Lightweight Medical BodyNets
Invited Paper

Foad Dabiri
University of California Los Angeles
3514 Boelter Hall
Los Angeles, CA, 90095
dabiri@cs.ucla.edu

Hyduke Noshadi
University of California Los Angeles
3514 Boelter Hall
Los Angeles, CA, 90095
hyduke@cs.ucla.edu

Hagop Hagopian
University of California Los Angeles
3514 Boelter Hall
Los Angeles, CA, 90095
shaitani@gmail.com

Tammara Massey
University of California Los Angeles
3256N Boelter Hall
Los Angeles, CA, 90095
tmassey@cs.ucla.edu

Majid Sarrafzadeh
University of California Los Angeles
3532C Boelter Hall
Los Angeles, CA, 90095
majid@cs.ucla.edu

ABSTRACT

Body area networks are becoming more and more popular in addressing health care application due to advances in sensing technologies and the fact the these networks lie within close proximity of the body. We have developed a general purpose wearable platform using lightweight embedded system to address various medical applications. This architecture is composed of tiny processors/microcontrollers equipped with non-invasive sensors. In addition, an on-body terminal enables the system to be reconfigurable and to communicate with medical enterprises. Since our architecture is made of software programmable blocks, it becomes a reconfigurable system. We introduce different levels of reconfiguration for body area networks and illustrate how reconfiguration can address several design challenges such as adaptability, reliability and power consumption. Finally, we formulate sampling rate assignment as a means of power reduction while meeting performance specification. Through our formulation, power dissipation can be minimized and at the same time, the desired accuracy of the system is achieved.

Keywords

Embedded systems, sensor networks, reconfiguration, health care

1. INTRODUCTION

Recent technological advances in embedded processing and wireless networking has enable lightweight sensor networks. Sensor networks consist of distributed devices connected of sensing elements which monitor physical or environmental conditions and have recently been used in medical application. These devices are usually lightweight embedded systems collaboration with each other through a wireless sensor. First we will review lightweight embedded systems the continue with body area networks and medical application.

1.1 Lightweight Embedded Systems

Light-weight embedded systems are often low-profile, small, unobtrusive and portable processing elements with limited power resources, which typically incorporate sensing, processing, and communication. They are often manufactured to be simple and cost-effective. Despite their low complexity, computationally intensive tasks impede light-weight embedded systems from being deployed in large collaborative networks in large quantities. Their sensing capabilities allow their seamless integration into the physical world, while their general-purpose architecture design yields notable advantages such as reconfigurability and adaptability to various applications and environments.

Light-weight embedded systems are gaining popularity due to recent technological advances in fabrication, processing power, and communication. Despite these advances, there are still significant scientific challenges for researchers to overcome in terms of power management, reliability, fault handling, and security. Unexpected or premature failures raise reliability concerns in mission-critical embedded applications such as medical devices. Failures not only can cause irreversible damages but often erode manufacturers' reputations and greatly diminish widespread acceptability of new devices.

In addition, failures in critical applications, like medical devices, often cause unrecoverable damage. The limited resources of an embedded system in terms of processing power, memory, and storage can often be mitigated by efficient communication that reduces the processing and storage load on an embedded device. In addition to effective communication, the challenge of fault handling and is difficult under demands for distributed processing and real-time input from the physical world. Especially in wireless communication, interference from environmental noise and channel collisions greatly affect system performance. Often power optimization techniques are essential in wireless communication to mitigate power loss from retransmissions. Likewise, security also poses a great challenge for light-weight embedded systems. These systems may be employed for critical applications where user or data security is a major concern. Due to the limited on-board processing capabilities and low energy consumption, lightweight protocols and algorithms are required to meet system specifications.

1.2 Medical Monitoring through BodyNets

The spreading wave of healthcare programs and patient management emphasize more involvement by patients themselves. This paradigm largely requires that patient information be readily available at the point of care, regardless of which physician the patient sees. The current proliferation of broadband wireless services, along with more powerful and convenient handheld devices, is helping introduce real-time monitoring and guidance for a wide array of patients.

A multitude of different devices have been developed allowing a patient to wear a set of sensors. Common designs include: wristbands for measuring pulse, body temperature, galvanic skin reactions, and electromyography (EMG) data [21], [24], [1]; chest and arm belts for physiologic monitoring [23], [15], [17], [20]; shoes for gait analysis [27], [22]; and photo plethysmographic ring sensors [2].It is important to note that many of these systems primarily focus on physiological data acquisition, and do not provide methods that assist healthcare providers (or patients) with data interpretation or diagnosis.

In order to create a realtime monitoring environment for healthcare applications, we have developed a wireless embedded architecture which individuals can be equipped with. Similar architectures have been proposed concurrent to our design which share the same general characteristics [8][7][9]. Main components shared by wearable architectures are 1) Sensors, 2)Transducers/Proccessors, 3) Local Server/ On-body Terminal and 4) Network infrastructure, both wired and wireless.

Our proposed wearable system is capable of communicating with remote servers in medical enterprises, therefore the remote serves are part of our proposed architecture. In section 2 we describe our proposed system architecture and moreover elaborate on the functionalities granted to this system.

2. SYSTEM ARCHITECTURE

Figure 2 illustrates the general architecture of our system which is also partially shared with other current designs [8][7][9]. Each constructive component in this architecture is briefly described in this section:

2.1 Sensors

The main purpose of an on-body sensor network in health care applications is monitoring vital signs and phycological signals as well as physical behavior of the person using this system. To accomplish this purpose, various types of sensors are employed to constantly measure related information. At this point, most of the sensor we have used (or plan to test) are non-invasive passive sensors. Table 2.1 summarizes the sensors we have used and potential applications that can utilize these sensors.

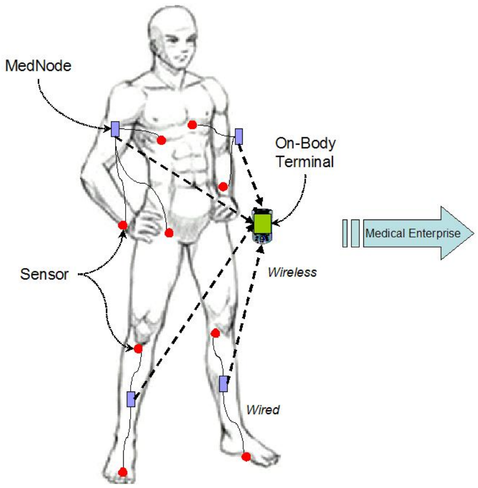


Figure 1: System architecture: Mednodes, sensors, on-body terminal and remote connectivity

2.2 Mednodes

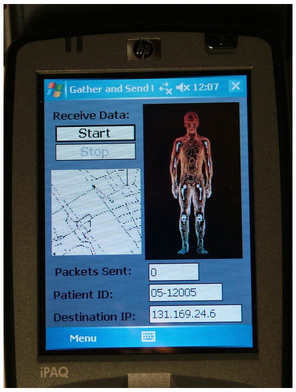
Mednodes are the main components of our proposed wearable architecture. A Mednodes is a stand alone component which consists of a processing unit, a sensor board and a local power supply which is a battery. We have used Mica2Dots and Mica2s from CrossBow [12] as the main processing unit in Mednodes. Motes in general are outfitted with multichannel ADCs (analog to digital convertor). Various sensors are interfaced to these channels through a sensor board. The sensor board operates as an interface to both excite a passive sensor and feed the sensed signal to an ADC channel. In some scenarios, a signal conditioning circuit is embedded onto the sensor board as well. The 'In Vivo pressure monitoring' device described in section 4.1 is one these cases. Mednodes are capable of wireless communication and support two wired communication protocols: I2C and UART. A coin cell battery is used to supply the Mednodes with power which immediately raises the power consumption challenges in system design. We will describe these challenges and propose methodologies to overcome them shortly.

2.3 On-Body Terminal

The on-body terminal is a personal server which is a programmable lightweight system, such as a cell phone or PDA. These devices serve as a means of data collection from sensors, local data processing/storage and data transmission to an enterprise. We use different off the shelf devices as part of our system, starting from PocketPCs, cell phones, and portable multimedia devices (e.g., iPod). Particularly we have used two different PocketPCs from HP (iPaq) which have WiFi connectivity and can connect to a local area network. Also they can support GSM mobile communication and in other words can be used as a cell phone. Among the above devices, cell phones inherit very strong characteristics suitable for medical applications. First, they are widely

Table 1: Various non-invasive sensors and theirs potential applications in health care

Sensor	Application
Accelerometers	Movement pattern recognition; Walking characterization
Piezoelectric	Impact detection and recording; Alzheimer's
Pressure	Foot pressure monitoring; Diabetics
Electrocardiogram	Hear beat classification; Ischaemia
Blood pressure	Pathophysiology; General diagnosis
Flex	Knee movement monitoring; Post-knee surgery
Galvanic Skin	Skin conductance; Emotional response; Arousal
Temperature	Diagnosis
Weight scale	Health factor



Packets Sent: 0
Patient ID: 05-12005
Destination IP: 131.169.24.6

Figure 2: GUI of the application running on the on-body terminal (OBT)

used among people: up to 90% of the world's six billion people will have mobile telephone coverage by 2010 [10] and at this point of time, there are already more than 1.8 billion mobile subscribers. The second obvious important feature of cell phones is the backbone network infrastructure that can support high data rate transmission.

2.3.1 On-body terminal specification

In our System the on-body terminal (OBT) acts as an intermediate connector between patient and physician. Patients interface with this medical monitoring system using the software running on OBT. Mednode communicate to the OBT through a gateway mote. Since the communication protocol used by Mednodes is not compatible with the communication protocols used in OBT, we utilize a gateway mote which the data from Mednodes wirelessly and transmits it to OBT serial connection (UART). OBT can transfer data to a remote server either using WiFi or can use Bluetooth for transfer data to a career medium opportunistically.

In order to consume less power on the OBT, the wireless radio is off by default. Our developed software has the capability to turn the wireless on and off based on application needs. The system can automatically turn on the radio and push the data out to its final destination which is a medical enterprise, and finally switch back to off mode. Bluetooth also can be used to transfer data in opportunistic fashion. In an environment where wireless LAN does not exist, other mobile patients wearing the same system, can be used as an od-hoc network to transfer data to a base station.

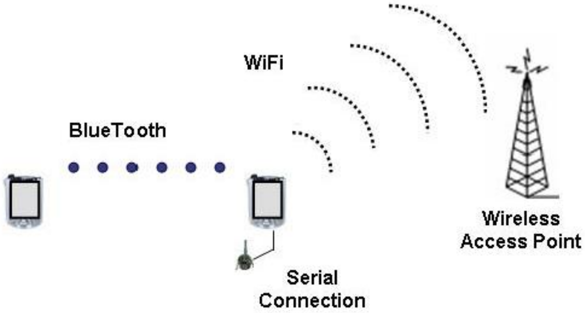


Figure 3: On-body terminal is can transmit data different destinations through Bluetooth and/or WiFi

2.4 Central Server at Medical Enterprise

The final leg in this design hierarchy is the central server. The server collects the data received from multiple on-body terminals and functions as a central storage unit. Moreover, extensive data analysis and processing are scheduled to be performed at the central server. One of the unique features of the server in our prototype is its capability to monitor the functionality of individual patients and to reconfigure on-body terminals and associated Mednodes according to realtime needs. In section 5 we will describe how we have implemented different levels of reconfiguration. Arguably, much of the emphasis in past work on healthcare applications of wearable sensors and devices has been driven by the hardware design and communication infrastructure; less attention has been paid to the problems of integrating such information into the medical enterprise. Thus, ideally, the server would execute several tasks including: 1) Data incorporation into a patient's electronic medical record (EMR); 2) Additional processing and/or signal analysis, given the additional computational processing power available on servers (and access to the patient's medical record); and 3)Reconfigure the processes on the hand-held device and of the Mednodes using interactions of healthcare providers and the EMR itself, (i.e., in a feedback loop).

3. SYSTEM CAPABILITIES

Lightweight embedded systems arranged in a body area network posses certain functionalities inherently, such as realtime sensing, and introduce new potentials functionalities, such as in-network processing. In this section, we depict a few of these capabilities in our system and elaborate on *re-*

configurability which is an essential and strong characteristic carried through the proposed architecture.

3.1 Real-time Sensing

Sensing and data acquisition is an essential attribute of any embedded sensor network specially body area networks tailored for medical monitoring and diagnosis. In our system, each Mednode utilizes 7 ADC channels which can be connected to multiple sensors. Data from these sensors can be sampled within a wide range of sampling frequency that can be set by the user or set by the system itself. The collected data is transmitted to the on-body terminal either in real-time or in chunks of data periodically. Furthermore, Mednodes can share each others' data to make a local decision. The ability to access the data sensed by other Medondes and the wireless communication ability, enable the system to perform *in-network processing.*

3.2 In-network processing

In this system, data gathered by the sensors can be processed within the network and only aggregated information is sent to on-body terminal. Mednodes are usually within wireless range of each other and can exchange information. While Mednodes have limited processing power , they can perform various functionalities in a distributed manner through collaborating with each other. The presence of an on-body terminal, which is a PocketPC, raises the following question: Why in-network processing is beneficial when PocketPC has much more processing power. In many applications, in-network processing can reduce the amount of data sent to the terminal significantly and plays a major role in reducing communication power consumption.

Moreover, in multi-hop networks, data transmission to a base station/terminal increases the power consumption of intermediate nodes and also can inject delay to the data processing procedures. Many of these issues can be addressed using in-network processing.

3.3 Mobile storage

The on-body terminal collects the sensed data from Mednodes and acts as a temporary storage unit. A potential benefit of the on-body terminal is the ability for the data to be remotely monitored by a healthcare provider without specific action by the patient - data can be "automatically" uploaded to the medical enterprise for review. Therefore, the stored/pre-processed data can be constantly uploaded to a destination through possible underlying wireless infrastructures. Three modes of wireless communication, Blootooth, wireless local area networks (WLAN) and general packet radio service (GPRS), can be used to enable such connections via the On-Body terminal device. GPRS [6] allows data rates of 115 kbps (0.115 Mb/s) and, theoretically, of up to 160 kbps (0.16 Mb/s) on the physical layer. EGPRS [6] (enhanced version of GPRS) is capable of offering data rates of 384 kbps (0.384Mb/s) and, theoretically, of up to 473.6 kbps (0.4736 Mb/s). In comparison, WLAN has a superior data rate (e.g., 54 Mb/s for 802.11g) and a cheaper connection cost, but smaller coverage area and limited availability.

3.4 Reconfiguration

Traditional embedded systems require flexibility and reconfigurability for effective operation. Devices used in medical systems require a greater amount of flexibility and usability from the end-user's perspective (i.e., physician, patient). Overly complicated operational requirements for wearable sensors/devices (e.g., changing sensors) will be a clear barrier to their adoption in point-of-care environments. Applications of wearable sensors in point-of-care environments require solutions that are easily and quickly customizable, and can evolve as requirements and standards change. Reconfigurable cores in our wearable systems can provide this required degree of adaptability and reconfigurability. However, flexibility in reconfigurable systems comes at the expense of configuration *time* and *power*, and can be a serious performance bottleneck when dynamic and realtime configuration is needed. From our preliminary work on a hardware-based solution [19] [18] [16], a potential answer comes from a new software-based architecture that will reduce the (re)configuration and synthesis time for systems that use reconfigurable logic. The most important component of our system is the software parameterized blocks (SPB). Mednodes are the target architecture for SPB. These blocks enable the system to be flexible; however, their basic structure remains fixed on the chip almost all the time and thus reconfiguration time is no longer a severe limitation. Furthermore, synthesis onto systems with a large number of SPBs is extremely fast. The SPBs can possess several parameters such that they can be complex enough to suit a range of applications.

In addition to SPBs, we plan to include fully reconfigurable fine grain logic for random logic implementation. Functions that cannot be covered by the SPBs will be mapped onto that portion of the hardware as well as the registers used for storage. Interconnect architecture will provide an interface between the SPBs and the fully reconfigurable logic. PLAs and FPGA are suitable architectures for this purpose but at this stage of development, an FPGA chip for example, can not easily be integrated to a small wearable system with minimum overhead.

Each individual software block can be parameterized in various ways. Noticeably, many medical applications involve specific classification and detection algorithms to identify a phenomenon (e.g., an ECG P wave). Such methods can be standardized such that by resetting known parameters, the application can switch between algorithms. To do so, first we will study the structure and the topology of each algorithm, clustering the functionalities and classifying input/output formats. Next, to create the standards, each cluster will be represented as a series of parameters and a list of software interfaces.

4. INSTANCES OF MEDICAL APPLICATION

We have designed and implemented several medical monitoring and assessment applications using our proposed architecture. These systems give the physicians the capability of continues patient monitoring and also will give the patients the flexibility of having their regular daily activities while being monitored.

Figure 4: The implanted Active pressure monitoring system; The pressure sensor is shown in this picture

4.1 In Vivo Active Pressure Monitoring System

Abnormal pressure applied to upper and lower urinary track, especially in the kidney, renal pelvis, bladder, and prostate, which is caused by obstruction between kidney and uterus, increases the risk of formulation of kidney stone, infection and eventually an irreversible damage to the kidneys [14].

We utilized our architecture to create an active implantable pressure sensing system [5], which continuously measures and monitors the pressure on the elevated pelvic and ureteral renal. This system actively monitors the pressure within the upper urinary track. The real-time sensed data (pressure) gets transmitted to a mobile device carried by patient. The mobile device sends the collected data to a remote central data bank and it will be used to analyze the patients condition. At this stage, we implanted the device inside a pig's body. Figure refpig shows an x-ray image of the pressure sensor after implant surgery.

4.2 Evaluation of Myotatic Stretch Reflex

It is possible to assess the severity and monitor the recovery of people who are suffering traumatic brain injury using myotatic stretch reflex. Usually the clinical evaluation of myotatic stretch reflex is based on Ashworth scale in between 1 to 5. However the this scale can be interpreted in different ways since it is not quantified. In [26] we used our architecture to build a system which achieves full quantification of myotatic stretch reflex. The system consists of a hammer and Mednodes attached to the hammer and patient's ankles. The Mednodes are equipped with 3-axis accelerometers. The input force is quantified based on the hammer's initial potential energy. Mednodes measure acceleration in both the impact and the ankle reflex. The reflex time, (Rt), is extracted from the collected data by measuring the time difference between the impact and reflex. This time interval can be used as a quantitative measure of myotatic stretch reflex.

4.3 Clinical Assessment System for Neurological disorder

Loss of upper limbs' functionality is a common symptom in people who suffer from stroke and spinal cord injury. To improve the patients' limb functionality, physicians assign exercises to the patients. In order to assess the exercises, which directly projects the motor functions, ordinal scale is being used. Quantified measurement of motor function is a fundamental contribution to better understanding of the neuromotor injuries.

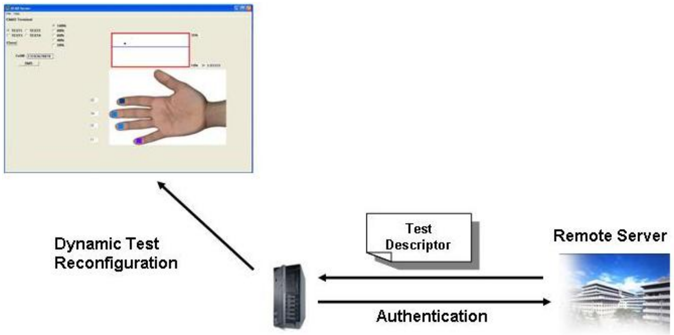


Figure 5: Reconfigurable CMAS

CMAS is a clinical movement assessment system which facilitates the quantification of motor effect following neuromotor injury [25]. The system has a PocketPC as the on-body terminal. The hand device has pressure sensors attached to it. The pressure sensors measure finger forces in flexion and extension. PocketPC displays the test procedure that patientd need to follow and records the data collected from hand device. PocketPC also sends the collected test results to a designated location. It is also possible to configure the program running on the PocketPC to transfer the collected test results to the physician for real-time or off-line monitoring/processing.

Before starting each test session, system first set appropriate parameters, tailored to each individual use by recording user's maximum voluntary contraction. After the calibration process, PocketPC runs a test and the patient should follow a moving cursor by applying the appropriate force to the hand device.

Based on the physician's assigned test for each individual user, the system is able to reconfigure itself to display new test case every time user decides to use the device. Physicians create new test case based on patient's improvement process. The remote system in the physicians site stored a test descriptor for each user associated with that server (The physician's patients). Every time the user launches the program, client side host system (either PocketPC or PC) connects to the server located in the physician's side and downloads test descriptor, is any exists. In the client side, the user interface describing the test is getting constructed dynamically based on the downloaded test descriptor. After the data collection is done, the transferred data is examined and the new test is constructed if nessesary

5. RECONFIGURATION IN BODYNET

5.1 Types of Reconfiguration

The architecture of our proposed system, enables us to use different reconfiguration approaches to achieve fault tolerance or to dynamically adapt the system to perform different functionalities without system redeployment. Most of the these approaches rely on software reconfiguration and utilizing SPBs described in section 3.4. Hardware reconfiguration is quite costly and power inefficient procedure, which makes it inappropriate for body area network.

5.2 Software Reconfiguration

We have the capability of performing software reconfiguration in our proposed architecture. Several software reconfiguration techniques has been utilized based on individual application need. Each of these techniques, which is described in the following sections has its tradeoffs with respect to flexibility , reconfiguration speed, and energy efficiency [3].

5.2.1 Full Image Upgrade

Initially, in our proposed architecture, the processing unites are programmed to perform set of tasks. It is sometimes necessary to change the whole application running on the processing unites during the operation. Full binary image update gives the user the capability of changing the program running on the processing unite without redeploying the system. To preform full image binary update we are using TinyOS [11], which is an operating system designed for sensor networks. TinyOS provides a bootloader and network protocol called Deluge [13], which facilitates the the full image update of a TinyOS application.

In full image update the desired new application transfers to the processing unit by replacing the old application binary image. It is also possible to have more than one image installed on the processing mote (As much as size permits), and reboot the desired image based on need. We perform both image upgrade and reboot over the air. Installing the full image in a node will cause the node's operation to interrupt, therefore the sensed and processed data stored in the node will be lost. This approach demonstrates very high update cost while allowing maximum flexibility.

5.2.2 Parameter update

In some of our applications the reconfiguration can be archived by changing some parameters inside the application running on the processing unit. For example it is sometimes necessary to change the sampling rate or data transmission rate in order to increase/decrease the precision or the power consumption. In digital signal processing applications, changing the coefficients of a digital filter will change the characteristics of the filter and therefore the application can adapt to different levels of filtering. Our proposed architecture is reconfigurable using parameter update. This parameters can be send to the processing node in the network (Mednode) using wireless medium. Once the parameters gets updated the system will reconfigure itself to a new operation mode without extra cost. This method reduces the reconfiguration cost compared to full image replacement but reduces the flexibility of the reconfiguration. The code below simply demonstrates how mode switching is an example of parameter update.

```
if(OperationMode == 0)
Call procedure A
else
if(OperationMode == 1)
Call procedure B
else
if(OperationMode == 2)
Call procedure C
...
```

5.2.3 Modular Binary Update

Some applications require to incrementally update the programm running on the mote. In addition it is desired to have the module injection and deletion without interrupting the operation to avoid critical data lost. We use SOS, an operating system designed at UCLA, which gives the user the capability of module insertion, removal and deletion which uses module linking and loading in run time [4].

5.3 Problem Definition

Lightweight embedded systems used in mobile devices rely on batteries as their power source. Wearable embedded systems utilize multiple lightweight components which all have their own battery. Noticeably, power consumption becomes one of the most important characteristics of such systems. Not only architecture level power optimization techniques can improve system life time in terms of power, but also dynamic reconfiguration in behavioral level can reduce power consumption. In this paper, we represent the problem of power consumption due to wireless communication and will propose a solution through dynamic rate assignment. Each Mednode in our architecture, samples data at a given rate $r samples/sec$. The sampled values create data packets and are transmitted wirelessly to the on-body terminal. Therefore there is a linear relation between the number of packets sent and the sampling rate in given period of time. Collected data will be use to detect/classify various events and characteristics. For instance, data from each individual ECG channel can be used for heart beat detection, QRS classification and extracting S-T elevation.

Each data channel (i.e. Mednode) contributes to a decision process in the on-body terminal or the server. The accuracy of the event detection and/or feature extraction process depend of the amount of information gathered which itself has a direct relationship with sampling rate; the higher the sampling rate is the more accurate the decision making process can be. At the same time, the overall accuracy of a process varies in time depending on the severity of conditions. As observed from above discussion, sampling rate has opposing effects on system characteristics in terms of power and accuracy. Hence we will formulate this problem in section 6 and introduce a methodology for dynamic sampling rate assignment.

6. DYNAMIC PARAMETER ASSIGNMENT

We consider a body area network with multiple Mednodes each connected to sensors. A sensor-Mednode pair is represented with a single node in this network abstraction. We model this network as set of nodes which are directly connected to the on-body terminal via wireless medium since in bodynets, all nodes are within wireless communication of each other. This topology creates a star network where the on-body terminal stands in the middle and collects the data received from other nodes. Assume we have $n+1$ nodes in this network (n sensors and one terminal node). For each node v_i we have the following parameters:

- r_i: sampling rate at which sensor readings are sampled.
- $d_i = \alpha r_i$: transmission rate, which is linearly dependant on sampling rate. For simplicity we assume $\alpha =$

1, which means each data is sent via a single packet.

- e_i: probability of packet loss in the communication channel between node i and the terminal
- $g_i(r_i)$: Accuracy function for node i; This function represents the accuracy of a detection algorithm G as a function of available data rate. Note that if the detection algorithm is used on the terminal as opposed to the node itself, the accuracy will be $g_i(r_i(1-e_i))$ since the available data for processing in the terminal is $r_i(1-e_i)$
- $P_i = K \times r_i$: power consumption of node i as a linear function of transmission rate due to wireless communication which dominates the total power consumption of individual nodes.

6.1 Problem Statement

In this section we formulate the problem sampling rate assignment for power minimization while meeting accuracy constraints imposed to the system. The total power consumption of the nodes can be represented as:

$$P_{total} = \sum_i P_i = \sum_i K \times r_i \tag{1}$$

Our applications such as heart beat detection or QRS classification, rely on the results of individual channels. The total accuracy of the system, Θ can be stated as a function of individual accuracies:

$$\Theta = \Phi(\vec{g}) = \sum_i \beta_i g_i(r_i(1-e_i)) \tag{2}$$

Generally speaking, Φ can be any nonlinear function depending on the application and algorithm. But in many voting based detection algorithms such as heart rate detection, it is a linear average of all individual accuracy function.

Now, the problem can be stated as:

$$minimize\ P_{total} \Rightarrow minimize \sum_i r_i \tag{3}$$

such that

$$\Theta \geq X\%$$

Where $X\%$ is the desired overall accuracy of the detection/classification algorithm.

On the other hand, the objective in Equation 1 minimizes total power consumption whereas since each node has its battery source, the result of rate assignment may over load one node and result in shutdown. Hence, if longer life time of individual nodes is desired, the objective function would be:

$$minimize(max(P_i)), \forall i \tag{4}$$

Objective defined in 4 guarantees maximizing system life time while meeting accuracy constraint. In this objective, the maximum power consumption is minimized and tries to evenly distribute power consumption.

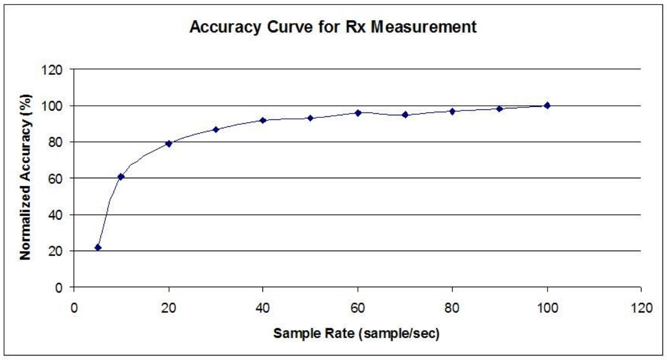


Figure 6: Accuracy of reflex time measurement vs data sampling rate. The curve has been normalized to the measurements of maximum sampling rate

Solution to the above optimization problem, sets individual sampling rates such that the overall accuracy of the system meets specification constraints. We observed that accuracy individual functions are convex with respect to sampling rate. Figure 2 shown the experimental results for reflex time detection in Myotatic Stretch Reflex device illustrated in section 6. Intuitively, an increase in sampling rate will not cause loss of information. Therefore accuracy functions is an non-decreasing function of sampling rate. On the other hand, intuitively, for large sampling rates, an small increase has less effect on accuracy compared to an increase when sampling rate is smaller. This qualitative argument indicates the convexity of accuracy function. This attribute enable solving the optimization problem efficiently.

7. SIMULATION RESULTS

To demonstrate the effectiveness of sampling rate assignment, we studied one of the algorithms used in myotatic stretch reflex device namely reflex time measurement. In this application, of the objectives is to measure the time difference between the instant the reflex hammer hits the knee and the physical reflex time. To measure this time interval (which we call it *Rt*), we placed two Mednode on the reflex hammer and the ankle of the person under test. Each Mednode was equipped with a 3-axis accelerometer. First, for different sampling rates shown in Figure 6.1 we measured *Rt*. For each sampling rate, we repeated the test 14 times and we ran this experiment on four people in our lab. Since we had no reliable frame of reference to compare the accuracy of our measurement, we placed a second pair of Mednodes which were set to sample at the maximum 100 $samples/second$ and normalized the accuracy of measurement by comparison to the 100 $samples/second$ case.

In our experiments, the on-body terminal was held at close proximity of the Mednodes to minimize the packet loss. After collecting the information, we simulated this system in the presence of different packet loss probabilities. Figure 7 summarized the simulation results. x-axis is the packet loss probability and y-axis represents the percentage of data rate compared to sampling at maximum of 100 $samples/second$. It is observed that by setting the appropriate sampling rate, a power reduction of 10% to 60% can be archived on average. Note that the desired accuracy was set to 90% and 95%

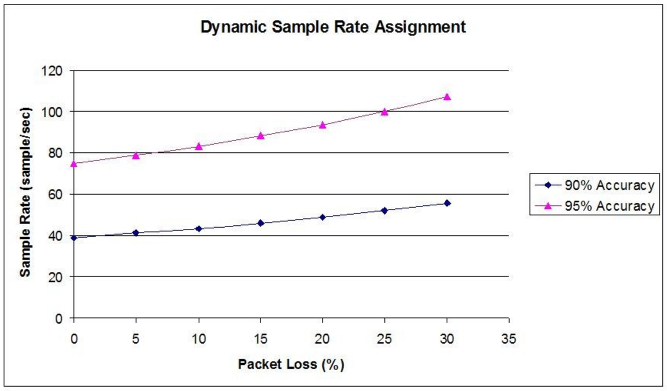


Figure 7: Optimal sampling rates for a given accuracy and packet loss rate

which are arbitrary numbers.

8. CONCLUSION

In this paper, we introduced a general purpose body area architecture targeting medical application. Our proposed platform utilizes lightweight embedded systems equipped with various sensor to monitor physiological signals. We reviewed three different applications which are made based on our platform. Furthermore, as a part of system capabilities, we expanded the concept of reconfiguration in body sensor networks and introduced new means of reconfigurations embedded in our system. Finally, the problem of dynamic data sampling rate assignment was formulated and through experimentation and simulation, we showed how effectively power savings can be achieved while meeting accuracy specifications of the application.

9. REFERENCES

[1] Oliver Amft, Gerhard Troster, Paul Lukowicz, and Corina Schuster. Sensing muscle activities with body-worn sensors. In *BSN '06: Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks (BSN'06)*, pages 138–141, Washington, DC, USA, 2006. IEEE Computer Society.

[2] Reisner A. Sokwoo Rhee Asada H.H., Shaltis P. and Hutchinson R.C. Mobile monitoring with wearable photoplethysmographic biosensors. *IEEE Engineering in Medicine and Biology Magazine*, 22(3):28–40, 2003.

[3] Rahul Balani, Chih-Chieh Han, Ram Kumar Rengaswamy, Ilias Tsigkogiannis, and Mani Srivastava. Multi-level software reconfiguration for sensor networks. In *EMSOFT '06: Proceedings of the 6th ACM & IEEE International conference on Embedded software*, pages 112–121, New York, NY, USA, 2006. ACM Press.

[4] R. Shea E. Kohler C. C. Han, R. Kumar and M. Srivastava. Sos: A dynamic operating system for sensor networks. In *Third International Conference on Mobile Systems,Applications, And Services (Mobisys)*, 2005.

[5] Foad Dabiri Tammara Massey Robert Tan Majid Sarrafzadeh Mani Srivastava PG Schulam Jacob Schmidt C.K. Lin, David Jea and CD Montemagno. The development of an in-vivo active pressure monitoring system. In *The 4th International Workshop on Body Sensor Networks (BSN 2007)*, 2007.

[6] Ericsson Communications. Edge: Introduction of high-speed data in gsm/gprs networks. In *Ericsson technical white paper*, Last accessed January 9, 2007.

[7] Koutsouris D. Dembeyiotis S, Konnis G. Integrating legacy medical data sensors in a wireless network infrastructure. In *Proceedings of 27th Annual International Conference of the Engineering in Medicine and Biology Society*, September 2005.

[8] Chris Otto Emil Jovanov, Aleksandar Milenkovic and Piet C de Groen. A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 2, 2005.

[9] Welsh M Juang RR Alm A. Gao T, Greenspan D. Vital signs monitoring and patient tracking over a wireless network. In *Proceedings of 27th Annual International Conference of the Engineering in Medicine and Biology Society*, 2005.

[10] 2007 GSM association study. www.textually.org/textually/archives/2006/10/013841.htm, Last accessed January 12.

[11] Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David E. Culler, and Kristofer S. J. Pister. System architecture directions for networked sensors. In *Architectural Support for Programming Languages and Operating Systems*, pages 93–104, 2000.

[12] http://www.xbow.com/.

[13] Jonathan W. Hui and David Culler. The dynamic behavior of a data dissemination protocol for network programming at scale. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 81–94. ACM Press, 2004.

[14] Piper I. Urethral pressure measurement. *Scandinavian journal of urology and nephrology. Supplementum*, 207:61–66.

[15] O'Donnell Lords A. Raskovic D. Cox P.G. Adhami R. Andrasik F. Jovanov, E. Stress monitoring using a distributed wireless intelligent sensor system. *IEEE Engineering in Medicine and Biology Magazine*, 22(3):49–55, 2003.

[16] R. Kastner, A. Kaplan, S. Ogrenci Memik, and E. Bozorgzadeh. Instruction generation for hybrid reconfigurable systems. *ACM Trans. Des. Autom. Electron. Syst.*, 7(4):605–627, 2002.

[17] Andreas Krause, Daniel P. Siewiorek, Asim Smailagic, and Jonny Farringdon. Unsupervised, dynamic identification of physiological and activity context in wearable computing. In *ISWC '03: Proceedings of the 7th IEEE International Symposium on Wearable Computers*, page 88, Washington, DC, USA, 2003. IEEE Computer Society.

[18] S. Memik, E. Bozorgzadeh, R. Kastner, and M. Sarrafzadeh. Sps: A strategically programmable system. In *Proceedings of Reconfigurable Architecture Workshop*, 2001.

[19] S. Ogrenci Memik, E. Bozorgzadeh, R. Kastner, and M. Sarrafzadeh. A super-scheduler for embedded reconfigurable systems. In *ICCAD '01: Proceedings of the 2001 IEEE/ACM international conference on Computer-aided design*, pages 391–394, Piscataway, NJ, USA, 2001. IEEE Press.

[20] Schmidt R. Perkuhn M. Reiter H. Lauter J. Thijs J. Musch G. Muhlsteff J., Such O. and Harris M. Wearable approach for continuous ecg - and activity patient-monitoring. In *EMBC '04: Proceedings of the 26th Annual International Conference of the Engineering in Medicine and Biology Society*, pages 2184 – 2187, 2004.

[21] Kazushige Ouchi, Takuji Suzuki, and Miwako Doi. Lifeminder: A wearable healthcare support system using users context. *icdcsw*, 00:791, 2002.

[22] S. Popovic M.R. Dietz V. Morari M. Pappas I.P.I., Keller T. Mangold. A reliable gyroscope-based gait-phase detection sensor embedded in a shoe insole. *IEEE Sensors Journal*, 4(2):268–274, 2004.

[23] Gerhard Troster Paul Lukowicz, Urs Anliker, Steven J. Schwartz, and Richard W. DeVaul. The weararm modular, low-power computing core. *IEEE Micro*, 21(3):16–28, 2001.

[24] Rohman S Santarelli C et al. Pollard JK, Fry ME. Wireless and web-based medical monitoring in the home. *Medical Informatics and the Internet in Medicine*, 27:217–219, 2002.

[25] David Jea Mani Srivastava Majid Sarrafzadeh Robert LeMoyne, Roozbeh Jafari. Cmas: Clinical movement assessment system for neuromotor disorders. In *Neuroscience 05, Symposium of Neuroscience*, November 2005.

[26] V. Reggie Edgerton Majid Sarrafzadeh Roozbeh Jafari, Devin L. Jindrich. Fully quantified evaluation of myotatic stretch reflex. In *Neuroscience 05, Symposium of Neuroscience*, November 2005.

[27] Morris S.J. and J.A. Paradiso. Shoe-integrated sensor system for wireless gait analysis and real-time feedback. In *Proceedings of the Second Joint Engineering in Medicine and Biology 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES*, pages 2468 – 2469, 2002.