Department of Electrical
and Computer Engineering
9th semester
T.U. Communication technologies for IoT
Wireless Sensors Network

Laboratory project:
Machine-to-Machine Communications:
Design of PHY and MAC layers for a WSN regarding the application

2023-2024

Gaël LOUBET
Lamoussa SANOGO
Eric ALATA
Daniela DRAGOMIRESCU
first_name.name@{laas; insa-toulouse}.fr

Table of contents:

General context

During this laboratory project, you will study, then design and implement a Machine-to-Machine (M2M) communication using Wireless Sensors Network (WSN) for different applications, each one with its own particularities and its own specifications. The design of the M2M communication has to take into account the specifications for each application, but also the constraints specific to WSN, such as energy efficiency. You will also seek to optimise its functioning.

This project focuses on the design of an energy efficient M2M communication at PHY (Physical) and MAC (Medium Access Control) layers. Student participation and autonomous work will be essential parts of this project.

- Organization:

Number of supervised sessions:	3
Duration of each session:	2h45
Group organisation:	6 students per group, with 2 students working on each layer, from different specialities

- Evaluation criteria:

A self-sufficient report -to be included in the portfolio- for each group containing everything that has been done during the laboratory project is required. Naturally an introduction and a conclusion are expected as well as a critical analysis of your work.

Not justified absences and delays will be considered for the evaluation.

Introduction – Machine-to-Machine Communications

What is Machine-to-Machine communications?

Machine-to-Machine (M2M) communication is to establish the conditions that allow a device to exchange information with a business application *via* communication network, as presented in Fig. 1. So, we often call it M2M which is shortened called for M2(CN2)M: Machine-to-(Communication-Network-to-)Machine.

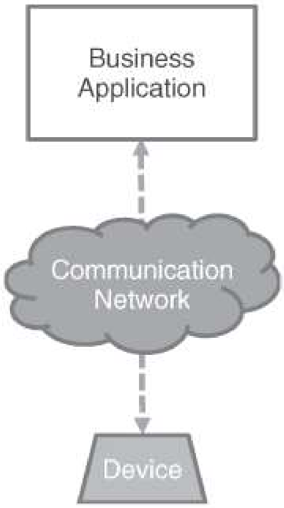


Figure 1: M2(CN2)M communications.

Mediated M2M relationship

M2M involves a group of devices communicate with a single application. Because of the limited capacities, the relationship is mediated by a gateway, as presented in Fig. 2. This M2M area network provides PHY and MAC layers connectivity between different M2M devices in the same M2M area network, then allowing M2M devices to gain access to a public network *via* a router or a gateway.

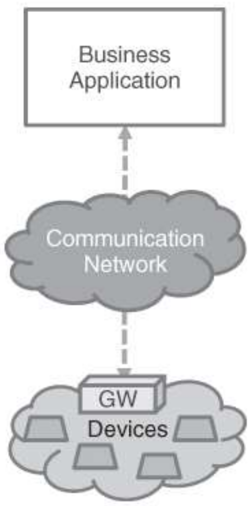


Figure 2: M2(CN2)M communications *via* a gateway.

M2M service layer

A set of architectures and processes will be functional separated. This kind of separated infrastructure will improve the cooperation between each network and protocol. Besides, the deployment of M2M applications could benefit from these building blocks which can accelerate the development, test and deployment life cycles, as presented in Fig. 3.

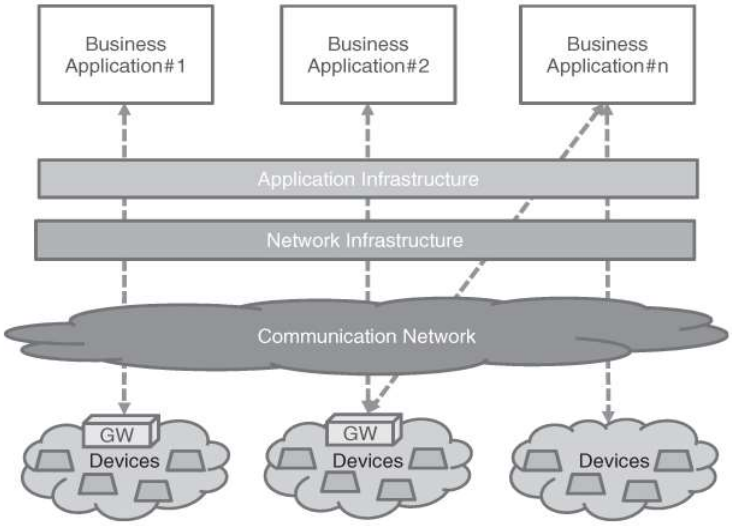


Figure 3: M2M network architecture

M2M system architecture

As presented in Fig. 4, the common M2M system architecture includes:

- M2M device: Device that runs M2M application, using M2M service capabilities and network domain functions with direct connectivity or using a gateway as a network proxy.
- M2M area network: providing PHY and MAC layers connectivity between different M2M devices connected to the same M2M area network or allowing M2M device to gain access to a public network *via* a router or gateway, such as IEEE802.15.X, Zigbee, Bluetooth, PLC or WIFI.
- M2M gateway: It is an equipment that has at least a WAN communication module (e.g. GPRS/UMTS) in addition to one or several communication modules that allow access to the M2M area network (e.g. Zigbee, PLC, etc.). The M2M gateway will provide access network, transport network and sometimes M2M applications.

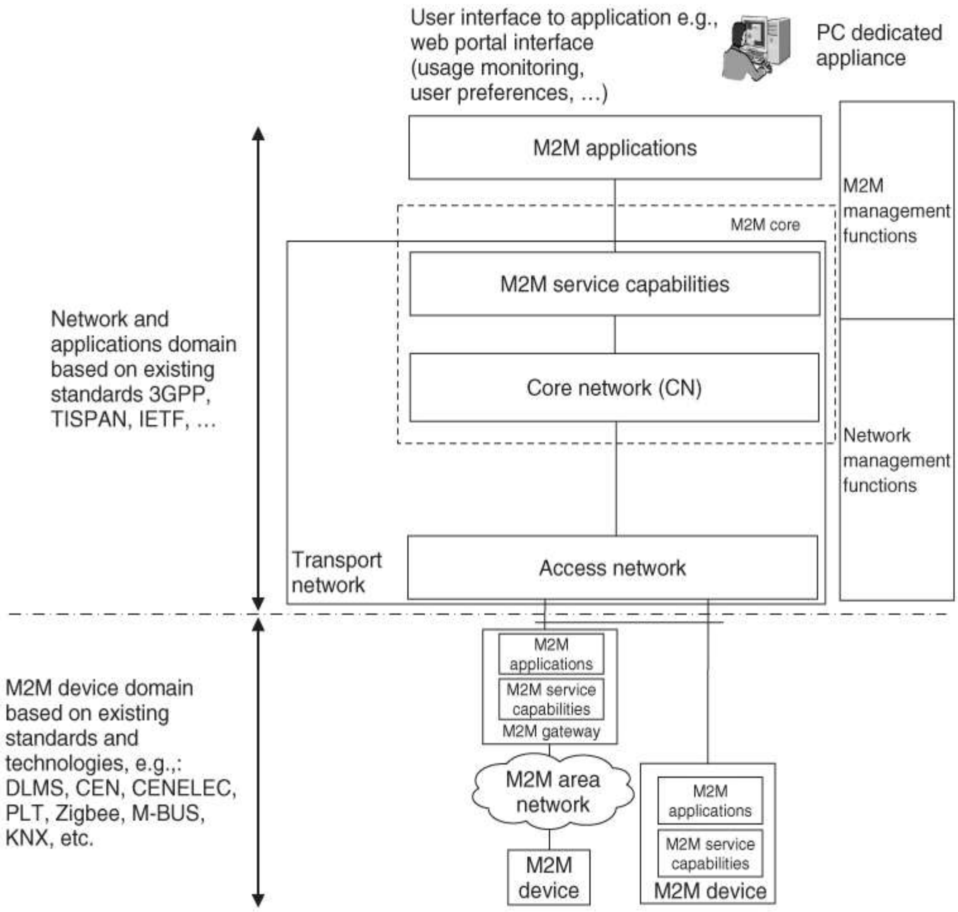


Figure 4. M2M system architecture

Future of M2M systems

In the future, for a real deployment of M2M systems, applications will increasingly focus on logic functions such as data mediation, security and device management. It will be based on a set of open, standardized and IT-friendly APIs. Horizontal platform has to provide access to more core network interfaces with services like location or QoS, without the burden of implementing protocols, because the complexity has to be hidden through the use of single open API. Devices have to implement an open interface toward the horizontal platform and service capabilities are exposed to device applications similar to the network side, as presented in Fig. 5.

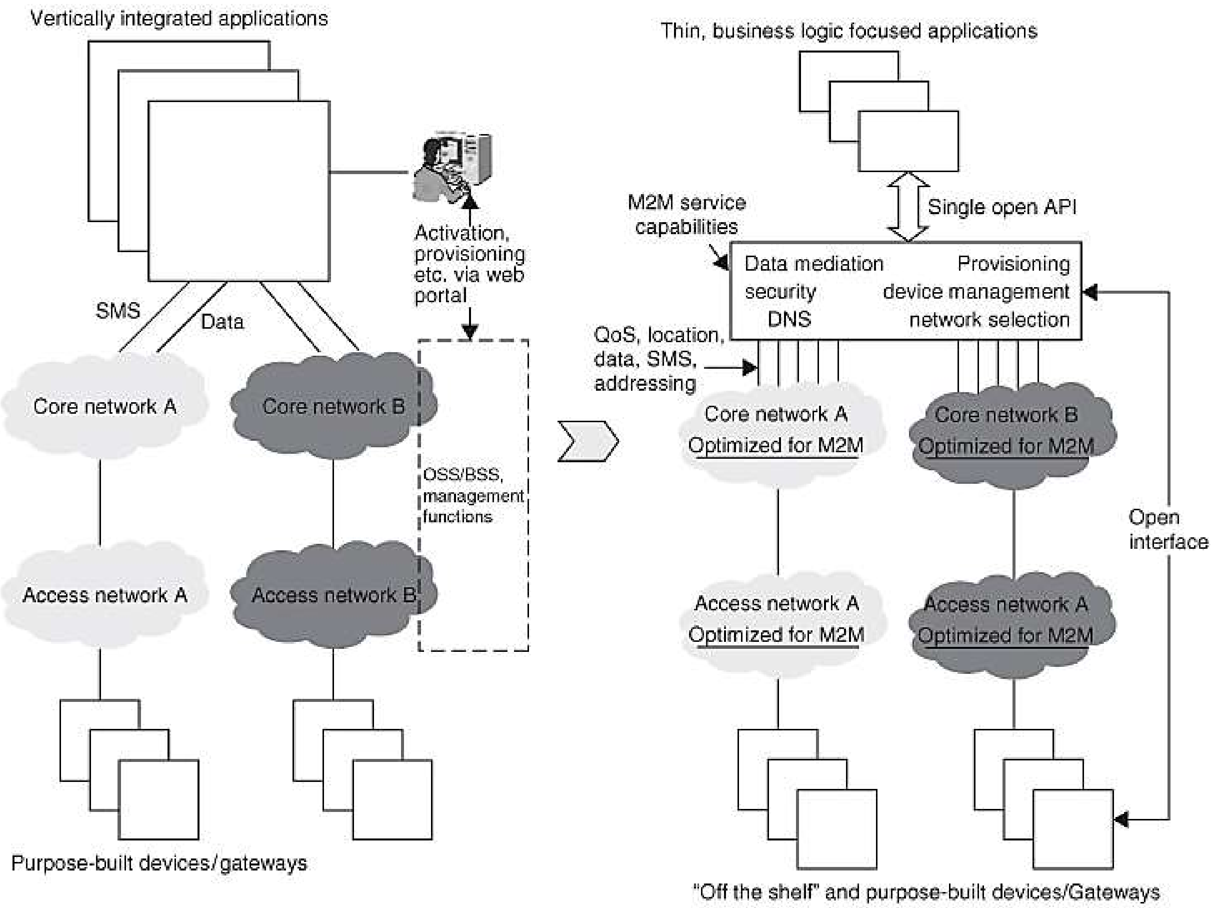


Figure 5: Key principles for future M2M deployment.

Introduction – Software Defined Radio

Software Defined Radio

For years now, more and more people and objects have been using wireless communication, particularly with the development of the Internet of Things. In addition, more and more radio communication systems are being developed. For example, each smartphone can communicate using: NFC, AM and FM radio (only receiving), GSM (2G), GPRS (2.5G), EDGE (2.75), UMTS (3G), HSDPA/HSUPA (3.5G), HSPA+ (3.75G), LTE (4G), LTE-A (4G+), 5G, Bluetooth/BLE, Wi-Fi. There are also an increasing number of standardised or proprietary solutions: Bluetooth/BLE (IEEE 802.15.1), cellular, LoRaWAN, LTE-M, NB-IoT, NFC (ISO/IEC 14443), RFID (ISO/IEC 18000), RuBee (IEEE 1902.1), SigFox, UWB (IEEE 802.15.3), Wi-Fi (IEEE 8902.11), WiMAX (IEEE 802.16), ZigBee (IEEE 802.15.4), Z-Wave, etc. As a result, systems are becoming increasingly complex, requiring more and more documentation, and the cost of equipment is rising.

Historically, a hardware-based approach has been favoured: each standard has its own optimised hardware. Thus, the hardware is well optimized: i.e. compact (small and light), very fast, low power consumption and cheap (in terms of unit cost for large quantities in production). Nevertheless, this hardware is specific, so: non-evolving, non-updatable and non-reusable; long and expensive to develop; hard to integrate (in terms of EMC, placement); and there are always more interfaces in the same object.

Thus, a new paradigm has emerged in recent years: Software Defined Radio (SDR). The objective is to employ a software-based approach, where a generic hardware architecture is used and each standard is processed with a specific software. This approach based on a generic hardware is flexible and modular especially scalable, updatable and reusable, and allows fast development and production. Nevertheless, this hardware is expensive, thus make the production cost very high. It is also hard to integrate in terms of size, weight, power consumption. As there is no highly effective generic treatment, certain parts of the software have to be duplicated and adapted to each standard.

Finally, SDR can be likened to conventional radio, except that much of the signal processing carried out by the hardware will be done by the software. The aim is to dematerialise the specific radio interface as much as possible to make it more general and more malleable for the software, in the knowledge that perfect dematerialisation does not exist today, and that a specific, immutable, reduced hardware part is still necessary. This need for dematerialisation has many origins, not least the proliferation of wireless protocols.

Universal Software Radio Peripheral

The generic hardware employed in SDR is called Universal Software Radio Peripheral (USRP), as presented in Fig. 6 and Fig.7.

Although its aim is to do away with specific hardware, we can still see that in the analogue part, an unchanging piece of hardware is still essential. It is simply a question of processing the signal before sending it *via* an antenna, with the aim of eliminating parasitic elements and amplifying it. It is worth remembering that an SDR requires no less hardware than a conventional radio, but often more. The advantage is that this hardware is not specific and can be reorganised according to requirements. The RF hardware is specific in the sense that it is adapted for these systems and unchangeable.

The *baseband processor* corresponds to the digital signal processing. This can be done by a microprocessor or FPGA or by a traditional PC.

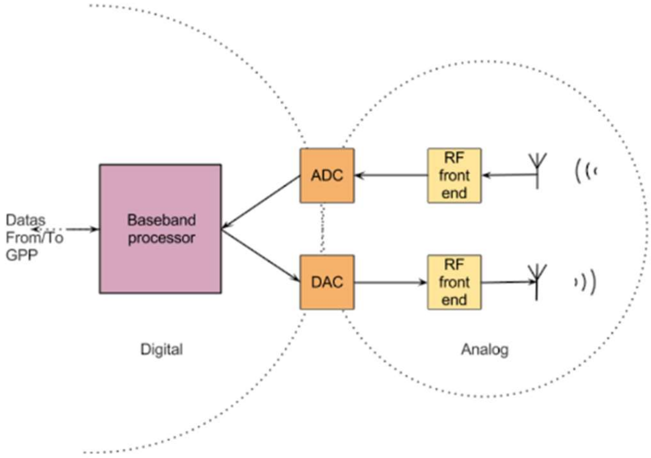


Figure 6: Simplified bloc diagram of an SDR.

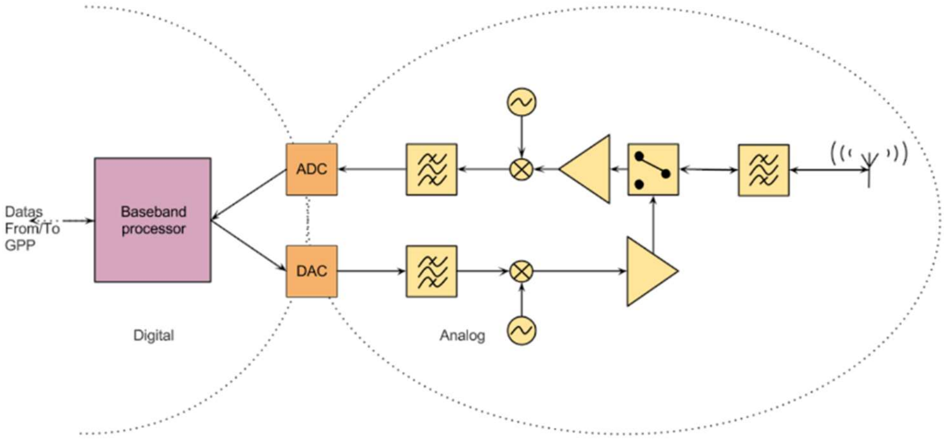
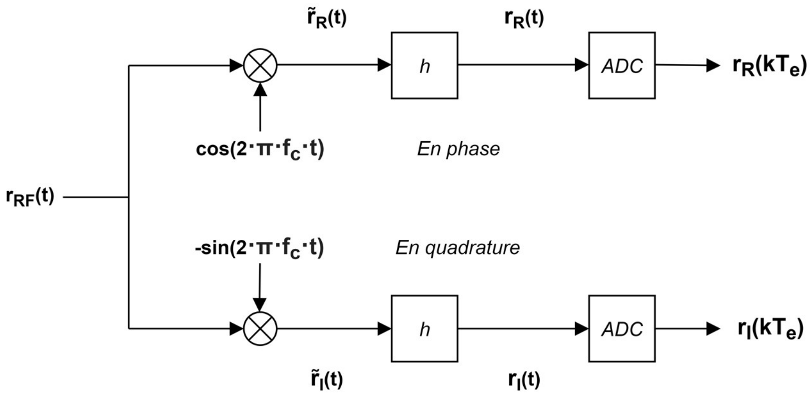


Figure 7: Simplified bloc diagram of an USRP.

In the case of the reception, the hardware consists of two layers, as presented in Fig. 7:

- An in-phase and quadrature demodulator (I/Q): to transpose the radio frequency signal into baseband and decompose it into electrical signals in phase and quadrature
- An analogue to digital converter (ADC): to sample signals in phase and quadrature

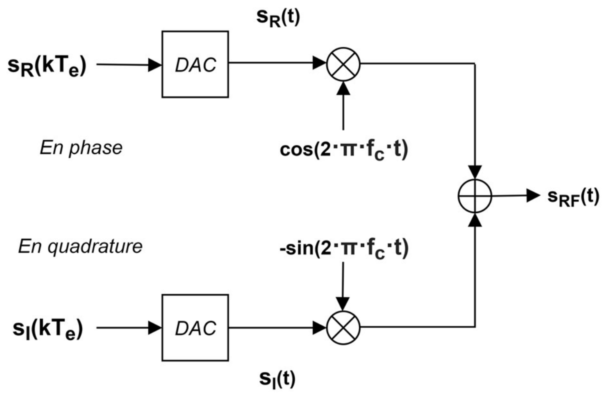


Récepteur

Figure 8: Simplified bloc diagram of the receiver.

In the case of the emission, the hardware consists of two layers, as presented in Fig. 8:

- A digital to analogue converter (DAC): to unsample signals in phase and quadrature.
- An in-phase and quadrature modulator (I/Q): to compose a 'high frequency' radio frequency signal from electrical signals in phase and quadrature.



Emetteur

Figure 9: Simplified bloc diagram of the transmitter.

During the laboratory project, two kinds of USRP could be employed, with suitable antennas:

- National Instruments B210 (requiring an RF daughterboard adapted to the frequency range):
  - Frequency band — 70 MHz – 6 GHz
  - Sample frequency — 15 MS/s (max. 61 MS/s)
  - Accuracy — 12 bits/S
  - Integrated or available processing devices — FPGA, CPU
  - Parallel implementation of several applications — Simultaneous Rx and Tx
  - Ability to update on the fly without stopping or rebooting
  - Price — >1000 €

- Analog Devices Adalm-Pluto:
  - Frequency band 325 MHz – 3.8 GHz
  - Sample frequency 65.2 kS/s to 61.44 MS/s
  - Accuracy 12 bits/S
  - Integrated or available processing devices CPU
  - Parallel implementation of several applications Simultaneous Rx and Tx
  - Ability to update on the fly without stopping or rebooting
  - Price >200 €

Other USRP are available at INSA, some of which are specifically designed for stand-alone deployment.

GNU Radio

GNU Radio is free, open-source software that lets you implement software radios using a block system for signal processing. It is distributed under the GPL licence. GNU Radio also respects the philosophy of Unix, as expressed by Mike Gancarz, and in particular the fact that it is necessary to "[...] write programs that do one thing, but do it well. [...]".

Although GNU Radio applications are in Python, it is possible for developers who use it to implement fairly complex real-time systems, as most of the signal processing libraries are programmed in C++, offering the possibility of using 16-bits floating point processor extensions where possible.

Today GNU Radio is distributed with its gnuradio-companion tool, originally third-party software. This tool offers a graphical interface that allows GNU Radio blocks to be manipulated like bricks that are placed in a space, configured and linked together. This interface greatly simplifies the use of GNU Radio, which would otherwise have to be handled directly with a python script.

gnuradio-companion allows you to generate these python scripts, which can be used on any computer that includes the GNU Radio libraries.

Work to do

Two main applications will be provided: WSN for aerospace and WSN for home (or health care) automation, but you are free to propose your own application. The specifications of each application will be established during the supervised sessions. These applications concern only static WSN: no mobility must be considered.

Each group (composed of 6 students, split in pairs) will choose one application. Each group will have to design and implement:

- the Physical (PHY) layer (and optionally, the sensor(s) connection)
- the Medium Access Control (MAC) layer
- the link to the gateway (in Wi-Fi or UMTS), i.e. the connection between the WSN and the gateway, and the gateway and a small server that you will set up.

Each group has to discuss for their protocol:

- the availability
- the security
- the quality of service (QoS): wait time, data rate, latency, etc.
- the power consumption

Design of the Physical layer

This laboratory project will use GnuRadio software and USRP. The students can install on their PC the GnuRadio software, if they wish to, or use INSA platforms (laptop or Raspberry Pi).

The students will be autonomous in the use of GnuRadio and USRP platforms. Please take the documentation from GnuRadio website and from USRP website and other dedicated blogs.

The main work will be focused on the communication dimensioning using GnuRadio and USRP platforms for sensor wireless communicating nodes, as well for the gateways which has to have at least two communication interfaces (one to WSN, the other to Internet, for example in Wi-Fi or UMTS).

Design of the Medium Access Control layer

To be able to choose the optimum MAC layer for your application, the first work you have to do is to establish the state of the art of the Mac layers for WSN. The students have to prepare, based on the scientific research literature, a complete list of the WSN dedicated Mac layers. Your research has to list also the main characteristics of each Mac layer in the context of WSN and at least the following characteristics: data rate, power consumption, number of supported nodes, QoS and its suitably to be used for WSN. You have also made a critical comparative analysis of these different Mac layers based on your knowledge and other available analysis found in the scientific literature with respect to their use for the WSN context.

The students provided a report including their detailed results of the bibliographic research and their analysis as assignment 2. They will use the results of this assignment **for their choice and implementation of the MAC layer in this project.**

Report

The students have to provide a complete report of the developed M2M communication layers, including their performances and their ability to respond to the provided specifications.

Applications

Smart medical home

A Wireless Sensors Network needs to be designed for aa smart home that can accommodate an elderly or sick person. We need to know where the person is and be able to detect if they have fallen without using a camera, which could frighten them away from their private life.

There are several types of node:

- Periodic collectors: These are state sensors. They can be temperature, humidity or light sensors, or a weather station. These nodes require a relatively low data rate and frequency.
- Actuators: These receive an order to be executed. They can be an electric shutter, a lamp, an electronic lock or a heating system. These nodes require low flow rates but must ensure a certain reactivity, depending on the operation to be carried out.
- Event-driven sensors: These are sensors that will occasionally transmit information. For example, when a user wants to open or close a shutter. These nodes need a low data rate but have a high time constraint in terms of responsiveness.
- Emergency sensors: These sensors can be linked to a fall detection system or an emergency call button to alert a rescue centre. These nodes are event-based collectors with very strong time constraints. The information they send back must be processed as a priority. In principle, they do not require a high data rate.

It is important to define the nodes that will be deployed in the house so that the actions that will take place in the house can be implemented. The following nodes can be considered.

Periodic collectors

- Temperature sensor: Each room will have a temperature sensor, which will then be used to manage heating and air conditioning. Energy management is also a priority.
- Interior light sensor: Each room will have an interior light sensor.
- Depending on other data such as the brightness outside, the presence of a person in the room and the position of the shutters, actions will be taken (opening the shutters, managing the lights).
- External light sensor: Each façade of the building will be fitted with a light sensor to obtain the level of sunlight in the building.
- Weather sensor: Provides information on weather conditions in real time (humidity, sunshine, wind, rain). This information can be used to orientate solar panels, activate wind power, open external blinds or close them in the event of strong winds.

Actuators

- External shutters: These are motors that raise and lower external roller shutters to protect the home or add thermal insulation to windows and doors.
- Interior solar protection shutters: These actuators are also motors that can be used to raise or lower interior shutters, which are not opaque and do not provide thermal insulation; they simply hide the sun to limit reflection, like a curtain. This system is widely used in office buildings to provide a certain level of comfort at the workstation.
- Heating system: The actuator is either a central, autonomous heating system, such as an automatic heating management system, or electric heaters that are controlled individually by the home automation control unit.

- Lamp actuator: These actuators are either lamp on/off switches that switch the lamps on and off, or dimmers that vary the brightness of the lamps. Dimming systems are used to achieve constant brightness in a room, such as an office. The lamp brightness level is controlled according to the information provided by the interior brightness sensors.
- Door opening: These sensors are motors that open the various doors in a home. For example, opening a garage door or a gate at the entrance to the property. Or automatically open the main door of a home adapted for people with reduced mobility.

Event-driven sensors

- Presence sensor: Each living area will have a presence sensor. Motion sensors can be installed in high-traffic areas such as corridors, entrance halls and toilets.
- Buttons: Buttons can be used to perform certain actions, such as forcing a light to come on, opening or closing the shutters in a room, or switching off the lights.
- Remote control: A remote control can be used to perform a range of actions similar to those performed using buttons. It can also be used to interact with the building's multimedia systems.

Emergency sensors

- Fall sensor: Each room can be fitted with a fall sensor. In addition, people at risk can be fitted with an on-board fall sensor in the form of a bracelet or necklace. The same equipment will incorporate an emergency call button for use in the event of health concerns.
- Unwanted presence detector: Sensors in the garden and in the home, particularly at doors and windows, can be deployed to enable the local authorities to intervene in the event of theft or other malicious events.
- Alarm deactivation: An alarm deactivation code box can send deactivation requests and messages if the wrong code is entered, notifying the user of attempts to enter the building.

Examples of scenario

A first scenario could be initiated by the arrival of a person in a room. The room is initially heated at low level, the lights are off and the shutters are closed. The arrival of a person in the room will trigger event-based information feedback. A number of actions will be taken based on the information regularly fed back by the room's internal and external brightness and temperature sensors. For example, if the light level in the room is too low, but it's very sunny outside, the shutters will be opened, allowing the room to be lit. In addition, the heating can be turned up while the sun warms the room, then back down to its initial setting once the desired temperature has been reached and maintained.

If the external light level is low, the lamp will be activated and the heating increased. Energy management is therefore a major issue, and depending on the weather conditions, the most energy-efficient solution will be implemented.

If the person becomes unwell, an emergency call must be possible, or a fall or unwellness detection system.

Ambient energy harvesting to recharge the nodes is permitted and should be considered.

Structure Health Monitoring for Avionic

This is a SHM (Structure Health Monitoring) application deployed on an aircraft. It requires the deployment of a network of more than 1,000 wireless sensors spread over the entire surface of the aircraft.

These sensors can be temperature sensors, strain gauges, pressure sensors or any other type you think necessary. The clocks between the communicating sensors must be synchronised so that the measurements can be correlated. By triangulation, we can then locate any structural faults in the aircraft. This management of local clock drift over time is essential in a sensor network, especially when deployed in an SHM context. Synchronisation must be as precise as possible.

As far as the autonomy of this network is concerned, the network of communicating sensors must be able to operate for several years (e.g. 3 years) by making regular flights of around ten hours.

Ambient energy harvesting to recharge the nodes is permitted and should be considered.

Examples of scenario

The aerospace industry is subjected to stringent requirements regarding aircraft safety and performance. This translates into both thorough testing during the development stages and regular monitoring of the aircraft's systems wear during its operational lifetime. Wireless Sensor Networks (WSN) promise quick and easy deployment and removal during development, but also constant monitoring of the aircraft's wear while operating. This will lead to increased safety and reduced development and operating cost. Developing energy efficient communication systems is critical to the successful development of WSN for aerospace monitoring applications.

Recent advances in microelectronics processes and digital communications have enabled new possibilities in this field. Wireless Sensor Networks have been proposed as a mean to monitor aircraft aging and improve maintenance operational burden and costs. We are proposing a new machine to machine (M2M) sensor network architecture, which employs existing cellular networks and a highly energy efficient wireless communication nodes.

The main application for such a system is an aircraft monitoring and maintenance scenario. Under a typical scenario, an airplane would be fitted with wireless sensors during manufacturing. These sensors would be placed at key location to monitor aircraft parts and systems subjected to aging or critical for the aircraft safety or performance. These sensors' locations, often inside the plane itself, would prevent them from communicating directly with each other or to a central monitoring system aboard the plane, let alone the aircraft manufacturer's or the airline's facilities. Relay nodes may be employed to make all the sensor nodes reachable from the gateway node by relaying data transmissions while also improving energy efficiency. Depending on the flight profile, network availability and the data itself, the gateway node could buffer data until arrival at the parking spot, or send data mid-flight if required. This system would improve upon the current Aircraft Communications Addressing and Reporting System (ACARS) which is severely constrained in terms of monitored systems and data rates. Indeed, most airlines limit maintenance data to the engines. The proposed system would improve overall aircraft maintenance and safety on a similar scale as ACARS did for engine maintenance. The location of the sensor nodes will severely constrain their energy budget, because of the difficulty of accessing and recharging a battery system, and because powering them with wire would defeat the benefit of wireless communications.

The most promising technology for powering these wireless nodes is advanced battery concepts with extended lifetimes coupled with energy harvesting. Either way, the power budget of these devices will be at least an order of magnitude smaller than what is common for wireless technologies readily available and in the milliwatts range.

The work during this lab classes has to focus on designing a M2M wireless communication system suitable for the sensor to gateway wireless communication offering maximum energy efficiency to provide monitoring services for the lifetime of the airframe.