A Secure And Scalable Telemonitoring System Using Ultra-Low-Energy Wireless Sensor Interface For Long-Term Monitoring In Life Science Applications

W. Zhang*[a], P. Passow*[b], E. Jovanov[c], R. Stoll[a] and K. Thurow[b]

a. Institute of Preventive Medicine, University of Rostock, Rostock, Germany
b. Center for Life Science Automation, Rostock, Germany
c. University of Alabama in Huntsville, Huntsville, Alabama, U.S.A

Abstract— **A scalable telemonitoring system for secure long-term monitoring of signals in life-science applications is presented that enables unobtrusive remote patient monitoring with small and lightweight general-purpose sensor nodes. Sensor data are sent to a retail smartphone via the ultra-low-energy wireless ANT protocol and forwarded over 2G, 3G or Wi-Fi to the cloud-based IPM-mHealth Portal that provides services for administration, communication, live visualization and data analysis. Additional protocols improve reliability and flexibility of the ANT interface and all wireless transmissions are encrypted by either SSL or the computationally lightweight Hummingbird 2 (HB2) cipher. On-sensor data compression is applied to minimize communication energy. Experimental data show performance and energy balance of the ANT-based sensor interface and quantify the HB2 encryption overhead.**

I. INTRODUCTION

Increasing life expectancy due to medical progress and decreasing birth rates lead to a change of the population structure in Germany. As illustrated in Fig. 1, the percentage of people over the age of 65 is steadily increasing while the number of persons under the age of 25 is declining [1]. This results in continuously rising costs for healthcare, especially for elderly people. One approach to this problem is the utilization of telemedicine systems to enable a relocation of the therapy to the patient's home and thus to shorten the expensive periods of hospitalization. A real-time remote monitoring system can provide valuable information for diagnosis and during therapy. Remote monitoring combines elements from different research areas, such as sensor communication technology, computer information technology, medical technology, and medical monitoring management [2]. Real-time data from wearable sensor devices are sent to smartphones over a short range wireless connection and forwarded to the remote monitoring platform via Wi-Fi, 2G or 3G. Here, the medical data can be displayed or stored in a database system. Conventional ECG monitoring, for instance, can provide important clues for the diagnosis of a patient's state of health. A trend in biomedical engineering research is towards wireless systems that concurrently monitor multiple physiological parameters, such as ECG, respiration and acceleration. Currently, the designer of a remote monitoring system faces three major challenges:

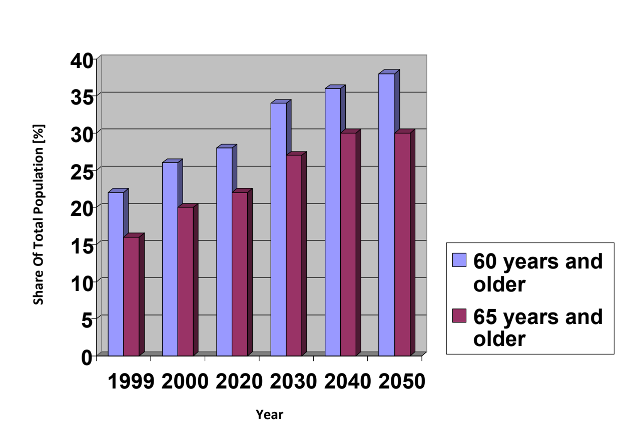

Figure 1. Projected Percentage of Elderly People in German Population (Source: German Federal Bureau of Statistics 2010)

- Minimization of the body-worn sensors' physical dimensions for easy wearing
- Minimization of the sensor devices' power consumption to maintain continuous monitoring of physiological parameters without interrupting the patient's daily activities [3]
- Improvement of scalability and compatibility of the remote monitoring system so as to identify different sensor equipment flexibly and provide an interface to enable interaction with the service and data of other medical information systems

In response to these challenges, we designed and implemented a remote monitoring system with sensor nodes that connect to retail smartphones via the proprietary ultra-low-power ANT protocol. The IPM-mHealth client software running on the smartphone collects sensor data from the wireless body area sensor network (WBASN) and forwards them over a WebSocket connection to the remote data center. On the server side, an open IPM-mHealth medical portal can be built to provide fast and efficient service for medical institutions, medical research centers and other related personnel. The server cluster is located in the Life Science Research Center of Rostock University in Germany. Doctors and researchers from different regions can access the collection of vital data stored in the database.

*Marked authors made equal contributions to this paper

II. SYSTEM STRUCTURE

The overall system structure, illustrated in Fig. 2, is based on a client-server-architecture that is designed as a tiered distributed system. The system consists of three major tiers, the WBASN, the data processing center and the IPM-mHealth portal [5].

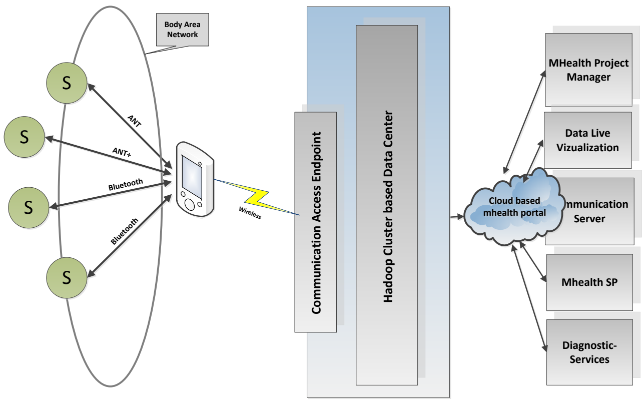

Figure 2. IPM-mHealth System Structure

Three Tiers

- The bottom layer WBASN is responsible for collection, calculation and transmission of vital data. Different wearable medical body sensors such as pulse oximeter, heart rate monitor, blood pressure monitor, thermometer, weighting scale, glucose meter etc. can be connected to the gateway via the short range wireless communication interface. The gateway functionality is provided by smartphones. The mHealth client running on the smartphones is responsible for the collection of sensor data and for sending the data to a remote data processing center.
- The data processing center based on a Hadoop server cluster is responsible for data collection, analysis, processing and data mining.
- The upper layer is the IPM-mHealth portal. It is in charge of the integration of heterogeneous modules and the interactions among them and it provides a unified application service for users.

III. SYSTEM DESIGN

A. Sensor Node Design

A WBASN usually consists of one or more body-worn wireless sensor nodes and a gateway. The usability of a remote monitoring system strongly depends on physical dimensions and battery life of the WBASN's components. Consequent low-power design enabled us to design a general purpose sensor node that operates for days or weeks on a single coin cell battery. The sensor node's main hardware components are the EFM32 Giant Gecko (Energy Micro) microcontroller and the nRF24AP2-8CH ANT transceiver (Nordic Semi). The EFM32 is equipped with the energy-efficient ARM Cortex-M3 core and provides extensive support for low-power applications. Special low-energy peripherals operate even in deep sleep mode where the controller draws little more than 1 µA. The internal time control is based on the RTC module that can wake up the CPU with 30 µs precision. A sorted queue of scheduled events is maintained instead of a tick-based mechanism. This implementation avoids unnecessary regular CPU wake-ups and leaves the EFM32 in a low-energy state for as long as possible. The remaining components of the prototype node are the temperature sensor TMP112AIDRLT (Texas Instruments), the vibration motor 304-103 (Precision Microdrives) and the PD2032 (Route JD) lithium-based coin cell battery. The prototype node is shown in Fig. 3. The node's general architecture including the flexible sensor interface that supports SPI, I²C as well as several other digital protocols is illustrated in Fig. 4. Analog signals can be fed to the EFM32's ADC.

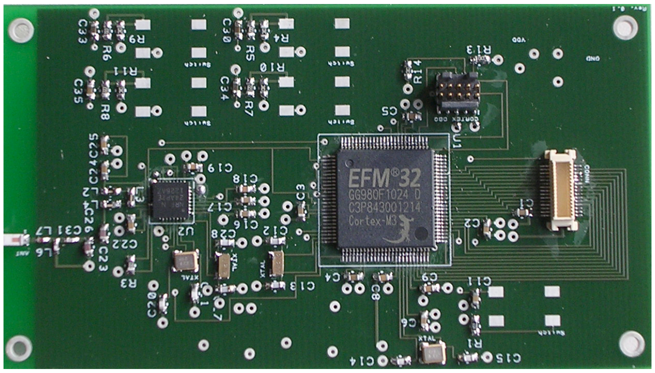

Figure 3. Sensor Node Prototype

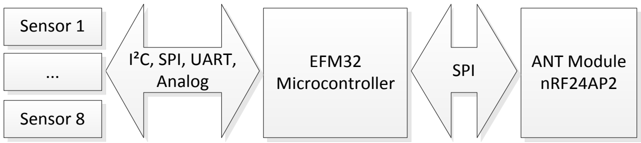

Figure 4. Sensor Node Architecture

For the processing of the sensor data, a software pipeline structure has been developed to facilitate insertion, modification and skipping of individual processing steps. Fig. 5 describes the pipeline's structure. When new sensor data is placed in the first stage buffer, the delta coded value is computed and passed to the Huffman encoder module. Here, the data is compressed and placed in a special container structure. When a container is full, it is forwarded to the next stage where a data packet is created. The routing algorithm then assigns the packet to one of the established ANT connections and stores it in the packet buffer. Here, the arrival of a transmission-ready packet triggers the statistics module that monitors the channel specific packet creation rate. This information can be combined with each channel's retransmission rate to optimize the corresponding ANT message period. The sensor node is designed to control up to eight different sensors and also up to eight parallel ANT channels. Since a Huffman code is optimal only for a specific distribution of values, there have to be several encoder instances, one for each sampled signal. Similarly, two dedicated HB2 state machines (TX and RX) are required for each ANT channel because of the HB2's stream cipher characteristic.

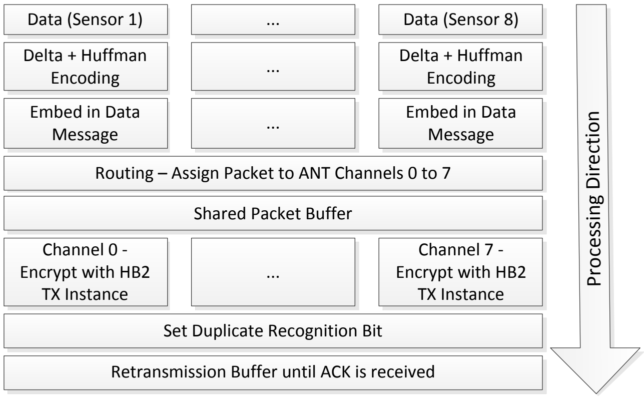
Data (Sensor 1)
...
Data (Sensor 8)
Delta + Huffman Encoding
...
Delta + Huffman Encoding
Embed in Data Message
...
Embed in Data Message
Routing – Assign Packet to ANT Channels 0 to 7
Shared Packet Buffer
Channel 0 - Encrypt with HB2 TX Instance
...
Channel 7 - Encrypt with HB2 TX Instance
Set Duplicate Recognition Bit
Retransmission Buffer until ACK is received
Processing Direction

Figure 5. Data Processing Pipeline Structure

Delta encoding followed by Huffman encoding is applied to reduce the required bandwidth of the wireless channel without loss of information. Delta encoding represents a value as the difference to the preceding value and Huffman encoding uses information on a specific signal's value distribution to assign the shortest codes to the most probable values and longer codes to less frequent values [8]. The combination of Delta encoding and Huffman encoding has been shown to be an efficient compressor for ECG signals in [6], where a compression ratio of 0.2 has been reported. Identical Huffman trees on both ends of the communication channel are required to enable the transfer of Huffman encoded data. Our sensor node can either gather the required statistical information locally and export it to the receiver side or import the distribution information for a known signal and generate the corresponding Huffman tree. Compressed data are stored in a container structure that consists of a 32 bit long code sequence and a 5 bit value that specifies the number of stored codes. A full container is embedded in a data message and sent via the assigned ANT channel. As the length of each Huffman encoded sample depends on the state of the signal, the number of packets that is required to transmit a given number of samples may vary significantly. Since ANT channels work with a constant transmission period, a control instance must be implemented to avoid the overflow or underflow of packet buffers. If a signal's value distribution results in a Huffman tree with branches that exceed the length of the data container, data compression is automatically deactivated.

B. Sensor Interface Design

The ANT protocol implemented in the nRF24AP2-8CH provides a basic data transportation service with error recognition that is roughly comparable to the data link layer of the OSI model. All higher layer protocols have to be handled by the application and must be mapped to ANT packets. Thus, the challenge in the interface implementation was to design robust and flexible protocols while at the same time saving as much of the 64 bits of ANT payload for the sensor data. For this purpose, the Message-ID logic was split into two levels. The level 1 Message-ID with a length of 3 bits is used to distinguish data messages from protocol messages related to sensor configuration, network management, encryption etc. The eight bits of the level 2 Message-ID allow the definition of up to 256 message types in each level 1 domain and thus enable the implementation of complex protocols. In data messages the level 2 ID is omitted in order to minimize the protocol overhead. Instead, a single flag is used to indicate whether a message is a standard data message or a so called "key frame" message. While standard data messages normally contain a number of Delta and Huffman encoded sensor samples, the "key frame" message, sent in regular intervals, provides a 24-Bit time stamp and a non-encoded sample that is required for the decoding of the following encoded samples. This mechanism also supports further compression of sensor data with lossy algorithms such as Skeleton [9]. Both types of data messages contain a 3 bit Sensor-ID that enables concurrent sensing with up to eight sensors, flags signaling encryption and compression, a single bit for duplicate detection, an 8 bit checksum to verify correct decryption and the lower eight bits of the sample number that in case of the standard data message is associated to the first sample in the container. Fig. 6 shows the general structure of a data message and a control message.

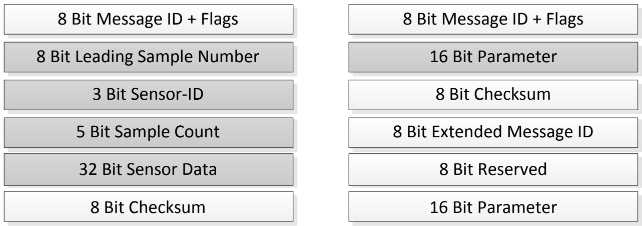

Figure 6. Message Structure Of Data Message (left) And Control Message (right) With Encrypted Parts In Darker Color

ANT channels provide a constant bandwidth that is determined by the corresponding message period. Usually, it is configured before the channel is established and not changed during operation. As mentioned before, the bandwidth demand of Huffman encoded data is not constant. Setting the message period according to worst case requirements is inefficient, so we developed the dynamic bandwidth protocol to dynamically change the message period of an open ANT channel. Bandwidth adaptions are initiated by the sensor node with the Set-Message-Period-Request that contains the 16 bit long proposed message period. The smartphone sends a Set-Message-Period-Response to confirm reception and then both devices reconfigure their ANT modules' message period. Subsequently the slave (smartphone) needs several message periods to resynchronize to the master (sensor node). In this time, no data can be transmitted and the master sends empty messages until it receives the Set-Message-Period-Acknowledge message from the slave that indicates successful resynchronization. Subsequently, data transmission is continued. Fig. 7 provides a visual representation of the protocol.

Using dynamic bandwidth control, the sensor node can minimize its power consumption in idle mode when it only broadcasts its own Node-ID. When one or more sensors are activated, the message period is decreased accordingly. A statistics module continuously monitors the channel specific packet arrival rate in the transmission buffer and computes the corresponding optimal message period.

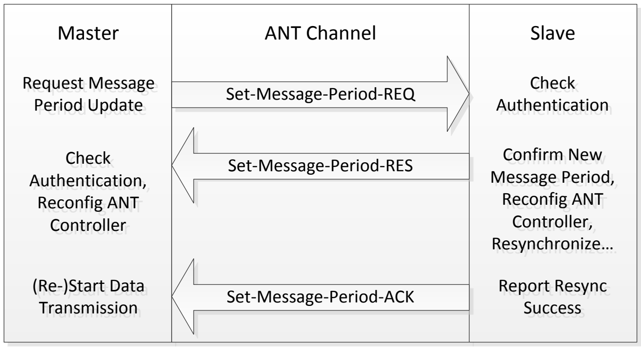
Master
ANT Channel
Slave
Request Message Period Update
Set-Message-Period-REQ
Check Authentication
Check Authentication, Reconfig ANT Controller
Set-Message-Period-RES
Confirm New Message Period, Reconfig ANT Controller, Resynchronize...
(Re-)Start Data Transmission
Set-Message-Period-ACK
Report Resync Success

Figure 7. Dynamic Bandwidth Control Protocol

In order to guarantee the confidentiality of potentially sensitive data that are sent over the wireless channel, all data messages are partially encrypted. Asymmetric ciphers, such as RSA and ECC require considerable computational effort and a corresponding amount of energy. AES is a widely used algorithm and more energy-efficient but its minimum block size of 128 bits would require two or more ANT packets to be encrypted and decrypted together. The increased latency and the at least doubled impact of packet loss disqualify the AES algorithm. HB2 is a computationally lightweight symmetric cipher with a block size of 16 bits [7]. It enables us to encrypt only parts of a message or to use different keys for protocol and sensor data. In data messages 48 bits are encrypted for transmission. Only protocol information and an 8-bit checksum are sent unencrypted. The checksum is generated using a non-invertible XOR-Rotate operation with an even number of operands to minimize the amount of exposed information [10]. The receiver uses this checksum to verify the correctness of the decrypted data. In configuration messages, encryption is only used for authentication. Hence, only 16 bits of a message are encrypted by the sender and checked against the checksum by the receiver. HB2 has a stream cipher characteristic as its internal state depends on the previously encrypted or decrypted data words. Both, sensor node and smartphone need dedicated HB2 instances for the encryption of outgoing messages and the decryption of incoming messages to keep the state machines synchronized. When a packet is lost, the state machines of sender and receiver have to be reinitialized.

The ANT protocol includes an acknowledged packet transmission service that comes with a penalty regarding energy consumption and the maximum achievable bandwidth but is still more energy-efficient than transmission error detection on application level. The message chosen for transmission is stored and retransmitted until either the ANT module confirms successful transfer or the specified maximum number of retransmissions is reached. In the latter case it is assumed that communication is currently not possible and the sensor node stops sending data. Since the acknowledged packet transmission service cannot distinguish a lost packet from a lost acknowledgment, unnecessary retransmissions can occur and create duplicate packets on the receiver side. These must be filtered out before decryption to maintain the synchronicity of the HB2 state machines. A single alternating bit in the header of each acknowledged message is sufficient for duplicate recognition. As no packet is truly lost, one arriving with its alternating bit in the wrong state must be a duplicate and can be ignored.

C. IPM-mHealth Middleware Design

Fig. 8 depicts the top view of the IPM-mHealth client middleware that is designed to connect different medical sensors and runs on a Sony Ericsson Experia Arc Smartphone. The IPM-mHealth middleware consists of the Universal Sensor interface module, the service module, the Communication module, the Security module and a Gateway proxy interface.

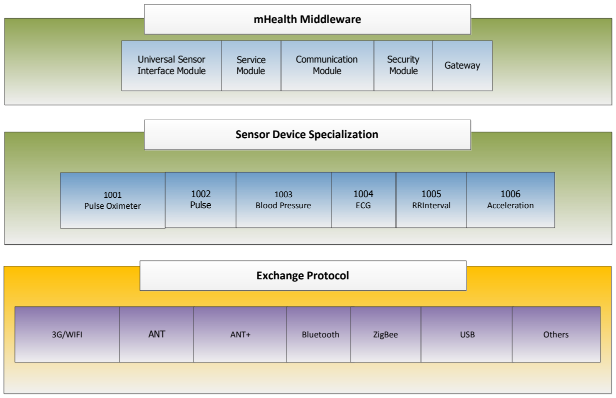

Figure 8. IPM-mHealth Middleware

- *Universal Sensor Interface Module (USIM):* In terms of the Universal sensor interface we define different objects, which can be used to represent service and parameters of different sensor devices. It comprises some of the configuration parameters of the medical sensor, the universal medical sensor data transmission format, and a Universal Sensor Transducer Interface (USTI). USTI is the basic object to enable the interactions between the IPM-mHealth middleware and different sensing devices. Here we use the medical device encoding standard including network key acquisition frequency, the sample array, the binary encoding rule (BER) and packet encoding rule (PER) [11].
- *Service Module (SM):* The service module is used to exchange data between sensor nodes and middleware. It has a data analysis interface, which

abstracts the general methods of analyzing sensor equipment. The system uses device specific analytical files to facilitate the integration of different sensors.

- *Communication Module (CM):* In order to enable the parallel connection of multi-sensor devices, we use the CM to define the main processes and sub-processes to realize the connection, combination and stop of the communication channel.
- *Security Module (SecM):* On the IPM-mHealth client we use the HB2 cipher to decrypt the sensor messages. On the server side, we use Secure Socket Layer (SSL) to encrypt the vital data and utilize 3P-AAA-SPs, Authorization and Accounting Service Providers to authenticate the client access [12].

A WebSocket connection and data interaction channel are established between the IPM-mHealth client and the remote data center. The WebSocket connection is established by the WebSocket agreement during the first "handshake". It is based on the underlying TCP/IP protocol. The WebSocket agreement is relatively simple. The client, using an ordinary browser, sends a request for the handshake to the server through port 80 or 443. The server will identify whether it is a WebSocket request or not according to HNP header. If so, it will upgrade the request to a WebSocket connection. After a successful handshake, it will enter the data transferring stage of the two-way connection. WebSocket data transmission uses a frame-based approach: 0x00 indicates the start of the data. 0xFF indicates when the data stops. The data is encoded as UTF8 and SSL is used for encryption.

Experimental results, presented in [13], showed that the WebSocket connection is real-time capable. The header message consumes 570 bytes and 50 milliseconds during the first start of a connection request and keeps the connection open to provide an exact and efficient data flow for communication.

D. IPM-mHealth Portal Design

The layered structure of the IPM-mHealth Portal is illustrated in Fig. 9. The modular development approach that has been used to develop each function enhanced the flexibility and scalability of the whole system. The IPM-mHealth Portal serves as a components bus and connects users to the implemented module. The user modules Admin Manager, Data Communication, Live Visualization Module and Data Analysis Module are the core functions of the system.

The holistic concept of the IPM-mHealth portal provides a modular, open mHealth platform with interfaces for the implementation of telemedicine prevention services. The open standards used for programming the platform enable the flexible integration of future third-party business models from the sectors of hardware, software and services. New value chains in the prevention area emerge, since in the future this concept will enable different suppliers to combine their individual products and services (sensors, mobile devices, electronic health records, etc.) to an integrated prevention product.

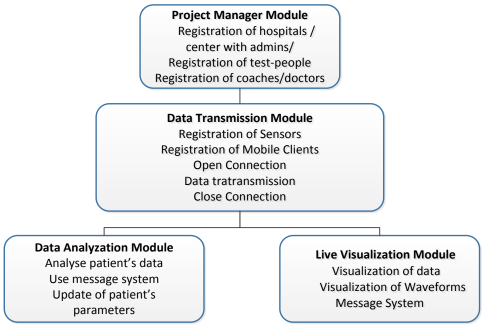

Figure 9. Modular Structure of IPM-mHealth Portal

IV. Experimental Results

A. ANT-Channel Characteristics

The range of possible applications of the telemonitoring system strongly depends on the performance of the ANT-based interface. Transmission errors were monitored in order to determine the available effective data rate at different message rates and to validate the duplicate recognition protocol. In Table 1, the terms 'Data Rate' and 'Net Data Rate' are used. The former specifies the total amount of successfully transmitted information while the latter only considers that part of a message that carries sensor data. The results show that on-sensor data compression is not only required to improve the node's energy balance but also because of strict bandwidth limitations.

Fig. 10 shows the sensor node's energy balance at different message rates in a per packet representation. Apart from the lowest rate, the results are nearly identical and indicate the effectiveness of the node's power management.

TABLE I. ANT INTERFACE - PERFORMANCE AND RELIABILITY

Parameter	**ANT Message Rate**	
	1 Hz	***50 Hz***
Total Messages	1,786	10,018
Total Retransmissions	38	704
Relative Retransmissions [%]	2.13	7.03
Total Duplicates	19	74
Relative Duplicates [%]	1.06	0.74
Data Rate [B/s]	62.64	2,975.12
Net Data Rate [B/s]	31.32	1,487.56

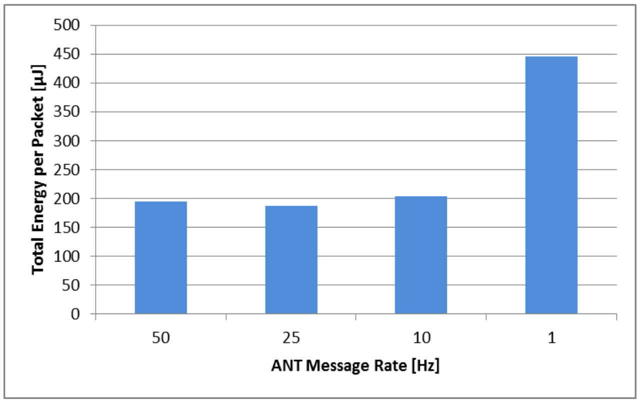

Figure 10. ANT – Energy per Packet at Different Message Rates

B. Hummingbird 2 Energy Balance

In order to determine the overhead caused by the encryption of sensor data with the HB2 cipher, identical sets of data were first transmitted as plaintext and subsequently in partially encrypted packets. The results, presented in Table 2, show a relatively low impact on the sensor node's total energy consumption.

TABLE II. HUMMINGBIRD 2 – ENERGY BALANCE

Parameter	Data Transmission Mode	
	unencrypted	*encrypted*
Total Messages	15,000	15,000
Total Energy [µJ]	2,836,833	2,923,253
Energy per Packet [µJ]	189.12	194.88
Relative HB2 Encryption Energy [%]	0	3.05

V. CONCLUSION

A secure and scalable telemonitoring system for medical applications has been presented. Small and lightweight wireless sensor nodes using the ultra-low-power ANT interface operate for days or weeks on a single coin cell battery. Data is encrypted with HB2 and SSL to protect confidentiality of sensitive medical information. Smartphones function as gateways between sensors and the cloud-based IPM-mHealth portal. Experimental results confirm the success of our low-power design approach.

REFERENCES

[1] Theoria cum praxi : Fünf Jahre Leibniz-Institut für interdisziplinäre Studien e.V. (LIFIS) Berlin: Trafo Verlag, 2007 (Sitzungsberichte der Leibniz-Sozietät 90) ISBN: 978-3-89626-690-3 pp.217-228.

[2] Holzmüller-Laue, S., Goede, B., Stoll, R. and Thurow, K., "A Highly Scalable Information System as Extendable Framework Solution for Medical R&D Projects. In: Adlassnig, K., Blobel, B., Mantas, J., Masic, I. (Ed.): Studies in Health Technology and Informatics: Medical Informatics in a United and Healthy Europe. Proceedings of MIE 2009 – The XXIInd International Congress of the European Federation for Medical Informatics, ISBN 978-1-60750-044-5),"2009.

[3] Yuce, M. R., "Implementation of wireless body area networks for healthcare systems," Sensors and Actuators A, vol. 162, pp. 116-129, 2010.

[4] Krco, S., and Delic, V., "Personal wireless sensor network for mobile health care monitoring," 6th Int. Conf. on Telecommunications in Modern Satellite, Cable and Broadcasting Service, TELSIKS 2003, 1-3 Oct., vol. 2, pp. 471-474, 2003.

[5] Neubert, S., Arndt, D., Thurow, K. and Stoll, R., "Mobile real-time data acquisition system for application in preventive medicine," Telemed J e-Health, Vol. 16, pp. 504-509, 2010.

[6] Marcelloni, F. and Vecchio, M., "An efficient lossless compression algorithm for Tiny nodes of monitoring wireless sensor networks," The Comp. J., vol. 52, pp. 969-987.

[7] Engels, D., Saarinen, M.-.O., Schweitzer, P. and Smith, E.M., "The hummingbird-2 lightweight authenticated encryption algorithm."

[8] Huffman, D.A., "A method for the construction of minimum-redundancy codes," Proceedings of the I.R.E. September 1952, S. 1098–1101.

[9] Kim, H., Yazicioglu, R.F., Torfs, T., Merken, P., Van Hoof, C. and Yoo, H.-J., "An integrated circuit for wireless ambulatory arrhythmia monitoring systems," Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Engineering the Future of Biomedicine, EMBC 2009 2009, pp. 5409-5412.

[10] Rivest, R.L., "The invertibility of the XOR of rotations of a binary word," International Journal of Computer Mathematics,88,2,281-284,2011,Taylor & Francis

[11] Hsu, J.-T., Hsieh, S.-H., Lo, C.-C., Hsu, C.-H., Cheng, P.-H., Chen, S.-J., Lai, F.-P. "Ubiquitous mobile personal health system based on cloud computing," 2011, IEEE Region 10 Annual International Conference, Proceedings/TENCON, art. no. 6129036, pp. 1387-1390.2011

[12] Ji, Z., Zhang, X., Ganchev, I., O'Droma, M., "A personalized middleware for ubiquitous mHealth services," 2012, IEEE 14th International Conference on e-Health Networking, Applications and Services, Healthcom , art. no. 6379465, pp. 474-476.2012

[13] Pimentel, V., Nickerson, B. G., "Communicating and Displaying Real-Time Data with WebSocket," Internet Computing, IEEE Volume: 16 , Issue: 4 Digital Object Identifier:10.1109/MIC.2012.64 Publication Year: 2012 , Page(s): 45 - 53