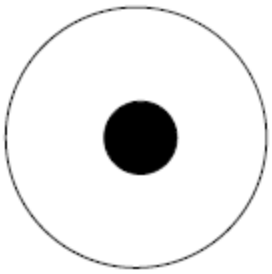
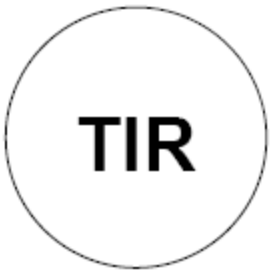




The IPSI BgD Transactions on Internet Research

Multi-, Inter-, and Trans-disciplinary Issues in Computer Science and Engineering

A publication of

IPSI Bgd Internet Research Society
New York, Frankfurt, Tokyo, Belgrade
July 2009 Volume 5 Number 2 (ISSN 1820-4503)

Table of Contents:

Pearls of Wisdom by the Honorary Doctor of Belgrade University:

Interview: "Development of Microprocessors"
Flynn, Michael 2

Creating Autonomous Sensors with Integrated Power Sources
Flynn, Michael 4

Belgrade University Research Efforts by Faculty of Electrical Engineering:

Programming Tools for WSNs
Crnjin, Aleksandar 8

Overview of Sensors for Wireless Sensor Networks
Rakočević, Goran 13

Medical Applications Based on Wireless Sensor Networks
Stanković, Stanislava 19

Processing of Medical Signals (ECG) in Wireless Sensor Networks
Ćirković, Predrag; Aleksić, Anđa 24

The IPSI BgD Internet Research Society

The Internet Research Society is an association of people with professional interest in the field of the Internet. All members will receive this TRANSACTIONS upon payment of the annual Society membership fee of €100 plus an annual subscription fee of €1000 (air mail printed matters delivery).

Member copies of Transactions are for personal use only

IPSI BGD TRANSACTIONS ON INTERNET RESEARCH

www.internetjournals.net

STAFF	
Veljko Milutinovic, Editor-in-Chief	**Marko Novakovic, Journal Manager**
Department of Computer Engineering **IPSI BgD Internet Research Society** **University of Belgrade** **POB 35-54** **Belgrade, Serbia** **Tel: (381) 64-1389281**	**Department of Computer Engineering** **IPSI BgD Internet Research Society** **University of Belgrade** **POB 35-54** **Belgrade, Serbia** **Tel: (381) 64-2956756**
vm@etf.rs	**ipsi.journals@gmail.com**

EDITORIAL BOARD		
Lipkovski, Aleksandar	**Gonzalez, Victor**	**Victor Milligan, Charles**
The Faculty of Mathematics, Belgrade,	University of Oviedo, Gijon,	Sun Microsystems, Colorado
Serbia	Spain	USA
Blaisten-Barojas Estela	**Janicic Predrag**	**Milutinovic, Veljko**
George Mason University, Fairfax, Virginia	The Faculty of Mathematics, Belgrade	IPSI, Belgrade
USA	Serbia	Serbia
Crisp, Bob	**Jutla, Dawn**	**Neuhold, Erich**
University of Arkansas, Fayetteville, Arkansas	Sant Marry's University, Halifax	Research Studios Austria, Vienna
USA	Canada	Austria
Domenici, Andrea	**Karabeg, Dino**	**Piccardi, Massimo**
University of Pisa, Pisa	Oslo University, Oslo	Sydney University of Technology, Sydney
Italy	Norway	Australia
Flynn, Michael	**Kiong, Tan Kok**	**Radenkovic, Bozidar**
Stanford University, Palo Alto, California	National University of Singapore	Faculty of Organizational Sciences, Belgrade
USA	Singapore	Serbia
Fujii, Hironori	**Kovacevic, Branko**	**Rutledge, Chip**
Fujii Labs, M.I.T., Tokyo	School of Electrical Engineering, Belgrade	Purdue Discovery Park, Indiana
Japan	Serbia	USA
Ganascia, Jean-Luc	**Patricelli, Frederic**	**Takahashi, Ryuichi**
Paris University, Paris	ICTEK Worldwide L'Aquila	Hiroshima University, Hiroshima
France	Italy	Japan

Interview: “Development of Microprocessors”

Flynn, Michael

1. *What is the essence of the contribution for which you received the Eckert Mauchly Prize?*

The award said: ``For important and seminal contributions to processor organization and classification, computer arithmetic and performance evaluation.". I guess the essence is my classification system for parallel processors and a number of related contributions to computer design, especially to the design of arithmetic units.

2. *What are the impacts of this contribution?*

The classification is easily grasped and provides a framework for creating both hardware and software models; simplifying the task of optimizing computation in large compute intensive applications.

3. *What are the applications of your contribtion that may change the everyday life?*

The classification work is generally applicable, but most useful in large scientific problems; such as weather modeling, fluid flow simulation (as in the design of air frames), etc.

4. *Can you shed more light on the last answer?*

5. *We learned a lot from your lectures in L Aquila. Can you tell us, what are the issues that we have to teach our kids, so they become creative when they finish studies?*

There's an important principle of intellectual humility. If an analysis or experiment provides unexpected results, do net become either discouraged or frustrated. An unexpected outcome is an invitation to invent and innovate by better understanding the subtleties, creating a larger framework for understanding.

6. *What are the major things to keep in mind, when you form a team for a scientific experiment, or similar?*

A team is like a mosaic or a stained glass window; the pieces (or members) must complement each other. Many teams fail because of an inability to provide a balance of contributions. Too many leaders is a worse situation than too few as it leads to antagonism among the members. A really good leader knows when to follow.

7. *What are the people to avoid, when trying to generate a break-through achievement?*

(see above)

8. *What is your oppinion about the impact of math?*

Well math is a central way we have to communicate ideas and abstractions. It is a formalized language. As with any language it expresses some ideas better than others; but it's the best we have.

9. *When targeting a major breakthrough, how sensitive one has to be about the direct interests of tax-payers?*

Many breakthrough cost little money. Many of the major advances in computing have been done on a "shoestring" (search engines, network routers, etc.). There are however some types of research facilities that are capital intensive (particle physics, silicon fabrication facilities, space exploration, etc.). These can be funded by governments where they feel it is in their national interest. There has to be a consensus that the research is worth the cost.

10. *What is the major driving force that motivates a person like you to continue to create and generate results after he-she receives such a big prize?*

I thought of it as a recognition for work done it the past; not as a goal in itself.

11. *For small nations like Serbian, what is your advice, which road to take, when it comes to science?*

Science and engineering are to the 21st century what the liberal arts were to the 19th. They provide the knowledge that an intelligent citizen needs to know to work, contribute and function in society. It is important to provide a quality math and scientific education for all young people.

As for government investment in research and development choose less capital intensive and more intelligence intensive projects.

12. *What road to take, when it comes to its general future development plans?*

(see above)

13. *Major impression about Serbian scientists that you met abroad?*

All were very well qualified and highly motivated.

14. *Major impressions from your visits to Serbia?*

I did limited touring; Belgrade still had many of the grand old 19th century residences. The countryside was quite pretty; we stayed overnight at an old monastery which I still recall with a great deal of fondness.

15. *Major impression about the University of Belgrade?*

I haven't seen it now for several years. I liked some of the old buildings.
I know that the faculty is very active and enthusiastic.

Creating Autonomous Sensors with Integrated Power Sources

Flynn, Michael

Abstract - *The great challenge for both sensor and SOC designers is providing a reliable power source. This usually involves a battery which often exceeds in bulk and size the sensor die itself .The alternative is to integrate the battery with the sensor on a single die. This approach has two major limitations; 1) a requirement for very low power consumption and 2) a finite sensor lifetime. The latter requirement can be mitigated by energy scavenging if the environment allows. Associated with this is the need for "wireless" sensors freed from the tether. The major architectural implication is design for extremely low power (order of 1 microwatt) and with a strict energy budget*

Introduction

A fast growing chip marketplace is autonomous chips. So far these devices have little processing power or memory but have RF communications and some type of self contained power source or power management system. The more elaborate autonomous chips also include or are built around a type of sensor. The simple versions include RFID chips [RFI], smart cards and chip implanted credit cards. The simplest is the passively power RFID. The chip simply reflects the source RF carrier and modulates it (using carrier power) to indicate its ID. More complex examples include patient monitoring alarm [GHANT] and the Smart Dust DARPA research program [DUST1, DUST2] of the '90s. Both of these used battery powered RF to broadcast an ID on a detected sensor input (Table1).

System	*Passive id*	*Active id*	*RF sensor*
Example	RFID, Smart Card {simple)	Smart card, Active RFID	Smart Dust; RFID + sensor
Power source	None	Short term battery	Battery
Maximum Memory	ROM ID (1KB)	R/W ID + parameters (2 KB)	
RF range (meters)	Passive; order of cm	Active 1-10	10 - 20
Compute	None	FSM	FSM

Table 1: Some autonomous chip examples (FSM represents a simple finite state machine or micro controller)

The various Smart Cards and Money Cards include VISA cards and Hong Kong's Octopus Card. All (except those that require contact) use a form of RFID. The simplest cards are passive without on-card writeable memory. Records are updated centrally. Implementation is frequently based on Java Card [JAV].

The Smart Dust [DUST1] project started in the early '90s and pioneered significant work in the sensor and RF areas. That project targeted sensor plus RF integrated into a form factor of the order of 1 mm^3 called motes. As a power source it relied on AA type batteries. That project was targeted at sensing an "event"; a moving object, a thermal signal, etc.

In this paper we consider tradeoffs in silicon technology, limits on batteries and energy, RF communications for sensor applications.

This paper was abstracted from M. J. Flynn, "Super SOC: putting the whole (autonomous) system on the chip (ASOC)," Proceedings of the Stamatis Vassiliadis Memorial Symposium, on the Future of Computing, published by Technical University Delft, NL, 2008

Technology

The main problem for useful an autonomous sensor is battery power or stored energy. In dealing with this issue recall two general relationships, relating silicon area, A, algorithmic execution time, T, and power consumption, P (in these expressions k is a constant):

1) $AT^2 = k$

This well known result [ULL] simply related area (the number of transistors) to the execution time required to complete an operation. The more area (transistors) used the faster (smaller) the execution time.

2) $P^3T = k$

This result [FLY] applies to dynamic power only and is based on voltage scaling augments. It's easy to see that as voltage is decreased power is reduced by the square but speed is reduced linearly. But the transistor charging current (representing delay) and voltage have a non linear relationship. This gives the cubic result. We can rearrange this as:

3) $P_2 / P_1 = (F_2 / F_1)^3$

So if we want to double the frequency we should expect the design to use 8 times more power. While the range of applicability of expression (2) is not precise, suppose we use it to project the frequency of a processor design that operates at a microwatt. The best power-performance design of today might consume one watt and achieve one Gigahertz (corresponding perhaps to 1,000 MIPS); this may be optimistic. Reducing the power by a factor of 10^6 should reduce frequency by a factor of 100 or 10 Megahertz.. Within the past two years a sensor processor has been built that achieves almost 0.5 MIPS per microwatt [ZHA]. While this is an order of magnitude away from our target of 10 Megahertz per microwatt, silicon scaling projections should compensate for the difference.

Powering the sensor

The key problems in forming robust sensors are energy and lifetime. Both relate to the power source, i.e. the battery. Batteries can be charged once or rechargeable (with varying recharge cycles). Rechargeable batteries can use scavenged energy from the environment. The capacity of the battery is usually measured in milliamp-hours; which we convert to Joules (watt-seconds) at 1.5 volt. Both capacity and recharagability depend on size which we assume is generally consistent with the size and weight of the sensor die (about 1 cm^2 surface area).

In table 2 we list 3 common battery types; the printed [POW, PUS] and thin film batteries [CYM] can be directly integrated into the sensor die (usually the reverse side); button batteries are external and are less than 1 cm in diameter.

Type	*Energy (J)*	*Recharge Y/N*	*Thickness (micron)*
Printed	2 / cm2	N	20
Thin film	10 /cm2	Y	100
Button	200	Y	500 stand alone

Table 2: Battery technology

Printed batteries are formed by printing with special inks in a flat surface; thin film batteries are deposited on silicon much as the sensor dies itself.

Energy may be scavenged from many sources (some are illustrated in table 3); usually the larger the battery format the more the charge. Much depends on the system environment as to which, if any, scavenging is suitable.

Source	*Charge rate*	*Comment*
Solar	65 (milliwatts)/ cm^2	
Ambient light	2 (milliwatts)/ cm^2	
Strain and acoustic	A force (sound) changes alignment of crystal structure, creating voltage	Piezoelectric effect
RF	An electric field of 10V/m yields 16uW/cm^2 of antenna	See [YEA]
Temperature difference (Peltier effect)	40 (microwatts/5^0 C difference)	Needs temperature differential.

Table 3 Some energy scavenging sources [MEN, ROU, PAR]

Assuming consumption of 1 uw (when active); the operational lifetime between charges is plotted in figure 1. Duty cycle can play an important role in expending the sensor serviceability. The assumption is that a passive sensor can detect an event and power up the system for analysis.

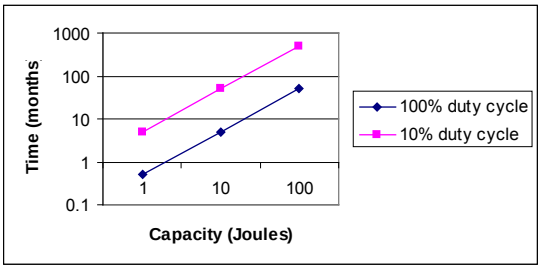


Figure 1: Maximum time between recharge for 1microwatt of continuous power consumption.

If we can configure the sensor to use of the order of 1 microwatt we should be able to incorporate a suitable battery technology especially if we have the ability to scavenge some addition energy.

RF Communications

The work of the Smart Dust program seems to the most pertinent [DUST1, DUST2]. That program demonstrated the integration of low power RF into an sensor chip. To summarize some of their many finding:

1) A feasibility study realized a transceiver achieving 100 Kbps over a 20 meter distance with an energy budget of 25 nJoules/bit. This corresponds to about 10^{11} bits / Joule/ Meter. One Joule of battery energy allows 100 Gbits to be transferred across one meter.
2) Communications with less than 1 milliwatt was not only feasible but likely to be commercialized. With typical duty cycle of less than 1% the average power consumption was between 1 and 10 microwatts.
3) There is a large data packet overhead (including startup and synchronization, start symbol, address, packet length, encryption and error correction). Short messages can have as little as 3% payload packet efficiency. It is better to create fewer longer messages.
4) As a result of (2) and (3) the system designer will want to minimize the number of transmission and maximize the data packet payload.

Conclusions

There's a whole new field to be explored based on the next generation of autonomously sensors and SOC chips. Transistor density improvements will enable close to a billion transistors per cm^2. This enormous computational potential has a major limitation: limited electrical energy. There is a new direction opening in

computer architecture, *nano computing*, to contrast with historical efforts in supercomputing. The target of this field is to produce the algorithms and architectural approaches for high performance at less than one millionth current levels of power dissipation; freeing the chip from external power coupling.

For untethered operation a form of wireless communication is required. This is another significant challenge for RF especially with a power budget also in the order of microwatts; but careful budgeting of messages to limit transmissions could be helpful

References

[RFI] RFID, RFID Journal, http://www.rfidjournal.com/

[DUST1] B.W. Cook, S. Lanzisera, K.S.J. Pister, “SoC Issues for RF Smart Dust”, Proc. Of the IEEE, Vol. 94, no. 6, June 2006

[DUST2] A. Molnar et al, “An ultra low power 900 MHz RF transceiver for wireless sensor output”, Proc. Custom Integrated Circuits Conference, 2004 p 401-404

[GHANT] Patient monitoring University Hospital of Ghent
http://www.rfidjournal.com/article/articleview/3120/1/1/

[JAV] Java Card Platform Specification, http://java.sun.com/products/javacard/specs.html

[ULL] J. D. Ullman, “Computational Aspects of VLSI.” Computer Science Press, 1984

[FLY] M. J. Flynn, P. Hung and K.W. Rudd, “Deep-Submicron Microprocessor Design Issues.” IEEE Micro Magazine, July-August Issue, 1999, pp. 11-22.

[ZHA] B. Zhai, et al, “A 2.60pJ/Inst Subthreshold Sensor Processor for Optimal Energy Efficiency”, VLSI 06

[POW] PowerID, Power Paper Corp.
www.powerpaper.com

[PUS] V. L Pushparaj et al, "Flexible Nanocomposite Thin Film Energy Storage Devices,"
www.newswise.com/articles/view/531241/

[CYM] Cymbet Corp, The POWER FAB (Thin Film Lithium Ion Cell) battery system,.
http://www.cymbet.com

[YEA] E.M. Yeatman, “Advances in power sources for wireless sensor nodes,”
Proceedings of 1st International Workshop on Body Sensor Networks, London, 2004

[MEN] S. Meninger, et al, “Vibration-to-electric energy conversion”, IEEE Trans. VLSI Systems, Vol. 9, No. 1, p 64-76, Feb. 2001

[ROU] S. Roundy, et al., "Power Sources for Wireless Sensor Networks," in Proc. 1st European Workshop Wireless Sensor Networks, Jan. 2004, pp. 1 - 17.

[PAR] L. D. Partain, “Solar Cells and Their Applications” (Wiley Series in Microwave and Optical Engineering), Pub. J. Wiley 2004

Programming Tools for WSNs

Crnjin, Aleksandar

Abstract - *As with all other computing devices, sensor nodes have to be programmed in order to be able to do something useful. The similarity, for the most part, ends there. There is scant similarity between programming a home computer with 2 gigahertz CPU and gigabytes of RAM and storage and programming a tiny device with just a dozen of kilobytes of RAM and an 8-bit processor running on just a few kilohertz. Still, programming sensor networks isn't all that similar to embedded devices either: Programming embedded systems typically amounts to writing an assembly program and loading it into programmable ROM of the embedded device, while a sensor node usually has an operating system (such as TinyOS or ConTiki) which provides support for basic I/O operations, operation of the radio transceiver and so on. This article introduces the reader to general issues in sensor network programming and focuses on two different programming paradigms: NesC programming, for TinyOS compatible sensor nodes, and the Java Squawk virtual machine, created for Sun SPOTs.*

1. General Issues in Sensor Network Programming

In personal computers, OS primitives are used to achieve hardware abstraction, thus lifting the burden of operating the peripherals directly from the programmer. Clearly, that approach is desirable in sensor nodes as well. Still, due to scarce, *scaled* resources of a sensor node, some compromises are required, most notably the need to conserve energy resources by providing an implicitly energy efficient programming model.

The first obstacle we have to overcome is to choose the right operating system for the sensor node. All conventional OSes for embedded systems (such as Windows CE or PalmOS) require a ROM capacity of a hundred kilobytes or more; sensor nodes, however, typically have only a few kilobytes. The most popular solution today is the TinyOS, developed by U.C. Berkeley. Another option is the Java Squawk virtual machine, developed by Sun Microsystems for their Sun SPOT project. This article focuses on these two paradigms.

2. The TinyOS Operating System

For sensor nodes, a specialized operating system had to be devised; one that would provide the necessary primitives to operate the sensor node hardware, but would also cope with the quite limited resources of a sensor node. These limitations include:

Not enough memory for stack. All TinyOS programs have to operate within a single context, as it is impossible to perform traditional context switching, due to a very small amount of memory available for stack. This also means that TinyOS programs can't rely on registers to save state.

Limited amount of memory. This means that what available memory there is has to be allocated carefully. Dynamic allocation of memory is prohibited; the TinyOS *components* are arranged into a *configuration* at compile-time, and individual components each get their preassigned portion of memory, the *memory frame.*

Limited amount of energy. Special care is taken to ensure that the battery power is conserved as much as possible. Busy waiting and interrupt polling is prohibited in all TinyOS compatible devices. TinyOS programs execute only in response to *events*; this is called the *event-driven* programming model.

2.1. TinyOS Component Model

To address these issues, the *TinyOS Component Model* was devised. The parts of code in this model are organized into distinct entities based on functionality. For example, there may be an entity for operating the integrated radio-transceiver unit. (In fact, there is such an entity in TinyOS, it's the **rfm** system component.) These entities are associated with a statically allocated memory portion (called *memory frame)* and interconnected with other such entities through *interfaces.* Such completely described entities are called *components.*

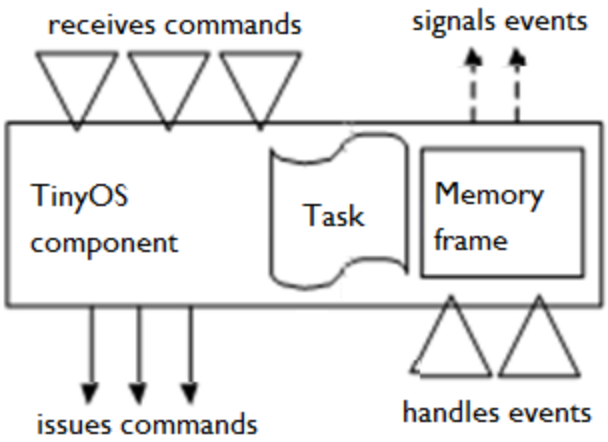


Figure 1: Block view of a TinyOS component

The *memory frame* is allocated at compile-time, based on total memory requirements of the component.
The *interface* consists of registered commands and events, through which components communicate. The main difference between the two is in the fact that events propagate upward in the connection chain (from device driver components, such as aforementioned **rfm**, through bottom level user components up to the top level user components) and commands propagate downward in the connection chain. One component may choose to implement several interfaces, each consisting of a number of commands and events. More on this will be mentioned later, in the chapter on component interconnection.
The *code* contained in an individual TinyOS component consists of:
Command handling routines, which they execute in response to *commands* issued from other connected components;

Event handling routines**,** which they execute in response to *events* signalled by connected components;
Tasks, which components themselves schedule for later execution.

System Components and Pre-defined Events

TinyOS comes with a certain number of *system components* which perform the function of device drivers. Examples include the already mentioned **rfm** transceiver-operating component, **Photo** and **Temperature** – the components for manipulating photo and temperature sensors, and so on.
In addition to pre-defined system components, there is also a pre-defined interface **StdControl,** consisting of events **Init, Start** and **Stop.** Each TinyOS component has to implement this interface by providing handlers. For example, an **Init** event handler defines the response of the component to the event of system initialization. A component could use this handler in order to initialize it's state to initial values. Actual execution of a program begins with handling of the **Start** event, in the mandatory **Main** component.

Component Configurations

Components are either **modules** or **configurations.** Each TinyOS "program" is a configuration of interconnected modules; each module is an encapsulation of code based on functionality, reminiscent of classes in object oriented programming languages. Each configuration and module is described in a separate **.nc** file. Modules are connected through *interfaces.* An interface is a collection of events a given component can signal or be notified of, and commands that a component can issue or obey. Interfaces are also defined in their separate **.nc** files. A configuration is formed by connected by "software wiring" of the interfaces. This is accomplished in two steps:

1. each module lists the desired interface in either it's **uses {…}** block, or it's **provides {…} block;**
2. a special line in the configuration code is added, which establishes an unidirectional link between an *interface provider* and *interface user***.** This is achieved with following line of code:

User.UserInterfaceName ->
Provider.ProvInterfaceName
or, alternatively:
Provider.ProvInterfaceName <-
User.UserInterfaceName

(The UserInterfaceName part may be omitted, if the interface names are the same.)

One example of an interface is the Timer interface, through which the system component for the on-board timer notifies the user component of regularly spaced *tick events.* The description of this interface is given in Timer.nc:

```
interface Timer {
  command result_t start(char type, uint32_t
interval);
  command result_t stop();
  event result_t fired();
}
```

From the source code, it should be clear that the interface specifies one event – fired() – through which the Timer provider notifies the user of timer ticks, and accepts two commands – start, which accepts the type and interval parameters and starts the timer, and stop(), which stops the timer.
The actual semantics of the words *provider* and *user* might not be immediately clear: The events and commands which are the part of the interface are specified in the interface's **.nc** file. If a module declares itself to be a provider of a given interface, then it's responsible to implement all commands listed by the interface, and acquires the right (which it may or may not use) to signal events to those users of said interface to which it is connected through wiring. Likewise, if a module declares itself to be a user of an interface, then it is responsible to implement event handlers for every single event specified by the interface, and it attains the right to issue commands to the provider of the used interface.
In other words, the provider can signal events to users, and users can issue commands to the provider.

NesC "Hello World" program: Blink

To give an overview of the issues involved, we consider a simple application that comes bundled with the TinyOS distribution, called **Blink.** As it name suggests, the Blink application uses the on-board timer to periodically change the state of a sensor node's LED, producing a blinking effect. The Blink application consists of four connected components: Main, BlinkM, SingleTimer and Leds, as seen in the configuration declaration (Blink.nc):

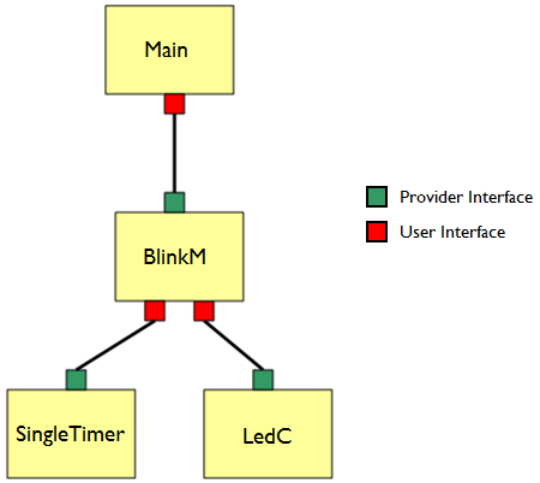
```
configuration Blink {
}
implementation {
  components Main, BlinkM, SingleTimer, LedsC;
  // connecting Main to SingleTimer and BlinkM through
StdControl
  Main.StdControl -> SingleTimer.StdControl;
  Main.StdControl -> BlinkM.StdControl;
  BlinkM.Timer -> SingleTimer.Timer; // connecting BlinkM to
SingleTimer
  BlinkM.Leds -> LedsC;          // connecting BlinkM to LedC
}
```

In this example, we can note the following:
Components Main, BlinkM, SingleTimer and LedC are used. Code for these components is elaborated in separate files (for example, BlinkM code is found in BlinkM.nc)
In the last four lines before the ending bracket, we can see how these components are interconnected – through "wiring" of their interfaces, using the notation User.Interface -> Provider.Interface.
BlinkM, the central module of the application, provides the StdControl interface, of which the implicit component Main is the user. The execution of the program "begins" with Main issuing the commands "Init" and "Start" to BlinkM. In response to these commands, BlinkM will start the timer, through the Timer interface. As BlinkM is notified of timer tick events, it toggles the LED, by issuing commands to the LedC component through the Leds interface.

```
//Implementation for Blink application:
// the red LED is toggled whenever Timer fires.
module BlinkM {
 provides {
   interface StdControl;
 }
 uses {
   interface Timer;
   interface Leds;
 }
}

implementation {
 //Handling of the Init command, issued by Main
 // just pass the Init command on to the Leds component
 command result_t StdControl.init() {
   call Leds.init();
   return SUCCESS;  // SUCCESS is always returned
 }
 // Handling of the Start command, issued by Main:
 // set the rate for the clock component.
 command result_t StdControl.start() {
 // Start a repeating timer that fires every 1000ms
   return call Timer.start(TIMER_REPEAT, 1000);
 }
 // Handling of the Stop command, issued by Main:
 // stop the timer
 command result_t StdControl.stop() {
   return call Timer.stop();
 }
 // Handling the Fired event, issued by Timer:
 // issue redToggle command to Leds
 event result_t Timer.fired(){
   call Leds.redToggle();
   return SUCCESS;
 }
}
```

The configuration of the Blink application can be summarized in the following diagram:



TinyOS: Conclusion

Programming in TinyOS is much simpler and quicker compared to the Full Custom model of assembly programming, and arguably easier, too. Still, the TinyOS programming model has suffered criticism for its still steep learning curve, especially for programmers accustomed to programming in established languages (such as Java).

3. Sun SPOTs and the Squawk virtual machine

The Squawk virtual machine, created by Sun Microsystems for their Sun SPOT technology, is one attempt at providing easier-to-use programming model to developers for sensor networks. Using Sun SPOTs and Squawk, a sensor network developer can write applications in a slightly modified versionof Java.
Squawk itself is a Java virtual machine running directly on SPOT hardware, without an underlying operating system. To facilitate execution of such sophisticated software, Sun SPOTs had to be designed with significantly more powerful hardware, compared to the TinyOS nodes (a Sun SPOT has a 180MHz 32-bit processor). This might mean greater energy consumption and less battery life, compared to TinyOS nodes.

Programming Sun SPOTs

As mentioned already, Sun SPOTs are programmed using a variation of Java language. SunSPOT applications are MIDlets; the main class of the SunSPOT application extends the javax.microedition.midlet.MIDlet class, so constructs typical for MIDlets, such as startApp(), pauseApp() and destroyApp(), form the skeleton of a Sun SPOT application. Therefore, an "entry point" for a Sun SPOT application is the startApp() method.
The whole functionality of the SPOT is abstracted using the EDemoBoard class. Typically, a programmer will obtain an instance of this class using EDemoBoard.getInstance() in the body of the startApp() method. Inputs (sensors and switches) and outputs (LEDs) are then reachable through Java interfaces, which are obtained using demoBoardInstance.get*xxx* () methods. For example, a temperature reading can be obtained using:

```
ITemperatureInput ourTempSensor =
EDemoBoard.getADCTemperature();
double celsiusTemp = ourTempSensor.getCelsius();
double fahrTemp = ourTempSensor.getFahrenheit();
```

A SunSPOT "Blink" application

As for TinyOS/nesC, we now present a Blink (Hello World) application for Sun SPOTs.

Sun SPOTs/Squawk: A Conclusion

Sun SPOTs certainly provide an easier programming model and as a result they are very popular within certain educational circles. However, their bulky size (compared to TinyOS-based platforms), higher price (for non-educational use) and higher battery consumption still make them somewhat impractical for some industrial applications.

```
package org.sunspotworld;
import com.sun.spot.sensorboard.EDemoBoard;
import com.sun.spot.sensorboard.peripheral.ISwitch;
import com.sun.spot.senso
            rboard.peripheral.ITriColorLED;
import com.sun.spot.util.*;
import javax.microedition.midlet.MIDlet;
import
javax.microedition.midlet.MIDletStateChangeException;

public class Blink extends MIDlet {
  private ITriColorLED [] leds =
EDemoBoard.getInstance().getLEDs();
  protected void startApp() throws
MIDletStateChangeException {
    System.out.println("Hello, world");
    ISwitch sw1 =
EDemoBoard.getInstance().getSwitches()[EDemoBoard.SW
1];
    leds[0].setRGB(100,0,0);       // set color to moderate red
    while (sw1.isOpen()) {         // done when switch is
pressed
      leds[0].setOn();           // Blink LED
      Utils.sleep(250);          // wait 1/4 seconds
      leds[0].setOff();
      Utils.sleep(1000);         // wait 1 second
    }
    notifyDestroyed();           // cause the MIDlet to exit
  }
  protected void pauseApp() {
    // This is not currently called by the Squawk VM
  }

protected void destroyApp(boolean unconditional) throws
MIDletStateChangeException {
    for (int i = 0; i < 8; i++) {
      leds[i].setOff();
    }
  }
}
```

4. References

[1] Philip Lewis, TinyOS Programming, 2006.
[2] TinyOS Documentation Wiki
[3] SunSPOT Developer's Guide, SUN Microsystems, 2009.

Overview of Sensors for Wireless Sensor Networks

Rakočević, Goran

Abstract - *In a wireless sensor network sensors play an important part, as sensing is one of its central roles. A number of characteristics are important when choosing the right sensor for an application. Sesnors can be clasiffied accoridng two to basic criteria: principal by which they function, and by the function the sensor performs. Mechanical sensors detect mechanical properties and actions. Temperature sensors are some of the most widely used, and a variety of temperature sensors exist.*
Chemical, bio, and radiation sensors play an increasing role in a many WSN applications.

Keywords - *wireless sensor networks, sensors, microsesnsor*

Introduction

A sensor is an electronic device used to detect or measure a physical quantity and convert it into an electronic signal. In other words, sensors are devices that translate aspects of physical reality into a representations understandable and processable by computers.

In a wireless sensor network sensors play an important part, as sensing is one of its central roles. Technology behind sensors, however, is not of major interest when considering sensor networks, with the emphasis being more on communication, network management, and data manipulation. Most sensors used in WSN systems have been developed independently of WSN technology, and these two fields continue to develop somewhat independently.

Nevertheless, any in-depth discussion of wireless sensor networks, especially when aimed towards providing the reader with a holistic picture of current capabilities and limitations of wireless sensor networks, must include sensors.

Sensor characteristics

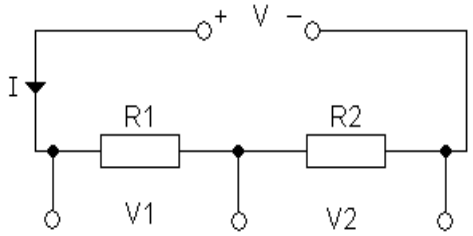
When choosing the right sensors for an application it is important to understand the basic characteristics of sensors found in the datasheets. Detailed explanation of of these characteristics is outside of scope of this article, so only a list is provided:

- Transfer Function
- Hysteresis
- Linearity
- Sensitivity
- Accuracy
- Dynamic Range
- Noise
- Resolution
- Bandwidth

Clasificaion

There are two basic ways to categorize sensors. First is based on the principal by which they function, and the second is based on the function the sensor performs.

Most sensors act like passive devices (i.e. capacitors or resistors). These sensor require external circuitry for biasing and amplification of the output signal.
Resistive sensors are devices whose resistance changes with the value of input signal being measured. These sensors can be used in a simple voltage-divider configuration (Picture 1). For more precise measurements a variety of configurations can be used (e.g. the Whetstone bridge circuit).



Picture 1: Voltage divider Legend: R1: Resistive sensor, R2: Reference resistor, V1: Voltage on the resistive sensor, V2: Voltage on the reference resistor, V: reference voltage.

Capacitive sensors produce a change in capacitance proportionate to the value of the measured input signal. Detection of this change is done quite similarly as with the resistive sensors, only in this case the impedance of the capacitor is observed, which means that an AC bias must be provided. Inductance based sensors can be observed in much the same way.

As opposed to these sensors some sensors produce their own bias voltage, and can directly be connected to an AD converter, or an amplifier if amplification is required.

Perhaps the more logical way to classify sensors is with regards to the physical property they measure. The most common categories include:

- Mechanical
- Thermal
- Electrical
- Magnetic
- Radiant
- Chemical and bio-chemical

Mechanical sensors

Mechanical sensors detect mechanical properties and actions. This includes (among other things) pressure, velocity, vibration sensors and accelerometers.

Pressure sensors
Pressure is one of the most important physical properties, and thus, pressure micro-sensors were the first micro-sensors developed and used by the industry. A wide variety of applications calls for a wide variety of pressure sensors, but most belong in one of three major categories.

Piezorezistive pressure sensors
Piezorezistive pressure sensors have a piezoresistor integrated in a membrane. Pressure is applied to the membrane, causing it to deform. This in turn, causes a change in resistance, proportionate to the applied force.

Capacitive pressure sensors
In capacitive pressure sensors (whether membrane or comb based) pressure is applied on the sensor surface, causing a membrane to deflect and the capacitance to change. These senosors generally have greater sensitivity and linearity, while exhibiting very little or no hysteresis. However, these sensors also have higher production costs when compared to piezoresistive pressure sensors.

Optical pressure sensors
Optical pressure sensors operate on the principal of the Mach-Zehnder interferometer. Laser light is brought into the sensor via an optical fiber. This light is split into two beams. One of the two beams crosses through one of the beams which is deformed by the pressure. This deformation changes the light's properties. The two beams are combined and brought to a photodiode. Different propagation speeds create a phase shift between these beams which is detected at the diode.

Position and Motion Sensors

Position sensors play an important role in a wide variety of applications. Numerous ways of detecting position are available, ranging from simple contact sensors to more complex contact-free ones. Position measurement can either be relative (displacement sensors) or absolute, linear or angular.

Accelerometers
Accelerometers are sensors that measure acceleration they are subjected to. Most are based on resistive or capitative and piezoelectric methods.

Resistive and capacitive accelerometers
With these micro-sensors an elastic cantilever with an attached mass is usually used. When the sensor is subjected to acceleration, a force proportionate to this acceleration deforms the cantilever. With piezoresistive sensors a piezoresistor is integrated into the cantilever, whose deformation causes a change in its' resistance. With capacitive sensors the cantilever acts as one electrode, with a electrode strip acting as the other. As the cantilever is deformed it is brought closer to the electrode strip, which in turn effects the capacitance between the two electrodes.

Resistive and capacitive accelerometers can be used to measure constant acceleration, such as that of earth’s gravity. They are generally used for measuring low frequency vibrations.

Piezoelectric accelerometers
Piezoelectric accelerometers are based on the piezoelectric effect. This means that an electric charge is created when the sensing material is

squeezed or strained. Several methods of straining of the material can be used, three of the basic being: compression, flexural, and shear, with the shear being the most common one. These accelerometers are generally durable, protected from contamination, impervious to extraneous noise influences.

Temperature sensors

Temperature sensors detect a change in a physical parameter (resistance or output voltage) that corresponds to a temperature change. Three basic types of temperature sensors are electro-mechanical, electronic, and thermo-resistive.

Electromechanical temperature sensors
These sensors are based on expanding or contracting properties of materials when subjected to a temperature change. Bi-metal thermostats are created by bonding two metals into a single strip of material. Different expansion rates of the metals create electro-mechanical motion when the material is subjected to a temperature change. In capillary thermostats the capillary motion of expanding or contracting fluid is used to make or break a number of electrical contacts.

Resistive Temperature sensors
Resistive temperature sensors are devices whose resistance changes with the temperature.

Thermistors
A thermistor is a type of resistor with resistance varying according to its temperature. They typically consist of a combination of two or three metal oxides that are sintered in a ceramic base material.

Thermistors can be classified into two types: positive temperature coefficient (PTC) and negative temperature coefficient (NTC). PTC devices exhibit an increase in resistance as temperature rises, while NTC devices exhibit a decrease in resistance when temperature increases.

The main disadvantage of the thermistor is its strong non-linearity. Cheap thermistors have large spread of parameters (“tolerance”) and calibration is usually necessary.

Resistive temperature detectors (RTDs)
Unlike thermistors that use a combination of metal oxides and ceramics resistive temperature detectors are made from pure metal (copper, nickel or platinum are usually used). RTDs are useful over larger temperature ranges, while thermistors typically achieve a higher precision within a limited temperature range.

As a RTD is a resistance device and it needs measuring current to generate a useful signal. Because this current heats the element above the ambient temperature ($P = I2.R$), errors can occur, unless the extra heat is dispersed. This forces us to choose a small-sized resistance device with a quick response or a larger resistance device and better heat release.

A second solution is to keep the measuring current low (usually between 1 mA and 5 mA).

Humidity sensors

Humidity is the amount of water vapor in the given substance (usually a gas). It is an important parameter in a variety of fields, including room air humidity in patient monitoring and exhibit perseveration in museums, meteorological observations, soil humidity in agriculture, and process control in the industrial applications.

Humidity can be measured as the absolute humidity (ratio of water vapor to the volume of substance), relative (compared to the saturated moisture level) or dew point (temperature and pressure at which the observed gas starts to turn into liquid). Most common humidity sensors are based capacitive, resistive, and thermal conductivity measurement techniques.

Humidity is the amount of water vapor in the given substance (usually a gas). It is an important parameter in a variety of fields, including room air humidity in patient monitoring and exhibit perseveration in museums, meteorological observations, soil humidity in agriculture, and process control in the industrial applications.

Humidity can be measured as the absolute humidity (ratio of water vapor to the volume of substance), relative (compared to the saturated moisture level) or dew point (temperature and pressure at which the observed gas starts to

turn into liquid). Most common humidity sensors are based capacitive, resistive, and thermal conductivity measurement techniques.

Capacitive RH Sensors
In a capacitive RH sensor, change in dielectric constant is almost directly proportional to relative humidity in the environment. Relative humidity sensors have three-layer capacitance construction and consists of thermoset polymer, platinum electrodes, and a silicon chip with integrated voltage output signal conditioning.

These sensors have low temperature coefficient, and response times that range from 30 to 60 seconds.

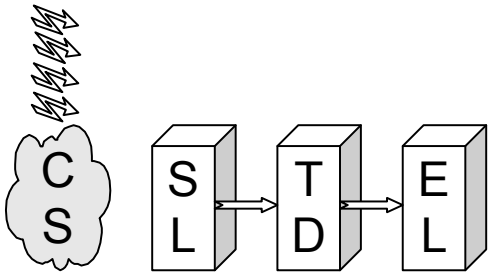
They offer near-linear voltage outputs, wide RH ranges and condensation tolerance, and are stable over long-term use. However, the capacitive effect of the cable connecting the sensor to the signal conditioning circuitry is large compared to the small capacitance changes of the sensor. This limits the distance from sensing element to signal conditioning circuitry.

Resistive Humidity Sensors
Resistive humidity sensors measure the resistance change in a medium such as a conductive polymer or a salt. Resistance usually has an inverse exponential relationship to humidity. Response times of these sensors is 10-30 seconds.
Resistive humidity sensors are small size, low cost, and are usable from remote locations.

Chemical sensors

Chemical sensors are detect the presence or concentration of particlar chemical elements or compounds in a given sample. A chemical sensor usually consists of a chemically sensitive film or a membrane and a transducer.

A chemical process occurring in or on a chemically sensitive film or membrane is causes a signal to be generated generation at the transducer. Examples of mechanisms commonly employed include host-guest binding, catalytic reactions or a redox process.
Chemical sensors have a wast variety of applications ranging form medical diagnostics and nutritional sciences through security to automotive industry.



Picture 2: Structure of a chemical sensor Legend: CS: Chemical substance, SL: Sensitive layer, TD: Transducer, EL: Electronics Explanation: Chemical substance reacts with the chemical layer. Reaction causes a signal to be generated generation at the transducer. The signal is then processed by electronics and converted into a format suitable for further processing.

Interdigital transducer sensors
Interdigital transducers using capitative measurement are often used in chemical sensors. Sensitive layer is used as the dielectric between two electrodes. The dielectric properties of the sensitive layer are changed when it interacts with certain substances, effecting the capacitance between the two electrodes.

Conductivity sensors
In these sensors the sensitive layer is used as a conductor of electricity. Interactions with certain chemicals (e.g. absorption of gasses) modifies the conductivity of this layer. There are two types of sensing layers: Metal Oxide and Conducting Polymers.
Metal Oxide sensitive layers are typically made of SnO2 doped with Pt or Pd. These sensors can operate at high temperatures (300-5000 C) which makes them especially suitable for combustion gases.
Conductive Polymer sensitive layers are usually based on pyrrole, aniline or thiophene. These sensor operate best at room temperatures. Compared to Metal Oxide sensors these sensors have lower power consumption, and faster response and recovery times. However, they are have lower sensitivity and are sensitive to humidity.

Optical chemical sensors
In optical sensors an optical waveguide is used as the sensitive layer. Chemical reactions between the waveguide and the target chemical substance cause a change in the optical

properties of the waveguide (e.g. the index of reflection). As a result the amount (or the wavelength) of the light striking the sensor on the end of the waveguide varies.

These sensors are highly sensitive, can handle small quantities, inexpensive, and easy to sterilize.

Majority (about 60%) of chemical sensors are gas sensors. Most commonly used chemical sensors include: O2, pH, CO, CO2, NOX, Methane, etc.

Ion sensitive FET sensor
An ion sensitive field effect transistor (ISFET) is an ion-sensitive field effect transistor used to measure ion concentrations in solution; when the ion concentration (such as pH) changes, the current through the transistor will change accordingly. Here, the solution is used as the gate electrode. A voltage between substrate and oxide surfaces arises due to an ions sheath.

An ISFET's source and drain are constructed as for a MOSFET. The gate electrode is separated from the channel by a barrier which is sensitive to hydrogen ions and a gap to allow the substance under test to come in contact with the sensitive barrier. An ISFET's threshold voltage depends on the pH of the substance in contact with its ion-sensitive barrier.
The surface hydrolization of OH groups of the gate materials varies in aqueous solutions due to pH value. Typical gate materials are Si3N4, Al2O3 and Ta2O5.

ISFET sensors are used in devices for continuos measurements like those for continuous measurement of pH value and gases in blood (O2, CO2).

Piezoelectric chemical sensors
Piezoelectric effect is the generation of an electric charge in a crystalline material upon subjecting it to stress. A piezoelectric chemical sensor is a piezoelectric oscillator that responds to changes in the chemical composition of its environment with changes of the resonant frequency, or wave speed.
Complex nature of these sensors make them unsuitable for a brief overview of operating principals, as is suitable for this book. However, as these sensors are undergoing a rapid expansion, readers are encouraged to turn to references for more detailed explanations.

Biosensors

Detection of presence and concentrations of bacteria, viruses, or molecules and molecular complexes like proteins, enzymes, antibodies, DNA, etc. is essential to a wide range of applications.

Traditionally, this has been done through time-consuming chemical analysis methods, that require laboratory conditions and employ expensive reagents and equipment. Technological advancements and introduction of micro-sensor technology to this field has led to development of biosensors.

Much like a chemical sensor, biosensor consists of three parts: a sensitive layer, transducer and electronic circuitry to process the signal from the transducer.

Sensitive layer in a biosensor is a biosensitive biological component, like enzymes, antibodies, cell membrane receptors, tissue slices, etc.

Radiation sensors

Ionizing radiation consists of subatomic particles or waves that are energetic enough to detach electrons from atoms or molecules, ionizing them. Exposure to radiation causes microscopic damage to living tissue, resulting in skin burns and radiation sickness at high doses and cancer, tumors and genetic damage at low doses. Therefore, monitoring radiation levels is imperative in many industrial applications where human interaction with radioactive materials exists, as well as in guarding against intentional or accidental exposure of wider population to radiation. Wireless sensor networks provide ideal infrastructure for these kinds of systems.

Geiger-Müller counter
Geiger counters are used to detect radiation usually gamma and beta radiation, but certain models can also detect alpha radiation. The sensor is a Geiger-Müller tube, an inert gas-filled tube (usually helium, neon or argon with halogens added) that briefly conducts electricity when a particle or photon of radiation temporarily makes the gas conductive. The tube amplifies this conduction by a cascade effect and outputs a current pulse.

Quartz fiber dosimeter
A quartz fiber dosimeter is a pen-like device that measures the cumulative dose of ionizing radiation received by the device. The device is mainly sensitive to gamma and x-rays, but it also detects beta radiation above 1 MeV. Neutron sensitive versions have been made.

Film badge dosimeter
The film badge dosimeter, or film badge, is a dosimeter used for monitoring cumulative exposure to ionizing radiation. The badge consists of two parts: photographic film, and a holder.

The film is sensitive to radiation and, once developed, exposed areas increase in optical density (i.e. blacken) in response to incident radiation. One badge may contain several films of different sensitivities or, more usually, a single film with multiple emulsion coatings. This allows for separate measurement of neutron, beta, and gamma exposure, and estimation of energy spectra. The holder may contain a number of filters that attenuate certain types of radiation, such that only the target radiation is monitored. To monitor gamma rays or x-rays, the filters are metal, usually tin or lead. To monitor beta particle emission, the filters use various densities of plastic.

Thermoluminescent Dosimeter
A thermoluminescent dosimeter, or TLD, is a type of radiation dosimeter. A TLD measures ionizing radiation exposure by measuring the amount of visible light emitted from a crystal in the detector when the crystal is heated. The amount of light emitted is dependent upon the radiation exposure.

References

[1] Jon S. Wilson, Sensor Technology Handbook, Elsevier Inc., 2005.
[2] Pavel Ripka, Alois Tipek, Modern sensors handbook, ISTE Ltd, 2007.
[3] Ian R. Sinclair, Sensors and Transducers Third edition, Elsevier Inc., 2001.
[4] Richard P. Buck, Erno Lindner, Wlodzimierz Kutner, György Inzelt, Piezoelectric Chemical Sensors, Pure Appl.Chem., Vol.76, No.6, pp.1139–1160, 2004.

Medical Applications Based on Wireless Sensor Networks

Stanković, Stanislava

Abstract - *Wireless sensor networks aim to improve different aspects of our lives. This technology is experiencing true expansion for the past decades. Main concept in these systems is a sensor node, small microprocessor integrated with number of sensors. The sensor functions imply the WSNs' course of utilization. Possibilities are enormous. This paper focuses on WSNs utilization in medicine. In order to present the current state in this research field, few particular projects have been selected and compared.*

Keywords - *Wireless Sensor Networks (WSNs), Sensor Nodes, Utilization, Existing Medical Applications (EXMAs), Healthcare.*

Introduction

Wireless Sensor Network (WSN) is a set of small, autonomous devices, working together to solve different problems. It is a relatively new technology, experiencing true expansion for the past decade.

Research in the field of nanostructures and sensors has brought real opportunities for development of WSNs. People have realized that integration of small and cheap microcontrollers with sensors can result in production of extremely useful devices, which can be used as an integral part of the sensor nets. These devices are called sensor nodes. Nodes are able to communicate each other over different protocols. Studies, in the field of communication protocols for wireless sensor networks, are particularly interesting, and rely on various network topologies. Issues, addressed by communication among nodes, include power management, data transfer, mobility patterns, etc.

Like it was mentioned before, WSNs present new technology. History of this concept begins at University of California (UC), Berkley, with Smart Dust project, which was funded by Defense Advanced Research Projects Agency (DARPA) [1]. The aim of this project was to develop self-organized, millimeter-scale hardware platform for distributed WSNs. Primary, this was a military application which resulted in development of relatively large sensor nodes. Later miniaturization process brought much smaller devices, with solid sensing and communication capabilities.

One of the key points in the history of wireless sensor networks was the implementation of energy-efficient software platform, Tiny OS, operating system, also developed at UC. Further development lead to implementation of different software platforms for WSNs.

Soon, people have understood that, by putting sensor nodes to work together, infrastructure improvement and problem resolving can be achieved in different fields. The opinion, which was generally accepted, points on the low cost of this technology and its endless benefits. Today, sensor nets are used in agriculture, ecology, tourism, but medicine is the area where they certainly meet the greatest potential.

Today's healthcare

Nowadays, healthcare system is highly complex. List of elderly people and people in need for continuos care increases every day. Medical staff faces with more and more challenges each year. This opens serious questions in the domain of interest, which must be answered in the best way possible.

Problem solving must include detail analyses of the current state in order to form good and functional system which resolves the satisfying number of issues. Wireless sensor networks can offer this kind of solution.

WSNs – Existing medical applications (EXMAs)

The stress of this paper present short overview of medical applications based on WSNs, being developed in large research sectors. Projects, we will be talking about, are simply unified as EXMAs, and further will be specified so. EXMAs are recognized as significant part of today's research activities in the area of medical appliances of sensor networks. By highlighting the essence of these applications we understand the way WSNs can facilitate medical proficiency.

Firstly, some important issues, EXMAs are facing and which must be resolved during the implementation or deployment phase, will be indicated. Then, the technologies WSNs rely on will be emphasized, and a comparison of few selected applications will be given.

EXMAs – Important issues

The main issues of medical applications, based on wireless sensor networks, can be divided into three categories:

- technical issues
- social issues, and
- personal issues

Three specified domains cover remarkable number of problems. First one, the domain of technical issues, is common to all applications depending on WSNs. Social and personal issues are more related to health applications.

Technical issues include power management, communication protocols, network's reliability, and security.

Power management is the most important part of any WSNs' infrastructure, because of the fact that energy is limited. Data sampling costs power as well as data processing, but the biggest energy part is consumed by data transfer. Designers of sensor nets must accurately define how the data will be transferred through the network, in order to preserve energy as much as possible.

Communication protocols are another type of technical issues. Well structured protocols aim to resolve all the problems engineers face with. This includes power management, but also reliability and security, and other problems like congestion avoidance, load balancing, real time services, etc.

The stated concepts are real problems in the world of sensor nets, and improvement in this area is crucial part of researches.

Security could be referred as technical, but also as social issue. Data security and privacy must be provided in order to achieve acceptance of WSNs' technology. People are not willing to share their privacy, so different encryption techniques are used to secure data transfer through sensor networks. In case of medical applications, security of transferred data can be compared to doctors' ethic.

Besides security, **social issues** refer to general sensor nets' impact on social life. This technology must be well fitted into existing infrastructures. Earlier, it was mentioned that first designs of sensor nodes were quite large and that it was not satisfying.

New devices are small, but miniaturization process aims to produce even smaller and well-designed nodes. Nowadays, people carry a number of devices, mobile phones, I-pods, GPS devices, so the list expansion should not be a problem, as long as new devices fit into the profile.

Personal issues have physical and psychic face. Sensor nets, in any way, may not harm patients' health. These issues are probably the most important aspect which must be resolved before any serious network implementation. At this point, it is important to have in mind that WSNs aim to improve patients' lives, and not to compromise them.

EXMAs – WSNs vs. existing infrastructures

It is very important to highlight one of the most powerful advantages of WSNs, their ability to fit into existing solutions, and not to interfere among them. Here, some networking infrastructures will be mentioned, and the hierarchy of WSNs and those infrastructures will be discussed (Figure 1).

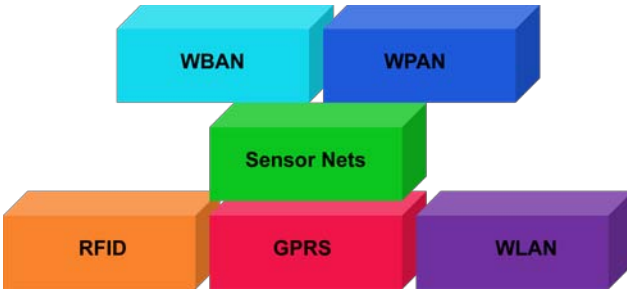


Figure 1: **Hierarchy of WSNs and existing infrastructures**

Today's face of networking is very different to the same face, just the couple decades before. Many new technologies have been developed and standardized. One of the advantages of wireless sensor networks is its compatibility with these technologies.

WSNs rely on existing infrastructures in order to provide real time services. Radio frequency identification (RFID) is low-power technology with number of benefits. Today, RFID tags are widely used for device tracking, so they can be used to keep track of sensor nodes.

Locating is an essential part in some projects based on WSNs, for example, emergency response applications. Global Positioning System (GPS) can be identified as basic supporting technology in this area.

Data, collected by sensor nets, need to be transferred and presented in some way. Choosing the Internet as representation portal seems like the most appropriate solution. This can be done over General Packet Radio Service (GPRS).

Internet connectivity can also be reached over wireless local area networks (WLANs/802.11). Representation of the same data on mobile phones can be achieved via Bluetooth technology.

Basic technologies, sensor networks rely on have been introduced. Combining of different frameworks leads to deployment of different appliances of sensor nets, wireless personal area networks (WPANs) or wireless body area networks (WBANs).

EXMAs – State of the art

Different projects, based on WSNs, are being developed, worldwide. These projects can mostly be referenced as academic researches, still finding their way into industry sectors. Here, couple of projects with different interests will be mentioned, and a corresponding comparison will be given.

The comparison will be done in terms of general description and expected contribution. This overview aims to discover the progress which has been made in the area of EXMAs, for the past few years. Purpose of this preview is to provide persons, willing to develop something similar, with well fundamental knowledge. Before development some research needs to be done, so this paper lists some of the projects already been developed in order to address things which must be taken into consideration.

UbiMon project

UbiMon (ubiquitous monitoring environment for wearable and implantable sensors) is the architecture for distributed mobile monitoring, developed at Imperial College London.

The aim of this system is to provide continuous management of patients under their natural physiological states so that transient but life threatening abnormalities can be detected and predicted [2].

The use of wearable and implantable sensors is the area of research with a promising future, and the UbiMon system presents the framework for integration of these sensors into the body sensor network (BSN).

UbiMon – contribution

The UbiMon system, besides continuous patients' monitoring in their natural environment, provides the architecture for collection, gathering and analyzing data from a number of biosensors (wearable or implantable). Another significant aspect provided with this architecture is the context awareness which enables the capturing of any clinical relevant episode.

Satire project

Satire (software architecture for smart attire) is a wearable personal monitoring service, developed through cooperation between University of Illinois and University of Virginia. It allows users to maintain a private searchable record of their daily activities as measured by motion and location sensors, which are two of the most popular sensing modalities in personal instrumentation [3]. The main idea is to deploy non-obtrusive wearable monitoring platform for data sensing, local storage and occasional upload in order to satisfy medical, safety, personal or entertainment reasons.

Satire – contribution

Major contribution of the Satire system is user transparency. Project's goal includes the implementation of the service which provides information collection, storage and uploads without explicit input or maintenance, required from the user. Satire presents new wearable platform limited by the currently available off-the-shelf components.

Alarm-Net project

Alarm-Net is a wireless sensor network for assisted-living and residential monitoring being developed on University of Virginia. It integrates environmental and physiological sensors in a scalable, heterogeneous architecture [4]. The focus of this network is to provide smart environment and to secure patients' monitoring on daily basis. This project aims to facilitate the concept of health care by including informal caregivers, such as family, friends, and patients' themselves in the process. Early detection and prevention of diseases insures costs' reducing with simultaneous care quality maintenance.

Alarm-Net – contribution

Contributions of the Alarm-Net system include development of scalable, heterogeneous network, based on custom hardware components; implementation of context aware protocols, informed by CAR analysis, which enable smart power management and dynamic alert-driven privacy depending on individual's activity patterns; design of SecureComm messaging protocol which supports user based decisions for selection of security policies.

BikeNet project

BikeNet is an extensible mobile sensing system, being developed through collaboration of Dartmouth College and Columbia University, for cyclist experience mapping leveraging opportunistic sensor networking principles and techniques. It represents a multifaceted sensing system and explores personal, bicycle and environmental sensing using dynamically role-assigned bike area networking [5].

BikeNet – contribution

Contributions of the BikeNet system include cyclist performance and fitness management by collecting data about cyclist's current and average speed, distance travelled, calories burned, along with time and location metadata; environment and experience mapping by emphasizing the healthiness levels in terms of pollution, allergens, noise, and terrain's roughness; long term performance trend analysis which enables the upload of data traces into personal repository and selective data sharing; data collection and local presentation customized by cyclists; data query and remote presentation on the web-based portal; disconnected operations which refer to opportunistic networking paradigm, or the fact that data uploading happens according to the opportunities that arise as a result of cyclists' uncontrolled mobility.

Conclusion

To conclude, the expansion of WSNs for the past few years will be emphasized. This certainly is a rising technology. Like it was stated before, great advantage of sensor nets is their compatibility to existing infrastructures. Another advantage lies in the fact that these networks meet a number of completely different appliances, agricultural, ecologic, and especially medical. Miniaturization of sensor nodes and overcoming of demonstrated issues will bring more sophisticated solutions and applications.

With all new technologies we face with problem of failure or success. Despite the promising features, risks of failure are always present. In order of overcoming this problem some

necessary points must be done. First of all, start with theoretical basses' learning, and then do some practical work in order to gain knowledge and to improve the state of the art.

Existing medical applications based on wireless sensor networks are still research projects with good potential for utilization. Great number of medical scenarios is being covered with these applications and that opens wide spectrum of benefits for caregivers. They should be able to provide the necessary care whether in emergency situations, or in some hospital environments, or in form of smart homes. A lot of effort needs to be done before the EXMAs' ultimate aim is reached. In this case, the ultimate goal implies the improvement of human lives.

For the end, from this very point, it can be said that the future of WSNs and their medical appliances looks highly promising.

References

[1] M. Yacoub (foreword), G.-Z. Yang (Ed.), *Body Sensor Networks*, London: Springer-Verlag, 2006, pp 24-26.

[2] "Ubiquitous Monitoring Environment for Wearable and Implantable Sensors", *in International Conference on Ubiquitous Computing (UbiComp)*, Tokyo, Japan, Sep 11-14, 2004.

[3] R. K. Ganti, P. Jayachandran, T. F. Abdelzaher, J. A. Stankovic, "SATIRE: A Software Architecture for Smart AtTIRE", in *International Conference on Mobile Systems, Applications, and Services (MobiSys)*, Uppsala, Sweden, June 19-22, 2006.

[4] A. Wood, G. Virone, T. Doan, Q. Cao, L. Selavo, Y. Wu, L. Fang, Z. He, S. Lin, J. Stankovic, "ALARM-NET: Wireless Sensor Networks for Assisted-Living and Residential Monitoring," *Technical Report CS-2006-11*, Department of Computer Science, University of Virginia, 2006.

[5] S. B. Eisenman, N. D. Lane, E. Miluzzo, R. A. Peterson, G-S. Ahn, A. T. Campbell, "BikeNet: a mobile sensing system for cyclist experience mapping"*, ACM Transactions on Sensor Networks* (under review).

Processing of Medical Signals (ECG) in Wireless Sensor Networks

Ćirković, Predrag; Aleksić, Anđa

Abstract - Combination of embedded systems and wireless technology provides a lot of opportunities for acquisition, processing and transmission of data. It is useful to apply these technology approaches for solving medical problems.

ECG is the record of the heart muscle electric impulses. Received and processed ECG signal could be analyzed, and results could be used for detection and diagnostics of heart diseases. One of the important medical problems is arrhythmia. In this work, the aim was to detect them in the easiest way by detecting R waves, and measuring R-R intervals. A part of the researching is, also, how to register the ECG signal in action and inaction in some enveloped area.

I. System Architecture

Wireless system for ECG monitoring is composed of two hardware applications: *Access Point* and *End Device* and one software application called *Doctor's Assistant*. If it is required to expand the range of wireless connection, it is possible to include specific number of *Range Extender* motes.

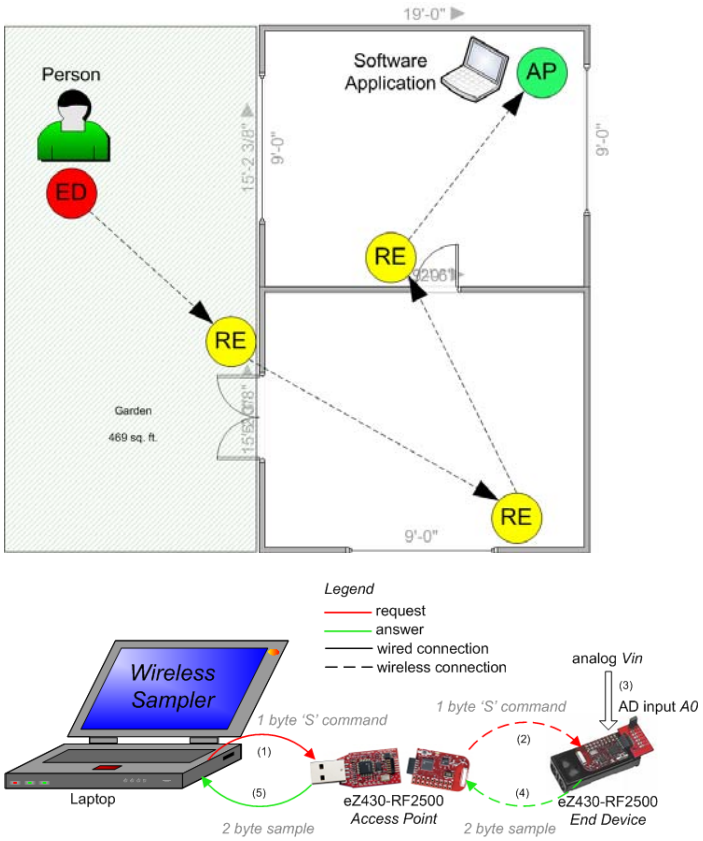
Hardware applications are developed in *IAR Embedded Workbench* environment for *eZ430-RF2500* motes with *MSP430F2274* microcontroller [1].

Software application

is developed in *National Instruments CVI* tool. It is installed on a PC, and represents a command interface between user (a doctor) and *End Device* mote. The user has ability to specify the measuring program, selecting between two measuring modes: *On Demand* and *Continuous Measuring*.

Predrag Ćirković,
School of Electrical Engineering
Belgrade, Serbia

Anđa Aleksić
Faculty of Medicine
Belgrade, Serbia



II. ECG (Electrocardiogram)

A. Generating of Normal Electric Impuls in Heart

Normal electric impulse derives from the *sinoatrial (SA) node*, which is located in the upper wall of right atrium. Sinoatrial node is a group of special muscle cells in heart, which are able to generate impulses and to induce *cardiomyocytes* to contract. Cardiomyocytes are the major structure of a heart. They build two atriums and two ventricles.

That impulse is called *depolarization*. So, ECG signal is a registration of depolarization spreading through heart muscle. SA node is also called *pacemaker*, because it is the leader of the heart beat rhythm. He produces 60-100 heart beats (cycles) per minute, and that is a normal heart rate [2]. Human physiology passes this way: primary impulse from the SA node is taken by inter-nodal ways that connect the SA node and *AV (atrioventricle) node*, which is placed between ventricles and atriums of a heart. Further, signal goes

by *Hiss's bundle* into ventricle muscle. These three elements: SA node, AV node and Hiss's bundle are components of the conduction system of a heart (Fig. 1) [3].

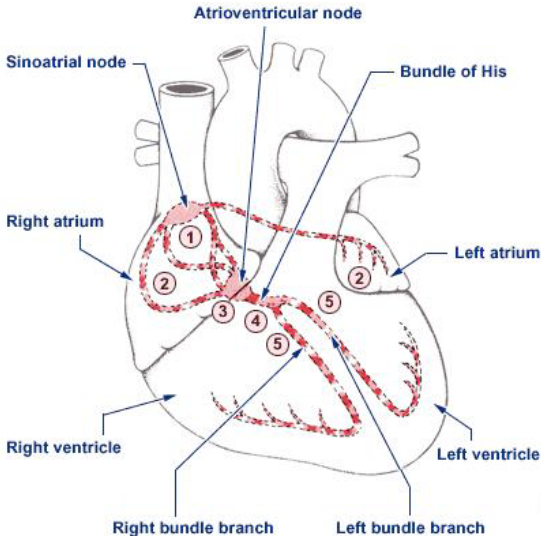


Figure 1: Heart and its conduction system

B. Heart Cycle

First, atriums become depolarized (stimulated by primary impulse from SA node). Depolarization is the main cause for atrium contraction. This phase corresponds to P wave on the ECG. Next, depolarization wave is conducted by inter-nodal passes to the AV node. That action is followed by small break which is caused by stopping impulse in AV node. The reason for that is ventricle preparing for action. Period that shows these two parts of the heart cycle is PQ segment of the ECG. At last, depolarization wave comes to ventricles and push them into contraction, QRS complex, and relaxation of ventricles, T wave. While ventricles contract, atriums relax. That is the whole heart cycle.

C. ECG Construction

ECG signal consists of: P wave, QRS complex and T wave (Fig. 2) [4] [5]. P wave shows us the activation of atriums (depolarization). QRS complex shows us the ventricles action (depolarization). T wave is a diagram of *repolarization* of ventricle muscle cells, coming back into stable stage for another contraction.

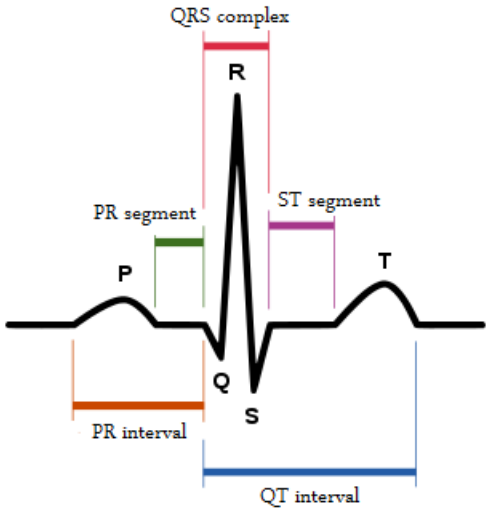


Figure 2: ECG signal.

D. Important Parameters from an ECG Signal

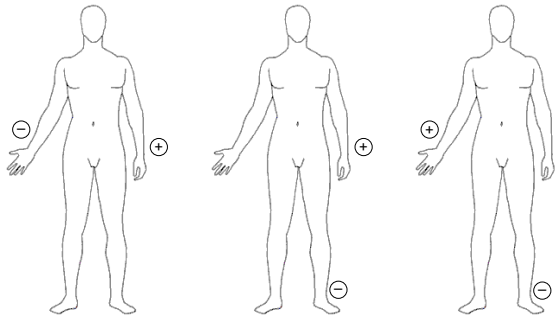
1) Appearance of waves and sprockets,

2) Duration of waves, sprockets, segments and intervals,

3) Amplitude of waves and sprockets,

4) Rhythm of heart cycles,

5) Un/presence of waves or sprockets.



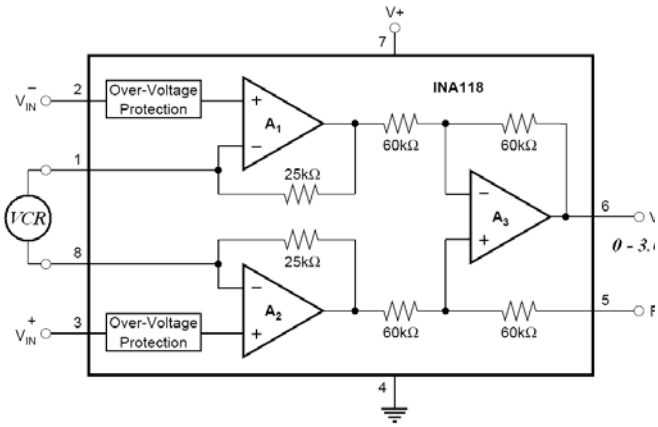
III. Analog Signal Processing

Electrical activity of the heart is detected by electrodes, which are fixed on specific places on human body.

Amplitude of detected signal is very low, usually between 0.5 mV to 4 mV [6], so the signal is very liable to noise influence. Instrumentation amplifier is used for bioelectric signal amplifying because of its characteristics: high input impedance, high common mode rejection ratio (CMRR) and changeable gain.

Gain is adjustable in range 900 – 6800, by next equation:

$$G = 1 + \frac{50K\Omega}{R_G} \qquad (1)$$



The analog circuit is shown on Figure x. And it presents the instrumentation amplifier in one pack which is battery powered and it is adjustable for…

In (1) R_G is variable resistance and it is implemented as a voltage controlled resistor (FET transistor) to get ability to adduce the gain (G) from the control interface, over the wireless communication. Transistor's gate should be connected to the output of the D/A converter of the microcontroller.

Instrumentation amplifier, INA118, is used because of its small size and very wide supply voltage range, usefully for battery supply [7].

Maximum amplified signal is in range from 0 V to 3.6 V.

MSP430F2274's 10-bits A/D converter, ADC10, is used for amplified signal discretization. Discrete signal is in range 0 to 1023 of unsigned short integer and it is stored to memory of microcontroller, block-by-block. Size of a sample block is about 400 B.

IV. FILTERING

ECG signal is liable to following types of noise [8]: *baseline wander* – LF noise, which arises from respiration and other physiology actions, *power line nose* – HF noise of 50 Hz/60 Hz and *muscle noise* – arises from action of other muscles in human body.

A. High Pass Filtering

Baseline wander noise is eliminated by HP filter. Cutoff frequency of the filter is equal to the lowest frequency of the slowest heart rate (about 40 bpm). It is about 0.67 Hz, so the chosen cutoff frequency is 0.5 Hz [8].

It is preferable to use a finite impulse response filter (FIR), to avoid phase distortion of the ECG signal.

There is a problem to design the HP filter with very low cutoff frequency and very tight transition area, because of its high order. Algorithm of filtering must be run in a real-time on a platform with very restrictive resources. This problem should be solved in appropriate way.

There is a proposal to use the algorithm for subtraction average curve from the original signal. Average curve is attained by averaging the ECG signal on 2 s interval. Hereafter, it is required to use the interpolation (resampling) of average curve to the length of ECG signal. It is possible to implement some of the fast algorithms of interpolation to increase the efficiency of filtering.

The third proposal is to use the median filtering. Problem with this is usage of very long median mask. There is a question mark over that algorithm be efficient enough.

HP filtering is shown on Fig. 1. Red line shows the mean value of the ECG signal, before and after median filtering.

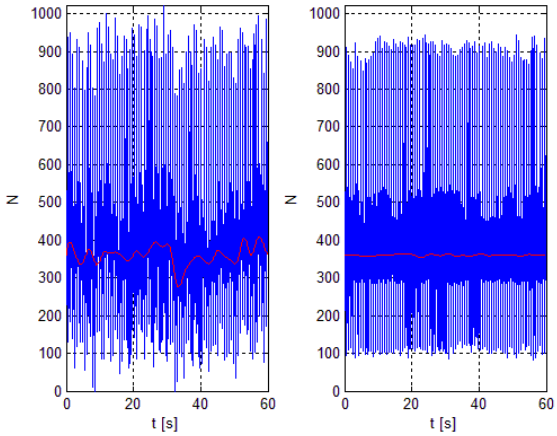


Figure 3: HP filtering with median filter.

B. Low Pass Filtering

Most of the power line noise is eliminated by instrumentation amplifier, because of its high CMRR. Additional low pass filtering is attained by using the low pass filter with 40 Hz cutoff frequency.

There is a proposal to use median filter instead of LP.

The examples of filtered signal with 3, 5 and 7 samples long mask are shown on Fig. 2.

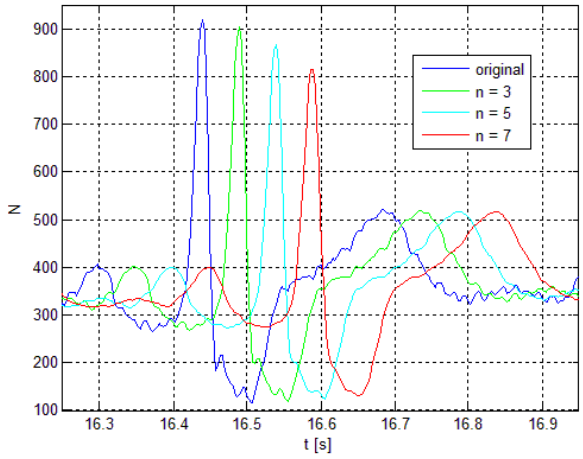


Figure 4: LP filtering with median filter.

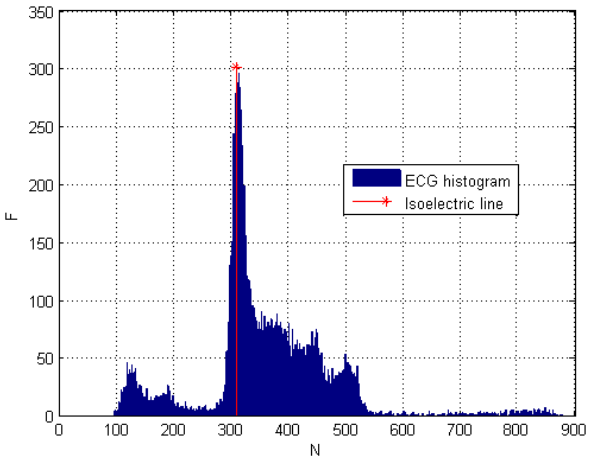


Figure 5: Histogram of an ECG signal.

V. COMPRESSION

It is required to do the compression of signal, before transmission of signal from the End Device to the Access Point. After the low pass filtering it is possible to decimate signal to reduce the sampling frequency and size of data package for transmission.

Histogram of ECG signal (Fig. 5) is counted for one of ECG signals taken from *physionet* database [9]. It shows that one set of values is occurred more frequently than others. Based on that conclusion, the *Huffman coding* is chosen as a lossless compression method.

This algorithm refers to packing array of words to continuous bit stream, where the shortest code is used for encoding the most frequent occurred value, and the longest code, for encoding the least frequent occurred value. There is a problem of rapid grow of code length for less frequent occurred values. So, it might be necessary to use some of modified Huffman encoding algorithms [10].

VI. COMMUNICATION PROTOCOL

Communication between the software application and the microcontroller of the Access Point is done over the serial interface called MSP430 Application UART with following specifications: virtual COM1 port, 8 data bits, without parity bit and with one stop bit.

Wireless communication is accomplished over the radio interface CC2500 placed on the eZ430-RF2500, which communicates with microcontroller over its SPI interface. All communication functions are supported by API of *SimpliciTI* protocol [11].

VII. ANALYSIS AND DIAGNOSTICS

Arrhythmia is any kind of heart beat rhythm changing. There are a lot of them, like bradyarrhthmia, tachyarrhythmia, paroxysmal atrium contraction, paroxysmal ventricle contraction, ventricle tachyarrhythmia, AV block (I, II, III degree...). They have typical appearance in the ECG signal. That characteristic could be used for analysing of signal and getting diagnose.

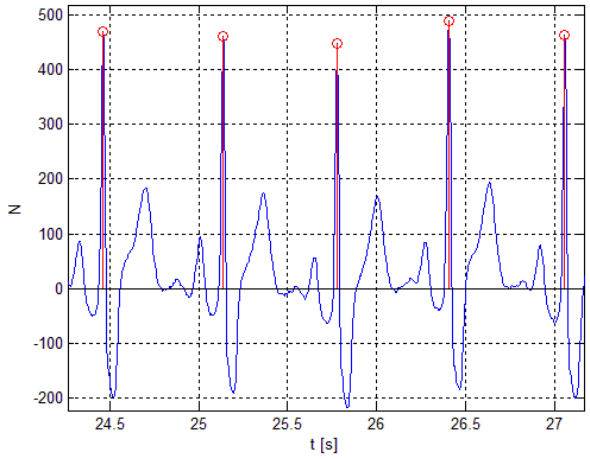


Figure 6: Time diagram of ECG signal with R waves marks.

It is very important to find the *isoelectric line* of an ECG signal, to establish signal for correct further analysis. Isoelectric line is defined by x axis value where histogram riches the maximum value (Fig. 5). That value is subtracted from all signal samples (Fig. 6).

Using the algorithm for peak detection [12], it is possible to find maximum values of R waves. Some changes of the algorithm are required to be possible to detect other important extreme values (P, Q, S and T). Till now, using the peak detection algorithm, the following parameters are determined: R-R duration and amplitudes of R waves.

Further analyses are accomplished to detect disturbances of an ECG signal, which enable getting adequate diagnose. It is possible to identify arrhythmia, based on found results of appearing of R waves. It could be calculated the mean variation of time intervals between R waves, expressed in percents.

$$arr = \frac{\sum |diff\,[t_R]|}{n-1} \cdot 100\%. \qquad (2)$$

In (2) arr is arrhythmia rated in percents, $[t_R]$ is array of time points (moments) where the *R* waves reach their maximum values and n is number of detected *R* waves.

$$marr = \frac{|\max(diff\,[t_R]) - \langle [t_R] \rangle|}{\langle [t_R] \rangle} \cdot 100\%. \qquad (3)$$

In (3) $marr$ is maximum arrhythmia rated in percents.

Formulas (2) and (3) are used for getting the diagnose for chosen signals from physionet database (16256 signal from *MIT-BIH Normal Sinus Rhythm Database (nsrdb)*, and 201 and 212 signals from *MIT-BIH Arrhythmia Database* [9]).

It could be concluded that these three signals have differences in duration of R-R intervals. If we know that normal heart rate is between 60 and 100 bpm, it also could be said that R-R intervals could last from 1 s to 0.6 s. It is known that arrhythmia means any change of heart beat rhythm, so if standard R-R intervals are registered in the ECG signal, that does not mean that it is not some kind of arrhythmia. That is proved by this research. It is shown that significant deviation of R wave appearance could show arrhythmia.

Table 1: Researching results.

Signal No.	16256	201	212
R-R [s]	0.63	0.67	0.66
Heart rate [bpm]	95	90	91
Arrhythmia [%]	1.46	9.86	2.57
Max var. [%]	9.27	51.8	15

VIII. Target Groups of Patients

A. People with heart disease

B. People from high risk group

1) Older than 55 years of any gender

2) Younger than 55years with cardiovascular disease (arrhythmia, acute myocardial infarction, stable angina pectoris, unstable angina pectoris, hypertension), diabetes mellitus, hypothyroidism, obesity.

References

[1] Texas Instruments Inc, "eZ430-RF2500 Development Tool User's Guide", 2008.
[2] D. Manojlović "Internal Medicine", chapter – arrhythmias 1998.
[3] The Merch Manual of Medical Information – Home Edition, p.80, edited by Mark H. Beers and Robert Berkow. Copyright 1997 by Merch & Co., Inc, Whitehouse Station, NJ.
[4] Wikipedia, Electrocardiogram, http://en.wikipedia.org/wiki/ecg.
[5] D. Dubin, "Rapid interpretation of ECG's", Romanov.
[6] X. Zhang, "A Design of ECG Amplifier", ECE 525 Project #1, 2003.
[7] Burr-Brown "Precision, Low Power Instrumentation Amplifier".
[8] M. Aksela, "ECG Filtering", 2004.
[9] PhysioBank Archive, ECG databases http://www.physionet.org/physiobank/database/#ecg.
[10] M. Popović, "Digital Image Processing", Academic mind, Belgrade, 2006.
[11] Texas Instruments Inc, "SimpliciTI Overview" http://www.ti.com/litv/pdf/swru130a.
[12] E. Billaouer, "peakdet: Peak detection using MATLAB" http://billauer.co.il/peakdet.html.

Reviewers by Countries

Argentina
Olsina, Luis; National University of La Pama
Ovando, Gabriela P.; Universidad Nacional de Rosario
Rossi, Gustavo; Universidad Nacional de La Plata
Australia
Abramov, Vyacheslav; Monash University
Begg, Rezaul; Victoria University
Bem, Derek; University of Western Sydney
Betts, Christopher; Pegacat Computing Pty. Ltd.
Buyya, Rajkumar; The University of Melbourne
Chapman, Judith; Australian University Limited
Chen, Yi-Ping Phoebe; Deakin University
Hammond, Mark; Flinders University
Henman, Paul; University of Queensland
Palmisano, Stephen; University of Wollongong
Ristic, Branko; Science and Technology Organisation
Sajjanhar, Atul; Deakin University
Sidhu, Amandeep; University of Technology, Sydney
Sudweeks, Fay; Murdoch University
Austria
Derntl, Michael; University of Vienna
Hug, Theo; University of Innsbruck
Loidl, Susanne; Johannes Kepler University Linz
Stockinger, Heinz; University of Vienna
Sutter, Matthias; University of Innsbruck
Walko, Zoltan
Brazil
Parracho, Annibal; Universidade Federal Fluminense
Traina, Agma; University of Sao Paulo
Traina, Caetano; University of Sao Paulo
Vicari, Rosa; Federal University of Rio Grande
Belgium
Huang, Ping; European Commission
Canada
Fung, Benjamin; Simon Fraser University
Grayson, Paul; York University
Gray, Bette; Alberta Education
Memmi, Daniel; UQAM
Neti, Sangeeta; University of Victoria
Nickull, Duane; Adobe Systems, Inc.
Ollivier-Gooch, Carl; The University of British Columbia
Paulin, Michele; Concordia University
Plaisent, Michel; University of Quebec
Reid, Keith; Ontario Ministry og Agriculture
Shewchenko, Nicholas; Biokinetics and Associates
Steffan, Gregory; University of Toronto
Vandenberghe, Christian; HEC Montreal
Croatia
Jagnjic, Zeljko; University of Osijek
Czech Republic
Kala, Zdenek; Brno University of Technology
Korab, Vojtech; Brno University of technology
Lhotska, Lenka; Czech Technical University
Cyprus
Kyriacou, Efthyvoulos; University of Cyprus
Denmark
Bang, Joergen; Aarhus University
Edwards, Kasper; Technical University Denmark
Orngreen, Rikke; Copenhagen Business School
Estonia
Kull, Katrin; Tallinn University of Technology
Reintam, Endla; Estonian Agricultural University
Finland
Lahdelma, Risto; University of Turku
Salminen, Pekka; University of Jyvaskyla
France
Bournez, Olivier
Cardey, Sylviane; University of Franche-Comte
Klinger, Evelyne; LTCI – ENST, Paris
Roche, Christophe; University of Savoie
Valette, Robert; LAAS - CNRS
Germany
Accorsi, Rafael; University of Freiburg
Glatzer, Wolfgang; Goethe-University
Gradmann, Stefan; Universitat Hamburg
Groll, Andre; University of Siegen
Klamma, Ralf; RWTH Aachen University
Wurtz, Rolf P.; Ruhr-Universitat Bochum
Greece
Katzourakis, Nikolaos; Technical University of Athens
Bouras, Christos J.; University of Patras and RACTI
Hungary
Nagy, Zoltan; Miklos Zrinyi National Defense University
India
Pareek, Deepak; Technology4Development
Scaria, Vinod; Institute of Integrative Biology
Shah, Mugdha; Mansukhlal Svayam
Ireland
Eisenberg, Jacob; University College Dublin
Israel
Feintuch, Uri; Hadassah-Hebrew University
Italy
Badia, Leonardo; IMT Institute for Advanced Studies
Berrittella, Maria; University of Palermo
Carpaneto, Enrico; Politecnico di Torino
Japan
Hattori, Yasunao; Shimane University
Livingston, Paisley; Linghan University
Srinivas, Hari; Global Development Research Center
Obayashi, Shigeru; Institute of Fluid Science, Tohoku University
Mexico
Morado, Raymundo; University of Mexico
Netherlands
Mills, Melinda C.; University of Groningen
Pires, Luís Ferreira; University of Twente
New Zealand
Anderson, Tim; Van Der Veer Institute

Philippines
Castolo, Carmencita; Polytechnic University Philippines
Poland
Kopytowski, Jerzy; Industrial Chemistry Research Institute
Portugal
Cardoso, Jorge; University of Madeira
Natividade, Eduardo; Polytechnic Institute of Coimbra
Oliveira, Eugenio; University of Porto
Republic of Korea
Ahn, Sung-Hoon; Seoul National University
Romania
Moga, Liliana; "Dunarea de Jos" University
Serbia
Mitrovic, Slobodan; Otorhinolaryngology Clinic
Stanojevic, Mladen; The Mihailo Pupin Institute
Ugrinovic, Ivan; Fadata, d.o.o.
Singapore
Tan, Fock-Lai; Nanyang Technological University
Slovenia
Kocijan, Jus; Jozef Stefan Institute and University of Nova Gorica
South Korea
Kwon, Wook Hyun; Seoul National University
Spain
Barrera, Juan Pablo Soto; University of Castilla
Gonzalez, Evelio J.; University of La Laguna
Perez, Juan Mendez; Universidad de La Laguna
Royuela, Vicente; Universidad de Barcelona
Vizcaino, Aurora; University of Castilla-La Mancha
Vilarrasa, Clelia Colombo; Open University of Catalonia
Sweden
Johansson, Mats; Royal Institute of Technology
Switzerland
Niinimaki, Marko; Helsinki Institute of Physics
Pletka, Roman; AdNovum Informatik AG
Rizzotti, Sven; University of Basel
Specht, Matthias; University of Zurich
Taiwan
Lin, Hsiung Cheng; Chienkuo Technology University
Shyu, Yuh-Huei; Tamkang University
Sue, Chuan-Ching; National Cheng Kung University
Ukraine
Vlasenko, Polina; EERC-Kyiv
United Kingdom
Ariwa, Ezendu; London Metropolitan University
Biggam, John; Glasgow Caledonian University
Coleman, Shirley; University of Newcastle
Conole, Grainne; University of Southampton
Dorfler, Viktor; Strathclyde University
Engelmann, Dirk; University of London
Eze, Emmanuel; University of Hull
Forrester, John; Stockholm Environment Institute
Jensen, Jens; STFC Rutherford Appleton Laboratory
Kolovos, Dimitrios S.; The University of York
McBurney, Peter; University of Liverpool
Vetta, Atam; Oxford Brookes University
Westland, Stephen; University of Leeds
WHYTE, William Stewart; University of Leeds
Xie, Changwen; Wicks and Wilson Limited
USA
Bach, Eric; University of Wisconsin
Bazarian, Jeffrey J.; University of Rochester School
Bolzendahl, Catherine; University of California
Bussler, Christoph; Cisco Systems, Inc.
Charpentier, Michel; University of New Hampshire
Chester, Daniel; Computer and Information Sciences
Chong, Stephen; Cornell University
Collison, George; The Concord Consortium
DeWeaver, Eric; University of Wisconsin - Madison
Ellard, Daniel; Network Appliance, Inc
Gaede, Steve; Lone Eagle Systems Inc.
Gans, Eric; University of California
Gill, Sam; San Francisco State University
Gustafson, John L..; ClearSpeed Technology
Hunter, Lynette; University of California Davis
Iceland, John; University of Maryland
Kaplan, Samantha W.; University of Wisconsin
Langou, Julien; The University of Tennessee
Liu, Yuliang; Southern Illinois University Edwardsville
Lok, Benjamin; University of Florida
Minh, Chi Cao; Stanford University
Morrissey, Robert; The University of Chicago
Mui, Lik; Google, Inc
Rizzo, Albert ; University of Southern California
Rosenberg, Jonathan M. ; University of Maryland
Shaffer, Cliff ; Virginia Tech
Sherman, Elaine; Hofstra University
Snyder, David F.; Texas State University
Song, Zhe; University of Iowa
Wei, Chen; Intelligent Automation, Inc.
Yu, Zhiyi; University of California
Venezuela
Candal, Maria Virginia; Universidad Simon Bolívar

IPSI Team
Advisors for IPSI Developments and Research:
Zoran Babovic, Darko Jovic, Aleksandar Crnjin, Marko Stankovic, Marko Novakovic

Welcome to IPSI BgD Conferences and Journals!

http://www.internetconferences.net

http://www.internetjournals.net

VIPSI-2009 AMALFI
Hotel Santa Caterina
Amalfi, Italy
March 5 to March 8, 2009

VIPSI-2009 BELGRADE
Belgrade, Serbia
April 2 to Aprli 5, 2009

VIPSI-2009 CROATIA - OPATIJA
Opatija/Abbazia
Villa Ariston
May 28 to May 31, 2009

VIPSI-2009 TIVAT
Tivat, Montenegro,
August 20 to August 23, 2009

VIPSI-2009 SLOVENIA
Lake Bled in Alps
Slovenia
September 24 to September 27, 2009

VIPSI-2009 VENICE
Venice, Italy
September 27 to September 30, 2009

VIPSI-2010 AMALFI
Hotel Santa Caterina
Amalfi, Italy
March 4 to March 7, 2010

VIPSI-2010 TIVAT
Tivat, Montenegro
June 3 to June 6, 2010

VIPSI-2010 VENICE
Venice, Italy
September 30 to October 3, 2010

VIPSI-2010 MILOCER
Milocer, Montenegro
December 30 to December 31, 2010

CIP – Katalogizacija u publikaciji
Narodna biblioteka Srbije, Beograd
ISSN 1820 – 4511 =
The IPSI BGD Transactions on Advanced Research
COBISS.SR - ID 119128844