Two ways of approaching sensor nodes design

Maximilian Nicolae, Radu Dobrescu, Dan Popescu

***Abstract*—** The paper presents two way of approaching sensor nodes design, one from the energy consumption point of view and the other from the high performance point of view. Both ways are taking in consideration the cost efficient criterion. The low energy consumption, low size and low cost are achieved classical using low power microcontrollers and optimizing the software written on them. The other approach was to use a digital signal processor (DSP) for signal processing, an audio codec for data acquisition and a 802.11g wireless access point for communication. In this case the entire network was seen initially as a Voice over IP (VoIP) mobile network, yet the information exchanged wasn't voice but measurements and commands with Quality of Service (QoS) inherited from VoIP. The usage of mature technologies of audio codecs and wireless LANs assured a high performance and cost efficient solution for mobile sensor networks. The platform's modularity and flexibility allow its usage for many applicative didactical activities and also in research, as logistic support for developing complex projects.

***Keywords*—** Mobile Sensor Networks, Digital Signal Processor, Harvard architecture, Direct Memory Access, audio codec, wireless infrastructure, radio modem.

I. Introduction

WIRELESS sensor networking is one of the most essential technologies for implementation of ubiquitous computing. Sensor networks will be applied in variant environments, i.e. health care, military, environment, warehousing and transportation management. The sensor nodes are usually scatted in a sensor field and data are routed back to the sink by multi-hop. These sensor networks usually share the same communication channel. Sensor nodes have limited in power, computational capacities, memory and short-range radio communication ability. The limited battery life of sensor nodes raises the efficient energy consumption as a key issue in wireless sensor networks [1], [2], [3]. There are four major sources of energy waste; collision, overhearing, control packet overhead and idle listening. Collision of transmitted packets increases energy consumption due to the follow-on retransmissions. Overhearing also spends unnecessary power since a node picks up packets that are destined to other nodes. Sending and receiving control packets consumes energy too. Idle listening meaninglessly consumes battery power by listening to receive possible traffic that is not sent [1].

The aim of the first approach is the implementation of a wireless sensor node with specific facilities for integration in a sensor network. The design offers an association of the data acquisition function with the communication function based on a low cost microcontroller structure and a radio modem. The performances of this device: low power consumption, reduced error bit rate, high computational capacities are fair enough to recommend it for use as a node in a wireless sensor network and justify its acronym: ISNN – Intelligent Sensor Network Node.

In a mobile sensor network the number of nodes may vary from few units to several thousands and could have a large density. According to the application which the network is designed for, there may be some constrains about nodes like overall dimensions, energy consumption, acquisition performance of measured quantities (resolution and sampling rate), etc. The nodes density, the information quantity spread by every node, communication channel multiple access, routing algorithms, all of this lead to loading and congesting the network and are very important factors for efficiency of the network]. If we also add the node price, we obtain a picture with enough complexity towards designing such a network and its nodes. References as [4], [5] and [6] discuss different procedures to optimize the power consumption.

Endowing sensorial nodes from a sensor network with DSP cores it could lead to high performance because of the possibility of parallel and distribute processing and thus reducing the quantity of information spread into the network (network load). If such a node uses for communication a mature infrastructure like 802.11 standard, will be obtained a solution for implementing mobile sensor networks with high performance end cost efficient. In the same way of obtaining high performances with low costs we suggest that on the information chain from the physical quantity to the numerical result, the acquisition part to be done by an audio codec.

Our second approach was to try to analyze some of these criterions and select those mature solutions which will ensure cost efficient implementation. Towards reducing the quantity of information transmitted by each node without compromising the performance of the system leads to a local signal processing at the node level. Concerning the communication medium multiple access, an 802.11g network ensure a mature solution, efficient from the costs and performance points of view, including the QoS.

Getting close to the monitored/controlled process, it can be used the same approach also for the acquisition and command part, therefore utilization of analog to digital and digital to analog converters used in audio codecs ensures high performances because of the domain in which the audio codecs are used and the high level of production diminish their price.

Manuscript received July 8, 2007. This work was partially supported by the Romanian Ministry of Education and Research under Grant PN2/CNMP – PC 11-040/2007 and Grant CEEX/INFOSOC - 72/2006.

The authors are with the POLITEHNICA University of Bucharest, Faculty of Control and Computers, Splaiul Independentei 313, Bucharest 7100, Romania (telephone: +40214029105, e-mail: max@aii.pub.ro).

II. NODE ARCHITECTURE

A sensor module consists of several functional blocks as seen in Fig. 1. Each block has a clearly delineated task in the sensor module's operation.

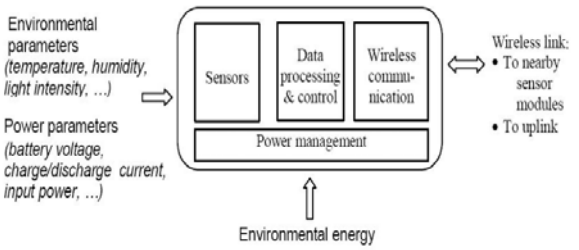

Fig. 1 Block schematic of a wireless sensor

The specific requirements for a particular block's tasks can differ depending on the application:

- A large-area low-density sensor network deployment will require a more powerful radio than a short range indoor or on-the-body sensor application.
- Applications where high data rates and complex signal processing functions are required will benefit from a more powerful signal processor.
- Sensors and associated sensor electronics will vary from application to application.
- For some applications the power can be provided by an appropriate primary battery, whereas others call for a complete power management system that can scavenge energy from the environment.

Therefore if a wireless sensor module architecture is to be suitable for a broad range of applications, it should be designed in a modular fashion. The blocks in Fig. 1 were implemented as distinct hardware layers which can be plugged together like Lego blocks to obtain a sensor module with the desired functionality. The data processing & control block can be reprogrammed in-system to provide this application-tailored functionality.

The target hardware architecture is based loosely and consists of power sources, SRAM memory, CPU core, general purpose I/O, RF communication unit and digital logic for the embedded operating system.

The mixed signal processing (both digital and analogical) of the microcontroller device can be used to interface to general purpose I/O, this will be necessary to connect sensors, actuators, memory and network interfaces (both RF and conventional) to the CPU. It can also be used to implement custom hardware accelerated user-instructions for the CPU core. Processing that does not map well to the CPU instruction set, or whose computationally requirements make it difficult to meet real-time performance constraints are candidates for implementation in hardware logic. Nodes requiring digital signal processing of audio/video data may require such functionality.

The ISNN software architecture responds to the requirements of the software framework of the whole network. The components of the framework provide the functionality of single sensors, sensor nodes, and the whole sensor network. According to these components, applications are classified into *sensor applications*, *node applications* and *network applications*. The software implemented on ISNN corresponds to the first to levels (Fig. 2).

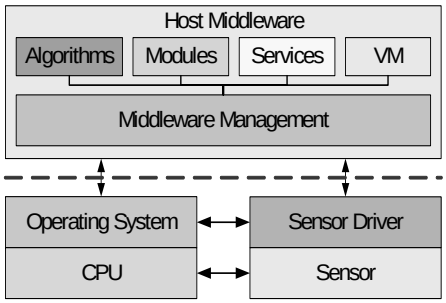

Fig. 2 ISSN software architecture block scheme

A *sensor application* contains the readout of a sensor as well as the local storage of data. It has full access to the hardware and is able to access the operating system directly. The sensor application provides essential basic functions of the local sensor node, which may be used by the node application. The *node application* contains all application specific tasks and functions of the middleware to build up and maintain the network e.g., routing, looking for nodes, discovering services, and self localization. The term *middleware* refers to the software layer between operating system and sensor application on the one hand and the distributed application which interacts over the network on the other hand. Primary objective of the middleware layer is to hide the complexity of the network environment by isolating the application from protocol handling, memory management, network functionality and parallelism. A middleware for sensor networks has to be· scalable, generic, adaptive and reflective. Resource constraints (memory, processing speed, bandwidth) of available node hardware require an optimization of every node application. Thereby, the application is reduced to all essential components and data types and interfaces are customized (*scalable middleware*). The components of the middleware require a generic interface in order to minimize customization effort for other applications or nodes (*generic middleware)*. The mobility of nodes and changes of infrastructure require adaptations of the middleware at runtime depending on the sensor network application. The middleware must be able to dynamically exchange and run components (*adaptive middleware*). Reflection covers the ability of a system to understand and influence itself. A reflective system is able to present its own behavior. Thereby, two essential mechanisms are distinguished – the inspection and the a adaptation of the own behavior (*reflective middleware).*

A node from a sensor network may be thought modular, on levels, towards being much flexible in implementation (Fig. 3). Thus, the node may be connected to a wireless network through an Access Point (AP) or an Ad-Hoc architecture with other nodes, but also may be connected to an Ethernet

network, GSM, 3G or ADSL only through small modification at the communication controller level. Towards reducing these modifications and increase the flexibility it can be chosen a specialized DSP for networks which will have network protocol stacks incorporated. Direct Memory Access (DMA), word width, many Arithmetical-Logical Units (ALU) and fast calculation with floating point numbers, recommend utilization of DSP processor as a best choice in this case.

The DSP processors have a Harvard architecture which implies two separate memory buses. This permits simultaneous accessing of two memory locations. In general use processor, one of the buses is dedicated for reading the instruction code, and the other for fetching the operands. The Harvard-DSP architecture allows the instruction bus to be used for operands reading. Because we need simultaneous

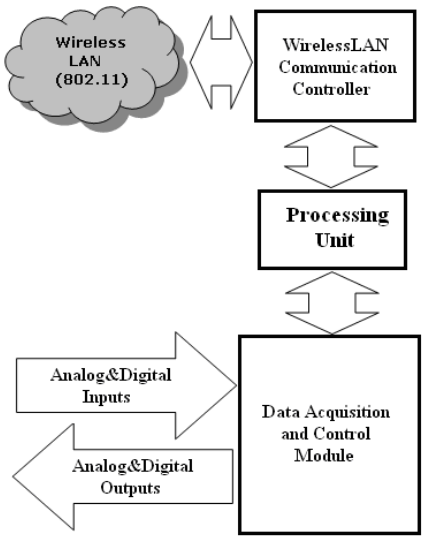

Fig. 3 Architecture of a wireless sensor node

three information values, the Harvard-DSP includes a cache memory which can store the common instructions which will be reused, leaving both buses free for the operands. This extension it's called Super Harvard ARChitecture (SHARC) [7].

There is also a disadvantage of the Harvard architecture. A DSP that processes 32-bit words requires at least 64 pins for each data bus, so that makes a total of 128 pins for exterior connection. The result is a very large sized chip, hard to place in dedicated circuits.

For the control and data acquisition module we propose using a dedicated audio codec who has analogue to digital and digital to analogue integrated converters with high resolutions (up to 32 bits), with a sampling frequency reaching hundreds of kHz, high sensibility for high frequencies too and incorporated digital filters [8].

III. Implementation Issues

There are three main hardware units incorporated in the ISNN structure: the communication unit, the central processing unit and the data acquisition unit. Each unit is single board implemented around a dedicated chip.

For the first approach the communication unit is based on the TRF6901 chip, an integrated circuit intended for use as a low cost FSK transceiver to establish a frequency-programmable, half-duplex, bidirectional RF link. The multichannel transceiver is intended for digital modulated applications in the free 868-MHz European band and the North American 915-MHz ISM band. The single-chip transceiver operates down to 1.8 V and is designed for low power consumption. The synthesizer has a typical channel spacing of better than 200 kHz and uses a fully-integrated VCO. Only the PLL loop filter is external to the device. The transmitter consists of an integrated VCO and tank circuit, a complete integer-N synthesizer, and a power amplifier. The divider, prescaler, and reference oscillator require only the addition of an external crystal and a loop filter to provide a complete PLL with a typical frequency resolution of better than 200 kHz. Since the typical RF output power is approximately 9 dBm, no additional external RF power amplifier is necessary in most applications. The integrated receiver is intended to be used as a single-conversion FSK receiver. It consists of a low noise amplifier, mixer, limiter, FM/FSK demodulator with an external LC tank circuit or ceramic resonator, a LPF amplifier, and a data slicer. The received strength signal indicator (RSSI) can be used for fast carrier sense detection or as an on/off keying demodulator.

The TRF6901 was easily interfaced to a baseband digital signal processor such as the Texas Instruments MSP430 ultralow-power microcontroller, that is the core of the processing unit in this case. The TRF6901 serial control registers are programmed by the MSP430, which performs also baseband operations in the software. According to the application requirements, one can choose the design complexity necessary to implement the system, a data rate and the coding scheme, and one can determine the number of required frequency channels. Wireless transceiver designs can be loosely classified into three groups according to performance and design complexity. Simple designs are often characterized by low data rates (10 kbps), very low cost targets, and short transmission ranges (under 100 m). Cost is the primary driver in making design choices. Complex designs have higher performance goals that may include higher data rates or longer transmission ranges (over 100 m). Performance largely determines design choices. Intermediate designs fall between the two extremes, where additional system costs are carefully traded against the benefits of increased performance. In the implemented solution the DSP contains additional software that ensure three optional supplementary signal processing functions: adaptive equalization for canceling inter-symbol interference, error correction to minimize the bit error rate and coded modulation (QAM type) for increasing the transmission speed.

The choice of coding scheme has important implications for several parts of the transceiver design, including loop filter bandwidth, frequency deviation, and operating the TRF6901 in learn and hold modes. Systems with low data rates (2.4 kbps to around 30 kbps) are often implemented with Manchester coding. There is a voltage change every bit (or

symbol) period, hence the dc content of the data is zero or constant. In FSK systems, this means that the bit rate is the same as the rate at which the transmitter toggles between frequencies (symbol rate, data rate, or frequency modulation rate). If Manchester coding is used, the TRF6901 can be operated in the learn or hold modes when receiving data. The training sequence, used during the learn mode, is still required to establish a reference voltage at the sample-and-hold capacitor. Systems with data rates higher than 30 kbps are often implemented with unipolar Non Return-to-zero (NRZ) coding. There is a voltage change only when the data changes from one to zero or vice versa, hence the voltage content of the bit stream and the frequency modulation rate depends on the data content. In FSK systems, this means that the bit rate is about twice the maximum frequency modulation rate. If NRZ coding is used, the TRF6901 must be operated in the learn mode while receiving the training sequence, and then the TRF6901 must be switched to the hold mode while receiving data.

The data acquisition unit has been implemented using the facilities of the MSP microcontroller. This chip contains a ADC with successive approximations with 12+2 bits resolution and an accuracy of ± 1 LSB.

The two boards are interconnected and installed in a special box together with the power supply. A top view of the realized ISNN is presented in fig. 4

Fig. 4 Top view of the ISNN

As we seen in the second approach the manufacturing process of such a platform implicates considerable efforts and an adequate technology which is rather hard to stand up at the project level. To overwhelm the eventual manufacturing errors there has been adopted as an initial solution, developing the equipment around a DSP development platform which guarantees the correct functioning assuring by this way a strictly functional approach. In this way, the development will focus strictly on the elaboration and testing aspects of the algorithms

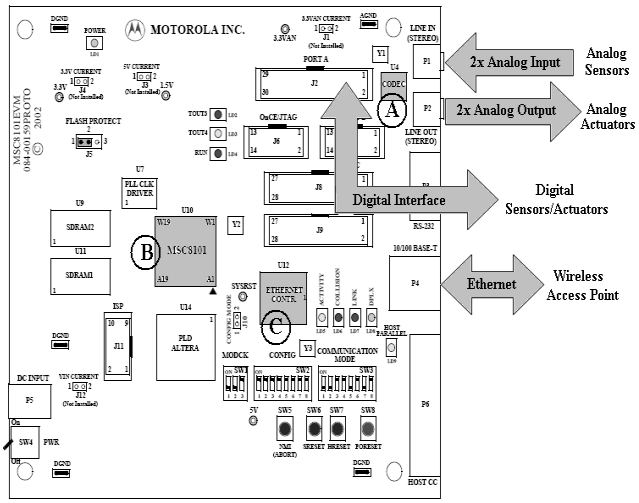

Fig. 5 MSC8101EVM developing board architecture:
A-stereo audio codec for analog interface,
B- DSP core, C- Ethernet controller

.

There has been chosen a development board for VoIP solutions (MSC8101EVM from Freescale), around a DSP (MSC8101) from the StarCore140 family, dedicated for networks, which incorporate more protocol stacks met in computer networks. The board also contains a stereo audio codec (TLV320AIC23PW from Texas Instruments) which allows acquisitioning of three analogical channels (Line-In Left, Line-In Right and Mic-In) and two analogical output commands (Line-Out Left and Line-Out Right). Communication will be done through an Ethernet Controller towards a wireless Access Point (see Figure 2)

This implementation is modular and allows independent developing of each processing and data transmission stage. By this way algorithms for the acquisition and analogical command can be developed and tested, ignoring the communication part and exploiting the audio codec at its limits. The DMA function the DSP disposes, the external 64-bit bus working at 100MHz and the high-capacity Flash memory (4MB) on the board allows capturing the signals on large time scales at a high sampling rate (Ex: at a 16 bit/sample resolution and a 96KHz sampling frequency, over 200s can be captured). All this is being made without keeping the processor busy, due to the DMA's functionality. Having the acquisition and command parts well covered (i.e. specialized functions libraries), high performance control algorithms can be developed by using the functionalities of the DSP core on the board. The fact that this disposes of four arithmetical-logical units (ALU), 16-bit word width, high addressing capacity and large memory which can contain interpolation maps for complex functions, this allows implementing advanced control algorithms, and generating an accurate analogical command due to the codec can assure control of complex processes ordered with analogical actuators.

Further, several communication infrastructures can be evaluated, from DSL networks up to wireless networks. Being a specialized DSP for networks, it has the some protocol stacks implemented as ATM, Ethernet, IP. Also there can be evaluated routing algorithms in mobile sensor networks, assuring some requests connected to real time operating protocols (RTP), priorities and QoS [9], [10].

The idea of the architecture and implementation has left from trying to use a mature and cheaper to cheaper technology like VoIP on the wireless infrastructure, but to have it spread instead of voice, data from sensors and commands to actuators. Even in the case of VoIP compression with loses is used, the corresponding methods and protocols can be exploited with a lossless compression. Hardware accomplishing can be seen in figure 3 in which the development board is interconnected through an Ethernet cable with RJ45 connectors at a wireless Access Point.

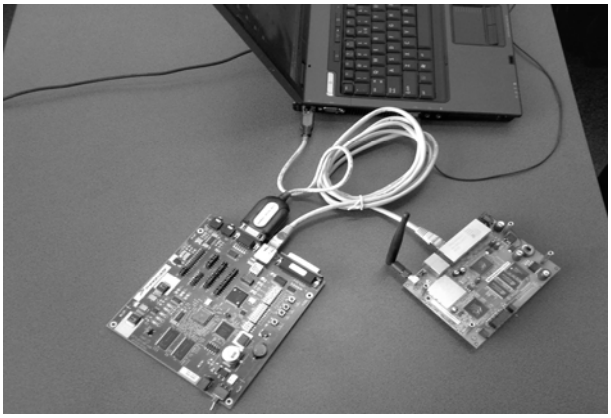

Fig. 6 Hardware implementation of sensor node with a high performance DSP core

This implementation is one for development and testing, so, according to the application such a nod is designed for, only components of the modules helping achieve the desired goal shall be kept, and will be integrated on a single board, by this way being able to satisfy the size related constrains. The fact that the modules are made to work independently allows fixing the errors easily, through isolation and then analyzing them, and the eventual fix in the end. More then that, performances can be measured independently at each level (acquisition, processing, and communication) and the adequate hardware and software development decisions can be taken

To develop the processing software that will run on the DSP, it has been used MetroWerks's Code Warrior developing environment (IDE). This disposes of a C compiler with a real time debugger.

IV. Test procedures and results

Once the solutions were hardware implemented, in order to continue, certain tests were imposed which made the subsequent developments depend on them. Those tests are referring at the possibility to capture data and to broadcast it radio, even through a wireless 802.11 network.

A. Testing the analogue acquisition and command

For the first approach the data acquisition and commands performance are given by the ADC module located inside the MSP microcontroller. Data acquisition and command possibilities for the second approach through the audio codec have been analyzed at this test. Therefore in order not to insert perturbations and noise from signal adaptors required for interconnecting sensors with the Line-IN, a method of using a signal generator has been chosen (Fig. 7). It has been set a voltage level specific to the audio signals for the Line In input. On the same line there has been connected a channel from an oscilloscope. On the line-out of the board has been connected another channel from the oscilloscope to observe the differences. For the first attempt a program was written meant to program the codec in order to gather data from a Line-in input channel at a pre-established frequency, with a certain resolution, and these samples, without being process, were sent back to the codec output buffer at a Line-Out channel. The comparison between the output signal and initial one from the input was being made using the oscilloscope, obtaining only a latency difference of the output signal.

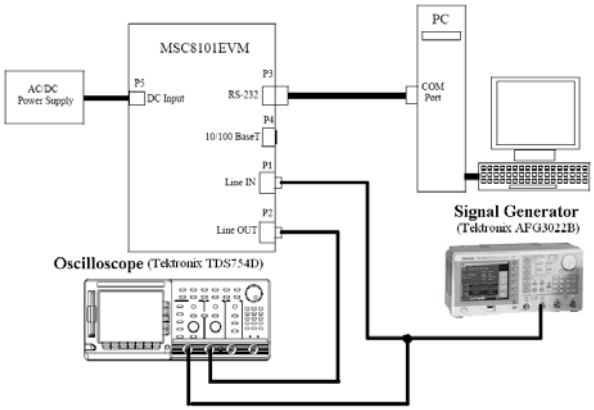

Fig. 7 Testing the acquisition/command module

For a more detailed analyze of the acquisition possibilities, to the initial program has been added the feature to change the audio codec's settings (16, 20, 24 or 32 bit resolution up to 96Khz sampling frequency) through an application that runs on the PC and communicates with the board through the RS232 serial port. Further there has been implemented the possibility to capture a sampled signal with a high resolution and frequency through the DMA so that the processor can deal with other tasks.

The results have been satisfying and allowed continuing the research in order to implement this solution for the acquisition and command part of the processes that require measuring some high fidelity parameters.

B. Testing the radio communication over 802.11 infrastructure

At this point there has been taken for granted that data is already acquired and processed and remain only to be transmitted at the distance. The testing model for the DSP architecture is illustrated in Fig. 8. The goal was to transmit previously captured signals, saved in the flash memory of the developing board through TCP/IP to a remote server. Here there was no more comparison done between the initial signal and the one received because the communication protocol itself assured error detection and retransmitting poor packets. Of high interest was the transmission rate, so when the board started to fill the Ethernet's controller transmission buffer a timer started, and when it received the last confirmation it stopped the timer, and the time obtained was related to the sent data quantity (measured in bits) and send it to the computer the obtained transfer rate.

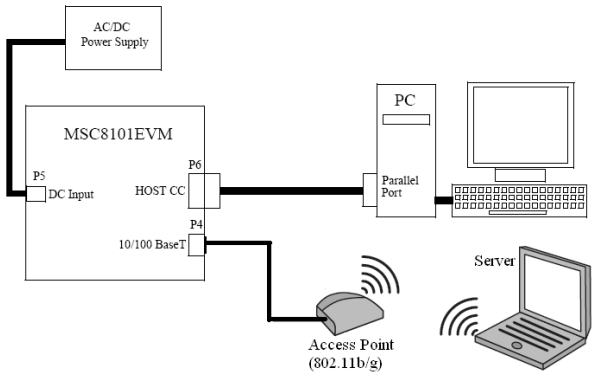

Fig. 8 Testing the communication module

The average result of several tests, with no connection problems between the Access Point and the Server, and without other 802.11 devices near by, from which the network would be overloaded and issue conflicts, was 50Mbps. Measurements have been performed for a single packet sent several times in the network, analyzing latency, time gauge between consecutive packets, jitter, etc. (Table I and II). Results have proven that packets do not reach in succession and the gauge can reach up to 10ms between two 160 bytes consecutive sequences.

TABLE I. SINGLE PACKET REMOTE TRANSMISSION PERFORMANCES FOR THE FIRST APPROACH

Packet position in frame	Sequence length [bytes]	Delay [ms]	Latency [ms]	Jitter [ms]
first	128	-	73.08	0.10
10th	128	3.87	73.11	0.11
100th	128	4.12	73.13	0.11
500th	128	7.11	73.15	0.12

TABLE II. SINGLE PACKET REMOTE TRANSMISSION PERFORMANCES FOR THE SECOND APPROACH

Packet position in frame	Sequence length [bytes]	Delay [ms]	Latency [ms]	Jitter [ms]
first	128	-	62.33	0.11
10th	128	3.15	62.35	0.14
100th	128	3.72	62.35	0.14
500th	128	6.68	62.36	0.12

A global comparing of the both implementation are presented syntactical in table III.

TABLE III. THE GLOBAL RESULTS COMPARED

Parameter	Classical implementation with mixed signal microcontroller (first approach)	High performance implementation with powerful DSP (second approach)
Power consumption of processing unit	1,2mW (low)	450mW (high)
No of ALUs/ Processing speed	1 / 4Mhz	4 / 300Mhz
Sampling rate / maximum resolution	7 kSps / 14 bits	96 kSps / 32 bits (provided by audio codec)
Maximum data transmission rate	256 kbps	50 Mbps
Cost	Cheap <10$	Approx. 50$ but reduced through our approach

V. TESTING COVERAGE STRATEGIES FOR MINIMIZING POWER CONSUMPTION

In order to minimize power consumption, the authors suggest a solution for configuring network topology without redundant working nodes, that means in other words an optimal coverage procedure. Our goal is to achieve a full coverage with a minimum number of working nodes. Working nodes are here defined as the nodes which are involved in sensing and transmitting sensory data. Given a set of sensors and a target area, no coverage hole exists in the target area, if every point in that target area is covered by at least *k* sensors, where *k* is the required degree of coverage for a particular application (Fig. 9). The sensor network should be connected at all times so that nodes are able to communicate with each other. As with the multiple coverage requirements, multiple connectivity is also desirable to guard against single link or

node failure partitioning the network

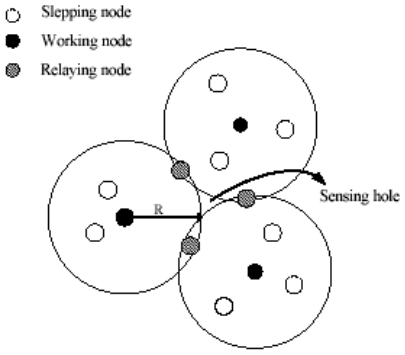

Fig. 9 The coverage principle

A simulation program was written to model larger scale networks and assess the performance of network build-up algorithms for creating the web topology. For this, we first assume a sensor network where: 1) the sensor nodes are all homogeneous and energy constrained and stationary; 2) the sensing range and the radio transmission range of sensor node are equal; 3) the sensor nodes are located arbitrarily in the dense network; 4) each sensor nodes has its unique identification and knows its position information, and knows its one-hop neighbor's position information using hello messages.

With a selected set of working nodes, there should be no sensing hole in the network to guarantee full coverage. In addition, all working nodes must be connected to the sink to forward the sensing data. For this, we introduce a concept of relay node which is defined as a node that provides interconnection between two working nodes. We consider two extreme scenarios for selecting working nodes in the sensor network (Fig.10).

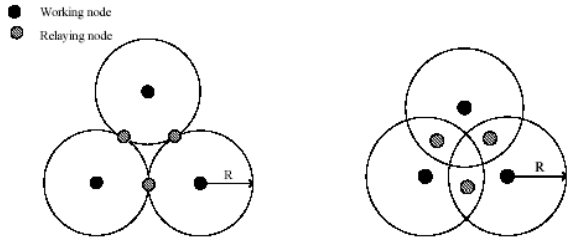

Fig. 10 Two scenarios for selecting working nodes

(i) If we select working nodes in such a way that their transmission ranges (ideally given by a circle) border on each other as seen in Fig. 10 (left), we need to place a relay node at the point of tangency of two ranges to give connectivity among working nodes. In this way, we can minimize the number of working nodes in the network. However, we will have a sensing hole (marked as shaded area) as shown in the figure.

(ii) If we select working nodes in such a way that they are very closely located as illustrated in Fig. 10 (right), we do not need to worry about coverage problem. But, we have to pay the cost for the redundancy of sensory data since the range of each working node is overlapped.

So, to configure the network topology without sensing hole or the redundant coverage problem, the working nodes should be distributed in such a way that their ranges intersect at two points making 60 degrees through the working node. At the same time, we have to put a relay node in each intersected area to obtain connectivity. In other words, the working nodes should be distributed so that their transmission ranges are intersected in hexagonal shape as shown in Fig. 11.

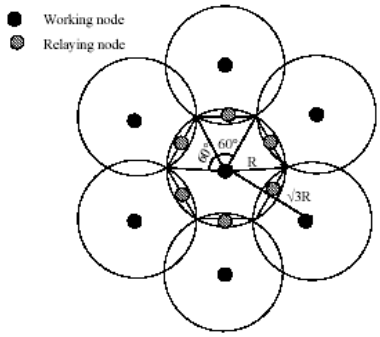

Fig. 11 Distribution of working nodes in a hexagonal shape

To evaluate our topology control algorithm, we carry out experiments by simulation to measure coverage and connectivity of the network. We use a network that has *n* nodes arbitrarily deployed in an area of 500 by 500, where *n* is in the range of [100, 1000]. The sensing range and transmission radio range are given 50. Fig. 5 shows the number of working nodes as we vary the number of deployed nodes in the network. As we count the number of working nodes which have connection to the sink in the sparse network with less than 500 deployed nodes, we can see that the number of working nodes is small. From Fig. 12, we can see that the random selection scheme generate too many working nodes. On the other hand, the network can be successfully configured with a small number of working nodes even in the dense network by our scheme.

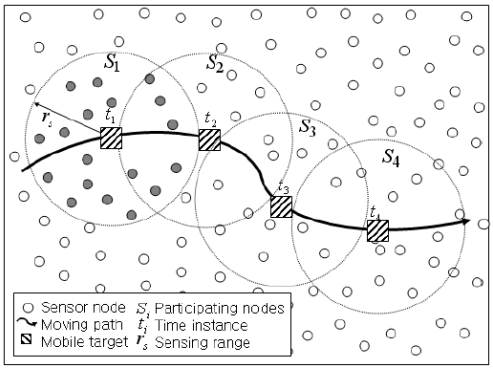

Fig. 12 A concept of tracking of a mobile object

This can be more clearly seen in the sparse network by

comparing with the number of working nodes (N_{hexa}) to be needed when the network topology is configured in such a way that all working nodes are connected in the hexagonal shape. This indicates that the network lifetime can be extended by turning off too many redundant nodes in our scheme. In this test, we do not consider the number of relaying nodes since it is simply dependent on how many working nodes are selected.

VI. Conclusion

The results presented in Table III are the key guide for selecting the proper design of wireless sensor network nodes. As it can be seen if the process meant to be observed and controlled has slow varying parameters and doesn't need extremely accurate measurements then the first approach its to be taking in consideration. The same choice applies when it is needed a large number of sensor nodes (thousands).

For the second approach the corresponding results were some of which we hope would be and so the tests have arrived with a confirmation that this approach creates possibilities to implement sensor acquisition solutions for mobile sensor networks on the 802.11 infrastructure using DSP core. Achieved performances are net superior to the Zigbee or Bluetooth sensor networks accordingly to the first approach, with a small nod number. Left on the list to check is the behavior in case of a high node density and the manufacturing costs of an involved node.

Both discussed ISNN architectures differs from previous work in being based explicitly on a hardware/software co-design approach supporting the deployment of novel programming language constructs directly onto the hardware in order to improve optimization. Several structures of distributed type were already tested in specific applications, aiming to focus on cooperative work between cluster agents [11], [12] and on integration of UPnP services [13], [14].

Using a mature infrastructure as 802.11 shall bring low costs together with a high reliability in exploiting. Using an advanced digital signal processor has the disadvantage of a high cost and higher energy consumption but has the major advantage of distributing the algorithms to the whole network level, by this way making it possible to implement an acquisition and a decentralized process control through a series of high-performance control and signal processing algorithms. Also network overload will drop due to the much lower required data volume to be transmitted. Using an audio codec for the acquisition and command part allowed using the DSP at it's fullest capacity, feed it with high resolution data and a high transfer rates.

In conclusion this architecture allows important developments and analyzes in domains like data acquisition from fast processes, remote control over such processes and real-time communication protocols for mobile sensor networks. We want to develop more of such nodes to form a network and then, according to the application, make an embedded node to diminish the gauge and to assure the highest possible energetically autonomy. We are currently completing feasibility studies on the components of our proposed architecture, prior to initial development work.

Our immediate research challenges are to determine appropriate abstractions for the construction and deployment of the embedded system architecture from hardware and software perspectives. We intend to evaluate our work against a range of applications in order to check the qualities of individual dedicated solutions. Future research addresses in particular appropriate error handling and resolving strategies within the real time applications for critical technical problems occurring in the embedded sensor network.

ACKNOWLEDGEMENTS

This work was partially supported by the Romanian Ministry of Education and Research under Grant PN2/CNMP – PC 11-040/2007 and Grant CEEX/INFOSOC - 72/2006

References

[1] G. Pottie. and W. Kaiser, "Wireless integrated network sensors" Communication of the ACM, 43, no. 5, 2000, pp. 51-58

[2] S. Olariu, L. Wilson, M. Eltoweissy and K. Jones, "Training a wireless sensor network", Mobile Networks and Appl., 10, 2005, pp. 151-167

[3] F. Akyildiz, W. Su, Y. Sankarasubramanian and E. Cayirci, „Wireless sensor networks: A survey", Comp. Networks, 38 (4), 2002, pp. 393–422.

[4] T.van Dam and K.Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks," SenSys, pp.171-180, 2003

[5] J. Cortes, S. Martinez, T. Karatas and Francesco Bullo, "Coverage control for mobile sensing networks," IEEE Transactions on Robotics and Automation, vol. 20, pp. 243-255, 2004

[6] C. Gui and P. Mohapatra, "Power conservation and quality of surveillance in target tracking sensor networks", Information Processing in Sensor Networks, 2005, pp. 246 – 253

[7] A. Nisbet and S. Dobson, "A systems architecture for sensor networks based on hardware/software co-design", In Mikhail Smirnov, editor, Proceedings of the 1st IFIP Workshop on Autonomic Communications, Springer Verlag, 2004

[8] R. Dobrescu, M. Nicolae, F. Stoica and R. Varbanescu, "Design of an Intelligent Sensor Network Node", Proceedings of the10th International Conference on Optimization of Electrical and Electronic Equipments OPTIM'06,Brasov, 2006, p. 71-78

[9] S. Baek; E. Choi and J. Huh, "Sensing information managing mechanism for context aware service in ubiquitous home network", Consumer Electronics, 2006, p. 51 – 52

[10] M. Kuorilehto, M. Hannikainen and T. Hamalainen, "A Survey of Application Distribution in Wireless Sensor Networks", EURASIP Journal on Wireless Communications and Networking, 2005, p. 774-788

[11] S. Mocanu, S. Taralunga – Cluster based simulations of Scale-Free Network Immunization Strategies, WSEAS Transactions on Computers, Issue 2, vol.6, February 2007, p. 268-274, ISSN 1109-2750

[12] Dobrescu Radu, Taralunga Sebastian, Mocanu Stefan.- Web Traffic Simulation With Scale-Free Network Models, Proceedings of the 7 th WSEAS Int. Conference on Applied Informatics and Communications, AIC 07, eds. Mingh Hung Le, Metin Demiralp, Valeri Mladenov, Zoran Bojkovic, ISSN 1790-5117, ISBN 978-960-8457-96-6, pp.277-282

[13] Dobrescu Radu, Dobrescu Matei, Nicolae Maximilian, Popescu Dan.- Using UPnP Services With An Intelligent Sensor Network Node, Proceedings of the 7 th WSEAS Int. Conference on Applied Informatics and Communications, AIC 07, eds. Mingh Hung Le, Metin Demiralp, Valeri Mladenov, Zoran Bojkovic, ISSN 1790-5117, ISBN 978-960-8457-96-6, pp.373-376

[14] R.Dobrescu, M.Dobrescu, M.Nicolae, D.Popescu: EMBEDDING WIRELESS SENSORS IN UPnP SERVICES NETWORKS , NAUN International Journal of Communications, Issue 2, Volume 1, 2007

Maximilian E. Nicolae. Born in 16.10.1980 in Bucharest, Romania. Dipl.Eng. degree in Automatic Control, in 2004 and Master degree in Open Systems Architecture, in 2005, both from the Faculty of Control and Computers of the POLITEHNICA University of Bucharest. Ph.D. student in Systems Engineering from 2006 at the "Politehnica" University of Bucharest.

Currently Asisstant Professor in the Department of Automation and Industrial Informatics of the Faculty of Control and Computers, POLITEHNICA University of Bucharest, specialized in data transmission, wireless communication and sensor networks.

Mr. Nicolae has obtained in 2007 a research grant for two years in order to finish the Ph.D. thesis on Mobile Wireless Sensor Networks.

Radu N. Dobrescu. Born in 11.01.1946, in Braila, Romania. Dipl.Eng. degree in Automatic Control from the Faculty of Control and Computers of the Polytehnical Institute of Bucharest, in 1968. Ph.D. degree in Electrical Engineering from the Polytehnical Institute of Bucharest, Romania, in 1976.

Currently Professor in the Department of Automation and Industrial Informatics of the Faculty of Control and Computers, POLITECHNICA University of Bucharest, head of the laboratory on Data Transmission and Industrial Communication. From 1992 Ph.D. adviser in the field of Automatic Control. Several scientific works in three main domains: Modern structures for the numerical control of machine tools and manufacturing flexible systems; Data acquisition, processing and transmission; Local area networks and industrial communication.

Prof. Dobrescu is a pioneer in fractal theory applications for medical image processing and biological systems modelling and simulation. Organiser of the three International Symposiums on Interdisciplinary Approaches in Fractal Applications (IAFA 2003, IAFA 2005 and IAFA 2007). IEEE Member since 1991, IEEE Senior Member since 2005.

Dan Popescu. Born in 18.07.1950, in Bucharest, Romania. Dipl.Eng. degree in Automatic Control from the Faculty of Control and Computers of the Polytehnical Institute of Bucharest, in 1974. Licence degree in Mathematics of the University of Bucharest, in 1980. Ph.D. degree in Control Engineering from the Polytehnical Institute of Bucharest, Romania, in 1986.

Currently Professor in the Department of Automation and Industrial Informatics of the Faculty of Control and Computers, POLITEHNICA University of Bucharest, head of the laboratory of Techniques for Experimental Data Processing. The scientific research activity is concretized on the following domains: acquisition and processing of signals of image type, signals acquisition from sensors and experimental data processing..

Prof. Popescu published 14 books (8 of them as unique author) and more than 100 papers in several specialty journals or in volumes from international conferences.