Definition for Personal Healthcare Devices*, USB Implementers Forum, Inc. Std., November 2007.

[Var03] U. Varshney, "Pervasive healthcare," *Computer*, vol. 36, no. 12, pp. 138–140, Dec. 2003.

[Var06] U. Varshney and S. Sneha, "Patient monitoring using ad hoc wireless networks: reliability and power management," *IEEE Communications Magazine*, vol. 44, no. 4, pp. 49–55, April 2006.

[Var07] U. Varshney, "Pervasive healthcare and wireless health monitoring," *Mob. Netw. Appl.*, vol. 12, no. 2-3, pp. 113–127, 2007.

[War05] S. Warren, J. Lebak, J. Yao, J. Creekmore, A. Milenkovic, and E. Jovanov, "Interoperability and security in wireless body area network infrastructures," in *27th Annual International Conference of the Engineering in Medicine and Biology Society (EMBS)*

2005, 2005, pp. 3837–3840.

[Web94] *Webster's Encyclopedic Unabridged Dictionary of the English Language*. New York/Avenel, USA: Gramercy Books, 1994.

[Wie06] G. Wiederhold and E. H. Shortliffe, *Biomedical Informatics, Computer Applications in Health Care and Biomedicine*, 3rd ed. Springer New York, 2006, vol. Unit I, ch. System Design and Engineering in Health Care, pp. 233–264.

[Wil00] S. Williams, "IrDA: past, present and future," *Personal Communications, IEEE*, vol. 7, no. 1, pp. 11–19, Feb 2000.

[Win98] M. Win and R. Scholtz, "Impulse radio: how it works," *Communications Letters, IEEE*, vol. 2, no. 2, pp. 36–38, Feb 1998.

[Yao05] J. Yao and S. Warren, "Applying the ISO/IEEE 11073 standards to wearable home health monitoring systems," *Journal of Clinical Monitoring and Computing*, vol. 19, pp. 427–436, 2005.

[Yu95] X. Yu, D. Dent, and C. Osborn, "The selection of weights precision for ballistocardiography classification," in *Neural Networks 1995, IEEE International Conference on*, vol. 5, Nov/Dec 1995, pp. 2485–2489.

[Yu96] X. Yu, "Ballistocardiogram classifier prototyping with plds," *Electronics Engineering*, vol. 68, no. 834, pp. 39–42, June 1996.

[Zah04] T. Zahariadis, "Evolution of the wireless PAN and LAN standards," *Computer Standards & Interfaces*, vol. 26, no. 3, pp. 175 – 185, 2004.

[Zak09] M. Zakrzewski, S. Junnila, A. Vehkaoja, H. Kailanto, A.-M. Vainio, I. Defee, J. Lekkala, J. Vanhala, and J. Hyttinen, "Utilization of wireless sensor network for health monitoring in home environment," in *IEEE Symposium on Industrial Embedded Systems SIES 2009 - Special Session on Health-care*, July 2009.

[Zig05] *ZigBee Specification 1.0*, ZigBee Document 053474r06, ZigBee Alliance, Inc. Std., Rev. 1.0, June 2005.

[Zig09a] Zigbee Alliance, "New Zigbee health care profile: Helping people live healthier and independent lives," www.zigbee.org, March 2009.

[Zig09b] Zigbee Alliance, "Zigbee wireless sensor applications for health, wellness and fitness," www.zigbee.org/healthcare, March 2009.

[Zim80] H. Zimmermann, "OSI reference model – the ISO model of architecture for open systems interconnection," *IEEE Transactions on Communications*, vol. 28, no. 4, pp. 425–432, 1980.

Publication 1

S. Junnila and J. Niittylahti, "Implementing USB Function Devices," In *Proceedings of the 18th IASTED International Conference on Applied Informatics (AI'2000)*, Innsbruck, Austria, Feb 14-17, 2000, pp. 579-582.

IMPLEMENTING USB FUNCTION DEVICES

SAKARI JUNNILA
Tampere University of Technology
P.O.BOX 553, FIN-33101 Tampere, Finland

JARKKO NIITTYLAHTI
Tampere University of Technology
P.O.BOX 553, FIN-33101 Tampere, Finland

ABSTRACT

In this paper, we discuss the possible software and hardware implementation schemes of a custom USB peripheral. First, an introduction to USB and it's future extensions, including different transfer modes, is given. We consider USB's suitability in data acquisition and the complexity of the required design work. This includes discussion upon developing Windows 98 drivers to the host computer. It is shown that it is possible to implement a high-speed data acquisition system using USB and thus connect medical devices to a PC. We also discuss the driver stack related problems found in transferring low-latency data. A design case is included to illustrate the presented ideas.

KEY WORDS

USB, device implementation, PC interface, device driver

1. INTRODUCTION

USB (Universal Serial Bus) [1] is a standard which replaces old PC interfaces, such as serial and parallel ports. It provides means to connect peripherals to a PC in a flexible way. There are several methods to implement a device that can be connected to USB. A common misconception is to think USB only as an improved serial bus (RS-232), and to underestimate the effort needed to implement USB functionality. Altogether, there has been very little research published on USB so far. Some ideas and possible applications based on the features of USB have been presented, but research evolving actual implementations has mostly been about power management [2] and overcurrent issues [3].

As a concrete design case, we present a system for collecting measurement information from medical devices to a host computer through RS-485 and USB. The system reads framed data from the medical devices by polling them at certain time intervals, and extracts the relevant information, which is then sent to the host computer via USB. Similarly, the control commands from the host computer are processed and framed, and then sent to individual measurement devices via RS-485 network. We review the possible implementation schemes together with a short look on commercially available components and HDL solutions. As final step in the architecture design, we present the chosen implementation, which uses a Hitachi H8 microcontroller and a Lucent USS-820 USB device controller.

After the hardware design, we review the software development process and present different approaches for the system software design. This includes the use of a commercial USB transport-layer protocol module, and the traditional implementation method chosen by us. We also present the steps in the system software development and the encountered problems. Finally, we conclude with discussion about the influence of high speed and low-latency requirements of our design to the system software.

2. USB

USB is a communication protocol oriented serial bus. The physical USB wiring is simple, consisting of two differential data signals (D+ and D-), power (Vbus) and ground (GND) signals. Almost all features of the bus are implemented in the complex bus communication protocol. USB has only one host, which initiates all data transfers. In cases where a device wants to send data to the host, it will have to wait for the host to send a header frame on the bus, thereby specifying the device. In USB terminology, a device is either a hub or a function connected to the USB. A hub is a device which provides additional attachment points to USB, and a root hub is a special hub directly connected to the host. A function is a device which provides capabilities to the system. In this paper we concentrate on the USB function devices.

Current USB specification (revision 1.1) [1] defines a full speed signaling rate of 12 Mb/s and a low speed signaling rate of 1,5 Mb/s. The header and bit stuff overheads limit the actual transfer speed little under the theoretical maximum, approximately to 10 Mb/s. The target speed of the next USB revision (2.0) is planned to be 40 times greater than the current maximum, i.e., 480 Mb/s. USB 2.0 is promised to be forward and backward compatible with the current peripherals, working with the existing cabling and connectors. A USB 2.0 hub detects the speed of the devices connected to it's downports, and selects the port speed according to which the connected device supports. A USB 2.0 device needs be connected to a USB 2.0 hub to be able to work at the maximum transfer speed. USB 2.0 also adds a concept of microframe, which

will be 1/8th of the 1 ms frame used by USB 1.1 devices. At the higher USB 2.0 data rates this will allow the use of smaller buffers.

For different data transfer needs, USB provides four different transfer modes: Control, Bulk, Interrupt and Isochronous. Control transfers are mainly used in device configuration and control functions. Bulk transfers are typically used for larger data transfers, and have no guaranteed bandwidth. Interrupt transfers are best for small data transfers, with guaranteed maximum latency. Isochronous transfers are suited for continuous real-time steady-rate data transfers.

2.1 APPLICATIONS

There are some problems in using USB in low-latency real-time applications. This is due to the unpredictability of Windows NT (and 2000) timing [4][5]. The Windows NT ISRs (Interrupt Service Routines) are serviced at small latencies, but only a small part of the driver code is run at the ISR level. ISRs schedule the more complex and time-consuming routines at the next lower priority level in the DPC (Deferred Procedure Call) queues. It has been shown [4] that the timing of the DPC queues is quite undeterministic, and in a Pentium class computer there can be delays in magnitudes of tens of milliseconds. The bulk of USB driver code is run in the DPCs, thus limiting it's use in low-latency real-time systems.

USB was originally targeted to medium and low speed devices. Typical devices include PC's input, telecommunication and audio devices. USB 2.0 enables the addition of high-speed devices, which are required for fast internet connections and video applications, thus making the USB truly universal. The flexible design of the USB protocol, i.e., support for both steady-rate data streams and short bursty data transfers, also makes it very suitable for all kinds of data acquisition systems.

3. HARDWARE IMPLEMENTATIONS

USB has received strong support from component manufacturers, and there are several useful components available for use in USB function devices. In general, a USB function can be implemented in software or in hardware. Software implementations can be done using a microprocessor and interface logic, or by means of specific interface controllers or microcontrollers. Hardware implementations of USB functions can be done with FPGAs (Field-Programmable Gate Arrays) or ASICs (Application Specific Integrated Circuits).

The most popular way to implement a USB function device is to use a USB microcontroller [6], which is the conclusion one also comes to when looking at the available components. In a microcontroller-based system, the dedicated USB-core, built in the controller, handles some or even most of the USB's functionality. In addition, the interface between microcontroller software, i.e., the application, and the USB-core, is typically quite good. Different actions in USB-core can directly activate routines in the microcontroller software with extremely low latencies. The downside of this approach is that the choice of the microcontroller core itself is limited. The USB-microcontrollers are typically based on some of the most familiar cores like the 8051. However, adding USB functionality to a system built around some exotic microcontroller or even a DSP (Digital Signal Processor), is more problematic. One would be either facing a change of microcontroller architecture or building a dual-controller (multiprocessor) system. The change of architecture is costly as one must renew all the development tools, and lose all the engineering experience gained from the earlier microcontroller architecture. The addition of an extra microcontroller to a system only for USB functionality is also a very questionable choice. A USB-microcontroller is a practical choice when the core is powerful enough to handle the application software part as well as the USB functions. It is an extremely good alternative when the microcontroller architecture and the development tools are already familiar to the designers.

An interface controller is probably the second most popular choice for USB function devices. An interface controller is basically a USB core that is placed on its own chip. The chip cannot run software, instead, it's functionality is programmed and controlled by setting and reading register values in the core via an external microprocessor. The interface controller and the microprocessor typically communicate via a serial or a parallel bus. The choice between a serial and a parallel bus is often decided by available PCB (Printed Circuit Board) area. Serial bus communication is more common in low speed USB function devices.

One of the more time consuming ways to implement a USB function device is to use a USB transceiver chip and control it with the application microprocessor. This requires implementing the USB functionality with software, and thus requires quite a lot of work in getting familiar with the USB specification. It is difficult to implement complex device functionality and USB protocol on a single microprocessor. This kind of approach could suit applications, where the device is not connected to USB in normal operation mode, but where the USB is needed for device configuration or data retrieval purposes. One such system might be a measurement device which would collect data independently, and would be only connected to a PC for the purpose of downloading the collected data for further processing.

HDL models of USB function devices are also commercially available. HDL models are best suited for mass production devices and systems, where the different functions of the system are already integrated on ASICs.

3.1 TEST DESIGN

Our test design was based on a medical system which has measurement devices communicating via an RS-485 bus at 500 kb/s. The measurement devices are controlled by an embedded system which is responsible for framing the commands to the measurement devices, and de-framing the responses. The data-collecting device has also other functionality. From the current implementation we know that the microprocessor cores in commercial USB microcontrollers were not powerful enough or lacked some relevant features (A/D channels and timers among others). Thus, we used the Lucent USS-820 interface controller as the USB core, and the Hitachi H8-series microcontroller. The H8 was chosen because we had all the necessary development tools for it, including the processor emulator hardware. The block diagram of our test design is presented in Fig. 1.

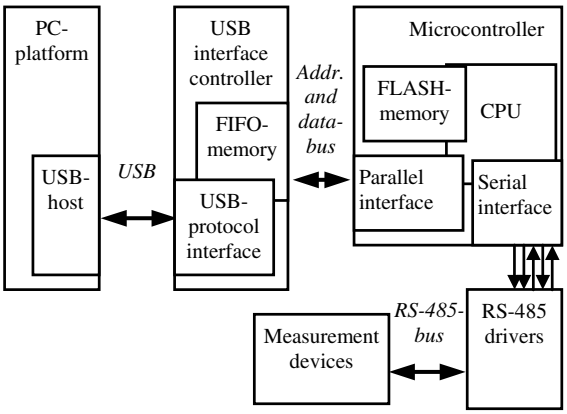


Figure 1. Block diagram of the test design

It is possible to implement the software for the USB function device without having to learn anything about the USB protocol itself. There are software packages [7] which allow the developer to link a software module to the application code, and use an API (Application Interface) for the USB communication. This kind of API solution is good for larger embedded systems, possibly running some RTOS (Real-Time Operating System). It allows one to change the hardware responsible of the USB communication, and still to use the same application software with changes only to the transport layer.

In our test device, we wrote all the USB functionality by ourselves. By far the most time consuming phase of the USB software design was the device enumeration. A bus analyzer would have been useful in this phase, but even an additional serial port for debug messages in our test device was a big help. The data transmit and receive functionality was not very difficult to implement. The USB software is mostly interrupt driven. The USS-820 has one interrupt signal which is raised when some non-masked interrupt inside the USS-820 arises. The interrupt signal is connected to H8's interrupt-in pin. This interrupt executes an interrupt handler that reads through the USS-820 interrupt registers to find out what interrupt occurred, and performs the appropriate action.

4. HOST SOFTWARE

We did our software development on Windows 98 OS, because Microsoft did not add USB support to Windows NT. USB drivers in Windows 2000 are based on the WDM structure used in Windows 98. For industrial applications, a stable and well-protected kernel is needed. The PC-based industrial applications of USB will probably run on Windows 2000, which is based on Windows NT kernel.

USB drivers are complex; Developing a USB driver from scratch without any prior knowledge of WDM (Windows Driver Model) or Windows NT device drivers is a demanding and time-taking task. Microsoft provides bulk and isochronous example drivers with the Windows 98 DDK (Driver Development Kit), which were used as base for our driver development. In our tests, we did not get the isochronous driver to work with data streams. The bulk driver worked well when transferring data at low rates, but as the driver was coded as non-reentrant, it would crash if back-to-back read and write sequences were issued too rapidly. We also found that we could do only one data transfer per frame for an endpoint, even if we had more transfers waiting and USB frame time available. The reason for this was later found to be in the UHCD.SYS (Universal Host Controller Driver) driver of Windows 98. It was implemented in a way that only allowed scheduling of one URB (USB Request Block) per pipe for a given frame. The Windows 98 Second Edition allows two URBs per frame. Thus a driver which uses many back-to-back reads or writes may not work as fast as expected. UHCD-drivers are used in systems which have USB hosts implemented in the UHCD-architecture. There is also an alternative OpenHCI-architecture (Open Host Control Interface). We did not have the chance to do any testing on an OpenHCI host.

Tools that assist in the driver development process are also available. These tools [8] generate the code required by the plug-and-play and initialization routines, leaving only the specific device functions to be coded into the driver. For devices which do not require complex or highly optimized driver functions, one can use one of the multi-purpose drivers commercially available.

In our test design, we would have wanted to read small-sized data packets on quite small time intervals. We implemented the system using WriteFile and ReadFile functions and bulk transfers, but found that USB stack caused too much overhead. Our solution was to use very light (2 byte) framing, and transfer several frames in one read or write operation. The additional framing and de-framing overhead in host and device end was still significantly less than that of the USB stack in the

previous case. The downside was the additional latency in data transfer between PC application and the data-collecting device.

5. CONCLUSIONS

We have found the USB function device implementation to be relatively easy. There is a wide spectrum of different components on the market, allowing USB to be added to most devices. USB does not significantly increase the hardware design complexity. The software design task is somewhat more difficult, depending on the implementation architecture. The design process is easier if the enumeration code is available from the component manufacturer. The most complex part is often the device driver development in the host system. In a USB function device development project, one most likely needs two separate teams: a hardware/embedded systems group designing the device, and a software group that will implement the device driver.

Real-time systems which require data transaction between software running on a PC and a USB device at very low-latencies are difficult to implement in Windows environment. In Windows environment, USB is well suited for data-acquisition systems which have only soft real-time requirements, such as our test design.

REFERENCES

[1] Universal Serial Bus Specification, Revision 1.1, Sep 1998.

[2] K. Lynn, Universal Serial Bus (USB) Power Management, *Proc. of Wescon/98*, 1998, 194-201

[3] P. Wiener, J. Witham, Meeting USB and IEEE1394 overcurrent protection requirements using PolySwitch devices, *Proc. of Wescon/97*, 1997, 442-475

[4] A. Baril, Using Windows NT in Real-time Systems, *Proc. of the Fifth IEEE Real-Time Technology and Applications Symposium*, 1999, 132-141.

[5] M.B. Jones, J. Regehr, The Problems You're Having May Not Be the Problems You Think You're Having:Results from a Latency Study of Windows NT, *Proc. of the Seventh Workshop on Hot Topics in Operating Systems*, 1999, 96-101

[6] T. Hathaway, V. Verma, Advanced Low Power System Design Techniques using the Universal Serial Bus Microcontroller, *Proc. of Northcon/96*, 1996, 270-274

[7] Moore Computer Consultants Incorporated, *A Guide to the MCCI USB Data Pump*, 1999, http://www.mcci.com/mcci/datapumpguide.shtml, (2000, February 10)

[8] Blue Water Systems, *WinDK Version 2.7 User Manual*, 2000, http://www.bluewatersystems.com/products_windk.html, (2000, February 10)

Publication 2

S. Junnila and J. Niittylahti, "A Patient Monitoring System Based on Standard PC Platform," In *Proceedings of the 15th International EURASIP EuroConference BIOSIGNAL 2000*, Brno, Czech Republic, June 21-23, 2000, pp. 348-350.

EuroConference BIOSIGNAL 2000
Brno, Czech Republic

A PATIENT MONITORING SYSTEM BASED ON STANDARD PC PLATFORM

Sakari Junnila, Jarkko Niittylahti
Digital and Computer Systems Laboratory, Tampere University of Technology
P.O.BOX 553, FIN-33101 Tampere, Finland
e-mail: sakari.junnila@tut.fi

Summary: This paper presents the use of PC-platform in patient monitoring application, where commercially available measurement instruments are connected to a standard PC. The interface media used for connecting the instruments was Universal Serial Bus and its characteristics were compared to other available PC interfaces. Finally, a prototype for the system was built.

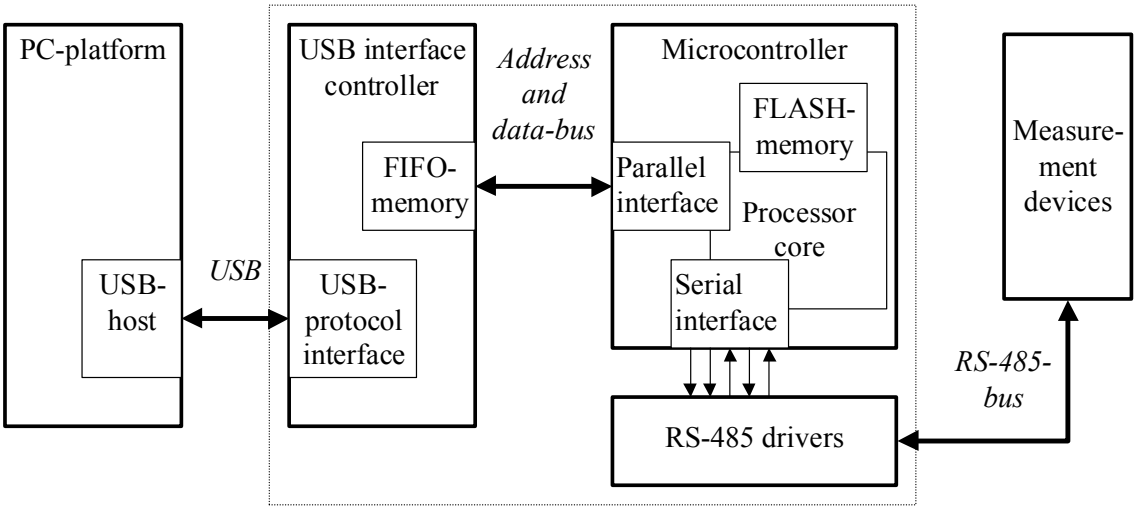
Custom-made computer systems have been a common solution in industry and medical applications until recently. As the home computer market has grown, the cost-effectiveness of the PC technology has improved. The PC's now have tremendous computing power which allows the use of more complex software algorithms instead of dedicated hardware solutions. Standardized application interfaces allow developers to use multipurpose GUIs like web browsers for their user interfaces. All this has lead to the move from custom-made computer systems to more generic PC-based platforms [1]. PC-based monitoring systems can also be used in home follow-up systems [2], where the ease of measurement device interfacing is a great factor.

The structure of a typical patient monitoring systems resembles a PC-platform in many ways: both have a display unit, a central processing unit, and interfaces for peripherals. A modern PC has different interfaces for connecting devices and peripherals. The internal computer buses like ISA and PCI are buses that are mainly used for connecting PCB-cards, and connecting devices to them usually requires opening the computer casing. In medical applications, some isolation circuitry is needed between the bus connection and peripherals that are used outside the computer casing. This is due to dangers of static charges and different electronic potentials. In traditional PC-based monitoring solutions AD-converter or other similar data acquisition cards are used [3]. In this paper, we present a system that connects measurement instruments to a standard PC platform, and we propose Universal Serial Bus (USB) to be used as connection interface.

Our approach was to keep the system as simple as possible, so it was more appropriate to use the standard interface connectors supplied by the PCs, which are designed to be more robust than the internal computer buses. Currently selling computers usually have RS232C serial port, Enhanced Parallel Port (EPP) and USB interfaces, and maybe also IEEE 1394 (a.k.a. Firewire) in near future. USB offers greater speeds than RS232C, and USB is more flexible and robust than the EPP. Also, EPP allows only one device to be connected, whereas USB can handle up to 127 devices, which leaves room for future system extensions. The IEEE 1394 is more complex and thus more expensive than the USB, and the additional speed it offers is not needed in our application. IEEE 1394 is also quite rare in current PC platforms and it is not required by the current PC-platform standards (http://www.pcdesguide.org). Among these facts, USB interface was found best suited for our monitoring application.

We have implemented a prototype system of the proposed system (Fig. 1). The prototype used an architecture where one device collects the measurement information from several measurement devices using a robust RS-485 bus. Each measurement device has it's own device number, and responds only to messages sent to that number. Because of this, all the modules can be connected to the same bus. The data from all the measurement devices is sent via USB to the PC. Both the PC and data collecting device have a dynamic memory structure where the information collected from the measurement instruments and the command needed to control them are stored. If a section of this memory is changed, the changed part is sent via USB to the other device e.g. the PC or the data collection device.

USB devices can be implemented in several ways, using FPGAs and ASICs, specific microcontrollers, interface controllers or by programming the USB functionality with software [4]. We built our prototype system using a register controlled interface controller that automates most of the protocol functions needed by the USB. The interface controller is supervised by a microcontroller via an 8 bit parallel bus. The microcontroller software handles the measurement instrument communication and frame en-/decoding along the USB functions. RS-485-interface is directly controlled by the microcontroller and it's serial communication module via appropriate bus driver circuitry.



The PC-platform chosen was a typical 266 MHz Pentium II system with USB. Because Microsoft did not add USB support to Windows NT 4.0, and the Windows 2000 was not then available, we had to use Windows 98 as our operating system for the software development and tests. The device drivers were implemented in the WDM architecture, and the GUI for our test application was done with Visual C++.

Based on its specification, the USB was found well suited for our application. During the prototype design and testing, we discovered that the implementation and specification of the USB were not always the same. One of the bigger problems found was that the Windows 98 USB stack could not deliver bulk-transfers to one endpoint more than once a frame, which contradicts the specification. The feature was partially fixed by Microsoft in Windows 98 Second Edition, which allows two transactions per frame. The OS USB stack was more complex than expected, and seemed to slow down the machine noticeably when loaded heavily (one bulk transaction initiated every frame). The interrupt- and isocronous transfers OS implementations are somewhat different that one could assume after reading the

specification. The stability of the operating system in error situations was found inadequate for critical medical applications.

A natural continuum of the research would be to connect measurement instruments directly to PC via USB, and thus eliminating the data collection device, e.g., using architecture similar to one presented by Varman [1]. This would require the redesign of the measurement instruments so that they could interface directly to USB. Compared to typical prices of measurement instrument devices, an additional USB chip would not have great effects on the final product pricing. Currently the data collection device is responsible for the isolation of the PC and the measurement instruments, so for such a system a USB hub-device with data and power isolation would most likely be required.

We also found that bigger data blocks can be transferred quickly via USB using the bulk transfers and the implementation in software and hardware is quite simple. USB is thus well suited as a download medium with data recording devices that currently use the RS232C serial bus.

Conclusions: A PC-platform was found suitable for certain patient monitoring applications. Because of safety and reliability reasons a custom monitor system will still have to be used in critical applications, but a PC-solution could be applied in some less-critical bedside monitoring systems. USB could also be used to relay commands to other independent patient care systems like infusion pumps. The current bottleneck in the PC-based monitoring system was found to be the device driver. A fast (low-latency) and reliable device driver is difficult to program because of the complex USB stack in the current Windows operating system. Hopefully the Windows 2000 will have a better USB stack than the Windows 98. The hardware implementation is in no ways OS dependent, so a future Linux implementation is possible, as USB support is already available in the current test releases of Linux kernels. As Linux kernel is available in source mode, it is possible to make a system based on it very stable and safe. USB itself is a robust bus that is quite well suited for medical applications. There are many commercial components available for USB, and device implementation is not very difficult.

References:

[1] Varman M, Ferris CD, Niccolai M, Varman M: “A Knowledge-based Design for a Computerized Compact Clinic for a Family Physician,” in Proc. *16th Southern Biomedical Engineering Conference*, 1997, pp. 46-49.

[2] Morlion B, Verbandt Y, Paiva M, Estenne M, Michils A, Sandron P, Bawin C, Assis-Arantes P: “A Telemanagement System for Home Follow-Up of Respiratory Patients.” *IEEE Engineering in Medicine and Biology*, July/August 1999: 71-79, 1999.

[3] Grasenick K, Maresch H, Pfurtscheller G, Trajanoski Z, Wach P, Kotanko P, Skrabal F: “Multiparametric PC-Based Monitoring System and Its Application in Studies of Cardiovascular Regulation,” in *Proc. 16th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 1994, pp. 926-927 vol.2.

[4] Junnila S, Niittylahti J: “Implementing USB Function Devices,” in Proc. *18th IASTED IC on Applied Informatics (AI’2000)*, 2000, to appear.

Publication 3

S. Junnila, J. Ruoho, and J. Niittylahti, "Medical Isolation of Universal Serial Bus Data Signals," In *Proceedings of the 9th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2002)*, Dubrovnik, Croatia, Sep 15-18, 2002, pp. 1215-1218.

Medical Isolation of Universal Serial Bus Data Signals

Sakari Junnila, Jarkko Ruoho, and Jarkko Niittylahti

Tampere University of Technology
Institute of Digital and Computer Systems
P.O.BOX 553, FIN-33101 Tampere, Finland
sakari.junnila@tut.fi, jarkko.ruoho@probatus.fi, jarkko.niittylahti@tut.fi

ABSTRACT

Modern PC-based patient monitoring systems are modular, where serial bus connects the measurement units and the PC. The bus is usually a derivate of RS-485 or RS-232 serial buses. Universal Serial Bus (USB) will probably replace RS-485 and RS-232 in some new applications. The USB specification does not specify anything about isolation, which is often required in medical communication applications. In this paper, we consider different isolation alternatives and present our implementation of medical isolation of USB data signals. We also discuss the problems we observed with maintaining compliance with the USB specification.

1. INTRODUCTION

10 years ago, a typical PC-based medical system consisted of a PC, an A/D conversion card, and an analog measurement system. Isolation was implemented either in the A/D conversion card or with an analog isolation amplifier between the A/D card and the measurement system [1, 2, 3, 4]. The analog isolation systems usually used transformers, which led to frequency constraints and meant that DC-signals could not be transmitted as such. As the standard PC interfaces developed faster, interest to use them instead of special interface cards grew. Possibility to use digital transfer was discussed by Metting van Rijn, Peper and Grimbergen [1], and use of parallel port by Mak, Escalona and Anderson [3]. Most modern systems implement the A/D conversion as close to the measurement as possible, and transfer the measurement data in digital form, which is much less susceptible to noise than the analog signals [5]. It would therefore be logical to use the fast standard PC peripheral interfaces to connect measurement devices to PC-based medical systems. For this purpose some kind of isolation of the interface is needed.

USB was not originally designed to be isolated, and thus it uses signalling that is more difficult to isolate than normal (RS-232 type) serial communication, which consists of only two different voltage levels. USB includes two data lines, a power line (+5 V), and a ground line [6]. The bi-directional half-duplex data lines use mixed differential and single-ended signaling. The data lines have both AC- and DC-information. Power isolation is not discussed in this paper as it can be implemented by using standard components. The isolation circuitry generates some power loss, which has to be taken into account when calculating the device power consumption and writing the device descriptors (the device information headers sent to the host when device is connected to the bus). The previous applies only to bus powered devices, which take their power from the USB.

In Section 2, we consider potential isolation design options. In Section 3, we present our solution. The results are analysed in Section 4. Finally, conclusions are drawn in Section 5.

2. ISOLATION

Medical isolation implies that there is no direct electrical connection between the patient and a device monitoring the patient. Medical isolation is used to improve patient safety and the quality of the measurement. Measurement quality is improved by the isolation by the removal of common mode noise and ground loops that could otherwise impair the measurement results. A design feature to be considered is that all measurement units should be isolated from each other to minimize patient leakage currents and from mains power supply to prevent electrical shocks under device failure situations [7]. Isolation between a medical device and the mains power supply must stand at least 4000 Vrms at 50 or 60 Hz for one minute, and the clearance between isolated sides must be 8 mm on the surface or 5 mm in the air [7]. This makes the components of the isolation barrier relatively large. For example, optoisolators with 5000 V isolation voltage are usually packaged in DIP-8 cases. Therefore, the number of optoisolators is an important factor when comparing the size of isolation circuitry in different approaches.

Many measurement applications require electrical isolation. Traditional industrial field buses are often serial, and isolation has been considered in their spec-

ification. USB interface components are very highly integrated, and thus the isolation cannot be done within the bus controller, at least when standard bus interface components are used. The cost of designing a specific isolated USB interface components is too high for low volume applications.

2.1. Optical isolation

Optical isolation can be done using optoisolators. An optoisolator consists of a light emitter and a detector. Information is transmitted as a light signal through the isolation barrier, without electrical contact. Typically, a light emitting diode (LED) is used as the light emitter and a phototransistor as the detector. In an optoisolator, these two components are coupled in one physical component. For bi-directional data transfer, two optoisolators are needed, one for each direction.

Isolation of USB with optoisolators has one major problem, the detection of the transmitted data. In slower isolation applications, one can detect the transfer direction and activate the corresponding optoisolator accordingly. However, the speed of USB 1.1 in fast mode, 12 Mbs, is too fast for simple microcontroller-based direction detection.

2.2. Transformer isolation

Transformer isolation is a very traditional approach. Typically, it is implemented with a transform ratio of 1:1. The information is transmitted as magnetic field over the isolation barrier. Transformers do not relay DC signals. This is a problem considering constant voltage levels and power lines. Transformers also cause some EMI and add a small capacitance over the isolation barrier.

2.3. Capacitive isolation

Capacitive isolation is based on the use of capacitors, which allow high frequencies pass through them. This feature can be used to filter out low frequency components, including DC-currents and 50 Hz power signals. However, capacitive isolation cannot be used as medical isolation, because fast, possibly high amplitude changes are able to pass through the capacitor.

3. IMPLEMENTATION

There are three possible approaches to implement the isolation. The first one is the isolation inside a hub. It would be easy to place optoisolators in the signal path between the hubs logical function and the downstream USB transceivers. This is because the signals inside a hub have normal CMOS logic levels. Unfortunately, all available USB hub circuits have transceivers integrated in them. This problem could be bypassed by implementing hubs functionality on an ASIC or an FPGA and using separate transceivers.

The second approach is to do the isolation on the USB cable. This can be done either using an analog discrete component circuit with optoisolators; using standard USB transceivers and logic level optoisolators; or using transformers and some additional logic. Unlike the transformer approach, these two active designs need some extra logic to tell which direction the data is moving in the bus.

The third possibility is to implement the isolation inside the USB device itself. Devices usually implement the USB connection with a separate USB device controller, e.g., Lucent's USS-820, or embedded in a microcontroller, e.g., Cypress EZ-USB series. These devices have embedded transceivers, and thus the isolation can only be implemented between the USB controller and the rest of the device. Full speed USB devices typically have parallel data-bus interfaces to the system, which are expensive to isolate. USB uses both single-ended and differential signalling, which makes it difficult to isolate using only one type of isolation and simple circuitry. Specification also requires that devices do not take over 100 mA from the bus when connected to USB.

3.1. Isolation implementation

Our implementation approach was based on the USB cable isolation alternative. This was described in the previous paragraph as the second alternative. We implemented the isolation at the device end, which gave us the possibility to use extra signals to communicate with the device. These extra signals simplified the isolation implementation, as can be seen later. We based our isolation on transformer components with additional optoisolators to handle special signals. It was considered to be the most inexpensive alternative, and the implementation price was one important design constraint. The implementation is similar to the one presented by Ghalary and Webster for their electrical impedance tomograph (EIT) [8]. The schematic for the implementation is presented in Figure 1.

The capacitors C_1 - C_4 prevent the saturation of the transformers and allow the open circuit voltage levels to be within the USB specification. Transformer 1 (Trans 1) carries the D+ signal over the isolation barrier, and transformer 2 carries D-. The optoisolator Opto 1 informs the USB device that it has been connected to the bus with the *device connect* signal of the USB controller, in our case. Optoisolator Opto 2 is used to inform the host that the connected device is a USB fast type device, by pulling up the D+ signal with a pull-up resistor R_1, as described in the specification [6]. Optoisolator Opto 3 relays low frequency signals over the isolation barrier. These are unable to pass through the transformer and the capacitor. This mainly means the 0-volt signal used for device reset. The diodes D_1 - D_4 clamp high and low voltage peaks, and thus help in keeping the off-load voltages of the signals in specified limits.

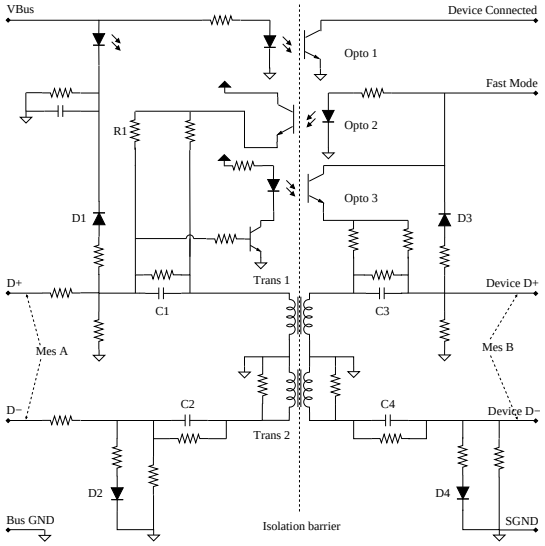


Fig. 1. Schematic for the isolation board.

3.2. Testing environment

Testing formed an important part of the design process. It was also a key part in comparing the design to USB specification. Our testing environment consisted of an P600 desktop PC with Windows 2000 OS, an USB-SUPI interface card developed in our previous research [9], a Hitachi H8-emulator connected to a P200 PC (Win98), and an USB isolation board. The USB-SUPI interface card is a Hitachi H8 processor card with Lucent USS-820 USB device controller. It was previously developed for interfacing patient monitoring system modules to USB, where the card is used to poll the modules and perform a protocol conversion. The card allows easy access to all USB related signals, which made it ideal for the isolation circuitry design and testing. The isolation board was connected to the host side signals of the USB-SUPI board, thus bypassing the original USB type-B connector. During the testing, we used a non-invasive blood pressure (NIBP) module (made by Datex-Ohmeda) connected to the USB-SUPI interface card to generate test traffic to USB.

4. RESULTS

We performed our measurements at two points. These points are the USB device connector before the isolation circuitry (Mes A) and the input pins of the device controller after the isolation circuitry (Mes B), as shown in Figure 1. Figure 2 shows the differential data signal (difference of D+ and D- signals) sent by the host PC at the USB cable device end before it has gone through the isolation circuitry. Figure 3 shows the data sent by the host PC after the isolation, at the data pins of the USB device controller. The data transmission from the device to the host PC is depicted in the next two Figures 4 and 5. Figure 4 shows the data sent by the device controller before the isolation, and Figure 5 shows it after the isolation.

The USB specification limits the cable delay to a maximum of 26 ns [6]. From Figure 6, it can be seen that the downstream delay is at maximum 15 ns. The upstream delay can be seen from Figure 5, when looking at the zero-crossover point of the signal, because the measurement was triggered from the measurement point B before isolation. The maximum upstream delay is about 45 ns. These delays were measured with a 3 m USB cable excluding the cable delay itself. Although the upstream delay was over the specification limit and the signal is clearly distorted, our test application (the NIBP module) worked with a 3 m USB cable. Longer, 5 m maximum length USB cable did not work, there were too many errors in transmission, which even crashed the computer regularly.

The design was not tested against the requirements of the IEC 601 standard. However, we used a winding with similar electrical characteristics to that of a winding that is approved for medical isolation transformers. Also, discrete component optoisolators approved for medical isolation are also available.

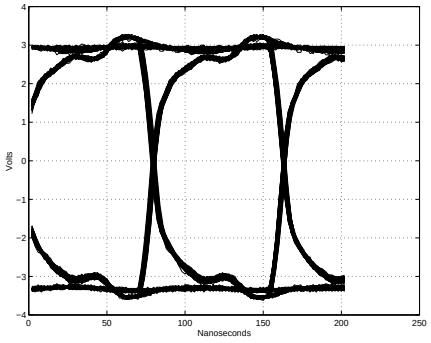


Fig. 2. Downstream data before isolation (Mes A).

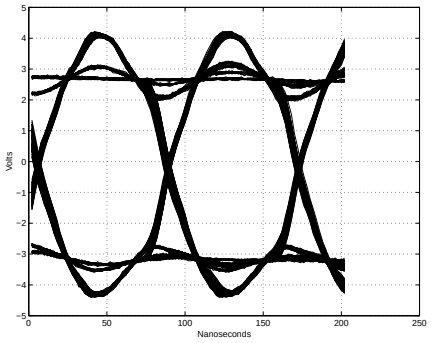


Fig. 3. Downstream data after isolation (Mes B).

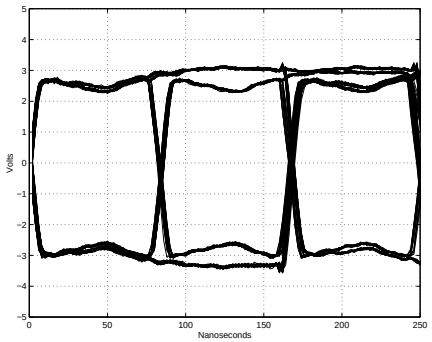


Fig. 4. Upstream data before isolation (Mes B).

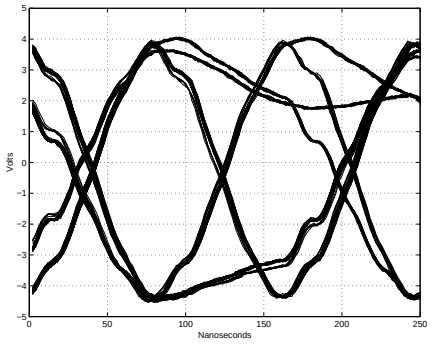


Fig. 5. Upstream data after isolation (Mes A).

5. CONCLUSIONS

In this paper, we presented ideas on how to implement medical isolation of USB 1.1 for use in patient monitoring applications. We briefly described three different approaches on how to implement the data isolation of USB, and then chose one approach and implemented it. The implementation was described, including the schematic of the implementation. Finally, we presented measurement results of the prototype implementation, which showed that our implementation works, although the signal is degraded especially at the device to host communication direction. The additional delay caused by the isolation circuitry shortens significantly the maximum length of the USB cable it is connected to. Measured upstream delay exceeds the specification limit, and the system does not therefore fulfill the USB specification requirements. We conclude that as such the developed system can be used in closed custom systems that utilize USB components and protocol, but are fitted with non-USB connectors.

6. REFERENCES

[1] A.C. Metting van Rijn, A. Peper, and C.A. Grimbergen, "The Isolation Mode Rejection Ratio in Bioelectric Amplifiers," *IEEE Transactions on Biomedical Engineering,* vol. 38, iss. 11, pp. 1154–1157, Nov. 1991.

[2] C. Housworth, D. Cook and E. Schroeppel, "A PC-based Pacing System for Collection of Intracardiac Electrograms," in *Proc. Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society*, Orlando, FL, pp. 1672-1673, Nov. 1991.

[3] M.H. Mak, O.J. Escalona, and J. Anderson, "PC-Based Cardiac Electrophysiological Studies: A Pacing System with Real-Time Screen Monitoring of Endocardial Electrograms," in *Proc. Computers in Cardiology*, Indianapolis, IN, pp. 389-392, Sept. 1996.

[4] R.A. Malkin, "Constructing a Multichannel Electrocardiography System From a Few Standardized, High-level Components," *IEEE Engineering in Medicine and Biology Magazine,* vol. 17, iss. 1, pp. 34–38, Jan.-Feb. 1998.

[5] M.K. Leverich, J.L. Silberberg, and S. Weininger, "Design and Development of a Multi-Channel Signal-Acquisition System for Recording an Apnea Waveform Database," in *Proc. IEEE 7th Symp. on Computer-Based Medical Systems*, Winston-Salem, NC, pp. 213-216, June 1994.

[6] Universal Serial Bus Specification, Rev. 1.1, Sep. 1998, http://www.usb.org.

[7] IEC 60601-1, Medical Electrical Equipment Part 1, General Requirements for Safety, 2nd edition, Dec. 1988.

[8] A. Ghahary and J.G. Webster, "Electrical Safety for an Electrical Impedance Tomograph," in *Proc. 11th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society*, Seattle, WA, pp. 461-462 vol.2, Nov. 1989.

[9] S. Junnila and J. Niittylahti, "Implementing USB Function Devices", in *Proc. 18th IASTED Int. Conf. on Applied Informatics*, Innsbruck, Austria, pp. 579-582, Feb. 2000.

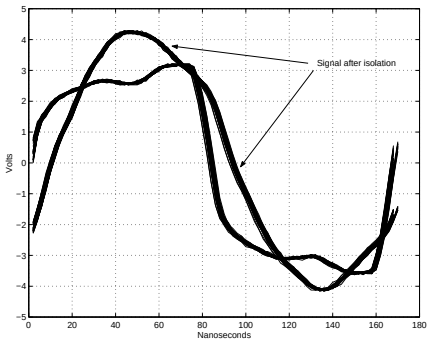


Fig. 6. Delay caused to the downstream data.

Publication 4

S. Junnila and J. Niittylahti, "Use of Bluetooth in Medical Systems," In *Proceedings of the 19th IASTED International Conference on Applied Informatics (AI'2001)*, Vol. 1, Innsbruck, Austria, Feb 19-22, 2001, pp. 488-494.

USE OF BLUETOOTH IN MEDICAL SYSTEMS

SAKARI JUNNILA and JARKKO NIITTYLAHTI
{sakari.junnila, jarkko.niittylahti}@tut.fi
Digital and Computer Systems Laboratory
Tampere University of Technology
P.O.BOX 553, FIN-33101 Tampere, Finland

ABSTRACT

In this paper, we discuss the suitability of Bluetooth to hospital environment, and describe possible applications for Bluetooth in medical systems. Our focus is on patient monitoring systems and communication networks between them. Bluetooths specification and features related to throughput, performance and interference are presented. Specified performance is compared to results from other Bluetooth performance studies. Communication setup and delays involved are discussed. We present the modern hospital environment, and discuss the issues the new wireless technologies may affect. Finally, we view the Bluetooth application areas in wireless medical systems.

KEYWORDS

Bluetooth, Hospital, Medical devices, WLAN.

1. INTRODUCTION

Bluetooth [1, 2] is a new wireless communication medium, which is designed to replace low-speed cable connections and enable ad hoc networking in electronic systems. In a hospital environment, where medical information and measurement systems are used, reduction in the number of cabling would be appreciated. However, there are still many situations where cable-based communication is still preferred over the wireless ones.

This paper presents the features of Bluetooth we find important in the design considerations of medical systems. We also look at work done by other researchers in this field [3, 4, 5, 6], and compare their results. Based on the Bluetooth features and our knowledge of medical systems and hospital environment, we present possible application areas for Bluetooth, and discuss the problems involved.

2. BLUETOOTH

Bluetooth [1, 2] is a short range radio communication network that was developed by the Bluetooth Special Interest Group (SIG). It was designed to replace low-speed cable connections and enable ad hoc networking in electronic systems. It works on the 2.4 GHz Industrial, Scientific and Medical (ISM) band and uses frequency hopping. In most countries, the frequency band is from 2400 MHz to 2483.5 MHz, but due to national limitation, some countries have a more limited band [1]. The design considerations and results presented in this paper focus on the full band Bluetooth system. Because Bluetooth devices designed to work in countries with a limited band will not work with devices that implement the full band, the Bluetooth SIG is working on harmonizing the Bluetooth band worldwide. The frequency band is divided into 79 or 23 RF channels spaced 1 MHz apart. 79 channels are used with the full 83.5 MHz bandwidth. The nominal frequency-hopping rate is 1600 hops/s, and a channel slot is thus 625 µs. Bluetooth uses a slotted time division duplex (TDD) scheme for full duplex transmission. The data transmission rate of Bluetooth is 1 Mb/s, which is the theoretical maximum data-rate for one channel. In theory, the whole Bluetooth spectrum has a capacity of 79 Mb/s, but because the channel hop sequence is non-orthogonal, the theoretical capacity cannot be reached. The required and nominal operating range of a Bluetooth device at the lowest power level is 10 m, with 1 mW (0 dBm) maximum transmitting power. At the highest power mode a Bluetooth device can transmit at 100 mW (20 dBm), which should give a range of up to 100 m.

Bluetooth supports point-to-point and multipoint connections. When two Bluetooth devices come to contact, they form a piconet. In a piconet, one of the devices is a master and others are slaves. Any Bluetooth device can become a master and communication can only be done between the master and a slave. Besides the master, a piconet can have maximum of seven active slaves. In addition, the piconet may contain slaves that are parked. A parked slave listens periodically the piconet communication for a wake-up instruction from the master. It can also send a message to the master if it wants to initiate an unparking procedure.

In Bluetooth, the payload data can be transferred in two different transfer types; synchronous connection-oriented (SCO) link and asynchronous connectionless link (ACL). SCO link can be considered as a circuit-switched connection between the master and a slave. It is a symmetric, point-to-point link between the master and one or more specific slaves. Master supports maximum of three SCO links in one piconet. A slave can handle three

links to a one master or two SCO links to two different masters. SCO packets are supposed to be used only for voice data, and contain no error checking mechanism or packet retransmission.

ACL link provides a packet-switched connection between master and all active slaves in the piconet. Only a single ACL link can be active between a master and a slave. Unlike SCO, which only supports single slot packets, ACL packets can also extend over 3 or 5 slots. A slave can return a packet to the master only if it was addressed in previous master to slave time slot. The packets may contain error checking and retransmission mechanisms. ACL link is the initial link between two devices. After it is made, SCO links can be established.

For power saving purposes Bluetooth has three useful modes; hold, sniff and park mode. In hold-mode, the ACL link is put on hold for specified time. It is typically used when there is no need to send data for certain time. The device behavior during sleep mode is not specified, e.g., the device can for example contact another piconet while in hold-mode. In sniff mode, master and slave negotiate a sniff interval and a sniff offset. After that, the slave only listens the channel at sniff interval periods, e.g., the master can only transmit to the slave at sniff intervals. The master can force a slave in to sniff mode, or slave or the master can request it.

Park-mode is used, when the slave does not have to participate in the channel, but it still should stay frequency-hop synchronized (FHS). When a slave is put in park mode, it loses its piconet address (active member address, AMADDR) and is assigned a unique parked member address (PMADDR), which can be used by the master to wake up the slave. The parked slave listens the channel at predefined time intervals for master commands. Like sniff-mode, the master can either force or request a park for a slave, or the slave may request a park mode.

2.1 COMMUNICATION SETUP DELAYS

Information itself does not care how it is transmitted. However, compared to cable based communication, wireless links have many issues, which have to be taken into account when designing systems based on them. There may be delays in setting up communication and the link in more susceptible to interference. The environment the device is used in affects the link performance.

In Bluetooth, the beginning state of devices is a standby state, where no information of the surrounding devices is available. In [1] it is stated, that in error free environment a device will have to spend 10.24 s to find all the possible new devices. After the inquiry, the device knows the identities of the devices that are around it and may attempt to establish communication links with them with the paging procedure. In a relatively static configuration, like a measurement session where Bluetooth enabled measurement devices are placed around a patient monitor, the master does not have to repeat the inquiry process if communication links would happen to disconnect. In this case, the information about the device identity can be used to re-establish the communication link with the paging procedure.

The connection establishment has yet another delay. A device does not listen all the time for Bluetooth paging messages. Normally, a unit that is in idle mode (standby state) wants to sleep most of the time to save power. It only wakes up at sleep time intervals T for 10 ms. T can range from 1.28 to 3.84 s [2]. Every time the unit wakes up, it scans at a different hop carrier. The Bluetooth wakeup hop sequence is 32 hops long and cyclic. The paging unit transmits cyclically an approximately 10 ms sequence in which 16 hop carriers are visited for duration of the sleep time. If that does not contact the idle unit, then the paging unit advances in the hop sequence, taking in to account that the sleeping unit also advances one hop carrier. Therefore, if the paging unit knows the identity of the sleeping unit it wishes to contact, then the maximum access delay will be twice the sleep time [2]. If the free running clocks in the devices are still synchronized, e.g., not much time has elapsed after the disconnection of the devices, then the average response time is reduced to half of the sleep time [2]. If a device does not know the identity of a device that it wishes to contact, then it has to use inquiry messages to find out what devices are present. This is the case when a new network in made, where as the previous case resembles a situation where interference has disconnected an existing connection.

In the standby state, the unit only scans for about 10 ms every couple of seconds. This gives a duty cycle under 1 percent. Once a connection is established, the unit can be put in park-mode, where the unit only listens to the channel at an even lower low duty cycle. In [2], it is estimated that a slave should only have to listen the channel approximately 126 µs. The beacon interval has a range of 0 to 2^{16}–1 slots [1]. In addition, a sleep interval, which is a multiple of the beacon interval, can be defined. Using the sleep interval, a parked unit can be set to listen to the channel only once in every 2^{24}–1 slots, almost once in three hours. With large sleep intervals, the power consumption can be decreased, but the response time of the device becomes too long for any dynamic and interactive system.

2.2 PERFORMANCE AND THE EFFECT OF INTERFERENCE

There are different data packet types available for ACL (Table 1). There is also a seventh ACL packet type, AUX1, which is DH1 packet without CRC and a 30 byte data payload. All DH and DM packets have CRC. The DM packets have simple 2/3-rate block code for data payload FEC, while DH packets are uncoded. The

maximum packet lengths are approximately 250 µs shorter than the multiple of the slot sizes to allow for the transmitter/receiver direction change.

Table 1. Properties of ACL packets.

Type	User payload (bytes)	Symmetric Max. Rate (kb/s)	Asymmetric Max. Rate (kb/s)		Slots	FEC
			Fw	Rev		
DM1	0-17	108.8	108.8	108.8	1	2/3
DH1	0-27	172.8	172.8	172.8	1	no
DM3	0-121	258.1	387.2	54.4	3	2/3
DH3	0-183	390.4	585.6	86.4	3	no
DM5	0-224	286.7	477.8	36.3	5	2/3
DH5	0-339	433.9	723.2	57.6	5	no

There are four different SCO link packet types, from which three are presented in Table 2. The fourth packet type is a combined voice and data packet DV. SCO packets are supposed to be used for voice data, and as such don't offer CRC and are never retransmitted. All the HV packet types offer 64 kb/s bi-directional transmission. It is noticeable that one HV1 link occupies the whole capacity of a piconet, HV2 occupies half of the capacity, and a HV3 link occupies one third of the capacity [5].

Table 2. Properties of SCO packets.

Type	User payload	Symmetric Max. Rate (kb/s)	Slots	FEC
HV1	10	64.0	1	1/3
HV2	20	64.0	1	2/3
HV3	30	64.0	1	No

In [5] Bluetooth radio network performance issues are studied using simulation models. In the simulations, varying number of one master and one slave piconets were placed in the same room. The simulations showed that in ACL links co-channel interference is much more dominate factor than interference caused by normal environment variables. In addition, the data payload errors were more probable than error in packet headers or access code. The main result of the simulations was that the throughput degradation caused by concurrent sessions is very small. According to the simulations, the DH5 packet type offered the best throughput even in interference situations. For SCO packets the simulations showed that HV1 packets cause the most interference to concurrent piconets. HV2 packets cause only half, and HV3 packets one third of the interference caused by HV1 packets.

In [6], the simulations were made to estimate Bluetooth performance in piconet and scatternet situations. One of the results was that multi-slot packets suit bursty traffic cases better, as they allocate bandwidth more dynamically. The simulation also showed that even with maximum number of active slaves and sending over two radio hops simultaneously, the delays in Bluetooth should be in order of hundreds of milliseconds with low to moderate loads and bursty traffic. From the figures and results presented in [6] estimation that with multislot packets and moderate loads average delays in scatternet should be less than 100 ms can be drawn.

In [3], actual performance measurements of Bluetooth in industrial environment were made. The study focused on bit errors in the data payload. The throughput and packet error rates were calculated based on the bit error rates measured. A very interesting result in the paper was that even in office environment with line-of-sight between two Bluetooth devices, the packet error rate rises sharply when the distance between devices increases from 4 to 6 meters. The DH packets suffer much more from bit error, and dramatic drop in throughput can be seen as the distance increases from 4 meters. The DM packets seem to work much better. In rough industrial environment line-of-sight tests, the bit error rate was noticeably higher, and the packet error rate was found to increase rapidly between 5 to 10 m even with DM1 packets (Table 3).

Table 3. Throughput of symmetric DM packets (kb/s) in rough industry environment test. [3]

Distance	DM1	DM3	DM5
5 m	101.5	166.1	130.5
10 m	66.7	12.1	1.1
15 m	30.6	0.0	0.0
Specification	108.8	258.1	286.7

In the same test situation, the SCO link quality was found to be acceptable for voice up to 10 m with HV1 packets, and up to 5 m with HV2 packets. Transmitting error sensitive data with SCO packets in these conditions would cause severe damage to the data. One of the conclusions drawn by the authors was, that that Bluetooth will not reach the specified throughput values (Tables 1 and 2), and that DH packets aimed at high data rate transmissions will only work at distances up to four meters. The authors also note that in non-line-of-sight conditions the performance is reduced in very unpredictable way.

It is noticeable that the simulations [5] give contradictory results compared to actual measurements [3]. Simulations suggest that DH5 packet should be used to get the best possible throughput, whereas measurement with real hardware shows the DH packets, especially DH3 and DH5, becomes almost unusable at distances over 5 m.

Frequency hopping used in Bluetooth has many benefits: low power density, data spread over large bandwidth, resistance to interference, jamming, multipath and interception. It makes Bluetooth more robust and less susceptible to interference. In [3], it is claimed that in a research done by Haartsen and Zürbes for Ericsson, it was shown that presence of direct sequence (DS) WLAN reduces Bluetooth throughput in the area of 22%. Similar

results have been obtained in simulations [4], where white noise, fading, multipath and inband interference effects on Bluetooth were estimated. The results were that (DS) WLAN degraded the packet throughput approximately 20-30 % depending on the noise power. The effect of different concurrent piconets was approximately 10-15 % degradation. The effects of white noise, fading, and multipath interference were very similar to each other. As the noise power relative to the signal power grows, the throughput decreases quickly, which would be expected. The paper also compared the bit error rate of the packets that were received. The affect of the WLAN to the bit error rate was found to be smaller than concurrent piconets, due to smaller power spectral density.

Microwave ovens operate also in the same band as Bluetooth, but the interfering bandwidth is smaller than the ISM band used by Bluetooth. Therefore, only a part of Bluetooth hop frequencies are affected. Use of microwave ovens near Bluetooth devices only degrades the throughput.

A Bluetooth device has a link supervision timeout counter, which can be set in the link manager protocol (LMP) [1]. The timeout value can be zero to 2^{16}–1 slots, e.g., the maximum duration of interference that blocks all Bluetooth traffic without a link disconnect is approximately 41 seconds. The default value for timeout is 20 s [1]. The timer is reset when a device receives a valid packet. In addition, the link management protocol will determine a link terminated, if response to a link management packet is not received in the maximum response time of 30 seconds. Therefore, a Bluetooth communication link should be able to stand short communication block outs without problems, if set up properly.

3. HOSPITAL ENVIRONMENT

Computer networks and wireless communication have become common in modern hospitals. A modern hospital environment (Fig. 1) typically has an internal local area network (LAN) to connect information systems and possibly patient monitors together. The LAN can be used to transfer information between a centralized patient monitoring station and individual patient monitors. A hospital may also have a wireless patient telemetry system, which gives patients more freedom for mobility while being monitored. Currently, most patient telemetry systems are sold as separate systems, having their own infrastructure and information database. A hospital also has patient care units that work outside the hospital, typically medical crews working in ambulances or helicopters. Communication between these units and the hospital is typically based on mobile phone networks. Mobile phone approach has limitations in data speeds, network coverage and reliability, and because mobile networks were designed to be used at ground level, some problems may occur with helicopter use. In addition, many hospitals have internal wireless telephone networks.

The next step in hospital communication development is the addition of wireless LAN (WLAN). The most likely places where WLAN will be used are handheld patient record interfaces that doctors and nurses can carry around the hospital and a patient monitor to LAN interfaces. The latter would allow the patient monitor to move with the patient, and remove cabling used to connect a patient

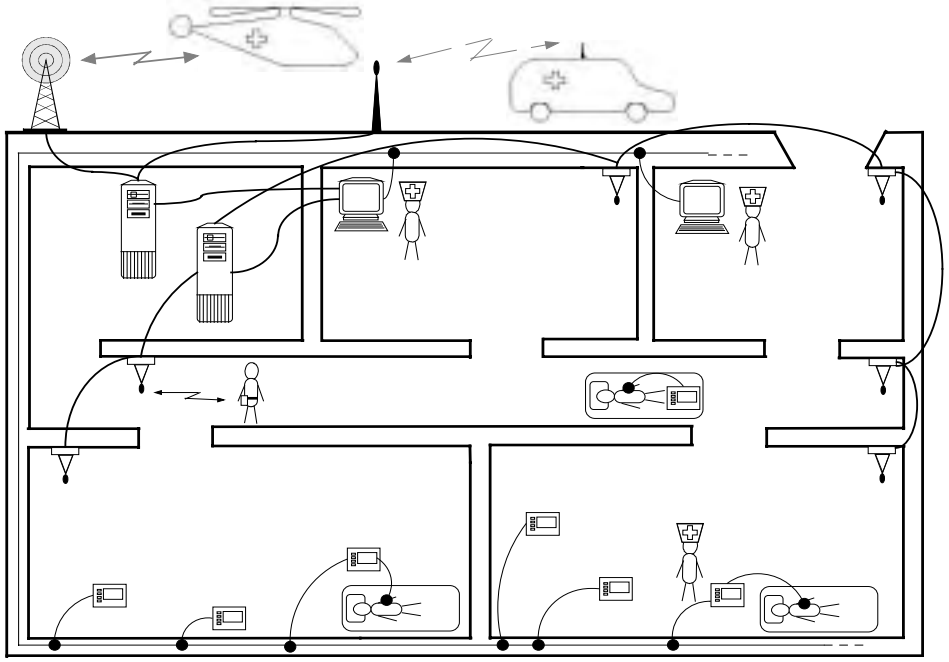


Figure 1. Modern hospital.

monitor to the LAN. The question that is addressed in this paper is should a wireless interface such as Bluetooth be added to an environment, which already has WLAN and possible an other wireless communication system for the patient telemetry.

3.1 PATIENT MONITORING SYSTEMS

A patient monitoring system is a system which measures parameters from the patient, uses different algorithms to calculate some physical values based on the measurements, and finally displays the values to the monitor user. A monitor can work as a stand-alone device or as part of a network of monitors where measurements from one monitor can be viewed on other monitor or in a centralized monitoring room.

A patient monitor measures different kinds of information. Some of it is very timing critical, for example heart rate measurement that is used to synchronize the defibrillator to the patients cardiac rhythm has very strict timing requirements. Other information, like EEG or body temperature, does not suffer severely from short communication breaks. Transfer buffers enable short communication breaks without data loss. Patient monitor may also handle patient records, i.e., what drugs are given to the patient or what treatments have been ordered. This kind of information has very loose timing requirements.

In general, the communication between measurement instruments of a patient monitor and the data processing unit is more critical than the communication to the user interface (Fig. 2). This is because the person who operates the patient monitor responds with a random delay in order of seconds, so small delays in data transmission are in most cases negligible. Functions that require critical timing are controlled directly by the data processing part of the monitor.

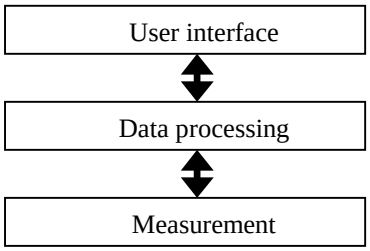


Figure 2. Communication in patient monitors.

4. BLUETOOTH MEDICAL APPLICATIONS

If the physical problems are left aside, there are some application areas for a dynamic radio link based interface such as Bluetooth. If we consider the modern hospital presented earlier (Fig. 1) we find several application areas for a mobile radio interface. First, when the patient is checked in to the hospital, a nurse fills the patients information in patient database. In the wireless hospital (Fig. 3), the nurse could have a Personal Digital Assistant (PDA) type of device, which could be used to input this information. The link to the patient database could be implemented with Bluetooth. This would probably be a cheaper way than a WLAN based system. If there is a hospital wide WLAN network installed, then the access to patient information database is preferable to do with WLAN.

A critically ill patient is usually placed in bed, and a patient monitor is connected to monitor him. In the wireless hospital, the patient monitors would have WLAN interface to the LAN-network, and they could be moved with the patient around the hospital, without having to connect/reconnect the patient to instruments. Ideally, this LAN-interface would be well suited for Bluetooth. Often the data is patient record information, or distant monitoring information, which does not have very strict timing requirements. The problem is that to guarantee the LAN access, Bluetooth LAN access points should be situated very near to each other. WLAN has typically been used for LAN access purposes in hospitals, but for application with low bandwidth requirements Bluetooth could be applied instead of WLAN. If the hospital has an existing WLAN infrastructure, adding a parallel Bluetooth network would not be practical. Bluetooth may suffer from longer communication breaks when the devices are moved around the hospital, due to slow communication setup procedures. The benefit of Bluetooth based access from patient monitor to patient records would be the flexibility in the communication medium between the record database and a Bluetooth equipped monitor. LAN access described before would be the most common inside the hospital, but outside a monitor could also connect to Bluetooth equipped mobile phone, and relay the data to the database via mobile phone network. The big problem of wireless patient monitors is of course the power supply, as fully equipped monitor usually uses quite much power for measurements. Outside the hospital and WLAN networks, in ambulances and helicopters, portable lightweight low-power monitors are already used. Mobile light patient monitors could benefit from Bluetooth.

In measurement devices, Bluetooth would replace the cable connection to the patient monitor. The battery-powered operation limits the use to measurement instruments that can function with low power. Ideally, wireless patient monitor with wireless Bluetooth instruments would be just what the hospital staff would like to have, as all the extra cables around the beds would be removed. With Bluetooth, the initial problem would be setting up the monitor environment. If there are several monitors next to each other, the staff has to manually set what instruments belong to what monitor. After the monitor configuration is set up, the communication itself could be implemented quite robustly: the system should

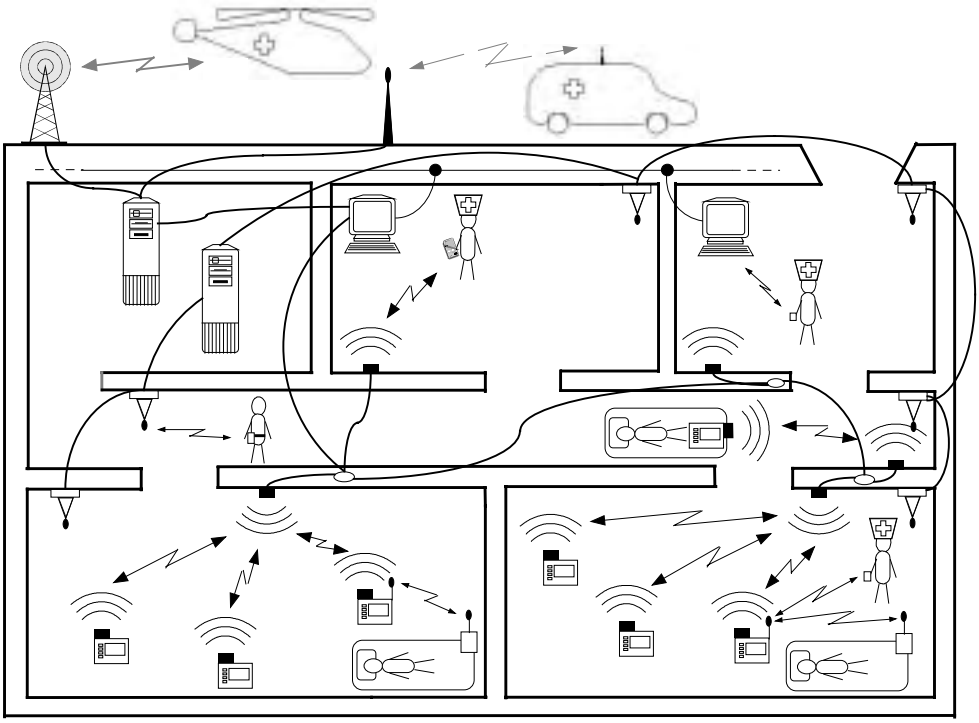


Figure 3. Wireless hospital.

recover easily from short communication block outs, and radio interference will only degrade the throughput. The data link between the monitor and a measurement instrument is not as reliable as cable based, and the timing is looser, which limits the possible applications. Possible Bluetooth based measurement applications could be body temperature and blood pressure measurement.

There are also many other devices around a patient monitor, which are not directly measurement instruments of the patient monitor, but which activities should be stored in to patient records. These devices are often from different manufactures, and lack common interfaces. Bluetooth could be used to collect information from these devices via a patient monitor to the patient records.

A major possibility for Bluetooth use would be in implantable measurement devices. A small, low power Bluetooth enabled device could measure few parameters implanted in the patient and the measurement results could either be monitored in real time or downloaded for processing periodically. The functionality of implantable devices could also be controlled outside the body.

Perhaps the most realistic Bluetooth application area in the near future would be a wireless user interface to a patient monitor and possibly further to patient records. Similarly to patient check-in situation, the nurse could carry with her a PDA-type device. When placed near a patient monitor, it would download the monitor user interface from the monitor dynamically, and then offer the services needed to control the monitor. The PDA-based user interface would also give the possibility to remove the physical user interface parts of the monitor, and thus make patient monitors more compact and portable. A patient monitor in usually connected to LAN, and thus to patient records, so in the absence of hospital wide WLAN-network, the PDA could be used to access patient records through the monitor. The patient monitors could then be thought as patient record access points. If the hospital were to have a WLAN-network installed, the WLAN would be the medium used to access the patient records. In WLAN environment, Bluetooth could be used to identify the patient. When the nurse would walk near to a patient connected to a patient monitor, the PDA would automatically open the patient record for that patient.

Finally, Bluetooth could be used as a personnel-tracking system. In a hospital equipped with Bluetooth enabled patient monitors, which are connected via LAN together, it would be possible to see from any patient monitor what Bluetooth devices are connected to it. If the hospital staff would have a unique Bluetooth device that they carry with them, they could be located always when they are next to a patient monitor.

5. CONCLUSIONS

The Bluetooth data transmission is very robust. Because of the small transmitter power, the performance degrades quickly as the distance between two units increases over a certain limit. For medical use, the complex communication setup, which involves random delays and unpredictable communication network structure, make the Bluetooth unsuitable for critical applications. The data transfer performance itself is

relatively good for a radio link, and Bluetooth would probably work in some static point-to-point applications, but much simpler solutions are available for this use. Environment equipped with WLAN decreases the Bluetooth throughput, but does not prevent its use.

The most promising application for Bluetooth in hospital environment would be a PDA based remote interface or database access device. It could also be used as low bandwidth LAN access point. Many of the application areas for Bluetooth are alternatives to existing WLAN-based solutions. Possibilities in implantable devices are interesting.

6. FUTURE WORK

There would seem to be a lack of real hardware measurements of Bluetooth performance. More detailed measurements, similar to work done in [3], are planned. Also, the authors would like to research more on WLAN and Bluetooth co-operation. Actual measurements with different WLAN systems are planned. On the medical device side, a prototype of Bluetooth PDA interface to patient monitoring systems is been looked at.

7. ACKNOWLEDGEMENTS

The authors would like to thank Mr. Kimmo Frondelius and Mr. Markku T. Roiha from Datex-Ohmeda for their valuable comments and ideas concerning current medical systems and Bluetooth application areas. Additional support was obtained from Nokia, VLSI Solution, and the National Technology Agency of Finland (Tekes).

REFERENCES

[1] Specification of the Bluetooth System, v. 1.0B, Dec 1999. (www.bluetooth.com)

[2] J.C. Haartsen, The Bluetooth Radio System, *IEEE Personal Communications*, 7(1), 2000, 28-36.

[3] U. Bilstrup and P-A. Wiberg, Bluetooth in Industrial Environment, *Proc. IEEE Int. Workshop on Factory Communication Systems*, 2000, 239-246.

[4] G. Koomullil, Performance Analysis of Bluetooth Systems in the Presence of AWGN, Fading, Multipaths, and Inband Interference, *Proc. Int. Conf on Consumer Electronics*, 2000, 294 - 295.

[5] S. Zürbes, W. Stahl, K. Matheus, J. Haartsen, Radio Network Performance of Bluetooth, *Proc. 2000 IEEE Int. Conf on Communications*, 2000, *1563-1567*.

[6] P. Johansson, N. Johansson, U. Körner, J. Elg, G. Svennarp, Short Range Radio Based Ad-hoc Networking: Performance and Properties, *Proc. 1999 IEEE Int. Conf on Communications*, 1999, 1414-1420.

Publication 5

S. Junnila and J. Niittylahti, "Wireless Technologies for Data Acquisition Systems," In *Proceedings of the International Symposium on Information and Communication Technologies (ISICT 2003)*, Dublin, Ireland, Sep 24-26, 2003, pp. 132-137.

Wireless Technologies for Data Acquisition Systems

Sakari Junnila and Jarkko Niittylahti
Institute of Digital and Computer Systems
Tampere University of Technology
P.O.Box 553, FIN-33720 Tampere, Finland
sakari.junnila@tut.fi, jarkko.niittylahti@tut.fi

ABSTRACT
This paper surveys currently available wireless technologies and considers their suitability for a six channel medical (BCG/ECG) measurement system. The main requirements for the wireless technology suitable for this application are capability to transmit data stream up to 60 kbps, low-power consumption, and low-interference to and from other devices.

KEY WORDS
Wireless Technologies, Data Acquisition, Medical engineering

1 Introduction

Wireless technology is replacing cables and generating completely new applications in industrial, medical, and other applications. New wireless technologies are designed to support media applications relaying audio and image data. Older wireless systems were usually designed for control applications with small data throughput. Data acquisition systems require more bandwidth than simple control systems, but they don't usually need speeds nearing 1 Mbps offered by the newer standards. In this paper, we survey the current technologies available for *mid-speed* wireless data acquisition systems.

Wireless technology for industrial sensors and wireless interfaces for IEEE 1451 sensor networks has been discussed by Brooks and Gilsinn et al. [1, 2]. Wiberg et al. presented survey of wireless industrial applications in 2001 [3]. Zahariadis et al. compared different technologies available for in-home broadband systems [4]. Davies compared Bluetooth to other Local Area Network (LAN) standards [5]. Most of the recent work has been on using Bluetooth or Wireless LAN (WLAN) technology for sensor (data acquisition) applications.

In the next section, we will shortly summarise the current status of European free frequency bands, e.g., what parts of the RF band can be freely used. The third section presents available technologies for these bands. In the fourth section, we present our application and discuss possible implementations. Finally, conclusions are drawn.

2 SRD frequency bands

The use of frequency bands in Europe is coordinated by the European Conference of Postal and Telecommunications Administrations (CEPT). It currently has 45 member countries. The telecommunications issues are handled by the European Radiocommunications Committee (ERC).

Table 1: Non-specific short range devices from 433 MHz to 5,9 GHz in Europe

Frequency range	Transmission power (ERP)	Channel spacing	Duty cycle
433,05-434,79 MHz (e) (ISM)	$\leq$ 25 mW		$\leq$ 10 %
433,05-434,79 MHz (e1)	$\leq$ 1 mW		No limit
434,04-434,79 MHz (e2)	$\leq$ 10 mW	Max 25 kHz	No limit
868,00-868,60 MHz (f)	$\leq$ 25 mW		$\leq$ 1 %
868,70-869,20 MHz (g)	$\leq$ 25 mW		$\leq$ 0,1 %
869,30-869,40 MHz (h)	$\leq$ 10 mW	25 kHz	No limit
869,40-869,65 MHz (i)	$\leq$ 500 mW	25 kHz	$\leq$ 10 %
869,70-870,00 MHz (k)	$\leq$ 5 mW		No limit
2400,0-2483,5 MHz (l) (ISM)	$\leq$ 10 mW		No limit
5725,0-5875,0 MHz (m) (ISM)	$\leq$ 25 mW		No limit

ERP = Effective Radiated Power
ISM = also designated as Industrial, Scientific and Medical (ISM) bands as defined in ITU Radio Regulations

The ERC recommendation (70-03) for Non-specific Short Range Devices (SRD) from 433 MHz to 5,9 GHz is summarized in Table 1 [6]. From Table 1, it can be seen that most of the sub 1 GHz bands are quite narrow or have narrow channels, or have tight duty cycle limits, which limits their use in faster and continuous data acquisition applications. This also partly explains why the 2,4 GHz band is currently so popular.

3 Wireless technologies

The wireless technologies offering data rates starting from few kilobits per second (kbps) upwards typically work in 433 MHz, 868 MHz or 2,4 GHz ISM bands. In addition, DECT has its own frequency range in the 1,8 GHz band (1880-1900 MHz). There is also an Ultra Wideband (UWB) technology, which uses a very wide band overlapping other bands. At lower frequencies, single channel transmission with FSK, ASK, or FM-modulation is typically applied, whereas in higher frequencies frequency hopping and spread spectrum multi-channel technologies are utilized.

3.1 Worldwide open standards

Bluetooth (www.bluetooth.org) is the leader in Personal Area Networking (PAN). It has a relatively complex protocol stack, which requires additional hardware resources to run. Available free protocol stacks are Linux-based, thus requiring a system capable of running Linux on the HW-platform. Commercial protocol stacks are available for smaller embedded systems.

One of the most used Bluetooth profiles so far has been the RFCOMM profile, which emulates a serial port interface. Hardware modules that provide the user with an RS232 (or RS485) interface while transmitting data wirelessly via Bluetooth are available from several manufacturers. These modules offer simple solution for wireless Bluetooth-based data transfer, which is easy to implement on top of an existing serial port based communication.

Zigbee (www.zigbee.org) was designed to be a low cost alternative to Bluetooth. The physical communication is developed as the IEEE 802.15.4 standard. Zigbee protocol is less complex than Bluetooth, estimated to be about 10 % of the size of a Bluetooth stack, although it also has

simple networking functionality. Transmission speeds are slower, but the range is larger. Zigbee has operating modes for three different frequency bands, from which the 868 MHz and 2,4 GHz bands can be used in Europe. The first radio chips using the IEEE 802.15.4 standard in 868 MHz were available at the end of 2002, and 2,4 GHz chips should come to market during 2003. The Zigbee protocol is expected to be completed in the beginning of 2004.

Wireless Local Area Networking (*WLAN, RLAN, Wi-Fi*) can be considered a mature wireless technology. Current development in WLANs is focusing on faster systems and use of higher (5 GHz) frequency bands. For wireless links, the speed of the much used IEEE 802.11b (11 Mbps) is more than adequate, making it one of the fastest wireless mediums on the licence free frequency bands.

HomeRF (www.homerf.org) Working Group disbanded in January 2003. HomeRF was targeted for home consumer devices, with an emphasis on voice and data communication.

3.2 Other technologies

WirelessUSB is a market name for a 2,4 GHz RF technology, which was designed by Cypress Semiconductor (www.cypress.com) to connect a wireless keyboard and mouse to an USB interface. The maximum data transfer rate is 217,6 kbps (half-duplex). WirelessUSB is not an open standard like USB, but it uses USB drivers in the PC end.

Spike is a technology developed by Eleven Engineering (www.elevenengineering.com) for wireless video game controllers. Like Zigbee, the standard has implementations for more than one RF band. The 900 MHz devices are already being produced, and 2,4 GHz devices should come to market during 2003.

Nanonet is the market name for a technology developed by a German-based company Nanotron Technologies (www.nanotron.com). NanoNET TRX is the first chip for data transmission from Nanotron using their new Multi Dimensional Multiple Access (MDMA) technology. The chip operates in the 2,4 GHz band. Volume production is expected to begin during 2003.

RF-232 is a market name for a technology developed by AeroComm (www.aerocomm.com), who manufactures temperature sensors with a 2,4 GHz FHSS transmitter.

The Digital Enhanced Cordless Telecommunications *DECT* (www.dect.ch) standard also includes digital data communication. DECT has its own reserved frequency band, which increases reliability due to minimal interference from other devices. DECT requires radio infrastructure (basestations), and as such is expensive for plain point-to-point applications. Due to long range and high transmission power, the commercial DECT HW-modules consume noticeably more power than short-range devices, e.g., Bluetooth.

Table 2 summarises the above wireless technologies.

3.3 UWB

Ultra-wideband (*UWB*) (www.uwb.org) devices operate by employing very narrow or short duration pulses that result in very large or wideband transmission bandwidths. Traditional radio systems see the UWB signal only as background noise, which should not interfere with their transmissions. It has been suggested that this may not be the case when operating in close proximity to UWB transmitters, but research in this area still suffers the lack of UWB devices. The Federal Communication Commission (FCC) has permitted the marketing and operation of certain types of devices incorporating UWB technology in the USA [7]. However, in Europe the regulatory process is only in the preliminary stages, and the final approval of the UWB technology might take place in the 2011 WRC at the latest. The basic problem with UWB

Table 2: Summary of wireless technologies

Technology	Transfer rate	Operation band (Table 1)
Spike	75.06 / 35.88 kbps (up/down)	l (and 915 MHz)
WirelessUSB	217,6 kbps	l
Zigbee	250 kbps	l, f (and 915 MHz)
DECT	552 kbps	- (reserved band)
Bluetooth	768 kbps	l
Nanonet	2 000 kbps	l
WLAN, IEEE 802.11b	11 000 kbps	l
WLAN, IEEE 802.11g	22 000 kbps	l (802.11b backward compatible)
WLAN, IEEE 802.11a	54 000 kbps	m

standardization is that UWB overlaps existing frequency bands, and it is impossible to use the traditional standardization method of allocating a small area of RF band to UWB use. Instead, UWB should be allowed to operate over the existing frequency bands with only some limitations on the signal strength in different parts of the frequency band. The main issue for the standardization body is to decide the signal strength limits in this *spectrum mask*. One problem is that in the RF band there are parts for which no transmission is allowed (art. 5.340).

3.4 RF transceivers

We use the term *RF transceivers* to describe small integrated transceiver chips, for which the user develops application specific protocol. Some of the radio parameters may be configurable either by hardware or by software. New transceivers are coming to market on daily basis, but transceivers capable of 50+ kbps data throughput in accordance with the ERC recommendations are not common. Table 3 presents three transceivers currently available.

In the 433 MHz band, a transceiver that uses more than 25 kHz band has to either transmit at 1 mW / 0 dB or at 25 mW with 10 % duty cycle (Table 1). RF Monolithics has developed a technology, which is able to transmit at high speeds with only 1 mW. Other transceivers tend to operate between 1 and 10 mW. In the 868 MHz RF band, the 869,7-870 MHz band is the only one enabling fast continuous communication, but none have yet surfaced. In the 2,4 GHz band, the focus has been on Bluetooth radios, but now the RF transceiver market seems to be growing. Motorola RF Data Modem is actually an implementation of the IEEE 802.15.4 specification, e.g., Zigbee radio, and it will support Zigbee network layers when they are finalised. Similar products (IEEE 802.15.4 based RF transceivers) are available from other manufactures.

Table 3: Fast RF transceivers currently available

Manufacturer and model	Frequency (band)	Transfer rate	Power supply and transmission current
RF Monolithics TR3100	433,92 MHz (e1)	576 kbps	2,2 - 3,7 V / 10 mA
Nordic VLSI NRF2401	2,4 GHz (l)	1 Mbps (burst) 10 kbps (cont)	1,9 - 3,6 V / 10,5 mA
Motorola RF Data Modem	2,4 GHz (l)	250 kbps	2,0 - 3,6 V / - (NA)

4 Data acquisition application and technology selection

The target application for our research is a ballistocardiography/electrocardiography (BCG/ECG) measurement device. The device will support at maximum 6 BCG/ECG channels with 16 bit resolution and 500 Hz sample rate. The measured signals are weak and sensitive to interference, requiring the device to be electrically isolated. This is achieved by making the device battery powered, which leads to requirements of low-power consumption on behalf of the RF circuitry. For EMC reasons, RF transmission power should be low. Higher operating speeds lead to more complex electrical design and increased interference. In order to minimise interference to the sensitive analogue measurement electronics, lower RF frequencies are preferred.

Adding RF circuitry to systems is not easy, it often requires knowledge and expertise. The more complete solution one can buy, the easier the design is. Lower frequency RF is in general easier to implement. One should carefully estimate the required throughput, and start searching for possible technologies from the lower end of the RF band. When making the final choice of the wireless technology, one should consider:

- Target environment - Is the use of RF bands limited, are some bands more preferable because of regulatory issues.
- The amount of integration - RF module requiring only external antenna vs. plain chip which requires board design and additional electronics.
- Physical chip/module parameters - Power consumption, operating voltage, interface to the rest of the system.
- How much does the chip/module handle the protocol - Implementing the protocol (e.g., Bluetooth higher layers) needs resources.
- Software support from the manufacturer.
- Price, availability, alternative components if the technology is discontinued.

Our target is a medical data acquisition system, which is to be used in hospitals. Use of ISM bands simplifies the acceptation process. Using a technology, which has already been accepted by the hospital (DECT and Bluetooth in our case) would make the acceptance even more simpler. We have limited resources for the RF circuitry design, so we prefer either complete RF module or highly integrated RF transceiver with an example design and evaluation/example board available. Power consumption has to be minimised. The low-level radio protocol should be implemented by the chip or by accompanying software. Finally, there should be an alternative component from other manufacturer which can replace the original RF with reasonable effort (second source).

Wireless medical applications require data encryption. With open standards, this is more relevant because data are transmitted using standard protocol. Some standards include data encryption. Custom systems implemented with RF transceivers are more difficult to listen, provided that the protocol is not made public. In general, UWB and frequency hopping technologies are more difficult to listen than single channel transmissions, especially if the hopping secuence is unknown (non-standard).

4.1 Choice of RF technology

The 433 MHz is a commonly used frequency band. It includes some high-power long-range systems among other traffic, so interference from other systems is likely. As explained in the end of the previous section, implementing mid-speed (and faster) continuously transmitting

systems in the band is difficult. One of the most promising possibilities is the new technology RF transceiver series developed by RF Monolithics.

The 868 MHz band has some interesting technologies, but the regulations are such that 50+ kbps continuous throughput is difficult to achieve.

The 2,4 GHz band offers several competing technologies. Speed is not an issue, most of the technologies offer more than adequate throughput. Possible alternatives include Bluetooth, Zigbee and 2,4 GHz RF transceivers. For Bluetooth chips, development boards and software are commercially available, but the technology itself is an overkill for most point-to-point data acquisition applications. Zigbee may be the choice of the future, but the protocol is still in development. Radio modules using the Zigbee (IEEE 802.15.4) standard and other RF transceivers provide light and cost-effective solutions. Frequency hopping used by most of the systems, the width of the 2,4 GHz band, and higher attenuation of 2,4 GHz signals combined with low transmission power reduce interference problems in the band.

Our first prototype is to be implemented with Bluetooth. Hardware modules and software support are available from several manufactures and there is enough speed to experiment with different protocol approaches (bursty vs. continous).

5 Conclusions

The current wireless market is very active. New technologies and techniques are introduced on a monthly basis. For mid-speed continuous data acquisition applications, the choices are limited, and suitable sub 2,4 GHz technologies for ISM bands are rare. If higher throughput is needed, then 2,4 GHz technologies are usually needed. Bluetooth is not the only choice in the 2,4 GHz band, but it can be considered as the best compromise for many applications. It has also been accepted for use in many hospitals along with DECT. Zigbee may replace Bluetooth in new mid-speed applications when completed. UWB technology acceptance in Europe is expected to take several years due to regulatory reasons.

References

[1] Thurston Brooks, "Wireless technology for industrial sensor and control networks," in *SIcon'01*, Rosemount, IL, USA, November 2001, pp. 73–77.

[2] James D Gilsinn and Kang Lee, "Wireless interfaces for IEEE 1451 sensor networks," in *SIcon'01*, Rosemount, IL, USA, November 2001, pp. 45–50.

[3] Per-Arne Wiberg and Urban Bilstrup, "Wireless technology in industry - applications and user scenarios," in *ETFA2001*, Antibes-Juan les Pins, France, October 2001, pp. 123–131.

[4] Theodore B. Zahariadis, Kostas Pramataris, and Nikolaos A Zervos, "A comparison of competing broadband in-home technologies," *IEEE Electronics Communication Engineering*, vol. 14, no. 6, pp. 133–142, August 2002.

[5] Anthony C. Davies, "An overview of Bluetooth wireless technology^{TM} and some competing LAN standards," in *ICCSC'02*, 2002, pp. 206–211.

[6] European Conference of Postal and Telecommunications Administration (CEPT), "ERC recommendation 70-03, relating to the use of short range devices (SRD)," December 2002.

[7] Federal Communications Comission, "FCC affirms rules to authorize the deployment of ultra-wideband technology," FCC NEWS, www.fcc.gov, February 2003.

Publication 6

S. Junnila, A. Akhbardeh, and A. Värri, "An Electromechanical Film Sensor based Wireless Ballistocardiographic Chair: Implementation and Performance", *Journal of Signal Processing Systems*, Vol. 57, Issue 3, 2009, pp. 305-320. doi: 10.1007/s11265-008-0307-2.

Journal article available at SpringerLink (www.springerlink.com).
doi:10.1007/s11265-008-0307-2

Publication 7

M. Armholt, S. Junnila, and I. Defee, "A Non-beaconing ZigBee Network Implementation and Performance Study," In *Proceedings of the IEEE International Conference on Communications (ICC 2007)*, Glasgow, Scotland, UK, Jun. 24-28, 2007.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

A Non-beaconing ZigBee Network Implementation and Performance Study

Magnus Armholt
Institute of Signal Processing
Tampere University of Technology
Tampere, Finland
Email: magnus.armholt@tut.fi

Sakari Junnila
Institute of Signal Processing
Tampere University of Technology
Tampere, Finland
Email: sakari.junnila@tut.fi

Irek Defee
Institute of Signal Processing
Tampere University of Technology
Tampere, Finland
Email: irek.defee@tut.fi

***Abstract*— In this paper we present our implementation of a layered ZigBee network and the performance test results using non-beaconing networks. Adding the implemented network layer to an existing IEEE 802.15.4 stack did not affect the throughput in the network. It is also concluded that with knowledge of the network structure an extra added delay between sending packets can reduce the number of packets lost due to channel access failure without decrease in throughput.**

I. Introduction

The ZigBee standard [1] is a standard for low-power consuming wireless devices, operating in the industrial, scientific and medical (ISM) radio bands, 868 MHz in Europe 915 MHz in the USA and 2,4 GHz in most countries of the world. The standard defines two types of devices; Full function device (FFD) and Reduced function device (RFD). An FFD usually operates as a coordinator or a router but can also act as an end-device. A RFD on the other hand is designed with power consumption in mind and can thus only operate as an end-device. The FFD is usually connected to a mains-power supply while the RFD is typically battery operated. A ZigBee network can be either beacon-enabled or non-beacon-enabled. In a non-beacon-enabled network all packets are sent using unslotted CSMA-CA. The supported network types are; star, mesh and cluster-tree topology. In star topology, the network in controlled by a single network coordinator. The mesh and cluster-tree networks extend the network onwards from the coordinator using router devices. Mesh, e.g., peer-to-peer networks provide multiple path options, which enhance reliability/scalability in the network. Cluster-tree networks utilize a hybrid star/mesh topology, which gives better support for battery powered nodes in the network as only the router nodes are responsible for data relay in the network. The routers in cluster-tree networks utilize a hierarchical routing strategy, and they may also employ beacon oriented communication, while mesh networks allow peer-to-peer communication and shall not emit beacons.

The ZigBee stack architecture, see Fig. 1, is inspired by the open systems interconnection (OSI) seven-layer model [2] and is divided into four distinct layers; the physical layer (PHY), the medium access control layer (MAC), the network layer (NWK) and the application layer (APP). I.e., the three lowest layers are implemented as such, and the APP layer combines the main functionality of the four higher layers. The two lower layers, PHY and MAC, are defined by the IEEE 802.15.4 standard [3]. The features and functionality provided by the MAC-sublayer are: network association and disassociation, acknowledgement frame delivery, frame validation, channel access (using CSMA-CA), beacon management (optional), and guaranteed time slot (GTS) mechanism for high-priority communications. The GTS mechanism is not used by the ZigBee NWK layer. In addition, the MAC sublayer provides support for implementing security mechanisms. Building on this base, the network layer adds functionality for discovering and maintaining routes, starting a network, functions for joining or leaving a network, and the ability for the coordinator to assign short address' to devices joining the network. The network layer can also secure the transmissions and synchronize devices within the network, but security was not implemented by us and the synchronization could not be fully implemented without MAC beaconing support. Finally, the ZigBee application layer adds binding to the device, meaning that more than one application can use the same device.

In especially science and education a layered implementation can be of interest due to the fact that it makes it easy to compare different implementations. This work focuses on implementing the ZigBee network layer from a layered point of view. Accessing the lower layers is done through the public functions described in the IEEE 802.15.4 standard and the interface to the application layer is constructed according to

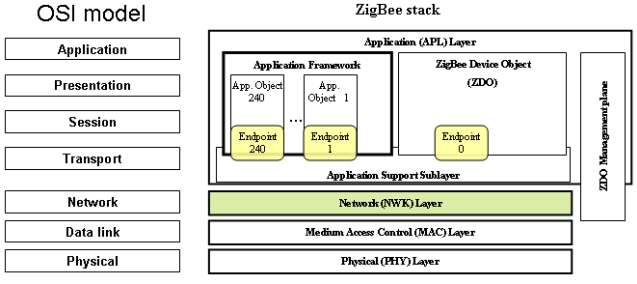


Fig. 1. The OSI model and the ZigBee stack.

the public functions in the ZigBee specification. The IEEE 802.15.4 stack used in this project does not support sending out beacons and processing other received beacons, hence the implementation is only tested for non-beaconing networks.

A ZigBee stack can be bought from several companies and a free stack is available from Microchip Technology Inc. for PIC microcontrollers. Also, Dr. Robert Reese [4] from Mississippi State University has developed a platform independent subset of the ZigBee stack, which is available for free for research purposes. This stack was not available at the start of this project. Other published research work in this area has mostly been based on simulations and mainly focused on the IEEE 802.15.4 MAC. Lu et al. [5] studied the performance of IEEE 802.15.4 in the NS-2 simulator and Lee [6] performed a indoor measurement study of the IEEE 802.15.4 performance. Petrova et al. [7] compared NS-2 simulations with indoor and outdoor IEEE 802.15.4 measurements. We could not find any existing studies incorporating the network layer.

In Section 2 we describe the implementation specific choices and issues. Section 3 is dedicated to the experimental setup and in Section 4 the results are presented. Finally, in Section 5 the conclusions are given.

II. Implementation

The hardware used in this project consists of three ZigBee radio boards; one Chipcon CC2420 v.1.1. and two Chipcon CC2420 v.1.2 development boards. To record the network traffic a packet sniffer was used, using Chipcon CC2400 v.2.0 evaluation board with a Chipcon CC2420 v.1.0 RF-module connected to it.

The implementation of the network layer is written in C and compiled with avr-gcc v.3.4.6. The underlying IEEE 802.15.4 MAC used was the Chipcon MAC stack v.1.3.0.

A. Timer issues

In the ZigBee standard it is said that the network layer shall request retransmissions of route requests after a certain time period. Also other parts of the network layer need to perform actions after some time has elapsed. This has led to the decision to implement a scheduler in the network layer. However, the microcontroller on the CC2420DB has four timers of which only two are 16 bit, and other two 8 bit. The IEEE 802.15.4 stack already uses one of these 16 bit timers for MAC timing. We felt that we should leave the other 16 bit timer for possible applications to use and opted to use the MAC timer for also the NWK stack. Because of this, the strict layered structure of this project was abandoned. The timer interrupt calls the timer interrupt routine of the MAC layer, which after processing the MAC functions, calls the network layer interrupt scheduler.

B. Reuse of network addresses

The ZigBee standard specifies two address assignment mechanisms; the tree address allocation and the higher layer address assignment. A ZigBee network address is 16-bits so the address space is finite and reuse of addresses is needed. When higher layer address assignment is used in this project, also reuse of address has been left to the higher layers.

For tree address allocation the ZigBee standard specifies two different equations for assigning addresses to router-capable children and end-device children. Addresses for router-capable children were assigned according to equation 1 where A_{parent} is the address of the current device, $Cskip(d)$ is a depth specific skip value, d is the depth of the current device, $1 \leq n_{router} \leq Rm$, and Rm is the maximum number of router-capable children a device may have.

$$A_{n_{router}} = A_{parent} + 1 + Cskip(d) \cdot (n_{router} - 1) \quad (1)$$

Network addresses are assigned to end devices according to equation 2, where $1 \leq n_{enddevice} \leq (Cm - Rm)$, and Cm is the maximum number of children a device may have.

$$A_{n_{enddevice}} = A_{parent} + Cskip(d) \cdot Rm + n_{enddevice} \quad (2)$$

The reuse of tree allocated address is implemented using a bitmap representing the n_{router} and another bitmap representing the $n_{enddevice}$. When a device leaves a network in such a way that the address is allowed to be reused then the bit in the bitmap representing the n_{router} or $n_{enddevice}$ is set in the corresponding bitmap. When assigning a new address to a joining device the bit map is checked and if a bit is set then the corresponding n_{router} or $n_{enddevice}$ is used instead of increasing the internally kept counter.

III. Experimental setup

A non-beaconing network was formed on channel 11, which is the first of the 16 2.4 GHz ISM-band channels, using three CC2420DBs; one coordinator having two children, from which one is acting as an end-device and the other one as a router-capable child. In all tests except test A, the packets were sent from the end-device via the coordinator to the router-capable child. The addressing mode was the 16-bit network addressing. The tests were conducted in office environment with a distance of about 40 cm between the devices. Throughout this paper, when talking about packet size, this refers to the actual data payload of the packet, discarding the headers and the footers. Also the throughput calculated is based on the actual data moved from point a to point b, saying that the size of the original data sent is divided with the total time, including the time for retransmissions.

A. Acknowledgment time

The first test conducted was to determine the acknowledgment time, meaning the time between packet sent and acknowledgment received. A non-beaconing network was formed and the device joined the network. 40 packets were sent from the device to the coordinator and the acknowledgment time was observed. The test was then repeated sending 40 packets from the coordinator to the device.

B. Throughput in burst mode

This test was conducted to study the effects of the network layer on the throughput. Throughout this report a burst mode means that the next packet is requested to be sent as soon as the acknowledgment from the previous packet is received. Once the non-beaconing network was up, an initial packet was sent to let the coordinator find a valid route to the router-capable device. When this was done, 40 packets were sent in burst mode with the router-capable device as the destination. The test was repeated 10 times and the traffic recorded with the packet sniffer. Retransmissions were utilized but if any packet was still lost (due to channel access failure) the test was discarded and repeated. This is because the occurrence of channel access failures was tested in the test presented in the next subsection.

To see the effects of the network layer, the same test was repeated using only the IEEE 802.15.4 stack. To make the comparison, all packets sent to the coordinator were requested to be sent to the router-capable device as soon as they were received at the network layer. The added functionality of the network layer over MAC layer in this test was to recognize the intended receiver of the packet and look up routing information for the intended receiver.

C. Packet loss due to channel access failure

When using unslotted CSMA-CA the transmission of a packet is first delayed with an initial random backoff time. The physical layer senses the channel and if the channel is busy an random backoff period is waited before the channel is sensed again. This procedure is repeated a couple of times but if the channel is busy for a longer period the MAC layer will drop the packet and indicate a channel access failure to the network layer.

This test was conducted to determine how often this occurs. A non-beaconing network was formed and an initial packet was sent to let the coordinator find a valid route to the router-capable child. Packets were sent in burst mode and the test was repeated 10 times.

D. Effects of steady flow

Reliability in ZigBee is implemented by retransmissions. However, retransmissions do not solve the problem with packets lost due to channel access failure. This test was conducted to try to find a waiting period between packets where packets do not collide and what throughput this will lead to.

To form a steady flow, a time was waited between the reception of the acknowledgment (for the previous packet) and the request for the next packet to be sent. A non-beaconing network was set up and an initial packet was sent to let the coordinator find a valid route to the router-capable child. Since the aim was to observe a loss of packets due to channel access failure the biggest packet size possible was used since this will occupy the channel the most. 70 packets were sent and the test was repeated 10 times.

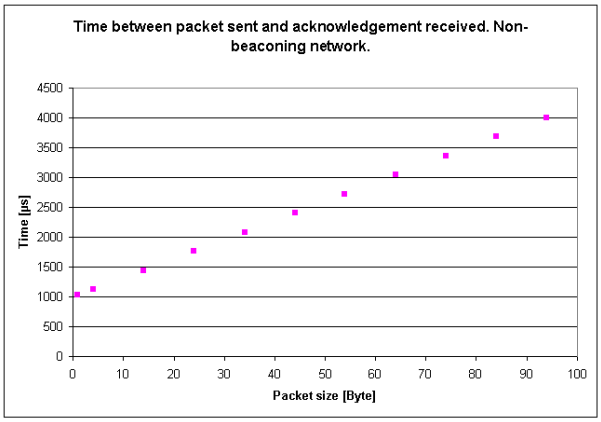


Fig. 2. Acknowledgment time in non-beaconing network.

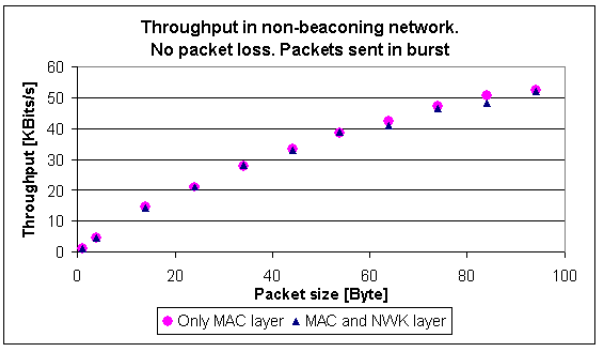


Fig. 3. Throughput in non-beaconing network, packets sent in burst.

IV. Results

The time used in calculations and reported in the results from experiment IV-A is the time recorded by the packet sniffer. In the case of experiment IV-A this means the time between the packet sniffer received the packet and the acknowledgment.

A. Measuring acknowledgment time

Measuring the acknowledgment time shows as expected an relationship between the packet size and the acknowledgment time. The results are visualized in Fig. 2. Acknowledgment is sent as soon as the packet has been received, before it is passed to the upper layers. Therefore, there is no difference in acknowledgment time when using more layers in the stack. Also worth noticing is that acknowledgments are sent without the use of CSMA-CA.

B. Throughput in burst mode

Figure 3 shows the results of the measurements. From the results it can be seen that there is no difference in throughput if only the IEEE 802.15.4 stack is used or if also the network layer is added.

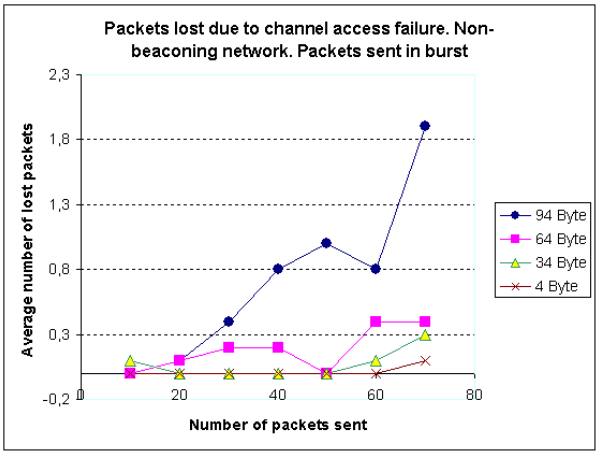


Fig. 4. Packet loss in non-beaconing network, packets sent in burst.

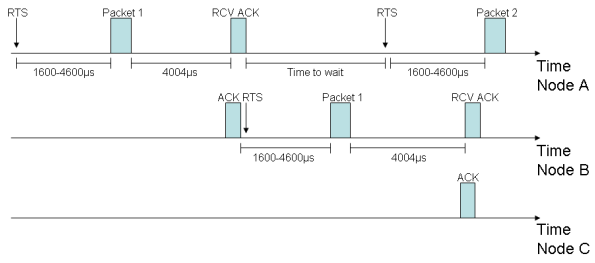


Fig. 5. Transmission time lines for node A, B and C.

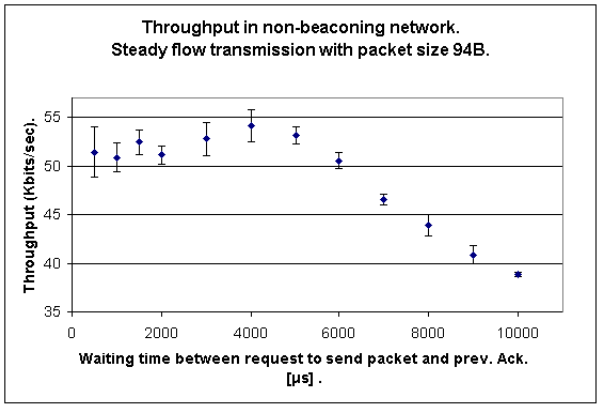


Fig. 6. Throughput in non-beaconing network, steady flow.

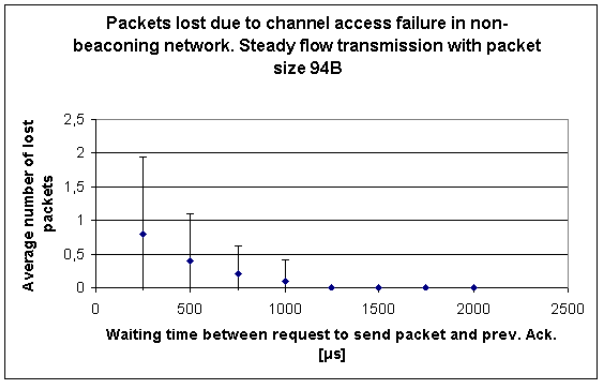


Fig. 7. Packet loss in non-beaconing network, steady flow.

C. Packet loss due to channel access failure

The results for the test are shown in Fig. 4. As expected, the bigger the packet size gets, the more packets are lost due to channel access failure. The small number of trials (10) causes fluctuations and dips in the curves, but the trend of the curves can be seen already from these measurements.

D. Effects of steady flow

Figures 6 and 7 illustrate the results. Analysing Fig. 6 a steady trend until 3000 μs can be seen, then an increase at 4000 μs and from 5000 μs a linear decrease.

This can be explained by looking at the acknowledgment time and the time between request to send and the actual transmission. Fig. 5 illustrates the transmissions and times for one packet, where node A is the source (end-device child), node B is the relaying node (coordinator) and node C is the destination (router-capable child). During the experiments the time between request to send (RTS) and actual transmission was observed to be random between 1600 μs and 4600 μs. These times depend on the MAC implementation of the CSMA-CA algorithm, which includes an initial random delay. Looking at Fig. 5 and the throughput in Fig. 6 it can be noted that the peak in throughput at 4000 μs is referred to the fact that 4000 μs is the time for an acknowledgment. When the time to wait is equal to the acknowledgment time then the next packet will be tried to be sent immediately after the acknowledgment is sent between the destination and the relaying node. When the time to wait is smaller than the acknowledgment time, then the next packet and the acknowledgment will both be trying to use the channel, leading to a backoff in the MAC layers CSMA-CA algorithm. After 5000 μs the difference between the acknowledgment and the exceeding time to wait is greater than the backoff time and hence the throughput gets lower.

Channel access failure occurs when the channel is busy for a longer time. When we add an extra delay between request to send packets, a time gap in between transmissions is created and any node waiting to send a packet can access the channel. Observing the number of recorded packets lost due to channel access failure, as visualized in Fig. 7, it can be noted that when sending as little as 70 packets a packet loss can be observed with up to 1200 μs waiting time. Packets are probably lost above this limit as well but they cannot be observed with this low amount of packets sent. In the scenario with only three devices a minimum worst case waiting time can be calculated by assuming that the RTS time for the relaying node of packet

1 is the maximum observed, 4600 μs, and the RTS time for the source node of packet 2 is the minimum observed, 1600 μs. Adding this time difference with the acknowledgment time, 4004 μs, gives a minimum worst case waiting time, 7004 μs. In the average case the RTS time is the same and they even out each other. Hence the peak of throughput when the waiting time is equal to the acknowledgment time. If the scenario is extended to four devices an another 3000 μs needs to be added to the waiting time to be certain that the next packet does not collide with the last acknowledgment. This scenario assumes that all devices are within reception range of each other.

V. Conclusions

This paper presented experiences made in implementing a layered ZigBee network layer and performance test results of this implementation in non-beaconing networks. Due to timer issues the strict layered implementation had to be compromised. The test results do not show significant impact on throughput when adding the network layer on top of the IEEE 802.15.4 stack. It can also be concluded that with knowledge of the network structure, adding a waiting period between sending packets can reduce the probability of channel access failure without a decrease in throughput.

VI. Discussion

What to do when a channel access failure occurs in a relaying device is in the ZigBee standard left to the implementer. When it occurs at the source of a packet the error is sent up to the application layer where a decision can be made. In a relaying node the error message can not be sent to the application layer since it is not the source of the frame. When the MAC layer has tried to send the packet and is about to indicate channel access failure, it drops the packet from the transmission queue and indicates an error to the network layer. To avoid losing the packet totally the network layer could implement an packet buffer having the same size as the MAC layer transmission queue and thus have a possibility to request the frame to be sent again. This however puts the frame at the end of the transmission queue which would lead to packets being transmitted out of order. Another solution would be to implement a way for the application layer to request missing packets, assuming that the data was fragmented over more than one packet due to packet size limitations.

A. Future work

Future work includes reducing the size of the network layer implementation and adding an application layer. Also, the behavior when a channel access failure occurs needs to be examined and evaluated in order to get reliable transmission.

References

[1] Z. Alliance, *ZigBee Specification*, 1st ed., ZigBee Standards Organisation, http://www.zigbee.org, June 2005.

[2] H. Zimmermann, "The OSI Reference Model - The ISO model of architecture for Open Systems Interconnection," *IEEE Transactions on communication*, vol. 28, pp. 425–432, April 1980.

[3] I. C. Society, *IEEE Std 802.15.4-2003*, The Institute of Electrical and Electronics Engineers, inc, http://www.ieee.org, October 2003.

[4] R. Reese, http://www.ece.msstate.edu/˜reese.

[5] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks," in *Proc. IEEE Int Conf on Performance, Computing, and Communications (PCCC 2004), Los Angeles, CA, USA*, 2004, pp. 701–706.

[6] J.-S. Lee, "An experiment on performance study of IEEE 802.15.4 wireless networks," in *Proc. IEEE Int Conf on Emerging Technologies and Factory Automation (ETFA 2005), Catania, Italy*, vol. 2, Sept 2005, pp. 451–458.

[7] M. Petrova, J. Riihijärvi, P. Mähönen, and S. Labella, "Performance study of IEEE 802.15.4 using measurements and simulations," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC 2006), Las Vegas, NV, USA*, vol. 1, April 2006, pp. 487–492.

Publication 8

S. Junnila, M. Zakrzewski, A-M. Vainio, J. Vanhala, and I. Defee, "UUTE Home Network for Wireless Health Monitoring," In *Proceedings of the International Conference on Biocomputation, Bioinformatics, and Biomedical Technologies (BIOTECHNO 2008)*, Bucharest, Romania, June 29-July 5, 2008, pp. 125-130.

UUTE Home Network for Wireless Health Monitoring

Sakari Junnila and Irek Defee
Department of Signal Processing
Tampere University of Technology
FIN-33720 Tampere, Finland
Email: {sakari.junnila,irek.defee}@tut.fi

Mari Zakrzewski, Antti-Matti Vainio, and Jukka Vanhala
Department of Electronics
Tampere University of Technology
FIN-33720 Tampere, Finland
Email: {mari.zakrzewski,antti-matti.vainio,jukka.vanhala}@tut.fi

***Abstract*—This paper presents a home sensor network for wireless health monitoring, including a wireless sensor network, client for controlling the sensor network, and a data storage server. A common software and hardware microcontroller-sensor interface was defined to enable joint use of sensor technologies developed in three different projects. IEEE 802.15.4 RF-transceiver based radio-boards and ZigBee network software were designed and built, along with a simple sensor network software on top of the ZigBee stack, to implement the wireless sensor network. Both commercial and custom made sensors have been interfaced to the sensor network. A set-up consisting of four sensors was developed and tested in a real home environment. The architectural overview of the system and main technical design choices are presented.**

I. Introduction

The UUTE project aims at integrating the technological expertise among Finnish and international research consortia and exploit these to form new wellness, health care and security service platforms for domestic environment. It is part of the international ITALH project which places emphasis on general assisted and independent living. The special emphasis in UUTE project is on providing support for selected groups of patients, such as cardiac patients and users of wheelchair, needing assistance in their disease management or to improve their rehabilitation. The focus is on integration of novel sensors, wireless communication, and service platforms to enable actual real-time continuous or on-demand measurements, data transfer and immediate clinical or safety oriented support and services.

The UUTE project concentrates on developing system approaches for home and assisted living environments with minimum system installation and removal costs made possible by the new wireless technology and robust software architecture. The support and services provided by the foreseen platforms would facilitate in adding general support for independent living, early treatment as outpatient after operations, good management of diseases or rehabilitation. During the project real life experiments with the integrated system will be conducted.

There are two other projects closely related to UUTE running in TUT (Tampere University of Technology) and belonging to the international ITALH project, namely the WISEPLA and the PUHVI projects. The WISEPLA project is concentrating on short-range wireless sensor platform for ambulatory and implantable applications, and the PUHVI project studies wearable well-being by integrating electronics and textile for monitoring a patient recovering from a hip operation. Although they all have a slightly different application domain and, thus, a bit different emphasis, all of the three projects concentrate on sensor development for WSN for health care purposes. To promote exchange of ideas and developed sensor devices, a common microcontroller/sensor hardware and software interface for all of the three projects was defined. Using it, a sensor developed in one of the three projects can be connected to any of the radios used in the three projects with small software modifications.

In the first part of the project, we have developed a so-called UUTE home network. It consists of UUTE sensor nodes and UUTE home client, which are connected via UUTE wireless sensor network (UUTE WSN). In addition, the UUTE home client is further connected to an external data storage server via a LAN/WLAN connection.

We start by presenting the common sensor interface and explaining the motivation for making it in Section II. Section III presents the current UUTE sensor nodes and sensor software implementation issues. The wireless sensor network implementation is described in Section IV. Section V talks about the home client and it's user interface. Finally, the results are presented in Section VI and conclusions are given in Section VII.

A. Related work

Wireless sensor networks have been one of the hottest research topics in the last years, and a huge number of publications have been written about them. Our research focuses on developing a wireless home monitoring sensor network, which is also a topic with much ongoing research and open issues. A recent article by Varshney on pervasive healthcare and wireless health monitoring [1] addresses many of the open issues, and presents work done in the field. A guest editorial [2] about m-Health, mobile computing, medical sensor, and communications technologies for health-care, is also a good introduction to the topic. Korhonen, Pärkkä, and van Gils presented infrastructure and usage models for wearable sensors and their future use in home health monitoring [3]. Previously,

978-0-7695-3191-5/08 $25.00 © 2008 IEEE
DOI 10.1109/BIOTECHNO.2008.19

Ogawa et al. have presented systems for automated home health monitoring [4], [5]. More generally, the usefulness and technical applications of home health monitoring have been discussed by Dishman [6], and Boric-Lubecke and Lubecke [7]. An article by Milenkoviv, Otto, and Jovanov [8] gives a good summary of issues related to implementation of wireless sensor networks for personal health monitoring with current technologies, while ad-hoc network related issues regarding wireless patient monitoring have been discussed by Varshney and Sneha [9]. As part of the most recent work on home health monitoring, Paganelli and Giuli [10] presented a model for home health monitoring and alerting, and also listed recent implementations of systems that support home-based assistance of chronic patients. Baker et. al. [11] present the most recent home health care wireless sensor network technologies, while paper by Ince, Min, and Tewfik [12] shows how wireless home health monitoring could be done using of-the-shelf components.

II. Common sensor interface

The driving force for the interface work was the coexistence of the related projects WISEPLA and PUHVI and the desire to use developed sensors across project borders. Because of different application domains, each project stressed application requirements differently: strict power management requirements with implantable sensors lead to choosing Texas Instruments's low-power microcontroller units (MCUs) instead of the Atmel ones used in our project; WISEPLA project aimed at studying the development of the physical layer of data transfer technologies, where as the PUHVI project was tied to using ANT radio protocol [13] by Dynastream Innovations Inc. for connecting to a commercial wrist watch interface; and in the UUTE project we set high priority for using the IEEE 802.15.4 technology based CC2420-radio chip because of the close co-operation with the international ITALH-partners. These distinct requirements resulted in a sensor interface independent both from the chosen MCU and the radio protocol, which is presented in Fig. 1.

A. Hardware architecture

Hardware (HW) utilizes a slot based architecture: a slot is a virtual block that provides a certain number of connections in the sensor interface. A sensor can reserve (i.e. be connected to) one or several slots, or a sensor can be connected only to a serial bus which is common to all the slots connected to MCU. The number of slots provided by the radio and MCU board is not delimited. The services offered to a sensor by a slot are defined in the Table I.

The HW implementation of the MCU/sensor interface for UUTE project is located on the UUTE radio board and consists of three slots operated by Atmel ATmega128L microprocessor.

B. Software architecture

MCU software (SW) implements an interrupt driven component based architecture in a way that it is divided into components that depend on the chosen MCU and components that are independent of the MCU. The sensor unit designer uses the HW interface drivers (colored with light green in Fig. 1)(a) to write a sensor driver (colored with light blue) for a sensor unit. The sensor driver provides functions to control the sensor unit, and the UUTE sensor node implementer uses these functions to control the sensor and to integrate the sensor unit code to the UUTE sensor node application code.

Properties per slot	
Analog inputs (ADC)	1 channel
Minimum resolution of ADC	10 bits
Conversion rate	250 samples/s/slot
Voltage range of ADC	0 - 3.3 V
Digital inputs	1 channel
Properties common to all the slots connected to radio and MCU board	
Serial interface	3 GPIO, which can be used as SPI/TWI/UART
Supply voltage	3.3 V

TABLE I
Common HW interface specifications.

III. UUTE sensors

In the first phase of our project, we have used four different sensor devices to collect data from the monitored subject (Fig. 2). These were 1) weight scale, 2) blood pressure monitor, 3) Heart rate/ECG monitor, and 4) an intelligent bed sensor monitoring sleeping activity. This particular sensor set was selected to promote the scenario work of the project presented in [17]. Excluding the HR/ECG sensor, these devices were commercial products with RS232 serial interface. The serial interface was used to connect the devices to the UUTE radio boards, and SW drivers were written to receive data from them.

The ECG/HR sensor was designed and built in the Institute of Measurement and Information Technology, TUT. It used the UUTE radio board with a ECG amplifier daughter board attached to it via the SPI interface. The ECG electronics were based on a earlier implementation presented in [14]. It measures the ECG, detects the R-spike, and transmits the heart rate interval. Complete ECG is not transmitted to save bandwidth and power, but future versions may include a full ECG transmission on-request.

The intelligent sleep sensor has been developed in VTT Technical Research Centre of Finland and it uses an sensitive EMFi [15] sensor element placed under the mattress to record a ballistocardiogram like signal generated by the mechanical activity within the body and body movements. The sensor is able to calculate heart rate and respiratory parameters from the measured signal. It sends an activity information packet once per minute.

The weight scale and blood pressure monitor are activated by the user, and send their measurement results each time a measurement has been performed.

A. Sensor node software

A current UUTE sensor has four states: *Undefined*, *Unconfigured*, *Waiting ACK*, and *Configured* (Fig. 3). The sensor

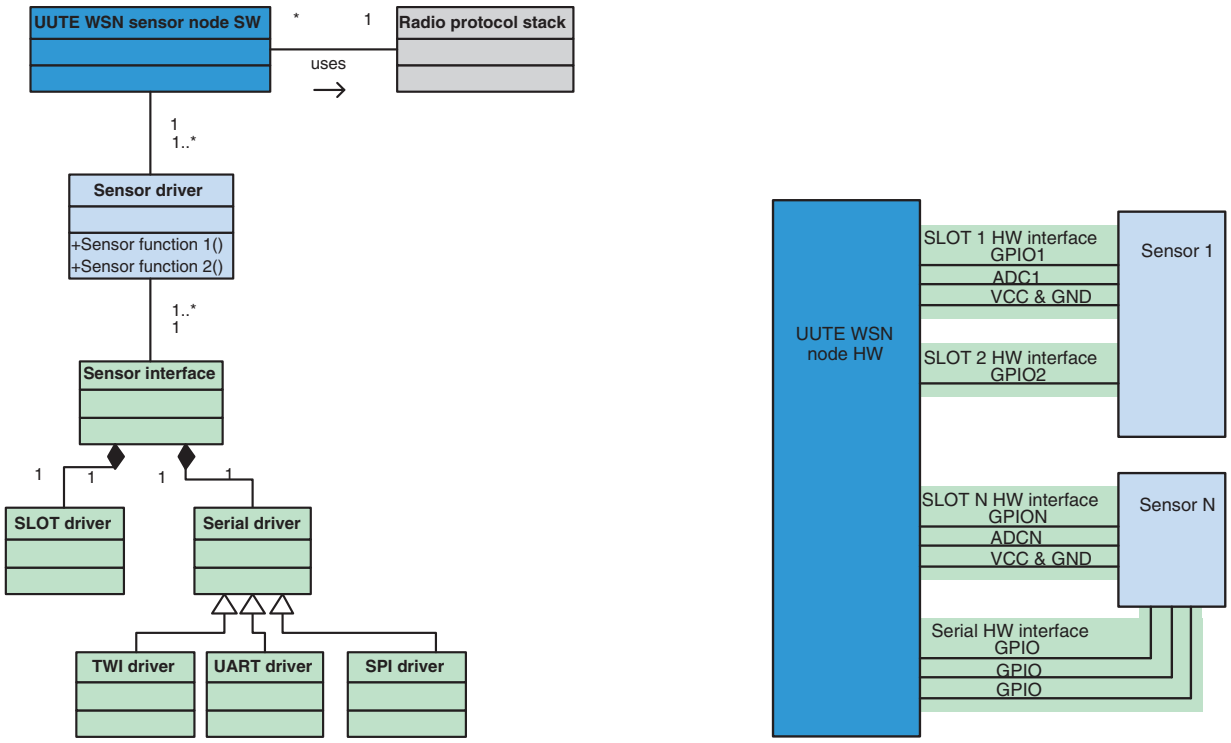


Fig. 1. The class diagram of the UUTE WSN sensor node software architecture (a) and an example of a sensor hardware configuration (b). The UUTE sensor node uses the radio protocol stack and a separate sensor driver for each sensor. These sensor drivers are written in a way that they are independent of chosen radio or microprocessor platform.

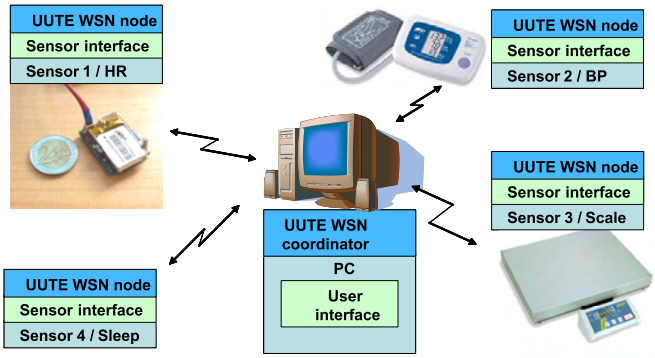


Fig. 2. UUTE home network and devices

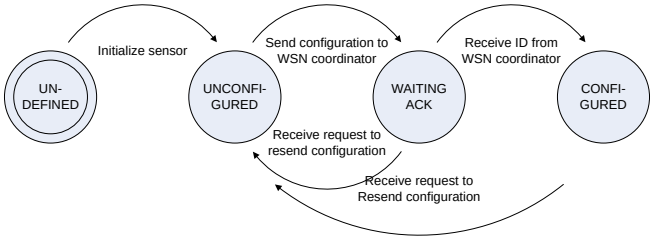


Fig. 3. UUTE sensor node states

starts in *Undefined*-state, and after initialization it changes to *Unconfigured*-state. The sensor then tries to send its configuration information to the UUTE WSN coordinator and changes to *Waiting ACK*-state. When it receives the device identifier, it switches to *Configured*-state, and is ready to operate.

UUTE sensors provide interrupt based UART drivers, and software implemented SPI drivers. The ATmega 128 MCU used for UUTE radio board also has an hardware SPI block, but it is used for communication between the MCU and the Chipcon CC2420 radio chip. Current UUTE sensors do not utilize analog inputs in the SLOT-interface.

At device start-up, the control is first given to the radio stack, which initializes itself and finally turns on the interrupts. Control is then given to the sensor driver, which initializes the interface drivers and then the sensor device itself, registers the sensor to the WSN coordinator, and finally starts to perform the measurements or wait for data from interface drivers depending on the sensor implementation. The radio stack is completely interrupt driven. The sensor driver only has to include functions to process received data from the WSN. Currently this means reception of configuration acknowledgment, but in the future it also includes parsing the commands sent to the sensor by the UUTE home client. To send data, the sensor driver calls directly the *nldeDataRequest* function of the Zigbee-network layer.

IV. UUTE WSN

The wireless sensor network (WSN) of the UUTE home network is based on partial Zigbee network implementation developed at Institute of Signal Processing, TUT, for the

Chipcon CC2420 radio and Atmel microcontroller [16]. All of the radio boards are currently using the CC2420 radio chip and Atmel ATmega 128L microcontroller. We use both the commercially available Chipcon CC2420DB evaluation boards, and our own smaller (6 x 5 cm) radio boards, equipped with one 64kB RAM-memory chip and PCB-antenna, and a piggyback UART-driver/connector daughterboard. The network supports routing, but at present it has only been used in piconet configurations. In piconets, all the UUTE sensor nodes are directly connected to the UUTE WSN coordinator. The UUTE WSN coordinator is attached to UUTE home client, a laptop PC, which collects the data from all the sensors, stores it and forwards it to the UUTE Server. The role of the UUTE home client is described in detail in Section V.

A. UUTE WSN coordinator

The UUTE WSN coordinator sets up a Zigbee network on a free Zigbee radio channel. It has a table of UUTE sensors connected to the network. Each new sensor joining the network is added to the table and given an ID number. The WSN coordinator communicates with the UUTE home client via serial port. Information about new sensors joining the network and data sent by the sensors is sent to the home client. The ID number attached to the data defines from which sensor the data is coming from. Currently, the home client cannot send commands to the WSN coordinator, nor the sensors, but this functionality will be added to the future versions. The coordinator radio unit is either a battery powered radio board identical to the sensor radio boards, attached via the serial port, or a radio unit equipped with USB connector, to which power is supplied via the USB connector.

B. Sensor communication

A simple sensor communication protocol has been implemented on top of the Zigbee stack. When activated, an UUTE sensor looks for available Zigbee networks, and if it finds one, it sends an configuration information message to the network coordinator defining its type. The UUTE WSN coordinator looks up its internal table to see if it already knows the sensor (based on its Zigbee address), and if not, it then adds the sensor to the sensor table and gives it an identifier number. The sensor receives this number, and is then ready to start to transmit its sensor data with the identifier number in the data packet header. When the UUTE WSN coordinator receives a data packet from a UUTE sensor it decodes the data from the data packet. The packet format is defined by the sensor type, which was sent by the sensor in the configuration information message, and was stored by the coordinator to a table. This proprietary protocol is very light and simple, but with limitations. We have looked at ISO/IEEE 11073 standard, for example, to find a more suitable medical sensor communication protocol, but we currently lack resources to implement such complex standards. Also, the Zigbee stack does not have transport protocol, especially fragmenting one, which is problematic with protocols sending bigger packets (like the ISO/IEEE 11073), as the Zigbee payload is only 92 bytes.

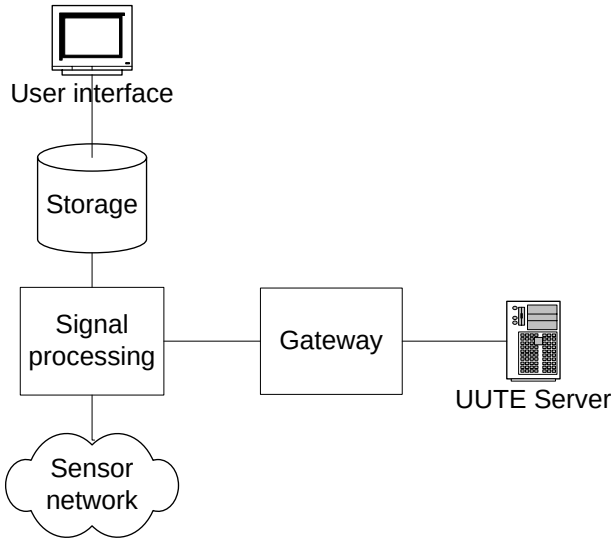


Fig. 4. UUTE home network architecture

V. UUTE HOME CLIENT

The central node of the network is the home client software running on a laptop. It has many roles. It acts as a processing unit for the data coming from the UUTE wireless sensor network; it stores the data locally; it acts as a gateway to the UUTE server and it provides a user interface. The software has separate parts for user interface and a model according to the MVC-model, and it is written in Java.

A. Signal processing

When a message arrives from the sensor network, it is first decoded. Then the payload is passed to device-specific object according to the device type field and device ID. The device-specific objects then parse the payload and construct human-readable values that can be presented to the user. When the data is parsed to the desired format, it is passed to the storage and the user interface. Each sensor device can have recommendation limits for the measured values, either upper, lower or both. These can be set by health care professionals to guide the user. Also, each sensor device can have corresponding limits for an alarm. These can be used for example to send alarm notifications to emergency services, if the measured values indicate an emergency.

B. Storing the data

At this stage of prototyping all the data was saved locally in the home client's hard drive. This was done for data gathering purposes and it also made it easier for the user to view the measured data. Each sample of incoming data is provided with a time stamp. This time stamp and the measured value were then converted into a text format and written to a file. We used plain text files, because they are easy to view using any text editor, and the amount of data wasn't very large. Each device

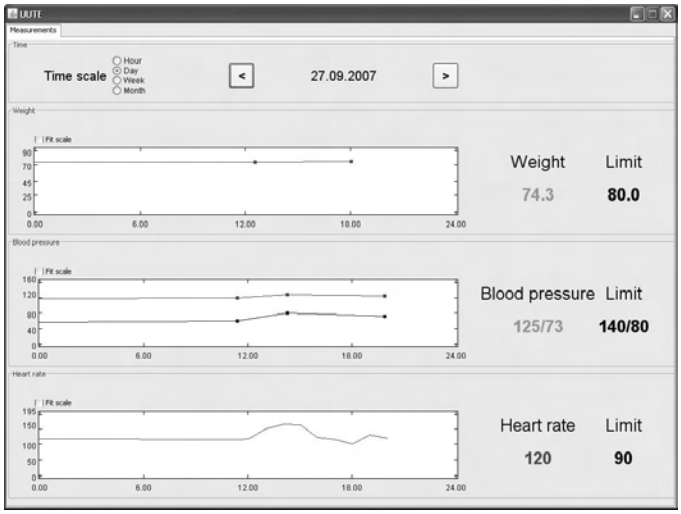


Fig. 5. UUTE home client user interface

has it's own directory, and each day has its own file. All the samples are stored right after they are received and parsed.

C. Gateway

The home client functions also as a gateway and sends the data to the remote UUTE server outside of the user's house. The data is transferred once per day, at a predefined hour, in this case at six in the morning. The software keeps track of what data have been already sent, and sends the new data using an SSL-secured socket connection. The server currently only receives the data and stores them. After that the secure connection is closed. Sending interval can be varied.

D. User interface

The most visible part of the home client software is the user interface that can be used to view the measured data. The user interface is updated in real time, and it can also be used to view all the past measurements. The measurements are drawn as a line graph, so trends can be easily seen. The graph has several options for the time scale: an hour, one day, one week, and one month. This is necessary because different measurements have very different needs for the time scale. For example, the weight is typically measured only once or twice a day, whereas Heart rate/ECG sensor provides a continuous stream of measurements. The scaling of the measurement value axis can be fitted to the view according to the current time scale selection, or it can be absolute.

The user interface shows always the latest measurement value in a numerical format. The color of the numerical representation depends on the set recommendation limits. If no limits are set, the color is normal black. However, if limits have been set, then the color is dependent on the measured value. If the value is in the recommended range, the color is green. Respectively, if the measured value is outside the recommended range, the color is red. This gives the user feedback of his or hers health.

VI. Results

The sensor network and home client were tested in a real home environment for one week. The test setting included the wireless sensor network, sensors for weight, blood pressure, heart rate and pressure sensor for bed, and a laptop with home client software. The data was transferred daily to UUTE server. This was a technical test to verify the operation of the sensor network and the home client, so usability was given only little consideration.

The test was mostly successful. The data was gathered properly to the home client and sent forth to the server. The home client was cut off from the Internet for some time during the test, but as the data was stored locally, no information was lost during the break. The only problem was that setting up the sensor network was somewhat difficult as the user was not very familiar with the system. However, the procedure will become easier in the later tests as we focus more on usability. The power consumption of the devices should also be optimized in the future versions.

The current system is limited to 255 sensors per WSN coordinator. As it is planned to have one coordinator per apartment, this should be sufficient. The limit comes from using 8-bit identifiers for WSN sensor identifiers. The ZigBee network uses 16-bit short addresses, so by modifying the sensor network code the 255 device limit could be increased to 65535.

The network delay in normal operation is short; the message from the sensor reaches the UUTE home client in less than one second, which is enough for home monitoring purposes and applications. The WSN is based on non-beacon network with CSMA-CA, so the message delay becomes longer and unpredictable when the network is heavily loaded or interfered, and packets may even end up being dropped. The effect of the network load has not been tested in this study. The current sensor network does not provide guaranteed data delivery mechanisms or support for different data priorities/quality of service. The non-beacon Zigbee network makes it also difficult to optimize power consumption, if two-way communication is desired.

In our project sequel, the setup will undergo usability tests consisting of three month test period by a senior user. In addition, the system is being further developed by adding more sensor devices such as embedded indoor positioning system and wearable pedometer in order to implement the most desired scenario of our assessment study [17]. One of the objectives of the project is to gain more insight of usability and service concept issues of assisted living and home health monitoring. This has been one of the explaining factors for developing the setup reported in this paper, although similar kind of systems have already been presented in the literature.

VII. Conclusions

UUTE home network serves as a good basis for WSN platform and proof of concept for the scenario work of the project [17]. We are currently working with UUTE projects sequel, UUTE2, and developing the platform further by adding

new sensors to network and increasing the functionality of the home network. The unique feature of the UUTE home sensor network design is that the sensor devices can be transferred to different radio hardware platforms to use different RF technologies with only small modifications, thanks to the common SW/HW interface.

REFERENCES

[1] U. Varshney, "Pervasive Health care and Wireless Health Monitoring," *Mobile Netw Appl*, 12, pp. 113-127, 2007.

[2] Guest Editorial, "Introduction to the Special Session on M-Health: Beyond Seamless Mobility and Global Wireless Health-Care Connectivity, "*IEEE Trans on Information Technology in Biomedicine*, Vol 8, No 4, Dec 2004, pp. 405–414.

[3] I. Korhonen, J. Pärkkä, M. van Gils, "Health monitoring in the home of the future", *IEEE Eng Med Biol Mag*, May-June, 2003, pp 66–73.

[4] M. Ogawa and T. Togawa, *Attempts at Monitoring Health Status in the Home*, in Proc. IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine & Biology, Lyon, France, Oct. 12-14, 2000, pp. 552–556.

[5] T. Tamura, T. Togawa, M. Ogawa, M. Yoda, "Fully automated health monitoring system in the home", *Med Eng & Phy*, Vol. 20, 1998, pp. 573–579.

[6] E. Dishman E, "Inventing wellness systems for aging in place", *IEEE Comput*, May 2004, pp 34–41.

[7] O. Boric-Lubecke, V. M. Lubecke, "Wireless house calls: using communications technology for health care and monitoring", *IEEE Microw Mag*, Sep. 2002, pp. 43-Ű48.

[8] A. Milenkovic, C. Otto, E. Jovanov, *Wireless sensor networks for personal health monitoring: Issues and implementation,* Computer Communications 29(13-14): 251-2533 (2006).

[9] U. Varshney, S. Sneha, "Patient monitoring using Ad Hoc wireless networks: reliability and power management", *IEEE Communications Magazine*, April 2006, pp. 49–55.

[10] F. Paganelli and D. Giuli, *An Ontology-based Context Model for Home Health Monitoring and Alerting in Cronic Patient Care Networks*, in Proc. Advanced Information Networking and Applications Workshops (AINAW'07), Vol 2, Niagara Falls, Canada, May 21-23, 2007, pp. 838–845.

[11] C. R. Baker, K. Armijo, S. Belka, M. Benhabib, V. Bhargava, N. Burkhart, A. D. Minassians, G. Dervisoglu, L. Gutnik, M. B. Haick, C. Ho, M. Koplow, J. Mangold, S. Robinson, M. Rosa, M. Schwartz, C. Sims, H. Stoffregen, A. Waterbury, E. S. Leland, T. Pering, and P. K. Wright, *Wireless Sensor Networks for Home Health Care*, in Proc. 21st International Conference on Advanced information Networking and Applications Workshops (AINAW'07), Vol 2, Washington, DC, May 21 - 23, 2007, pp. 832-837.

[12] N. F. Ince, C-H Min, A. H. Tewfik, *Integration of Wearable Wireless Sensors and Non-Intrusive Wireless in-Home Monitoring System to Collect and Label the Data from Activities of Daily Living*, in Proc. 3rd IEEE-EMBS International Summer School and Symposium on Medical Devices and Biosensors (ISSS-MDBS 2006), MIT Cambridge, MA, USA, Sept. 2006, pp. 28–31.

[13] www.thisisant.com

[14] A. Vehkaoja, J. Verho, J. Lekkala, *Miniature Wireless Measurement Node for ECG Signal Transmission in Home Area Network*, in Proc. 28th Annual International Conference of the IEEE-EMBS 2006 (EMBC 2006), New York, NY, USA, Aug. 2006, pp. 2049-2052.

[15] K. Kirjavainen, "Electromechanical film and procedure for manufacturing same," *U.S. Patent No. 4654 546*, 1987.

[16] M. Armholt, S. Junnila, and I. Defee, *A Non-beaconing ZigBee Network Implementation and Performance Study*, in Proc. IEEE International Conference on Communications (ICC 2007), Glasgow, Scotland, UK, Jun. 24-28, 2007.

[17] O. Kenttä, J. Merilahti, T. Petäkoski-Hult, V. Ikonen, I. Korhonen, *Evaluation of Technology-Based Service Scenarios for Supporting Independent Living*, in Proc. 29th Annual International Conference of the IEEE-EMBS 2007 (EMBC 2007), Lyon, France, August 23-26, 2007, pp. 4041–4044.

Appendix A

Errata

Publication [P3]

Page 2, section 2.1 Optical isolation, paragraph 2, first sentence should read: Isolation of USB with optoisolators has one major problem, the detection of *the transfer direction of the transmitted data.*