INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Wireless Sensor Based on Bluetooth Technology

Hervé MATHIEU

N° 0289

December 2003

THÈME 3

ISSN 0249-0803

Wireless Sensor Based on Bluetooth Technology

Hervé MATHIEU (herve.mathieu@inrialpes.fr)

Theme 3: human-computer interaction, image processing, data management, knowledge systems

Technical report n° 0289 – December 2003 - 56 pages

Abstract: This document discusses a project which aimed of building a wireless sensor device. The objective of the project is to master a new sensor approach. Target applications are robotic systems including medical robots.
This project is decomposed in two phases. These two phases are reported in the main parts of this document (*overview* and *proof-of-concept prototype*).

The first phase deals with concept and general architecture. The following actions take place:

- We present target applications, and the kind of sensor that can be used in them.
- Existing similar research projects were investigated: Usually, documents related to these projects are open documents. That means that one has an idea about how work has been done, and what can be done (limits) in terms of technology. The drawbacks are that final products do not exist or that they are not described in any document.
- We investigated also off-the-shelf (commercial) wireless sensors: This class of information is useful to know what is reachable in terms of final products. However, you do not know how company did it: the "Commercial product" term implies usually a secret process.
- We list the electronic modules, which can be candidate for a wireless sensor device. That implies RF (Radio Frequency) modules, micro controller modules, energy technology, and sensor devices.
- We report uncertainties about technology and functionality that should be cleared up within a prototype project.

The second phase is related to a proof-of-concept prototype. The goal of this prototype is to know if the concept, as defined during the first phase, works in real.
We report how we successfully built a wireless sensor. Aspects presented are as follows:

1. About the test bed itself: specification, hardware description, embedded software, purchased list and cost, are given.
2. About the development environment: tools for programming and debugging the test bed, and experiments performed are described.
3. We report also the experiments results and discuss the prototype.

Off-the-shelf modules have been bought when existing. We tested the concept by using a dual axis inclination sensor. We chose this sensor because we had a previous experience with it. When designing the test bed, we tried, as much as possible, to get close to a usable product. Therefore, we took care of the sensor dimensions, the power consumption, etc.

Keywords: sensor, MEMS, wireless, Bluetooth, micro controller

Unité de recherche INRIA Rhône-Alpes
655, Avenue de l'Europe, 38330 Montbonnot-St-Martin (France)
Téléphone : +33 4 76 61 52 00 – Télécopie : +33 4 76 61 52 52

Wireless Sensor Based on Bluetooth Technology

Author:

Hervé MATHIEU

Research Engineer at INRIA Rhône-Alpes
Hardware & Software systems integration

Email: herve.mathieu@inrialpes.fr
Web home page: http://www.inrialpes.fr/sed/people/mathieu/Welcome.html

6.1 Sensor providers 44
6.2 Micro controller module providers 44
6.3 Wireless sensor providers (including Bluetooth & other RF technologies) 44
6.4 Bluetooth chip or module providers 44
6.5 Bluetooth stack providers 44
6.6 Other Bluetooth related companies 44
6.7 Other RF solutions 44
6.8 Battery related web sites 44
7 Annexes 45
7.1 Interface module schematic 45
7.2 Interface Module PCB component and net lists 46
7.3 Mounting instructions 48
7.4 Component list and cost 51
7.5 Project milestones 52
7.6 Wireless Sensor datasheet 53

In this kind of robotic systems, all the sensor devices are closed to the controller (less than 2 meters). Wires have to go through structure and moving joints. This makes wiring quite hard. Wiring grounds properly is also a big issue. We have to take care of this for avoiding ground loop effects.

Wireless sensor technology would help. It would result in dramatic hardware simplification, and would make the whole system more robust.

2.3.2 Medical applications

Robotic systems are used for medical applications, especially for surgical procedures. Medical robotics aims at improving the capabilities of physicians to perform surgical procedures. Some existing applications are listed here.

1. A typical robot-assisted procedure is a biopsy under CT imager control. The procedure consists in the following steps:
    - Step 1: taking a CT scan where target is supposed to be located,
    - Step 2: target and entry-point identifications are made using CT slices. Target is usually between 20 and 100 millimeters depth inside the body. The skin entry-point represents the hole where the linear tool (needle, catheter or bone drill) will be inserted. This two-point identification gives three parameters in case of a linear tool: two angles for tool orientation and one distance (depth) for insertion.
    - Step 3: manual or robot-assisted placement typically includes the following three decoupled tasks: (1) touch down with the tool tip on the entry-point, (2) orient the tool by pivoting around the skin entry-point, and (3) insert the tool into the body along a straight trajectory.

    Performing this robot-assisted procedure requires registration of robot and imager. Registration process consists in finding a 4x4 matrix. This matrix represents how coordinates change between imager coordinate system and robot coordinate system. Registration requires several points of interest which are paired in both coordinate systems. Pairing is always a critical phase.

    Adding sensors improves the registration process. For instance some of the points of interest described above allow calculating what the vertical direction for both coordinate systems is. An inclination sensor gives directly this information Therefore less points of interest have to be paired. This helps reducing matrix resolution complexity, and so makes the registration process more robust.

    Adding inclinometer sensor on a robot improves also safety. It can be used as redundant encoder, because it gives information about robot orientation too. Both information coherencies can be checked, thus avoiding hazardous robot behavior due to motors encoder dysfunction.

2. During a total knee replacement, procedure includes machining bones. Some robotic systems perform this procedure, like ROBODOC from Integrated Surgical Systems (ref [web01]). A typical problem is doing registration between the robot and the patient's body. A common registration is based on localizers like Optotrack or Polemus from North Digital (ref [web02]). Basically, several tracker devices are attached to the body, and the others to the robot. All tracker devices are seen by the localizer, and so they are represented in a unique coordinate system. This makes possible registration between the robot and the patient's body.

    The drawbacks for using common localizers are the following: (1) tracking failure due to occlusion (tracking devices commonly used cameras), (2) a bunch of wires around patient due to wired tracker devices. These active trackers (versus passive trackers) improve precision, (3) a large area inside surgery room is dedicated to the localizer system.

    Adding wireless inclination sensor improves procedure. It helps on positioning when occlusions occur. Mixing tracking and sensor outputs improves precision. Hence passive (non-wired) tracker could be considered.

Oceana Sensor (ref [web10]) is a manufacturer of sensing systems for a wide variety of applications including machinery health, seismic monitoring or impact detection. They instrument sensors with Bluetooth interface. Products are clearly end-products, not for OEM.

Millennial Net (ref [web13]) produces low-power wireless sensors. These products are used in industrial automation, building automation, supply-chain management, automatic meter reading, security, and other low-data-rate applications. The company's primary product is the "i-Bean". It combines analog and digital interfaces, a radio transceiver, and a microcontroller for sensor signal processing, control and networking. Millennial Net uses a portfolio of radio technologies including ultra-low-power narrowband solutions as well as IEEE 802.15.4 wireless personal area network (WPAN) components.
Milennial Net was created in year 2000, and traces its roots to MIT.

Sensicast Systems (ref [web11]) develops and markets low-power, battery-operable, wireless sensor network solutions for the commercial, industrial, agricultural and security industries. Star product is H900 Wireless SensorNet System. It is based on 900 MHz RF band. Sensicast Systems, Inc. was created in 2002.

One can notice that most of the companies were created in the last four years, and are located in USA. They are working together with world class universities. Some products are based on Bluetooth technology, others on 900 MHz or 2.4 GHz RF bands.
Open products (OEM market) are used in network and communication research programs. OEM products usually are based on non Bluetooth technology. Research groups working on this product class claim that the Bluetooth protocol is too complex and too slow during the connecting process. Therefore they choose RF technologies which could provide an easier access.

Products really designed to be a sensor device are end products. Some of them are based on Bluetooth technology. Some companies claim that Bluetooth technology makes products easy to integrate, because of wide use of this technology. "Any PDA or Laptop allows connecting you".
Bluetooth based products do not really fit the specification described in the "target applications" section: their dimensions are generally too wide. They are designed to be used with a particular sensor, not with the one you would like to use.

At the end, we get the following conclusions:

1. This is the beginning of the wireless sensor history.
2. It makes sense to design a home-made wireless sensor.
3. Among wireless technology used in projects or products reviewed, Bluetooth technology is a good candidate in our context. Bluetooth technology is available on existing computer via dongles, and it is accepted by a lot of industrial companies. Debug phase and experiment integration would benefit.

Two documents help considering Bluetooth technology as a good candidate for a wireless sensor device.
The first document (ref [6]) described how a Bluetooth based module was built. The second document (ref [7]) presents an interesting overview of Bluetooth based sensor architecture.

2.5 Wireless sensor device architecture

We decide to specify the sensor device choosing off-the-shelf electronic modules when possible. Of course, sensor compactness is not optimum, nor power consumption, but time development will be reduced significantly.

The project goal is clearly not developing a 2mm x 2mm x 2mm sensor device as Berkeley final goal within "smart durst" research project. We do not have access to MEMS technology as developer, and people involved in this project are not so many. The project goal is developing a hand-size (50mm x 50mm x 50mm) sensor device.

This wireless sensor device consists on a box containing (1) a sensor based board, (2) a micro controller card, (3) a wireless communication card, (4) and a power supply stage.

The following figure (ref [Figure 1]) presents a device architecture based of functional blocks. Most of the blocks are also materialized with a electronic module.

- **Power supply (energy)**: this part really depends on other modules. During development phase, an external power supply has to be provided first. An exhaustive calculation for power consumption has to be done. Then, a technology shall be chosen.
  Anyway, about technology, Li-ion battery is optimum in terms of power versus weight. Primary battery might be considered also.

The following sections give some details about the technologies listed above. For each of them we list what has to be demonstrated with a prototype (second project phase).

2.6 Sensor

2.6.1 What kind of sensors?

As described in section about target applications, sensors of interest are:

- Accelerometers, which are used as inclination sensor too. Accelerometer sensor is a member of absolute family sensor. It deals with physic laws like gravity field or earth magnetic field. This kind of sensor can be easily integrated into the wireless sensor box. Therefore packaging and sterilizing are much easier.
- Force sensor which are usually based on piezo-electric technology. This kind of sensor has to be attached to some hardware (structure where you need measuring). Packaging is a big issue. It can not be generic, because it depends on structure shape. Also it is much harder to sterilize because of separated pieces.

This kind of sensors has the following common features:

- Frequency is between 10 and 1000 Hertz
- Signal dynamic is between 8 and 16 bits

To be consistent with, we should choose a 16 bits micro controller and a communication link throughput faster than 16 kilo bits per second. In fact most of 8 bits micro controller deal with 16 bits data with no really drop out performance.
We can notice that Bluetooth bandwidth is over 16 kilo bits/sec. In fact throughput is between 128 kilo bits/sec and 1000 kilo bits/sec, depending on Bluetooth configuration.

Other sensors like pressure, temperature or humidity have the same features, but they are not considered for the moment.

Sensors like CMOS video camera are much more complex. They need much higher resources. Micro controllers involved in this class of sensor are usually 32 bits. The communication channel bandwidth required is over one mega byte per second.

MEMS sensors are an emergent technology. They have low power consumption, and are very small. Their precision is now relevant for most applications.

How to define MEMS technology? (Extract from www.memsnet.org (ref [web06]))

Micro-Electro-Mechanical Systems (MEMS) is the integration of mechanical elements, sensors, actuators, and electronics on a common silicon substrate through micro fabrication technology. While the electronics are fabricated using integrated circuit (IC) process sequences (e.g., CMOS, Bipolar, or BICMOS processes), the micromechanical components are fabricated using compatible "micromachining" processes that selectively etch away parts of the silicon wafer or add new structural layers to form the mechanical and electromechanical devices.

2.6.2 Extensions to no-sensor devices

The general architecture shows a communication module plugged with a micro controller module. We may consider using theses bricks to make another king of device, for instance a remote controller device.
Powering a low power laser beam is an example. This kind of device is controlled by a simple digital output. This makes it easy to integrate.

We report in the next table (ref [Table 2]) modules which fit the specification. MICROCHIP PIC micro controller family is a very interesting candidate for this kind of application. But we did not find any integrated module with mandatory dimensions.

Company	**Model**	**Micro controller**	**Size (mm)**	**Power supply**	**IO**	**Software Dev Tool**	**Cost (euros)**
PHYTEC (ref [web07])	phyCORE-ARM7	Atmel AT91M55800A	60x53	3.3 Volts, 110 mA (typ)	AD:8x11bit Ethernet DA:2x10bit	C compiler (www.keil.com)	295
PHYTEC (ref [web07])	phyCORE-ST10F168	ST10	60x53	5 Volts, 110mA (typ)	AD:10bit	C compiler (www.keil.com)	165
RENESAS (ref [web09])	Module AE3664	H8/3664F	40x27	5 Volts, 50mA (typ)	AD:8x10bit	GNU C compiler www.kpit.com/	60

Table 2: micro controller modules

2.7.3 To be demonstrated with a prototype

- The analog inputs dynamic range has to be consistent with sensor dynamic range.
- The power consumption must be really low. For instance, power management must be easy to implement.
- The flash memory must be large enough for the developed program. This point is crucial, but the problem is that we are not able to know in advance what will be the final compiled code size. For instance concerning the Bluetooth module associated code, its size depends on what kind of Bluetooth functionalities we need. A full control of the Bluetooth stack means hundreds of code lines. But a minimum configuration is done by only five code lines.

2.8 Bluetooth technology

Note: a special award for the http://www.palowireless.com/infotooth/tutorial website (ref [web30]). This is clearly the best website we found about Bluetooth technology.

The following websites (ref [web15], [web16], [web18], [web28] and [web29]) have been also read for this part.

2.8.1 Technology Overview

Bluetooth is of course not the only wireless technology available. Other technologies like WLAN, HYPERLAN, DECT exist. The main features of the Bluetooth technology are: short range RF (<10 meters for 1 milli-Watt version), low cost, up to 8 connected nodes together (*piconet*). It fits projects working on local sensor devices networks.

A large majority of the project reviewed uses either Bluetooth technology or wireless technologies based on the same RF specification (2.4 GHz, spread spectrum, frequency hopping).

Bluetooth™ is the codename for a technology specification for low-cost, short-range radio links between mobile PCs, mobile phones and other portable devices, and connectivity to the Internet.
Unlike many other wireless standards, the Bluetooth wireless specification includes both link layer and application layer. Radios that comply with the Bluetooth wireless specification operate in the unlicensed, 2.4 GHz radio spectrum ensuring communication compatibility worldwide. These radios use a spread spectrum, frequency hopping, full-duplex signal at up to 1600 hops/sec. The signal hops among 79 frequencies at 1 MHz intervals to give a high degree of interference immunity. Up to seven simultaneous connections can be established and maintained. The Bluetooth specification contains the information necessary to ensure that diverse devices supporting the Bluetooth wireless technology can communicate with each other worldwide. The document is divided into two sections: a Core Specification (Volume I) and Profile Definitions (Volume II).

Note about power classes: Each device is classified into 3 power classes, Power Class 1, 2 & 3.

- Power Class 1: is designed for long range (~100m) devices, with a max output power of 20 dBm,

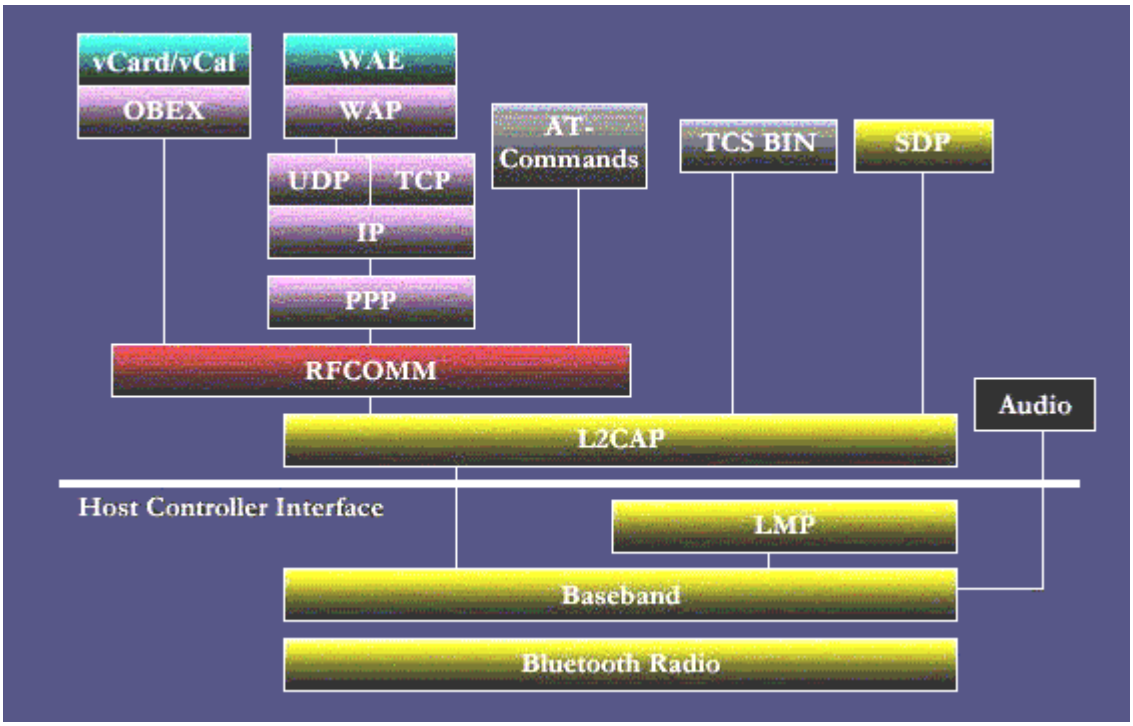


Figure 2: Bluetooth specification protocol stack

1. **Radio** The **Radio** layer defines the requirements for a Bluetooth transceiver operating in the 2.4 GHz ISM band.
2. **Baseband** The **Baseband** layer describes the specification of the Bluetooth Link Controller (LC) which carries out the baseband protocols and other low-level link routines.
3. **LMP** The **Link Manager Protocol (LMP)** is used by the Link Managers (on either side) for link set-up and control.
4. **HCI** The **Host Controller Interface (HCI)** provides a command interface to the Baseband Link Controller and Link Manager, and access to hardware status and control registers.
5. **L2CAP Logical Link Control and Adaptation Protocol (L2CAP)** supports higher level protocol multiplexing, packet segmentation and reassembly, and the conveying of quality of service information.
6. **RFCOMM** The **RFCOMM** protocol provides emulation of serial ports over the L2CAP protocol. The protocol is based on the ETSI standard TS 07.10.
7. **SDP** The **Service Discovery Protocol (SDP)** provides a means for applications to discover which services are provided by or available through a Bluetooth device. It also allows applications to determine the characteristics of those available services.

First profiles defined are:

1. GAP, generic access profile
2. SDAP, service discovery application profile
3. CTP, cordless telephony profile
4. IP, intercom profile
5. SPP, serial port profile
6. HP, headset profile
7. DNP, dial-up networking profile
8. FP, fax profile
9. LAP, LAN access profile
10. GOEP, generic object exchange profile
11. OPP, object push profile
12. FTP, file transfer profile
13. SP, synchronization profile

The document (ref [10]) gives an interesting overview of battery technology. The following websites (ref [web34], [web35], [web36] and [web37]) describe existing products.

We consider using a lab power supply for developing the test bed. In a second step, primary battery will be used in our first experiment. But the specification will take into account secondary battery as power supply. In fact we do not consider embedding a battery loader. So the design does not depend on technology used. Both primary and secondary battery are considered as voltage generator.
The challenge for the sensor is to have the capability for rising up an alert when energy is low. Then the strategy is different according to the battery technology. Li-ion battery output voltage is almost linear with respect to discharge. We do not know how to deal with the primary battery technology.

2.9.2 Power saved strategies

A major challenge for a sensor device is being awake as many hours as possible. Power saved strategies might be activated when possible for extending autonomous period.

First power saved strategy consists on activating module power saved. Most of micro controller chips do have this feature. Bluetooth modules seems having this feature too. But, because of module complexity, it is not clear what kind of power saved feature may be activated. It might also depend on Bluetooth configuration. This has to be investigated.
Sensor modules usually are “non intelligent” chip with no power saved feature.

Then, an action to switch off modules may be considered. The Micro controller might cut power of unused modules.

At the end, we shall consider the capability to switch off the whole system. This implies: (1) to get an external action for switching on the system, for instance a push button, (2) the capability to switch off the module by using a button or by using a programmed watchdog process.
This strategy step is important for applications meaning. The wireless sensor device would not be an autonomous sensor device anymore. It is clearly a limitation for some applications. It has to be considered as last chance for saving energy.

2.9.3 To be demonstrated with a prototype

- An exhaustive power consumption calculation must be done. Depending on the battery technology used, we will estimate how long a wireless sensor device can be powered (life cycle). Different power saved strategies might be tested.
- The capability for avoiding power supply failure when the battery charge is empty.
- Batteries are usually packaged in a metallic box. It could disturb wireless transmission. This has to be evaluated.

2.10 Packaging

The packaging is a big issue, but it might be finalized only when all internal parts will be defined.
The general requirements for packaging are:

- Hand-size container: box sizes must be less than 50mm x 50mm x 50mm
- Sterilizable
- Radio Frequency free way compatible. It implies avoiding metallic parts.

Most of plastic boxes are suitable.

2.11 Conclusion (overview part)

We present target applications where wireless sensor could improve dramatically the process. In other words we explain why wireless sensors were so interesting.

3 Part two: proof-of-concept prototype

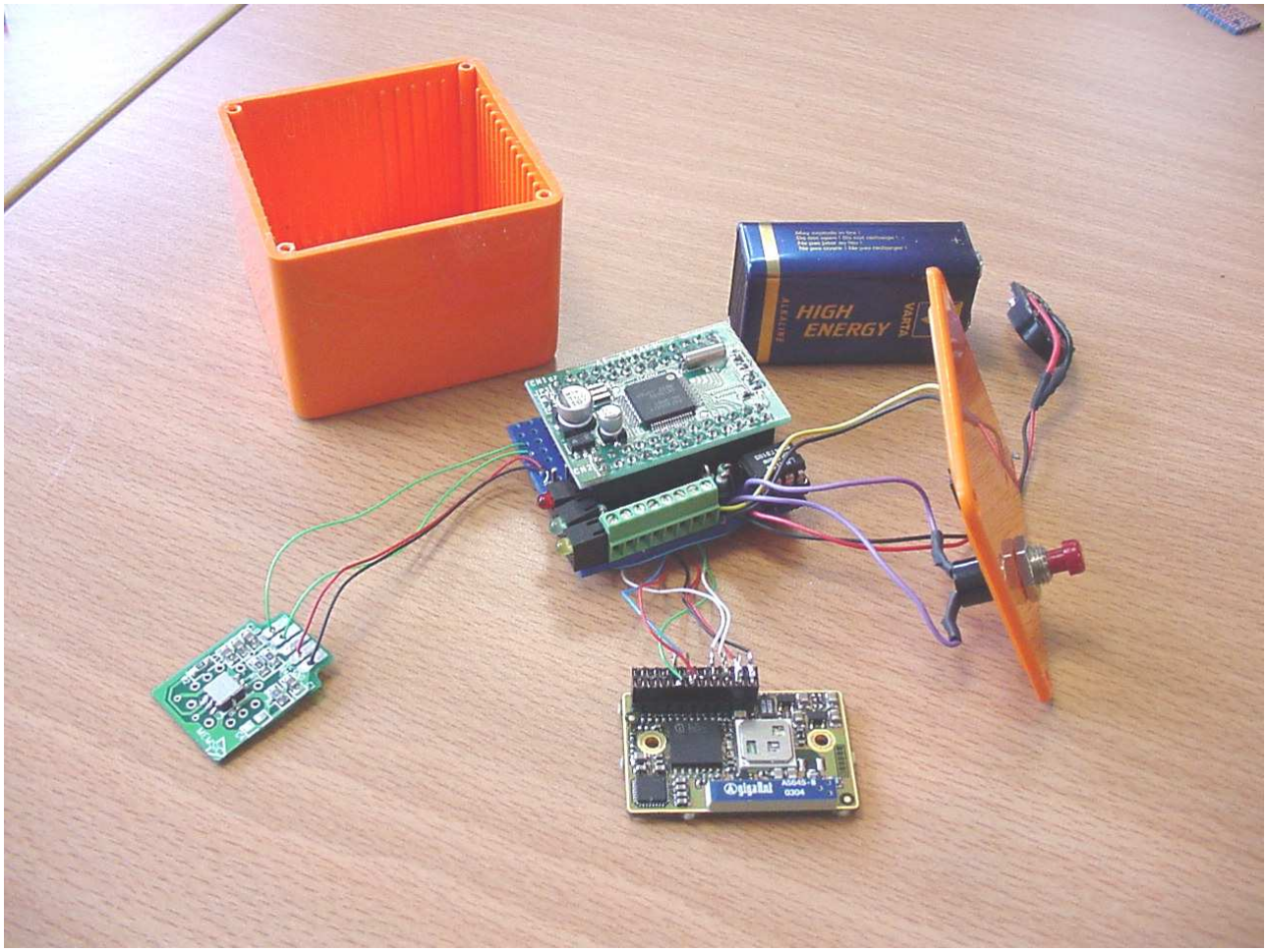


Figure 3: the prototype

3.1 Introduction

The objective for the test bed experiment consists on validating wireless sensor concept, as defined in the first part (overview). Roughly, it consists on packaging inside a small box, a sensor, in order to evaluate its communication performance over a wireless channel, and its power consumption.

In practice, test bed experiment consists on a remote dual axis inclination sensor (ref [Figure 3]).

This document is composed of following parts:

- First it gives how the sensor device works.
- Then the off-the-shelf modules used in test bed are detailed.
- It reports also the interface module specification. This module has been developed for inter-connecting the off-the-shelf modules.
- The embedded software and the development environment are described.
- Energy and packaging specifications are presented.
- At the end, the results concerning power consumption and communication experiments are reported.

This document may be considered as:

- An electric and mechanical manual for building a sensor device.

General comments about monitor are:

- <error> may replace <ok> or <local time> when error occurs on micro controller module program.
- During the analog scan stream mode ('f', 'F'), the timer frequency is changed from 4 Hertz to 122 Hertz. Therefore checking local time before-during-after the analog scan stream mode may induce time discontinuities.
- Except for the stream mode, the communication is initiated by host computer. So a very simple protocol based on "send a command, wait for answer" may be used. The stream mode makes the host communication state machine a bit harder. Multi-threaded processes might be programmed for avoiding conflicting returned messages.

When the watchdog is enabled (default mode), the module turns off its power if no command is received during a programmed period.
In details, a timer interrupt handler decrements a counter. When an incoming command occurs; the counter is updated with a predefined value. When the counter reaches zero, the program turns off a digital output, thus cutting the power supply.
Remark: The power supply will be turned off even if a client is connected. Criteria used deals only with incoming commands.

3.3 Off-the-shelf devices description

3.3.1 Sensor module: MEMSIC mxr2999m

The MXR2999 (ref [Figure 4] and [11]) is a low cost, dual axis accelerometer fabricated on a standard, submicron CMOS process. The MXR2999 measures acceleration with a full-scale range of ±1 g and a sensitivity of 1000mV/g at 25°C. It can measure both dynamic acceleration (e.g. vibration) and static acceleration (e.g. gravity). The MXR2999 design is based on heat convection and requires no solid proof mass.

The MXR2999 provides a ratio metric analog output that is proportional to 50% of the supply voltage at zero g acceleration. The typical noise floor is 0.2 mg/$\sqrt{Hz}$ allowing signals below 1 milli-g to be resolved at 1 Hz bandwidth.

The inclination measurements (X and Y) are given for +-60 degrees range and at 30 Hertz frequency update.

The MXEB-002_L (ref [Figure 4] and [12]) is a 26 mm x 19 mm board. It is developed by MEMSIC to make easier the integration of the MXR2999 sensor chip. The MXEB-002_L pin out is reported in table (ref [Table 5]).

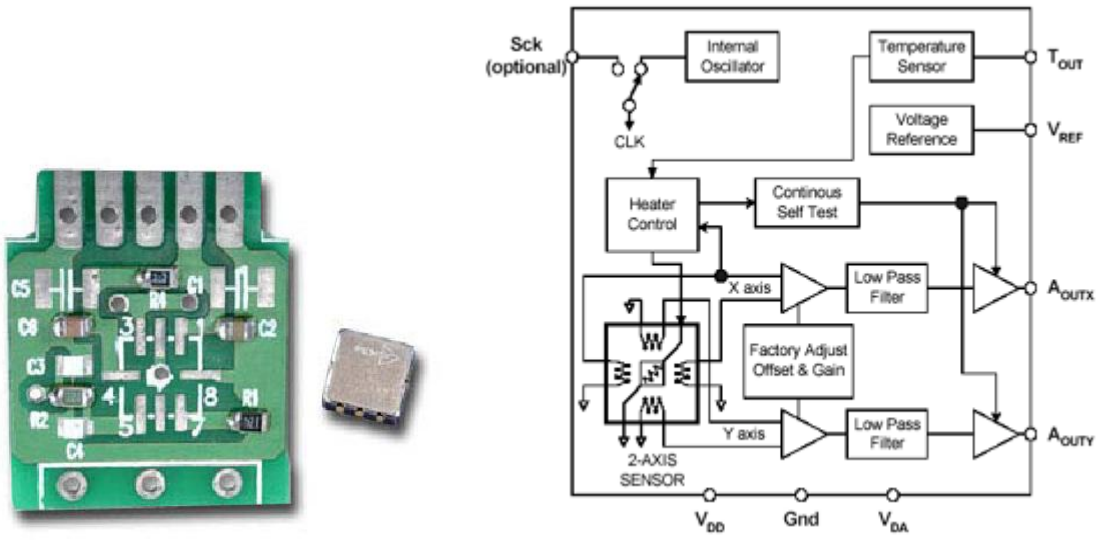


Figure 4: MXEB-002 L Basic Board (with MEMSIC IC on the side) &
MXR2999GL/ML Functional Block Diagram

• 5V operation • 16MHz H8/300H 16-bit CPU core • 32 kHz sub-system oscillator • Wide range of low power operating modes • 8 - 32kBytes ROM, 512 - 1kBytes SRAM • 32kBytes single supply Flash, 2kByte SRAM (Flash version only) • 1 x 16-bit timer with 4 capture/compare	• 2 x 8-bit timer • 1 x watchdog timer • 1 x USART • 1 x I²C • 8 channels 10-bit A/D • 37 I/O pins (8 x 20mA ports) • 11 External Interrupts • On-chip debug port

Table 6: H8/3664F main features

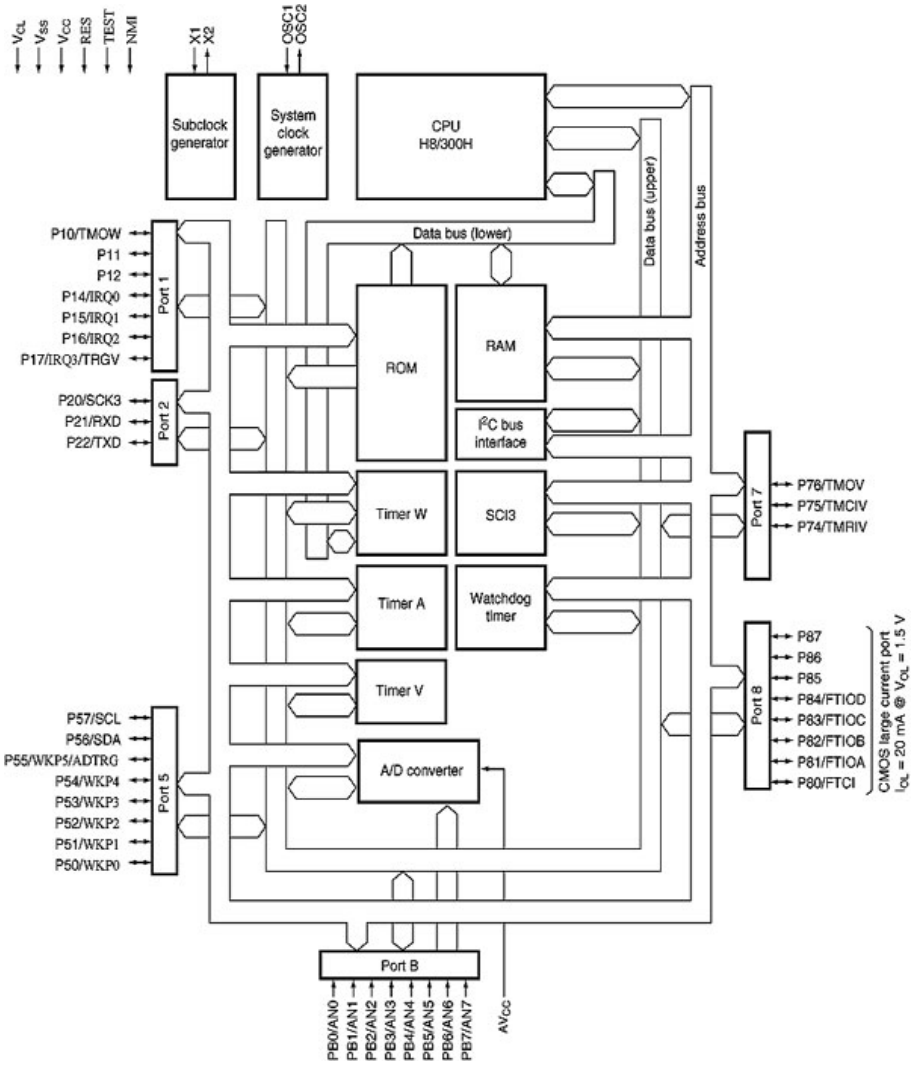


Figure 5: H8/3664F general architecture

The **AE-3664FP** board (ref [Figure 6]) is a 40mm x 27mm module, based on H8/3664F Renesas micro controller.

Please refer to Renesas "Hitachi Single-Chip Microcomputer H8/3664F Hardware Manual" (ref [14]) for a complete hardware description.

Software development tools

HEW 2.2 (High-performance Embedded Workshop) from Renesas is a flexible code development and debugging environment for applications target at Renesas microcontrollers. It provides an up-to-date "look and feel" with all of the features you would expect from a modern development environment.

The Flash Development Toolkit (FDT 2.2) is used for flashing programs on the micro controller. HEW GUI allows FDT access.

KPIT GNU compiler/linker: KPIT Cummins proposes an installable GNUH8 cross-compiler tool chain based on freely available source code from the Free Software Foundation (www.gnu.org) to help those who wish to build embedded system applications for Hitachi's H8 series of targets. GNUH8 is fully integrated inside HEW.

3.3.3 Bluetooth module: ConnectBlue OEMSPA13i

The OEMSPA13i (ref [Figure 7] and [16]) (cB-0701-01 technical module id) is a small size Bluetooth module based on the Infineon (former Ericsson Microelectronics) PBM 990 80 baseband controller and the PBA 313 05 (0dBm) radio.
The PBM 990 80 has on chip SRAM and FLASH stacked in the same package.

The main features are: short range (0 dBm), logic-level UART and RS232, internal antenna, 3.3-6 VDC power supply.

The module state modes are either the "Configuration mode", or the "Data mode".
The "Configuration mode" is used for configuring the Bluetooth parameters. This configuration may be achieved using AT commands (ref [17]) or ECI protocol (ref [16]). A Microsoft Windows wizard based on AT commands allows configuring part of the Bluetooth parameters. Other parameters must be modified using AT Commands or ECI protocols directly.
The "Data mode" is a totally transparent serial channel mode. Each character sent or received goes through the Bluetooth module.

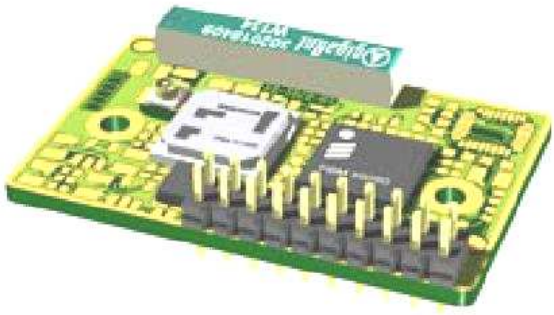


Figure 7: OEMSPA13i ConnectBlue Bluetooth module

All OEMSPA13i IO pins (except RESET) are located on a dual 2 millimeters connector (ref [Table 8]). Hardware RESET is located on J3 connector (pad on PCB).

- Serial configuration is 19200 bauds, 8 bits, 1 bit stop, no flow control
- Device name: “inria ra - ws 001”
- “Always connected” feature disabled

Software layers

AT commands are a set of ASCII commands recognized by the Bluetooth module. An escape sequence sent over serial line to Bluetooth module changes its internal mode. It switches from “data mode” to “AT commands mode”. Then the host configures the Bluetooth module using AT commands. “Data mode” is switched back using another specific AT command.

The ECI protocol (ref [Figure 9]) is delivered as a C library. It runs on the host with an interface towards all the Bluetooth functionality embedded in the Serial Port Adapter. The ECI protocol is a lightweight protocol. It provides access to the profiles GAP, SPP, DUN, and LAN as well as some other miscellaneous functions. It supports true multi-point. ECI protocol is delivered with source code, and so can be compiled on different target.

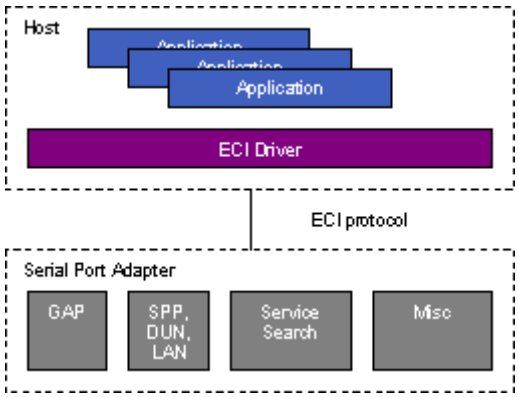


Figure 9: ECI protocol architecture

3.4 Energy

The original idea was to design an “always on” wireless sensor. In other words, the wireless sensor is powered when battery is connected, and stops working when battery is discharged.

3.4.1 Power consumption calculation based on datasheets

The H8/3664F micro controller needs a 5Volts/50mA power supply. A 7805 (5Volts/150mA) regulator is located on this module. Either a regulated 5Volts/50mA or 7-20Volts is used for powering the module. When 7-20Volts power supply is used, the regulated 5Volts may be used as output for supplying other modules.

The OEMSPA13i Bluetooth module needs a 3-6Volts/100mA power supply. Intra module voltage is 3.0 Volts.

The MXR2999 sensor module needs a 5Volts/4mA power supply.

At the end, the power consumption calculation gives 5 Volts / 154 milli-Amperes for the three off-the-shelf modules. The on board 7805 is not strong enough for supplying all modules. For the very first experiment, we decided to add a 5 Volts regulator (7805) on the interface module.
In fact, experiments show us that power consumption never get over 90 milli-Amperes. So we give up using an additional regulator component.

3.4.2 Power saved mode

The micro controller and the Bluetooth module have such a power saved mode. The strategy for getting this mode is presented in the “software” section.

- A mechanical interface for screwing Bluetooth module (two 2.5 millimeters diameter through holes).
- A 7x1 connector for the Bluetooth signals: two pins for the Bluetooth module power supply, three pins connector for the Bluetooth module LEDs output, and two pins for the RS232 communication.
- An interface to the micro controller module: two (2.54 millimeters pad, two rows, 13 columns).
- A power stage: two pins connector for the battery (+ and -), a static relay for the software control of the power, two resistors coming with the static relay.
- A connector for the dual axis sensor: Power, Ground, Xout and Yout signals.
- Four LEDs coming with four resistors. Three of them are dedicated to the Bluetooth module LEDs. The fourth LED is used as a micro controller status display (Diagnostic LED). A jumper allows powering or not the Bluetooth LEDs. This jumper may be installed for debug purpose. Removing the jumper provides energy saving.
- A 74HCT08 (4 AND logic chip) for adapting the 3 Volts CMOS outputs (Bluetooth side) to the 5 Volts CMOS inputs (micro controller side).
- Two extension connectors for the future uses: analog inputs, digital inputs and outputs, power and ground.

3.5.2 Interface module design

The next figure (ref [Figure 10]) represents the interface module schematic. The extensions mentioned in the above section are not represented. Pin out is given except for VCC and GND signals.

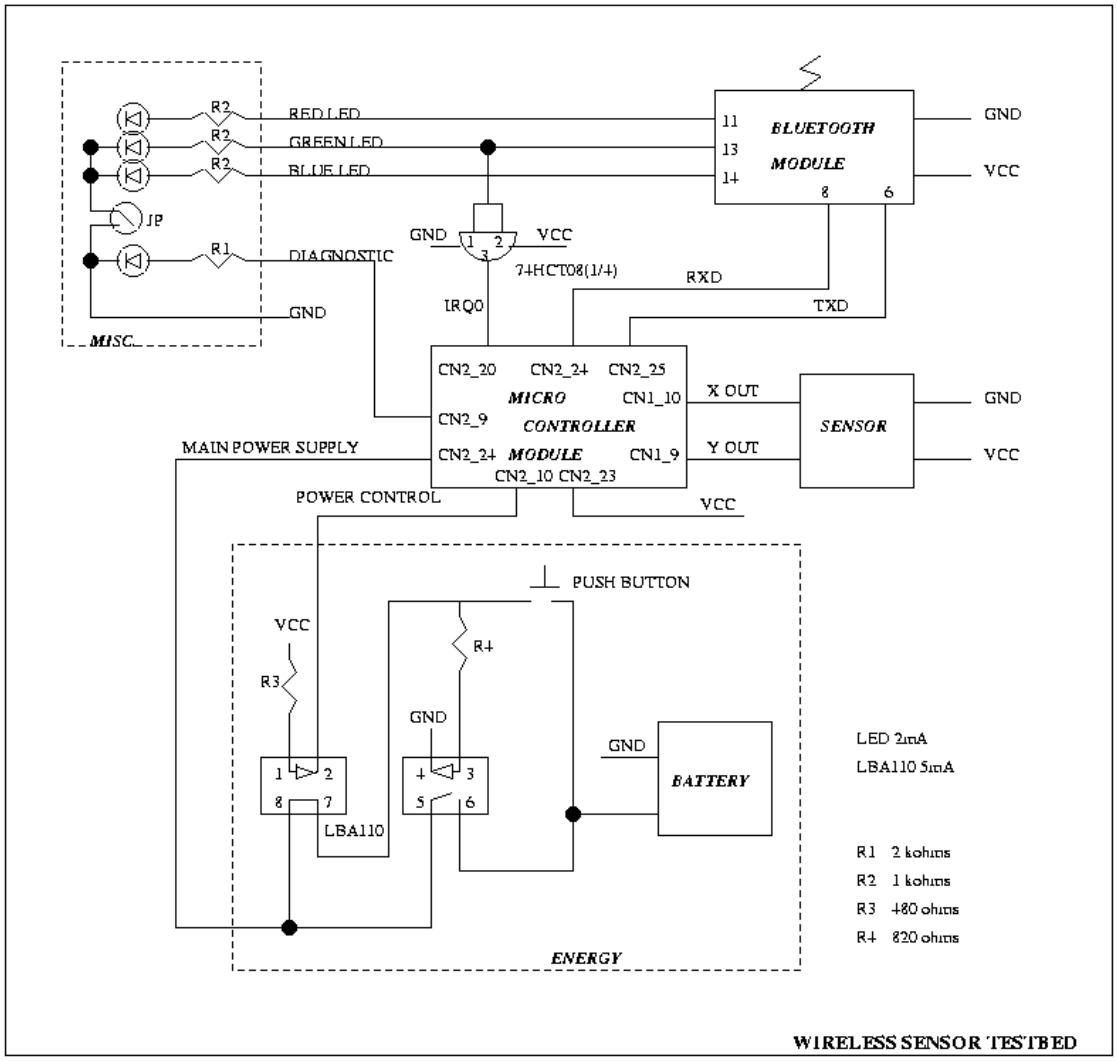


Figure 10: schematic layout

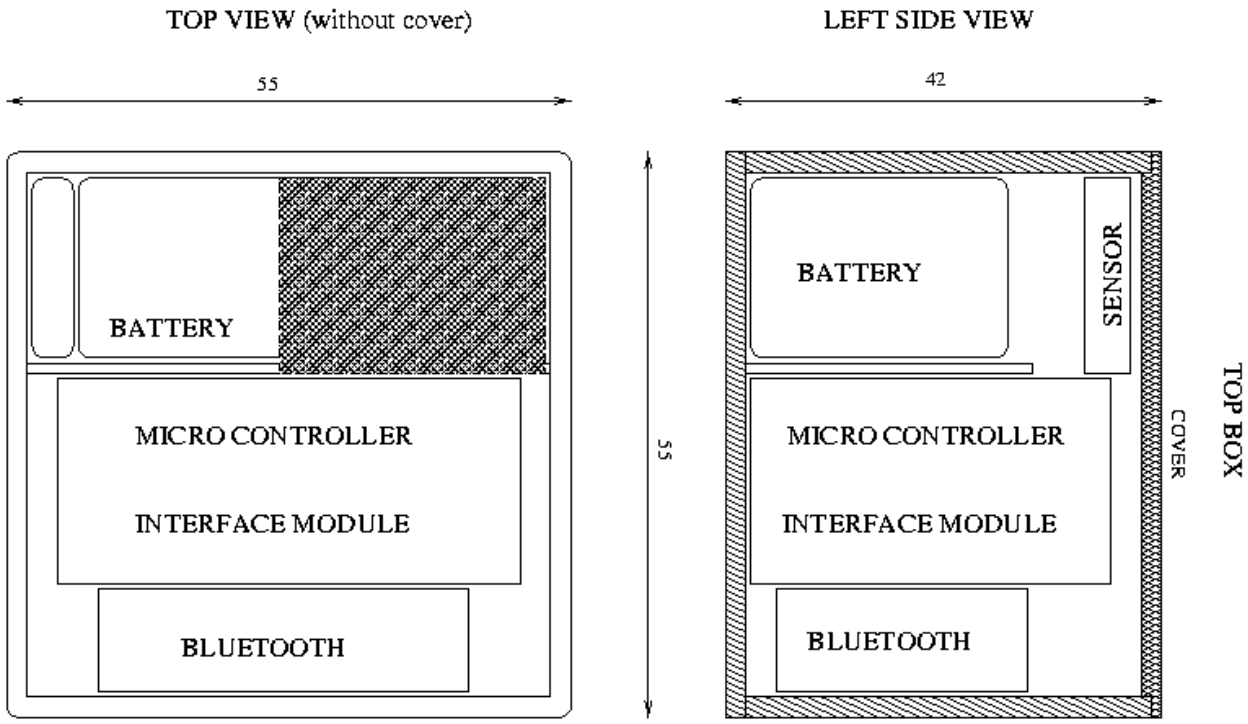


Figure 12: module packaging

The modules mounting instructions are given in the document annex (ref [3]).

The sterilization process is quite easy with a hermetic plastic box. Only the push button and the diagnostic led could induce problems. Fortunately, theses components are commonly used in medical applications. For the prototype we did not take into account sterilization process, and we choose standard components.

3.7 Embedded programs

Before talking about the final version of embedded program, let describe briefly the test procedures we performed. These programs implement incrementally features.

Hello World – learning development tools

From the hardware point of view, the micro controller is plugged on the development board (H8 Tiny Board TERA2). A RS232 cable connects development board to a PC running Microsoft Windows XP.
The developed program uses the H8/3664F RS232 hardware resource. The complete software development toolkit was validated: compiler, linker, flash toolkit. On the PC side, we use a HyperTerminal application. We validate also the serial cable.

Timer & Analog reading – using micro controller features

From the hardware point of view, the micro controller is plugged on the development board (H8 Tiny Board TERA2). A RS232 cable connects development board to a PC running Microsoft Windows XP. MEMSIC dual axis inclination sensor is wired on the development board (H8 Tiny Board TERA2) using the mirrored micro controller connectors.
With this program, we master two of the major micro controller features: timer and analog converter.

Bluetooth module test – configuring and testing module

- A new electronic signal has been connected between the Bluetooth module GREEN LED pin and the micro controller IRQ0 pin. In fact this signal goes through a 74HCT08 for electric compatibility. The Bluetooth outputs are 3 Volts CMOS, the micro controller inputs are 5 Volts CMOS.
- The GREEN LED signal goes to active mode when a connection occurs. It leaves active mode only when the client is disconnected. Note that the BLUE LED behavior is almost the same except that the BLUE LED signal flicks during communication.
- The micro controller goes to standby mode when the function "sleep" is called. It wakes up for different reasons. One of them is an incoming interrupt.
- An interrupt handler attached to an IRQ0 input pin sets (1) or resets (0) a flip-flop semaphore depending on the interrupt edge is rising or falling. The main C function is based on an infinite loop, waiting for an incoming command over RS232. A test on a semaphore state has been added into this loop. The followings actions happen within a session:
    1. At power on, the micro controller does its initialization, then calls the "sleep" function and enter "standby" mode.
    2. When a client connects, the Bluetooth module wakes up and enters the connecting mode. Then GREEN LED signal output goes to high state and the micro controller interrupt rises up due to a rising edge input. This makes the micro controller leaving "standby" mode. And during the next loop program, the "sleep" function will be bypassed, thanks to the semaphore state.
    3. When the client disconnects, the Bluetooth module switches GREEN LED signal output to low state, then it enters "stop" mode. Micro controller interrupt rises up due to a falling edge input. The semaphore changes its state, and during the next loop program, the "sleep" function will be called. Thus makes entering the micro controller to "standby" mode.

Final program

The final program consists on a C program with the following parts:

- Timer interrupt handler: it manages local time, watchdog, and synchronous analog scans requests.
- External interrupt handler: It is connected to the Bluetooth module for going or leaving the power saved mode (micro controller "standby" mode).
- UART interrupt handler: It receives incoming characters.
- Main program: First of all, it initializes the internal registers. Then an infinite loop waits for an incoming character (a command). The command is treated, and an answer is sent within the loop.

The C code is about 1000 lines. The file produced is 7 kilo bytes Motorola S-Format type, and 2.75 kilo bytes on the micro controller flash.

3.8 Program running on host

A test program running on Microsoft Windows station provides an access to all features available on the module.

This test program is developed using Microsoft Visual C environment on Microsoft Windows XP. Because this is a Console type program, it can be ported easily on other operating systems like Linux.

The program accesses a Bluetooth dongle connected over an USB port. CreateFile/CloseHandle/ReadFile/WriteFile basic Microsoft Windows functions are used for the USB communication. The communication channel number (COMx) is given by a Bluetooth utility program. This utility program provides several Bluetooth features (discovery mode, connecting device) which are not implemented inside the program. The utility program provides also Microsoft Windows systems features like mounting Bluetooth connection channel on a Microsoft Windows COM channel.

The program consists on two sub-programs. An argument given when program is called activates one of the sub-programs.

The first sub-program is used for measuring communication time. It is described within the "embedded programs" section.

The second sub-program consists on a menu providing functionalities described within the "How does the sensor device work?" section.

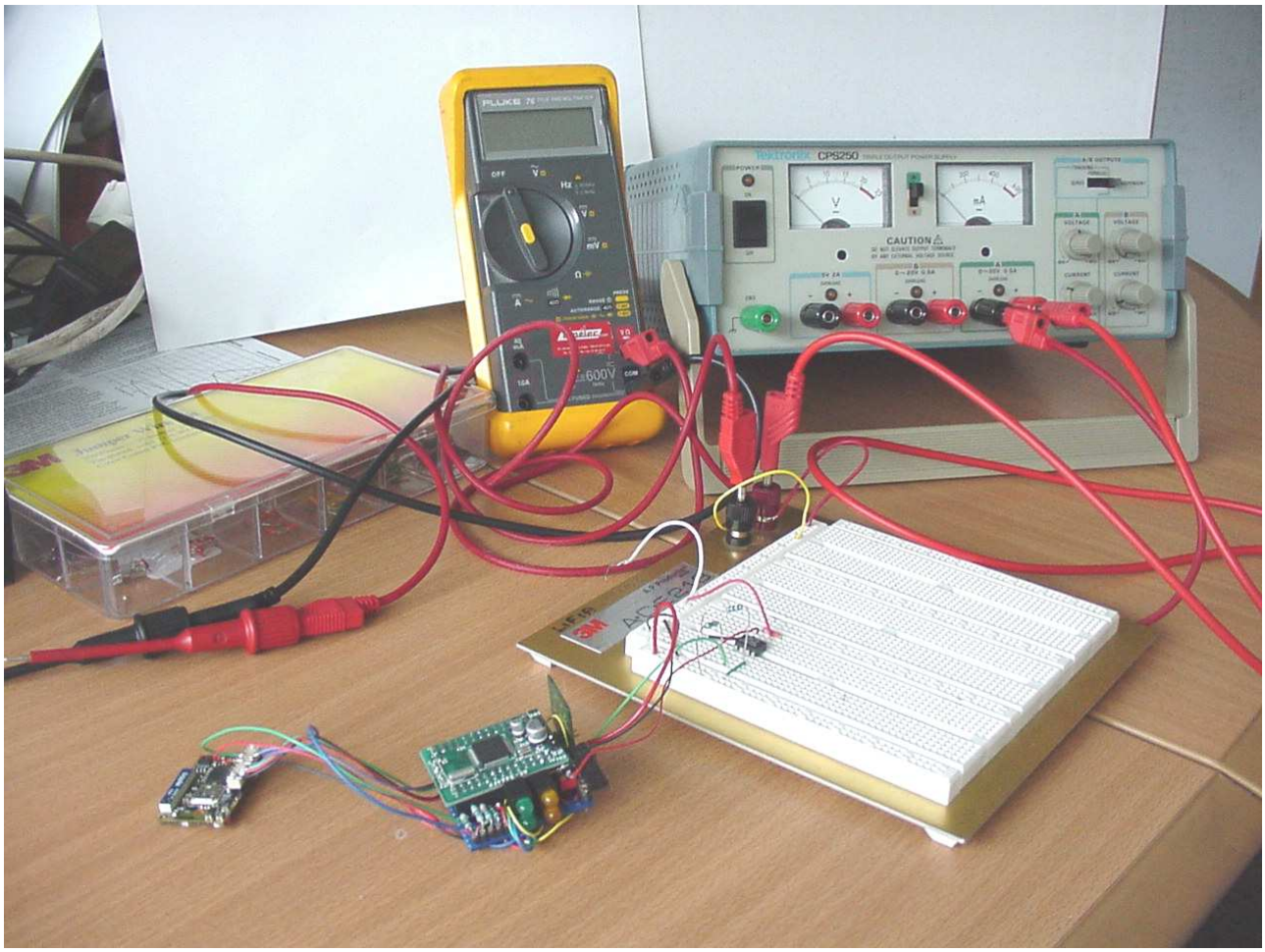


Figure 13: experimental table

The first objective was to contain the whole system in a “small” box. We succeed with a 55mm x 55mm x 42mm box. The total weight is 120 grams. The weight distribution is reported in the next table (ref [Table 10]).

Name	**Weight (gram)**
Battery	48
Micro controller module	9
Bluetooth module	5
Box	35
Sensor module	3
Interface module	15
Misc. (wires, insulator,…)	5
Total	120

Table 10: weights

The second objective concerned the power consumption. The following table (ref [Table 11]) reports the power supply measurements performed in the different device configurations. The lines match the incrementally building steps. First the micro controller is alone, then the sensor is added, then the Bluetooth module is connected. At the end, the whole system including the LEDs and the “auto power off” feature is tested. At the same time we tested the power saved modes. It is reported as “Idle” or “Stand by” for the micro controller, and “Not Connected”, “Connected, or “Stop Mode” for the Bluetooth module.

5. The PC receives the analog scan result.

The resulting bandwidth is about 30 Hertz.

The asynchronous protocol is as follows:

1. The PC requests for an analog scans.
2. The micro controller receives the command.
3. The micro controller performs analog scans and sends back the result. Scans are synchronized using the micro controller timer, thus allowing different output frequencies. At the same time, the micro controller carries on receiving incoming commands.
4. The PC receives the incoming analog scan results.
5. When the PC decides to stop, it requests the micro controller for stopping sending analog scan results.

The second protocol uses more micro controller and PC resources, because it needs several processes running in concurrent mode. It has been successfully implemented on the micro controller and the PC.
And, it should be preferred for applications when the bandwidth is critical.

The next two figures (ref [Figure 14]) summarize both protocol timings.

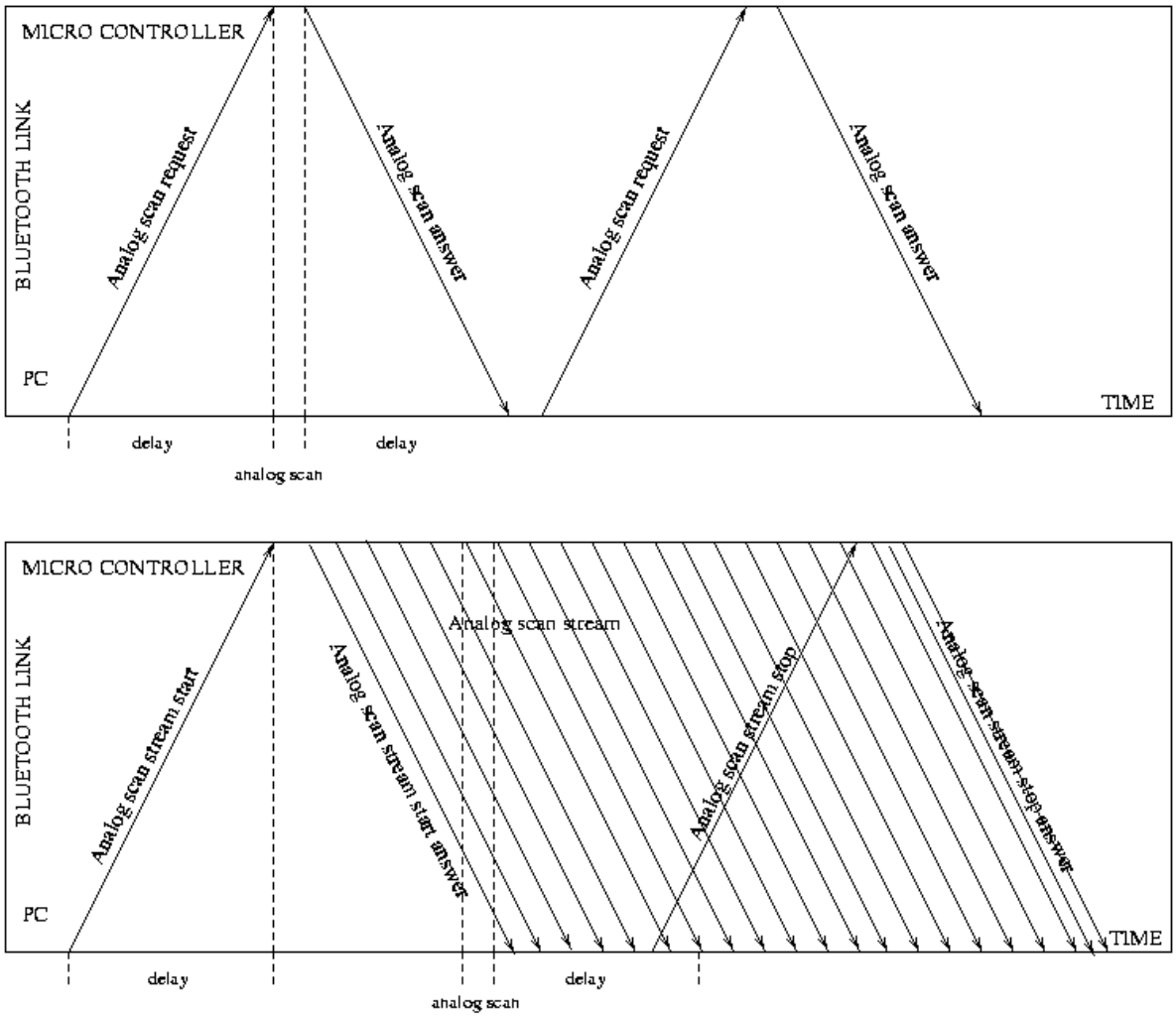


Figure 14: communication protocols (top: synchronous, bottom: asynchronous)

3.10 Prototype limitations

The sensor device main limitation concerns its accuracy.

4 Conclusion

We successfully build a wireless sensor. Main features are reported in the document annex (ref [annex 6]).

Our contributions are as follows: (1) we have reviewed existing research or commercial wireless sensor devices; (2) a prototype has been built and successfully tested.

Therefore, research groups have now the opportunity to use the wireless sensor technology within their applications. The tests performed are relevant and the limitations are clearly identified, thus making the integration process clear.

The prototype developed is not a final or a commercial product. The making process was very intuitive. Also components and modules placement is clearly not efficient.

But the final step for getting a usable product is not so high. All modules have been validated. The desired functionalities have been implemented with success. The package must be optimized in order to fit the application. The interface board has to be cleaned up, and a new version might use integrated technology like CMS components.

The wireless sensor cost is about 300 euros. The purchase list is given in the document annex (ref [annex 4]). A 40 euros Bluetooth dongle allows working with the device from any computer. Half of the device price is due to the Bluetooth module. This module includes a useful software stack that provides an easy access to all Bluetooth functionalities. But the cost can be reduced significantly by using cheaper Bluetooth modules and developing the missing software stack. At the end we would get an affordable sensor device.

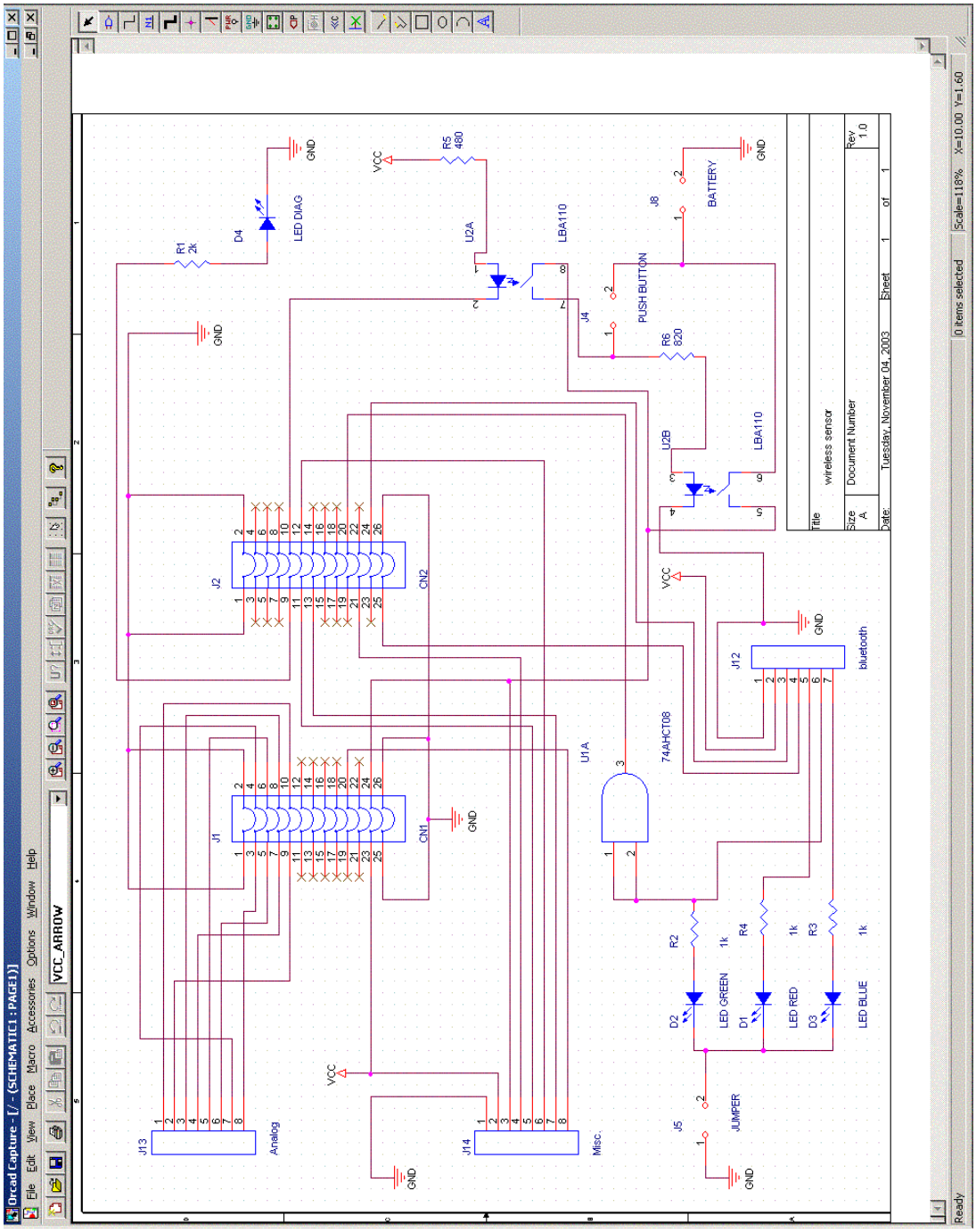
Further works will concern: (1) getting a “more industrial” wireless sensor device version, then (2) including this new sensor class within applications.

During this project we mastered; a micro controller module based on the H8/3664F RENESAS chip, and a Bluetooth module. Research teams working in virtual reality and in robotics are now interested for using these bricks in their applications.

2003
- [16] Serial Port Adapter: 2nd Generation: User Manual 1.3
  ConnectBlue
  2003
- [17] Serial Port Adapter: 2nd Generation: AT Commands
  ConnectBlue
  2003

7 Annexes

7.1 Interface module schematic



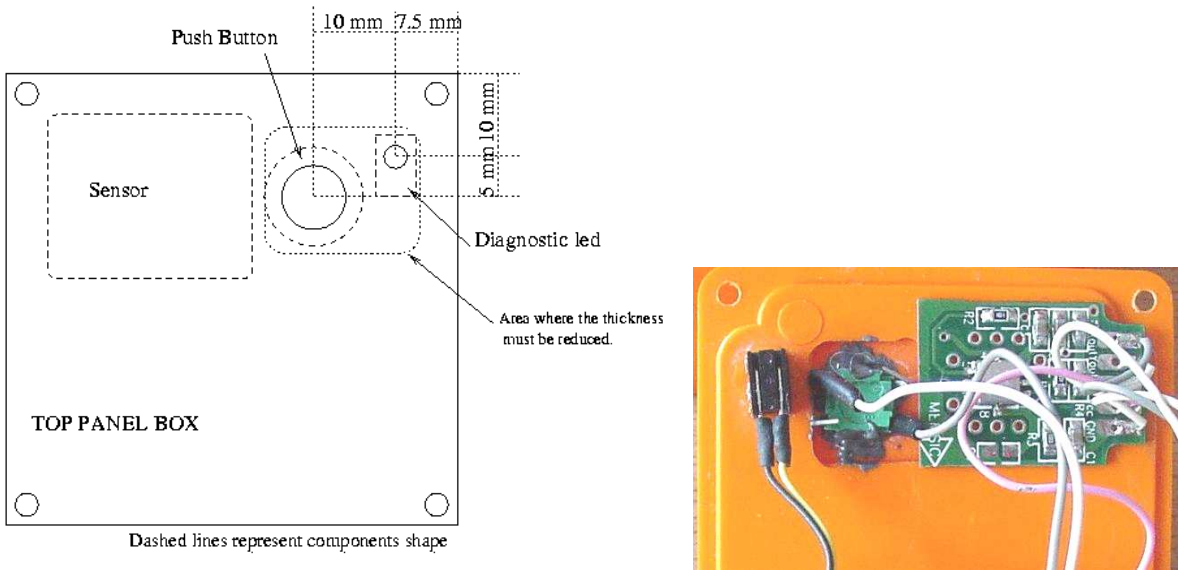
Net	**Connections**
1=GND	BAT(1) LBA(4) LD(2) S(1) BT(1) HCT(7) CN1(1,2,25,26) CN2(1,2,26) JP(2) IO(1)
2=VCC(5Volts)	R3(1) S(2) BT(2) HCT(14) CN1(23) IO(2)
3=BAT+	BAT(2) PB(1) LBA(6)
4	PB(2) R4(1) LBA(7)
5=POWER	LBA(8,5) CN1(24) IO(3)
6	R4(2) LBA(3)
7	LBA(1) R3(2)
8=PWR_CTL	LBA(2) CN2(10)
9=LED_DIAG	CN2(9) R1(1)
10	R1(2) LD(1)
11	JP(1) LR(2) LG(2) LB(2)
12	R2R(2) LR(1)
13	R2G(2) LG(1)
14	R2B(2) LB(1)
15=LED_R	R2R(1) BT(5)
16=LED_G	R2G(1) BT(6) HCT(1,2)
17=LED_B	R2B(1) BT(7)
18=RXD	CN2(24) BT(3) (TXD for the Bluetooth module side)
19=TXD	CN2(25) BT(4) (RXD for the Bluetooth module side)
20=IRQ0	HCT(3) CN2(20)
21=AN0	ANA(1) CN1(10)
22=AN1	ANA(2) CN1(9)
23=AN2	ANA(3) CN1(8)
24=AN3	ANA(4) CN1(7)
25=AN4	ANA(5) CN1(3)
26=AN5	ANA(6) CN1(4)
27=AN6	ANA(7) CN1(5)
28=AN7	ANA(8) CN1(6)
29=IRQ2	IO(4) CN2(21)
30=IO0	IO(5) CN2(11)
31=IO1	IO(6) CN2(12)
32=IO2	IO(7) CN2(13)
33=TMOW	IO(8) CN1(20)

Table: net list

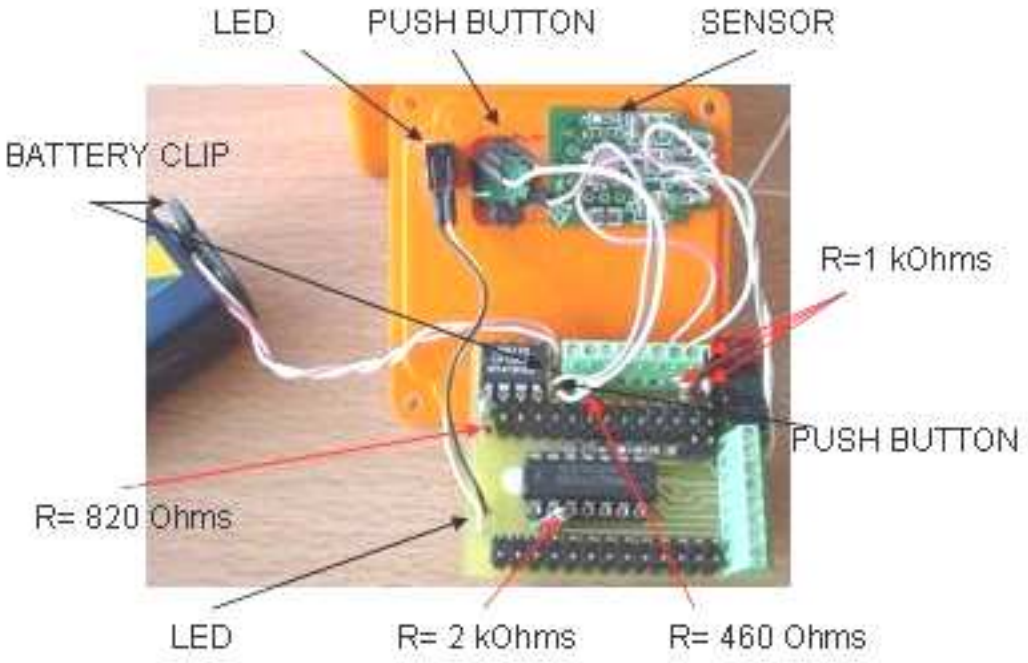
- Screw the Bluetooth module on the interface PCB, then the Bluetooth connector can be plugged on the Bluetooth module (ref [Annex 3 – Figure 3]).
- Screw wires coming from the sensor in the PCB connector. (ref [Annex 3 – Figure 2]).

Final mount: (ref [Annex 3 – Figure 5]).

- Plug battery using the battery clip.
- Insert the battery, then the insulator plate.
- Insert the devices (Bluetooth module + Interface module + Micro controller module).
- Screw the top panel on the box using the four screws.



Annex 3 – Figure 1: top panel (left: outside, right: inside)



Annex 3 – Figure 2: interface module PCB placement

7.4 Component list and cost

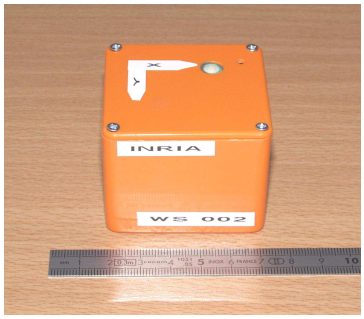
We list in the following table any software and hardware items used for building the wireless sensor.

Name	**Company**	**Reference**	**Distributor**	**Reference**	**Price (euros)**
Micro controller module	Renesas	AE-3664FP	Microtronique (ref [web08])	AE-3664FP	39
H8 tiny board	-		Microtronique		
H8 development board	-		Microtronique		30
Software H8	Renesas	HEW 2.2			0
Flash Toolkit	Renesas	FDT 2.2			0
GNU H8 C	KPIT	wingnuh8v0303coff			0
BlueTooth Module	ConnectBlue	OEMSPA13i	Eurodis (ref [web21])		158
Software ECI API	ConnectBlue	CB BTEDK01	Eurodis		305
Sensor	MEMSIC	mxr2999m	Eurodis		20
Sensor PCB	MEMSIC	MXEB-002 L	Eurodis		
Interface module PCB	INRIA	Inria ws	Teleph	Inria ws	7
Static relay	CLARE	LBA110	Radiospares		8
Logic And	Texas Instrument	745(A)HCT08	Radiospares	221-0883	0.7
LED 2 mm (x4)	Dialight	551-1(123)07	Radiospares	197-126	0.61x3
Clips for battery			Radiospares	172-5996	0.5
Connector I/O (x2)	Phoenix Contact	MPT 0,5/8-2,54	Radiospares	220-4327	2x2
Connector Bluetooth (2 mm pitch)			Radiospares	180-6578	5
Push button					
Cube box			Radiospares	281-6936	3
USB Bluetooth dongle					40

Table: purchase list

The prototype price is about 300 euros.
The only device you need to work with the sensor device is a Bluetooth dongle connected to a PC. A dongle can be purchased in any computer accessories store. Its cost is about 40 euros.

7.6 Wireless Sensor datasheet



The "Wireless sensor" device provides remote sensing in an efficient way. No physical connection is necessary. Any off-the-shelf Bluetooth dongle connected to a PC allows interfacing this device.

Dimensions: 55mm x 55mm x 42mm (cube box made in ABS plastic).

Weight: 120 grams (including 48 grams for battery).

Energy:

- Primary battery type PP3 9 Volts,
- Autonomy better than one month (depending on day-by-day use),
- Auto power off feature.

Technologies used:

- Bluetooth module based on Infineon chips (Power Class II),
- 16-bits micro controller from Renesas (formerly Hitachi); H8/3664F,
- MEMS (Micro Electro Mechanical Systems) dual axis inclination sensor.

Performance:

- +/-60 degrees dual axis inclination measurement,
- Resolution better than 0.5 degree arc,
- Bandwidth better than 100 Hertz,
- Distance between PC and device up to 10 meters (Bluetooth class II).

Software development (C language):

- Micro controller is ready for working. Source codes are available. Renesas software development toolkit is free. Therefore embedded program can be modified very easily for fitting application.
- A Windows Console type program with full device functionalities support is available as source code.

Medical application:

- Sterilizable package,
- Read in the news: a Bluetooth based device received a FDA approval.

A technical report written in English includes a detailed hardware and software description.

Other packaging can be designed easily.

Extensions to other sensors are possible, thanks to the eight analog inputs available on the micro controller.

Extensions to other devices, including LASER or motor remote controllers, are possible, thanks to digital inputs and outputs available on the micro controller.

Copyrights: INRIA – 2003